CPR: Retrieval Augmented Generation for Copyright Protection

Aditya Golatkar Alessandro Achille Luca Zancato
Yu-Xiang Wang Ashwin Swaminathan Stefano Soatto
AWS AI Labs
{agolatka,aachille,zancato,yuxiangw, swashwin,soattos}@amazon.com

Abstract

Retrieval Augmented Generation (RAG) is emerging as a flexible and robust technique to adapt models to private users data without training, to handle credit attribution, and to allow efficient machine unlearning at scale. However, RAG techniques for image generation may lead to parts of the retrieved samples being copied in the model's output. To reduce risks of leaking private information contained in the retrieved set, we introduce Copy-Protected generation with Retrieval (CPR), a new method for RAG with strong copyright protection guarantees in a mixed-private setting for diffusion models. CPR allows to condition the output of diffusion models on a set of retrieved images, while also guaranteeing that unique identifiable information about those example is not exposed in the generated outputs. In particular, it does so by sampling from a mixture of public (safe) distribution and private (user) distribution by merging their diffusion scores at inference. We prove that CPR satisfies Near Access Freeness (NAF) which bounds the amount of information an attacker may be able to extract from the generated images. We provide two algorithms for copyright protection, CPR-KL and CPR-Choose. Unlike previously proposed rejection-sampling-based NAF methods, our methods enable efficient copyright-protected sampling with a single run of backward diffusion. We show that our method can be applied to any pre-trained conditional diffusion model, such as Stable Diffusion or unCLIP. In particular, we empirically show that applying CPR on top of unCLIP improves quality and text-to-image alignment of the generated results (81.4 to 83.17 on TIFA benchmark), while enabling credit attribution, copy-right protection, and deterministic, constant time, unlearning.

1. Introduction

Foundation model users may need to adapt large-scale Diffusion Models to their use cases, like personalization, editing, content creation etc. However, fine-tuning the model on the user data is often not an option. This is in part due to the steep cost of fine-tuning models, but also because user data is a mutable entity: new data is constantly added, and low-quality data may often be filtered out. Moreover, data owners may, at any point, change their mind and demand that their data be removed.

Retrieval Augmented Generation (RAG) has emerged as a promising method to handle these situations. Rather than using the user data to fine-tune the model, supporting samples are retrieved from it at inference time to guide the generation of new samples. Data may be easily added or removed from the retrieval data store without changes to the model, and users may access different subsets of the data based on their access-right. However, RAG methods are double edged, direct access to retrieved reference images often significantly improves the quality of generated samples but as we depict in Fig. 1, RAG models are prone to copy information from the retrieved examples into the model output, potentially resulting in significant leak of private information. We formalize this observation in Sec. 5 and we show that applying RAG on top of a public model, while retrieving private user data at inference time, cannot satisfy even the weaker notion of privacy.

To remedy this, we introduce Copy-Protected Generation with Retrieval (CPR). CPR retrieves multiple private examples from the private user data pool. Information from all these samples is combined to generate a "private" diffusion flow which uses common information of those samples while discarding any unique and identifiable information. The resulting private flow is then optimally combined with the "public" flow generated by the base model to generate new outputs which still benefit from the retrieved samples, but minimize the risk of information leak.

In particular, we show that our method satisfies the recently proposed notion of copyright protection using Near Access Freeness (NAF) [62], a relaxation of differential privacy aimed at protecting specific attribute of the training data. Differently from previously proposed methods like [62] that realize NAF with a computationally expensive rejection sampling method, CPR does so by construction during the generation. Hence making our method significantly faster than the previous baselines and while also keeping

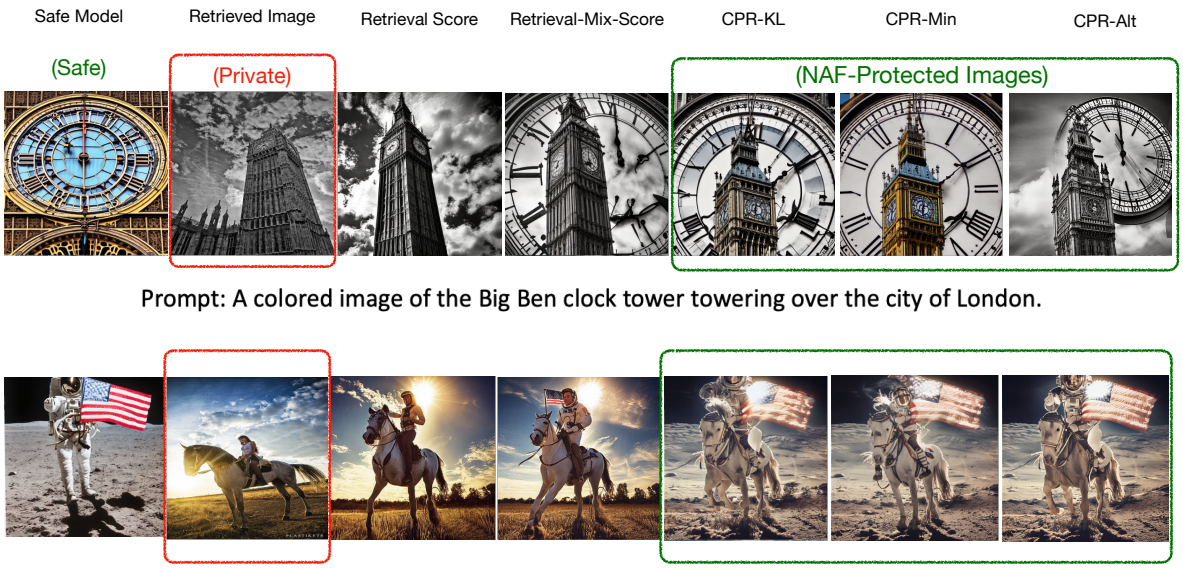


Figure 1. **RAG vs CPR image generation.** Images generated using the given prompt for a fixed random seed using different methods. *Safe Model*: Pre-trained model with no access to the retrievable data store, *Retrieval-Score*: Image generated using Eq. (7), *Retrieval-Mix-Score*: Image generated using Eq. (8), *CPR-KL, CPR-Min, CPR-Alt*: Images generated using our algorithms in Algorithm 1 Sec. 5.2.1 and Algorithm 2 Sec. 5.2.1. Images generated without CPR bear more resemblance to the retrieved image, compared to the CPR generated images, which are different from the retrieved image, while preserving the underlying concept in the prompt (for example the astronaut seems to be on Moon, Big Ben is more textured with different design).

inference cost deterministic.

Theoretically, we formally prove in Lemmas 1 and 2 that CPR offers strong protection guarantees by ensuring that the generated samples contain at most k-bits of unique information about retrieved samples, where k can be tuned by the user as required by the application. Empirically, we show that CPR can use private data to improve quality of the generated images ($81.4 \rightarrow 83.17$ TIFA score) while maintaining privacy guarantees on the retrievable data.

The rest of the paper is organized as follows. In Sec. 2 we provide a study of the relevant related works in RAG and privacy. In Sec. 3 we define the necessary notations, and in Sec. 4 show how to perform RAG with pre-trained text-to-image diffusion models by formulating inference over mixture of public-private distribution. In Sec. 5 we provide our CPR algorithm, along with its theoretical guarantees, provide empirical evaluation of our method in Sec. 6, followed by some discussion in Sec. 7.

2. Related Works

Retrieval Augmented Generation Retrival Augmented Generation (RAG) methods have been successfully applied to large language models (LLMs) [24, 31, 42, 45, 54]. RAG has been shown to outperform even LLMs trained jointly on the training set and the retrievable data pool. RAG have also been explored for image synthesis [2, 7, 9, 50, 53, 67, 69]. However, rather than reusing existing models, current methods require training of retrieval-specific architectures which — unlike the standard text-to-image diffusion models [46, 47, 49, 68] — can be prompted with several retrieved images along with conditional information, such as text, as inputs. Instead, we explore RAG using more generic pre-trained text-to-image diffusion models. [2, 50, 69] train a diffusion-based image retrieval model that can be prompted with latent image embeddings, while [9, 67] use autoregressive generative models inspired from LLMs.

Image Manipulation Several recent works [4, 18, 25, 35, 40, 41, 51, 63] have provided methods for image manipulation, editing by either fine-tuning or changing the cross-attention values at inference. With an appropriate retrieval function, and database such methods can be used to perform retrieval augmented generation by merging diffusion scores [12, 23, 38]. However, the manipulation methods significantly lower inference speed. Instead, we opt to use the unCLIP model [47] to generate a backward flow using the retrieved images, and merge it with the flow generated by a base text-to-image diffusion model at inference [12, 23, 38, 49].

Privacy Recent works [5, 6, 56, 57] have shown that such

models are able to memorize their training data. This raises several privacy challenges, including:

Unlearning: Machine unlearning [3, 20] enables users to delete their data from the weights of trained models [17, 43]. [3, 14, 19, 23, 34, 39] provide training methods which makes unlearning efficient, for example by breaking the dataset into multiple shards and training separate models on each, followed by ensembling at inference. Despite their improved privacy utility trade-offs compared to a single model, such approaches still require frequent retraining/fine-tuning. On the other hand, we propose splitting the training dataset into a core safe dataset [21] used to train a core model, and a user owned private data store used to retrieve samples. This allow instantaneous forgetting of any private sample without having to retrain the model. This strategy also improve performance (alignment), and enables easy continual learning by simply adding new data to the data store. Moreover, privacy of data which are never retrieved is completely preserved, unlike unlearning or differential privacy [15, 22] methods which mix information about the entire dataset in the weights.

Copyright Protection: Memorization in foundation models also increases the risk of copying, style mimicry and copyright [5, 6, 52] at inference. [62] proposed a definition for copying in generative models using near access freeness (NAF), and provided the CP-Δ algorithm for copy-protected generation. CP-Δ uses two generative models, trained on two disjoint splits of the data, and then at inference samples from the product and the minimum of the two distributions. However, using it directly out-of-the-box for diffusion models is challenging. Instead, they propose another algorithm, CP-k, based on rejection sampling. Diffusion models however tend to have slow inference speed, and adding rejection sampling further aggravates the speed problem. To address this, we introduce CPR, which provides a method to sample using CP-Δ (satisfies NAF) in a single run, without the need for rejection sampling.

3. Preliminaries

Let $p_0(x_0)$ be a data distribution over images, which we seek to model using a diffusion model [27, 58, 60]. Score based diffusion models models [60] define a (variance preserving) forward flow through a SDE, which transforms the distribution $p_0(x_0)$ at time $t = 0$ in a reference distribution $p_1(x_1) = N(0, I)$ at time $t = 1$:

$$dx_t = -\frac{1}{2}\beta_t x_t dt + \sqrt{\beta_t} d\omega_t \tag{1}$$

where x_t is the diffused input at time t, $d\omega_t$ is a standard Wiener process, and β_t are time varying coefficients (in practice implemented through linear or cosine scheduling), which determine the transition kernel and amount of noise added over time. The intermediate result $p_t(x_t)$ of the diffusion process at time t equivalently expressed as the result of applying a Gaussian kernel $p_t(x_t|x_0) = \mathcal{N}(x_t; \gamma_t x_o, \sigma_t^2 I)$ to $p_0(x_0)$, resulting in $p_t(x_t) = \int_{x_0} p_t(x_t|x_0)p_0(x_0)dx_0$, where $\gamma_t = \exp(-\frac{1}{2} \cdot \int_0^t \beta_t dt)$ and $\sigma_t^2 = 1 - \gamma_t^2$.

The forward process in Eq. (1) can be inverted through a corresponding backward process [37, 60]. In particular, this process can be used to generate samples of $p_0(x_0)$ starting from a sample of $p_1(x_1) = N(0, I)$:

$$dx_t = \big(-\frac{1}{2}\beta_t x_t - \nabla_{x_t} \log p_t(x_t)\big)dt + \sqrt{\beta_t} d\omega_t \tag{2}$$

where $\nabla_{x_t} \log p_t(x_t)$ is the score function of data distribution at t. Efficiently computing the score function is difficult. Instead, it can be approximated $\nabla_{x_t} \log p_t(x_t) \approx s_\theta(x_t, t)$ using a deep network $s_\theta(x_t, t)$, *i.e.*, a diffusion model. In practice, diffusion models are often trained to take additional inputs $s_\theta(x_t, t, c)$ in order to model a conditional distribution $p_0(x_0|c)$, where the conditioning c provides additional information about the sample to generate, such as textual prompts [11, 26]. Given samples of the joint distribution $p_0(x_0, c)$, a diffusion model $s_\theta(x_t, t, c)$ can be trained by minimizing the score-matching objective:

$$\mathbb{E}_{(x_0,c)\sim p_0(x_0,c)} \mathbb{E}_t \big[\|s_\theta(x_t, t, c) - \nabla_{x_t} \log p_t(x_t|x_0)\|\big] \tag{3}$$

Directly generating samples using the backward flow modeled by $s_\theta(x_t, t, c)$ can result into poor alignment [12, 27, 58]. This can be improved through classifier-free guidance [26], which uses the modified score:

$$s_\theta(x_t, t, \phi) + \lambda(s_\theta(x_t, t, c) - s_\theta(x_t, t, \emptyset)),$$

where the hyper-parameter λ controls the *guidance scale*, and $\emptyset$ denotes that no conditioning is fed to the model.

4. Mixed-Privacy RAG

In this section, we introduce a method for privacy-enabled RAG that is based on the notion of mixed-privacy [21, 22].

Let $D = \{x^i, c^i\}_{i=1}^N \sim p(x, c)$ be a safe training dataset — meaning that samples are considered public in the differential-privacy sense (see [21, 22]). We assume D is used to train a core public diffusion model $s_\theta(x_t, t, c)$, that accepts c as conditioning information. We shall also assume that c is the output of a CLIP encoder $c = \text{CLIP}(\texttt{<prompt>})$ fed with either a text prompt or an image prompt. Furthermore, let $D_{\text{private}} = \{x^i, c^i\}_{i=1}^M$ be a private dataset which may require frequent unlearning, or may require privacy or copyright protection. We shall consider D_{private} as our data store for retrieval.

Retrieval At inference time, given a user prompt c_{test} we retrieve a set of m relevant examples $D_{\text{retr}} =$

		Number of Retrievals														
		1					3					5				
Model	Safe Model	Retrieval -Score	Retrieval -Mix- Score	CPR-KL	CPR-Min	CPR-Alt	Retrieval -Score	Retrieval -Mix- Score	CPR-KL	CPR-Min	CPR-Alt	Retrieval -Score	Retrieval -Mix- Score	CPR-KL	CPR-Min	CPR-Alt
TIFA (COCO)	85.5	74.46	84.47	86.68	85.89	86.73	80.07	86.01	84.46	85.91	86.44	80.93	86.1	86.76	86.57	86.58
TIFA (Non-COCO)	77.43	58.81	76.66	79.57	78.1	79.72	67.77	78.25	79.02	78.27	78.80	69.21	78.46	78.94	78.66	78.81
TIFA (Avg)	81.4	66.45	80.49	83.16	81.92	83.17	73.81	82.06	82.67	82.02	82.57	74.96	82.21	82.78	82.25	82.63

Table 1. **Improved text-to-tmage alignment with retrieval**: We compute the TIFA score [29] which measures the text-to-image alignment on a set of prompts (Higher is better). We use a subset of MSCOCO [36] (2k images with high aesthetic score) as the private data store. We show that simply using the retrieval-score (in Eq. (7)) is not enough to improve alignment, instead using the retrieval-mixture-score (in Eq. (8)) is important to generate aligned and well composed images. CPR-KL, CPR-Min, CPR-Alt, meaningfully improve the text-to-image alignment across different retrieval settings compared to the base model while protecting the private data store.

$\{(x_i, \phi(c_i, c_{\text{test}})\}_{i=1}^m \subset D_{\text{private}}$ to aid the generation process. For simplicity, we simply retrieve the closest m samples based on L_2-CLIP similarity score:

$$\text{score} = \|c_{\text{test}} - c_i\| + \|c_{\text{test}} - \text{CLIP}(x_i)\|.$$

Note however that in D_{retr} we are modifying the prompt of the retrieved samples through a function $\phi(c_i, c_{\text{test}}) = c_i + c_{\text{test}}$ in order to align them better with the user prompt.

Mixture-of-Distribution Retrieved samples are used to improve the generation of new samples. Formally, the goal of CPR is to modify the sampling backward process in order to generate samples from a weighted mixture of the distribution of D and D_{retr} [12, 23, 38]:

$$p(x|c) = w_0 p_D(x|c) + w_1 p_{D_{\text{retr}}}(x|c) \tag{4}$$

where the weights $w_0 = \lambda$ and $w_1 = 1 - \lambda$ allow the user to control the contribution of the retrieved samples at inference time through an hyperparameter $0 < \lambda < 1$.

Mixture-of-Score To sample from this mixture distribution, we need to compute its score function $\nabla \log p_t(x_t)$ at time t (see eq. 2). From Sec. 3 we have:

$$p_t(x_t|c) = \int p_t(x_t|x_0)\left[w_0 p_D(x_0|c) + w_1 p_{D_{\text{retr}}}(x_0|c)\right] dx_0 \tag{5}$$

where $p_t(x_t|x_0) = \mathcal{N}(x_t; \gamma_t x_0, \sigma_t^2 I)$ is a Gaussian kernel. The following proposition expresses the score of the mixture as a function of the score of the individual components:

Proposition 1. *Let $p_t(x_t|c)$ be as in Eq. (5), then $\nabla_{x_t} \log p_t(x_t|c)$ is given by:*

$$\nabla_{x_t} \log p_t = \hat{w}_0^t \nabla_{x_t} \log p_D^t(x_t|c) + \hat{w}_1^t \nabla_{x_t} \log p_{D_{retr}}^t(x_t|c)$$

where we have defined:

$$\hat{w}_0^t = w_0 \frac{p_D^t(x_t|c)}{p_t(x_t|c)}, \quad \hat{w}_1^t = w_1 \frac{p_{D_{retr}}^t(x_t|c)}{p_t(x_t|c)}.$$

and $p_D^t(x_t|c)$ denotes the forward flow of the distribution $p_D(x_t|c)$ at time t (and similarly for $p_{D_{retr}}^t(x_t|c)$) and $p_t(x_t|c) = p_D^t(x_t|c) + p_{D_{retr}}^t(x_t|c)$.

While $\hat{w}_0^t$ and $\hat{w}_1^t$ could be computed exactly, we find that treating them as fixed hyper-parameters simplifies the implementation and performs well. The scores $\nabla_{x_t} \log p_D(x_t|c)$ can be approximated empirically by a diffusion model $s_{\theta_0}(x, t, c)$ trained on D. However, we do not have a model trained on the retrieved data D_{retr} to estimate $\nabla_{x_t} \log p_{D_{\text{retr}}}(x_t|c)$. To solve the issue, recall that such a diffusion model s_{θ_1} that minimizes the loss:

$$s_{\theta_1} = \arg\min_{s_\theta} \mathbb{E}_{(x_0,c)\sim p_{D_{\text{retr}}}} \mathbb{E}_{x_t}\Big[\|s_\theta(x_t, t, c) - \nabla_{x_t} \log\Big(\int p_t(x_t|x_0) p_{D_{\text{retr}}}(x_0, c)\Big)\|\Big] \tag{6}$$

Since $|D_{\text{retr}}| \ll |D|$, we expect the minimizer θ_1 to be a small small perturbation $\theta_1 = \theta_0 + \Delta\theta_1$. However, fine-tuning $s_{\theta_0}(x, t, c)$ to find such $\Delta\theta_1$ for every inference request is computationally prohibitive.

Instead of fine-tuning, we approximate the expected behavior of s_{θ_1} through prompting. Textual inversion and prompt tuning have been shown to perform comparably to fine-tuning on small datasets while using orders of magnitute less parameters [18, 35, 55, 65]. However, despite the reduction, it is still cumbersome to fine-tune at inference. Instead we propose manually modifying the user prompt c_{test} using the CLIP embeddings of the retrieved samples, and define the *retrieval-score function*:

$$\hat{s}_{\theta_0}(x_t, t, c_{\text{test}}) \triangleq s_{\theta_0}\Big(x_t, t, \frac{1}{m} \sum_{x_i \in D_{\text{retr}}} \text{CLIP}(x_i)\Big) \tag{7}$$

We visualize in Fig. 1 the results of sampling with Eq. (7). This definition is motivated by the following proposition, which bounds the distance of Eq. (7) from the optimal.

Algorithm 1: CPR-KL

Input: $\nabla_{x_t} \log \int q_t(x_t|x_0)q^{(1)}(x|c)dx_0$, $\nabla_{x_t} \log \int q_t(x_t|x_0)q^{(2)}(x|c)dx_0$, T, N, c_{test}

Output: x_0

$x_T \sim \mathcal{N}(0, I)$

for $t = T \cdots 0$ **do**

  for $i = 1 \cdots N$ **do**

    $x_t = x_t + \frac{\epsilon_t}{2} \cdot \frac{1}{2}\Big(\nabla_{x_t} \log \int q_t(x_t|x_0)q^{(1)}(x|c_{\text{test}})dx_0 + \nabla_{x_t} \log \int q_t(x_t|x_0)q^{(2)}(x|c_{\text{test}}dx_0)\Big) + \sqrt{\epsilon_t}z$ where $z \sim \mathcal{N}(0, I)$

  $x_{t-1} = x_t$

Algorithm 2: CPR-Choose

Input: c_{test}, $\widetilde{s}(x_t, t, c; q^{(1)})$, $\widetilde{s}(x_t, t, c; q^{(2)})$, J, reverse-update(x_t, s_t)

Output: x_0

$x_T \sim \mathcal{N}(0, I)$

for $t = T \cdots, 0$ **do**

  if $t \in J$ **then**

    $s(x_t, t, c_{\text{test}}) = \widetilde{s}(x_t, t, c_{\text{test}}; q^{(2)})$

  else if $t \notin J$ **then**

    $s(x_t, t, c_{\text{test}}) = \widetilde{s}(x_t, t, c_{\text{test}}; q^{(1)})$

  $x_{t-1} = \text{reverse-update}(x_t, s(x_t, t, c_{\text{test}}))$

Proposition 2. *Assume that s_θ is Lipschitz in θ and c. Let $s_{\theta_0+\Delta\theta_1}(x_t, t, c)$ be the optimal solution to Eq. (6) and let D_{retr} the private samples retrieved using c_{test}. Then*

$$\|s_{\theta_1}(x_t, t, c) - \hat{s}_{\theta_0}(x_t, t, c_{test})\| \leq l_\theta\|\Delta\theta_1\| + l_c\Big\|c_{test} - \frac{1}{m}\sum_{x_i \in D_{retr}} \text{CLIP}(x_i)\Big\|$$

Above result shows that we can approximate the optimal diffusion model trained on retrieved data using the engineered prompt $\frac{1}{m}\sum_{x_i \in D_{\text{retr}}} \text{CLIP}(x_i)$, which only requires computing the CLIP embeddings of the retrieved images. Combining Proposition 1 and Proposition 2, we finally obtain an expression for the score function of retrieval-augmented mixture of distributions, which we call *retrieval-mixture-score*:

$$s_{RAG}(x_t, t, c_{\text{test}}; D_{\text{private}}) \triangleq \hat{w}_0 s_{\theta_0}(x_t, t, c_{\text{test}}) + \hat{w}_1 s_{\theta_0}\Big(x_t, t, (1/m)\sum_{x^i \in D_{\text{retr}}} \text{CLIP}(x^i)\Big) \quad (8)$$

Eq. (8) allows us any pre-trained CLIP-based diffusion model to generate retrieval augmented samples without any further changes (see Fig. 1). In Tab. 1 we show that Eq. (8) improves text-to-image alignment (TIFA goes from 81.4 to 82.21). Parallelly, the retrieval-mixture score function already has immediate application to privacy, since it makes it trivial to unlearn examples contained in D_{private} in constant time: these samples are not used to train any parameter, and hence can be forgotten by simply removing them from disk. However, samples retrieved at inference time can still leak private information, which tackle this next.

5. Copy-Protected Generation

In this section, we will provide algorithms for copyright protected generation — in the Near-Access Freeness sense of [62] — using our mixed-privacy RAG method.

Near-Access Freeness Let D_{private} be a set of private samples, whose information we want to protect, and let Δ be a divergence measure between probability distributions, such as the KL-divergence Δ_{KL} or thr max-divergence Δ_{max} (that is, the Renyi Divergence as $\alpha \to \infty$). Let safe : $D_{\text{private}} \to \mathcal{M}$ be a function which maps a sample $x^i \in D_{\text{private}}$ to a generative model trained without using that x^i. The Near Access-Free (NAF) criteria is defined as:

Definition 1 (NAF Definition 2.1 in [62])**.** *We say that a generative model $p(x|c)$ is k_c-near access-free (or k_c-NAF) on a prompt c with respect to $D_{private}$, and Δ,* safe*, if for all $x^i \in D_{private}$ we have $\Delta\Big(p(x|c)||\,\text{safe}_{x^i}(x|c)\Big) \leq k_c$.*

In practice, safe_{x^i} can be a model trained with `leave-one-out`, or be `sharded-safe` [62], or simply be the safe core diffusion model $s_{\theta_0}(x_t, t, c)$. The above definition says that to perform safe generation the output sample must be close in distribution to a model which did not have access to the private samples in D_{private}.

5.1. CPR-KL

We first report here Theorem 3.1 from [62] which provides a simple procedure to generate NAF-protected samples with respect to KL-divergence.

Theorem 1. *(Theorem 3.1 [62]) Given a dataset $\widetilde{D}$, and copyrighted samples $\mathcal{C} \in \widetilde{D}$, split $\widetilde{D}$ into two disjoint shards D_1, D_2, and train two generative models $q^{(1)}, q^{(2)}$ on each respectively. Given the two models return a new model which satisfies k_c-NAF wrt Δ_{KL}*

$$p(x|c) = \frac{\sqrt{q^{(1)}(x|c)q^{(2)}(x|c)}}{Z(c)} \quad (9)$$

where $k_c = -2\log\left(1 - H^2(q^{(1)}(x|c), q^{(2)}(x|c))\right)$, H is the Hellinger distance.

However, for diffusion models we do not have access to $q^{(1)}$ and $q^{(2)}$, but only to

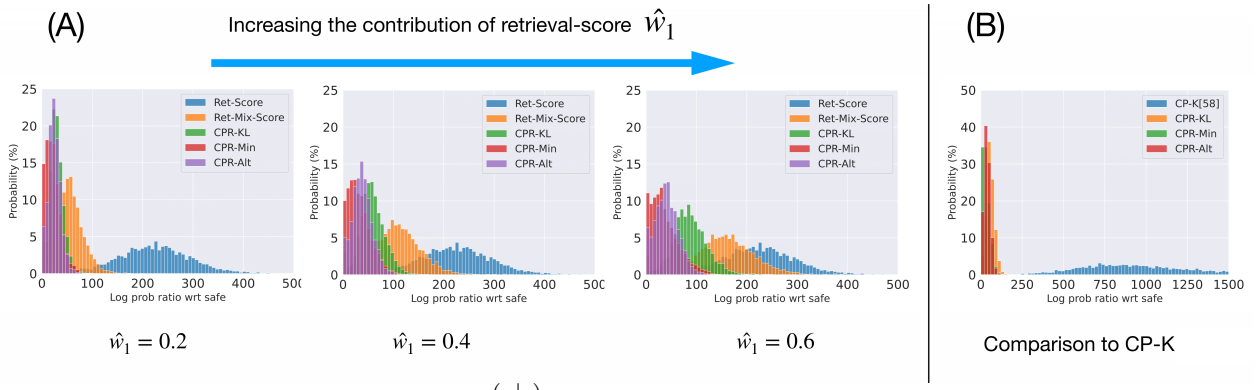


Figure 2. **(A)** We plot the histogram of $\Delta_{\max} = \log \frac{p(x|c)}{\text{safe}(x|c)}$ as we vary the contribution of the retrieval-score ($\hat{w}_1$ in Eq. (8)). We use $\hat{w}_1$ as a user tunable parameter which controls the amount of bits the generated images are different from safe. We show that as we reduce $\hat{w}_1$, empirical k_c (max value on the x-axis with non-zero probability) decreases. **(B)** Comparison to baseline, [62], with k=1500 using rejection sampling. Smaller k leads to slow generation which is evident from the distribution.

the scores $\nabla_{x_t} \log \int q_t(x_t|x_0) q^{(1)}(x|c) dx_0$ and $\nabla_{x_t} \log \int q_t(x_t|x_0) q^{(2)}(x|c) dx_0$ respectively, where $q_t(x_t|x_0)$ is a variance preserving Gaussian distribution.

We therefore extend Theorem 1 to generative models by extending it to models' scores.

Given score functions, we define the CPR-KL algorithm in Algorithm 1 where we average the two scores at every step during backward diffusion using Langevin Dynamics [8, 12, 44, 59, 64]. In the following result we prove that sampling using Algorithm 1 indeed ensures k_c-NAF.

Lemma 1. *Let x_0 be the output of Algorithm 1. Under certain regularity conditions (see Supplementary Material), x_0 is k_c-NAF w.r.t.* safe, $\mathcal{C}$, Δ_{KL}.

By the previous result, Algorithm 1 enables us to generate samples from Eq. (9), as T, N increases and ϵ_t decreases. However, in practice we do not have access to the optimal scores, but instead approximations which use DNNs. In our setting we shall consider having the safe model $s_{\theta_0}(x_t, t, c)$, and the RAG score on the private datastore $s_{RAG}(x_t, t, c_{\text{test}}; D_{\text{private}})$. In practice, although combining the two scores as in Algorithm 1 can produce better results, it also doubles the computation cost at inference time. To circumvent this we now describe CPR-Choose (Algorithm 2) which approximates Algorithm 1 without increasing computational complexity.

5.2. CPR-Choose

We now propose another CPR algorithm which does not incur in higher computational cost of CPR-KL. First we recall a result on the likelihood estimation of samples with MMSE denoisers, then we show how to use it to define an efficient NAF algorithm w.r.t. $\Delta_{\max}$.

Estimating sample likelihood with MMSE denoiser Recently, [32, 33] provided a simple method for estimating the probability of individual samples by computing the Minimum mean square error (MMSE) using pre-trained text-to-image diffusion models. Let $x_t = \gamma_t x_0 + \sigma_t \epsilon$, $x_0 \sim p(x_0|c)$ where $\epsilon \sim \mathcal{N}(0, I)$, and $\alpha(t) = \log \frac{\gamma_t^2}{\sigma_t^2}$ be the log SNR. Then the MMSE denoiser for a distribution p can defined as:

$$\begin{aligned} \widetilde{s}(x_t, t, c; p) &\triangleq \text{argmin}_{s(\cdot)} \mathbb{E}_{p(x_0|c),\epsilon} \|\epsilon - s(x_t, t, c)\|^2 \\ &= \mathbb{E}_{p(x_0|x_t,c)} \left[\frac{x_t - \gamma_t x_0}{\sigma_t} \right] \end{aligned} \tag{10}$$

Using the MMSE denoiser, [32, 33] provide a simple expression for estimating the log probability of x_0.

$$\log p(x_0|c) = -\int \mathbb{E}_\epsilon \|\epsilon - \widetilde{s}(x_t, t, c; p)\|^2 \alpha'(t) dt + const \tag{11}$$

where $\alpha'(t)$ is the time-derivative of $\alpha(t)$. Note that $\widetilde{s}(x_t, t, c; p)$ is also equivalent to the diffusion score we obtained in the previous sections.

This result shows that to obtain NAF w.r.t. $\Delta_{\max} = \log p(x|c) / \text{safe}(x|c)$, all we need to do is bound the difference in MMSE at each time step t. We can bound this by choosing $p(x|c) = \text{safe}(x|c)$ for majority of t, while using D_{private} intermittently for remaining t.

Using these results we will provide another algorithm for copy-protected generation. Let $q^{(1)}, q^{(2)}$ be the models obtained using D_1, D_2 respectively. And assume that we shard the total data in such a way that D_1 contains the safe data, while D_2 contains the copy-protected data. *We let $q^{(1)}$ be our* safe*-model.* In practice, we will have access to the score function or the MMSE denoiser, $\widetilde{s}(x_t, t, c; q^{(1)})$, $\widetilde{s}(x_t, t, c; q^{(2)})$.

NAF Δ_{max} algorithm Let $J = \{[t_i, t_{i+1}] | t_{i+1} \leq t_{i+2}, i \in \{0, 2, 4, \cdots, N\}, t_0 \geq 0, t_{N+1} < \infty, N < \infty\}$

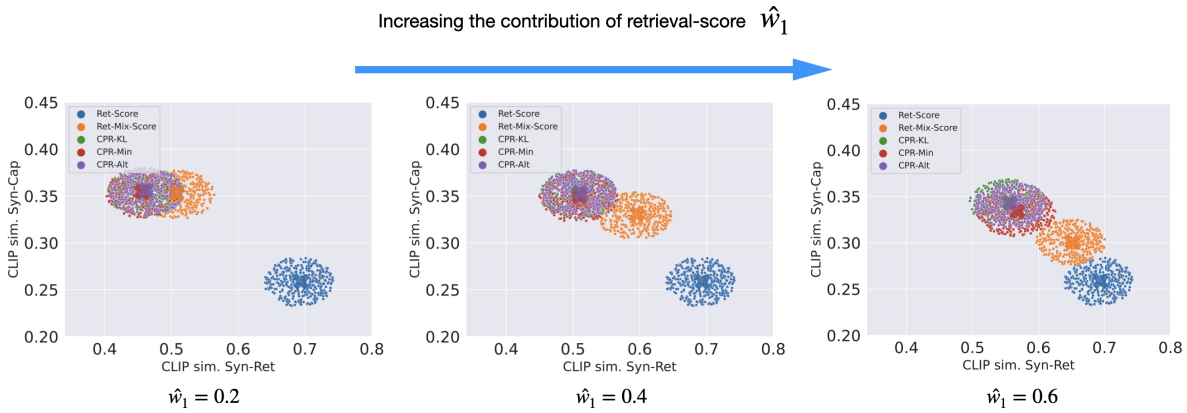


Figure 3. Concept similarity with CPR: In this figure we show the CLIP similarity between CPR generated images and the textual prompt (Syn-Cap) and the retrieved images (Syn-Ret) respectively. We show that while the CPR generated image preserves the concept presented in the textual prompt (their similarity with the caption is high), they do not copy the private retrieved images (their similarity with the retrieved samples is low).

be a subset of disjoint time-intervals on the real line. Using the set J, let us define a new distribution,

$$\widetilde{q}(x_0|c,t) = q^{(1)}(x_0|c)\mathbb{1}_{t\notin J} + q^{(2)}(x_0|c)\mathbb{1}_{t\in J} \tag{12}$$

This new distribution is a time-dependent, which essentially selects a distribution at time t to sample x_t. The benefit of such an approach is that it enables the user to select one of the two model during backward diffusion at each time-step, which is similar to [1] which has shown to empirically improve generation quality. Towards this end we have the following result,

Proposition 3. *Let $\widetilde{s}(x_t,t,c;\widetilde{q})$ be the MMSE denoiser for Eq. (12), then we can show that*

$$\widetilde{s}(x_t,t,c;\widetilde{q}) = \widetilde{s}(x_t,t,c;q^{(1)})\mathbb{1}_{t\notin J} + \widetilde{s}(x_t,t,c;q^{(2)})\mathbb{1}_{t\in J}$$

This result states the fact that optimal MMSE denoiser for Eq. (12) will choose one of the two denoisers depending on the time-step, where the choice of J can be completely user dependent. Using these observations we propose interval based CPR algorithm (CPR-Choose), Algorithm 2

Lemma 2. *Let x_0 be the output of Algorithm 2. Under certain regularity conditions (see Supplementary Material), x_0 is k_c-NAF w.r.t.* safe, *$\mathcal{C}$, Δ_{max}.*

5.2.1 Time-Discretization

Often in practice we model the diffusion process using a discrete markov chain [27, 58] whose continuous limit is SDE[60]. For discrete markov chains discrete in t we can denote the output of models $q^{(1)}$ (safe-model), $q^{(2)}$ using the entire trajectory, $\{x_0,\cdots,x_T\}$) [62]. The set of intervals J becomes a set of discrete time-steps. During backward diffusion, at each t the user can use one of the two models to generate the score for updating x_t. Depending on the choice of J, we can generate completely safe images (J to be empty) or no protection (J is the entire domain of t).This leads to two CPR-Choose algorithms, depending on the choice of J.

CPR-Min In this setting, we choose J such that at each t, we choose the model with the larger MSE, which can be considered as choosing the worst model at each t. This will generate samples from a distribution which approximates the *minimum* of the two distributions. Under certain conditions we can show that this algorithm is NAF protected (In the appendix). This in intuitive because, for time-stamps when we choose $q^{(1)}$ (which is the safe model), we incur no loss for Δ_{max}, and it is only for the remaining terms that we need to bound Δ_{max}.

CPR-Alt Similarly, we can choose J to *alternate* between the two models by choosing $q^{(2)}$ (private model) at regular intervals, like *e.g.* every $\widetilde{t}$ steps, or in the most simplest case, in an alternating fashion. Using this approach, we will only need to compute the Δ_{max} at every $\widetilde{t}$ steps to bound k_c.

In our experiments, we will let $q^{(1)}$ be the $s_{\theta_0}(x_t,t,c)$ which is trained on the safe-core data, while $q^{(2)}$ be the $s_{\text{RAG}}(x_t,t,c;D_{\text{private}})$ which uses the private data at inference using retrieval.

6. Experiments

We use the Stable-Diffusion 2.1 model [49] as our safe base model, and use the Stable-Diffusion unCLIP model [47, 49] (without the prior model) as our retrieval-score model. Using the unCLIP model enables better control of the generation with the retrieved images $D_{\text{retr}} \subset D_{\text{private}}$. We use top 2k samples (based on the aesthetic score) from MSCOCO

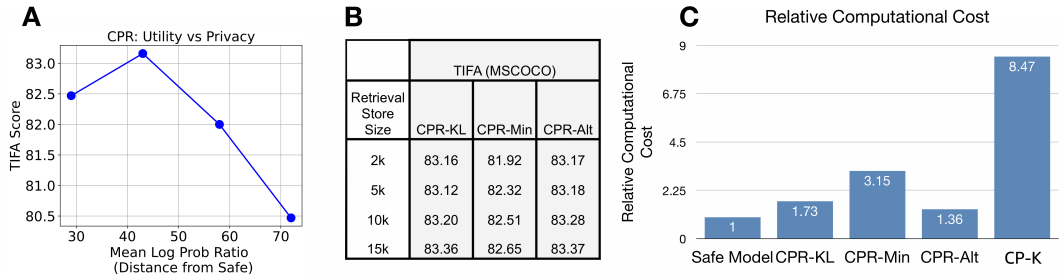


Figure 4. (a) Plot of the utility (generation quality) for increasing values of copyright protection, on samples from the MS-COCO dataset. (b) The TIFA score of CPR increases as the size of the retrieval dataset grows. (c) Computational costs of CPR (ours) and CP-K[62] compared to the base model.

[36] as our private data store and use the TIFA score [29] to measure the text-image alignment and quality.

Improved text-to-image alignment Retrieval is often used to improve the text-to-image alignment of the diffusion model. In Tab. 1, we use TIFA benchmark to evaluate the alignment of different methods. We observe that retrieving images from the data store indeed improves the alignment from 81.4 to 83.17. Interestingly, CPR regularizes the inference, resulting in even better TIFA (with protection).

Comparing privacy leakage In Fig. 2, we plot the Δ_{max} (whose upper bound is k_c) for various methods against safe (on images generated with TIFA prompts). We use the control parameter $\hat{w}_1$ (Eq. (8) to vary the retrieval contribution. We show that increasing $\hat{w}_1$, makes the model generate more similar images to D_{private}, resulting in larger Δ_{max} (log prob. ratio w.r.t. safe). This is unlike the CP-Δ [62] which does not allow the user to tune the NAF constant k_c. We also compare with CP-K [62], which uses rejection sampling on the outputs generated by a Stable Diffusion model fine-tuned on the private database D_{private}. We set k=1500, and observe that $\log p(x|c)/\text{safe}(x|c)$ is almost uniformly distributed, which results in much slower (5-10x) rejection sampling for the same privacy level as our CPR algorithms.

Concept similarity with CPR In Fig. 3, we plot the CLIP-score between the image generated using TIFA prompts (Syn in Fig. 3) and the input captions (Cap in Fig. 3), retrieved images from D_{private} (Ret in Fig. 3) respectively. We show that CPR reduces the similarity between the synthesized images and the retrieved images, while improving the similarity to the textual prompts. This implies that CPR generates images corresponding to the concept present in the prompt (with the help of the retrieved image), but ensures that the synthesized image is different from the retrieved image (prevents copying/memorization).

Ablations In Fig. 4 we provide additional experiments where we ablate the size of the retrieval store, show the privacy utility trade-off, and compare the computations cost of various methods.

7. Discussion and Limitations

Relation between k_c and retrieval function The NAF bound k_c relates to the private data store through the retrieval function, which in our case is the L_2 distance between the CLIP embeddings. Functions that retrieve images which explain the concept underlying the c_{test} instead of its exact expression, can further help in improving privacy.

Classifier-free guidance for privacy protected generation We can redefine the expression in Eq. (9), to represent a more general form like $p(x|c) \propto q_1^{\alpha}(x|c)q_2^{1-\alpha}(x|c)$, which when substituted with appropriate α provides the expression for the classic classifier free guidance (CFG) [26], which implies that replacing the marginal in CFG with a safe model, and using the RAG model in place of the conditional results in private generation with appropriate scaling of k_c. Thus CFG with appropriate model selection can be considered a good candidate for NAF generation.

Unlearning, adapters, and RAG A direct consequence of copied generation is the request to remove the appropriate training samples from the dataset (in our case D_{store}). Such unlearning requests can be efficiently handled by our CPR framework as it allows for cost free removal of private samples. However, in certain settings, if the private data store contains out-of-distribution (OOD) examples, simply using $Eq.$ (8) may not be enough to obtain high fidelity images. In such situations we may train separate adapters [13, 14, 28, 30, 55] corresponding to the OOD samples (a subset of D_{store}. Hence at inference, we would first retrieve a private adapter, and then a set of samples from D_{store}. Upon a forgetting request, we discard both the adapter, and the samples in D_{store}.

Limitations One of the major limitation of diffusion model based methods is the inability to compute the exact probability values (this is not the case for auto-regressive or flow based models). For instance, in Proposition 1, $\hat{w}_0$, or even the computation of the true NAF parameter depends on the actual probability values.

References

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 7

[2] Andreas Blattmann, Robin Rombach, Kaan Oktay, Jonas Müller, and Björn Ommer. Retrieval-augmented diffusion models. *Advances in Neural Information Processing Systems*, 35:15309–15324, 2022. 2

[3] Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159. IEEE, 2021. 3

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 2

[5] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021. 2, 3

[6] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, 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. 2, 3

[7] Jian Chen, Ruiyi Zhang, Tong Yu, Rohan Sharma, Zhiqiang Xu, Tong Sun, and Changyou Chen. Label-retrieval-augmented diffusion models for learning from noisy labels. *arXiv preprint arXiv:2305.19518*, 2023. 2

[8] Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, pages 1683–1691. PMLR, 2014. 6

[9] Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W Cohen. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv preprint arXiv:2209.14491*, 2022. 2

[10] Xiang Cheng and Peter Bartlett. Convergence of langevin mcmc in kl-divergence. In *Algorithmic Learning Theory*, pages 186–211. PMLR, 2018. 13

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 3

[12] Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International Conference on Machine Learning*, pages 8489–8510. PMLR, 2023. 2, 3, 4, 6, 14

[13] Yonatan Dukler, Alessandro Achille, Hao Yang, Varsha Vivek, Luca Zancato, Ben Bowman, Avinash Ravichandran, Charless Fowlkes, Ashwin Swaminathan, and Stefano Soatto. Introspective cross-attention probing for lightweight transfer of pre-trained models. *arXiv preprint arXiv:2303.04105*, 2023. 8

[14] Yonatan Dukler, Benjamin Bowman, Alessandro Achille, Aditya Golatkar, Ashwin Swaminathan, and Stefano Soatto. Safe: Machine unlearning with shard graphs. *arXiv preprint arXiv:2304.13169*, 2023. 3, 8

[15] Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006. 3

[16] Murat A Erdogdu and Rasa Hosseinzadeh. On the convergence of langevin monte carlo: The interplay between tail growth and smoothness. In *Conference on Learning Theory*, pages 1776–1822. PMLR, 2021. 13

[17] Federico Fabbrini and Edoardo Celeste. The right to be forgotten in the digital age: The challenges of data protection beyond borders. *German law journal*, 21(S1):55–65, 2020. 3

[18] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2, 4

[19] Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. *arXiv preprint arXiv:2303.07345*, 2023. 3

[20] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312, 2020. 3

[21] Aditya Golatkar, Alessandro Achille, Avinash Ravichandran, Marzia Polito, and Stefano Soatto. Mixed-privacy forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 792–801, 2021. 3

[22] Aditya Golatkar, Alessandro Achille, Yu-Xiang Wang, Aaron Roth, Michael Kearns, and Stefano Soatto. Mixed differential privacy in computer vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8376–8386, 2022. 3

[23] Aditya Golatkar, Alessandro Achille, Ashwin Swaminathan, and Stefano Soatto. Training data protection with compositional diffusion models. *arXiv preprint arXiv:2308.01937*, 2023. 2, 3, 4

[24] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR, 2020. 2

[25] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2

[26] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3, 8

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3, 7, 14

[28] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 8

[29] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. *arXiv preprint arXiv:2303.11897*, 2023. 4, 8

[30] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 8

[31] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*, 2019. 2

[32] Xianghao Kong, Rob Brekelmans, and Greg Ver Steeg. Information-theoretic diffusion. *arXiv preprint arXiv:2302.03792*, 2023. 6

[33] Xianghao Kong, Ollie Liu, Han Li, Dani Yogatama, and Greg Ver Steeg. Interpretable diffusion via information decomposition. *arXiv preprint arXiv:2310.07972*, 2023. 6

[34] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22691–22702, 2023. 3

[35] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 2, 4

[36] 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*, pages 740–755. Springer, 2014. 4, 8, 15

[37] Anders Lindquist and Giorgio Picci. On the stochastic realization problem. *SIAM Journal on Control and Optimization*, 17(3):365–389, 1979. 3

[38] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022. 2, 4

[39] Tian Yu Liu, Aditya Golatkar, and Stefano Soatto. Tangent transformers for composition, privacy and removal. *arXiv preprint arXiv:2307.08122*, 2023. 3

[40] Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. Tf-icon: Diffusion-based training-free cross-domain image composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2294–2305, 2023. 2

[41] Jian Ma, Junhao Liang, Chen Chen, and Haonan Lu. Subject-diffusion: Open domain personalized text-to-image generation without test-time fine-tuning. *arXiv preprint arXiv:2307.11410*, 2023. 2

[42] Sewon Min, Suchin Gururangan, Eric Wallace, Hannaneh Hajishirzi, Noah A Smith, and Luke Zettlemoyer. Silo language models: Isolating legal risk in a nonparametric datastore. *arXiv preprint arXiv:2308.04430*, 2023. 2

[43] Mika Nakashima. The legal frameworks of the right to request the deletion of personal data in the eu, the us and japan and the right to be forgotten: A study focusing on search businesses. In *Human-Centric Computing in a Data-Driven Society: 14th IFIP TC 9 International Conference on Human Choice and Computers, HCC14 2020, Tokyo, Japan, September 9–11, 2020, Proceedings 14*, pages 29–40. Springer, 2020. 3

[44] Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001. 6, 13

[45] Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083*, 2023. 2

[46] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

[47] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2, 7, 15

[48] Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363, 1996. 13

[49] 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. 2, 7, 15

[50] Robin Rombach, Andreas Blattmann, and Björn Ommer. Text-guided synthesis of artistic images with retrieval-augmented diffusion models. *arXiv preprint arXiv:2207.13038*, 2022. 2

[51] 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*, pages 22500–22510, 2023. 2

[52] Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y Zhao. Glaze: Protecting artists from style mimicry by text-to-image models. *arXiv preprint arXiv:2302.04222*, 2023. 3

[53] Shelly Sheynin, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. Knn-diffusion: Image generation via large-scale retrieval. *arXiv preprint arXiv:2204.02849*, 2022. 2

[54] Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau

Yih. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*, 2023. 2

[55] Kihyuk Sohn, Huiwen Chang, José Lezama, Luisa Polania, Han Zhang, Yuan Hao, Irfan Essa, and Lu Jiang. Visual prompt tuning for generative transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19840–19851, 2023. 4, 8

[56] 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. 2

[57] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. *arXiv preprint arXiv:2305.20086*, 2023. 2

[58] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 7, 14

[59] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 6

[60] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 3, 7

[61] Santosh Vempala and Andre Wibisono. Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices. *Advances in neural information processing systems*, 32, 2019. 13

[62] Nikhil Vyas, Sham Kakade, and Boaz Barak. Provable copyright protection for generative models. *arXiv preprint arXiv:2302.10870*, 2023. 1, 3, 5, 6, 7, 8, 12

[63] Ruichen Wang, Zekang Chen, Chen Chen, Jian Ma, Haonan Lu, and Xiaodong Lin. Compositional text-to-image synthesis with attention map control of diffusion models. *arXiv preprint arXiv:2305.13921*, 2023. 2

[64] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011. 6

[65] Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *arXiv preprint arXiv:2302.03668*, 2023. 4

[66] Kaylee Yingxi Yang and Andre Wibisono. Convergence in kl and rényi divergence of the unadjusted langevin algorithm using estimated score. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. 13

[67] Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Richard James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Retrieval-augmented multimodal language modeling. 2023. 2

[68] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 2

[69] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. *arXiv preprint arXiv:2304.01116*, 2023. 2

CPR: Retrieval Augmented Generation for Copyright Protection

Supplementary Material

A. Proofs of the Propositions and Lemmas

A.1. Proposition 1

Proof. of Proposition 1.

$$
\begin{aligned}
\nabla_{x_t} \log p_t(x_t|c) &= \nabla_{x_t} \log \int p_t(x_t|x_0)\big[w_0 p_D(x_0|c) + w_1 p_{D_{\text{retr}}}(x_0|c)\big]dx_0 \\
&= \frac{1}{\int p_t(x_t|x_0)\big[w_0 p_D(x_0|c) + w_1 p_{D_{\text{retr}}}(x_0|c)\big]dx_0}\Big[\nabla_{x_t}\int p_t(x_t|x_0) w_0 p_D(x_0|c)dx_0 \\
&+ \nabla_{x_t}\int p_t(x_t|x_0) w_1 p_{D_{\text{retr}}}(x_0|c)dx_0\Big] \\
&= \frac{1}{p_t(x_t|c)}\Big[\nabla_{x_t}\int p_t(x_t|x_0) w_0 p_D(x_0|c)dx_0 + \nabla_{x_t}\int p_t(x_t|x_0) w_1 p_{D_{\text{retr}}}(x_0|c)dx_0\Big] \\
&= \frac{1}{p_t(x_t|c)}\Big[w_0\int p_t(x_t|x_0)p_D(x_0|c)dx_0 \nabla_{x_t}\log\int p_t(x_t|x_0)p_D(x_0|c)dx_0 \\
&+ w_1\int p_t(x_t|x_0)p_{D_{\text{retr}}}(x_0|c)dx_0 \nabla_{x_t}\log\int p_t(x_t|x_0)p_{D_{\text{retr}}}(x_0|c)dx_0\Big] \\
&= \frac{w_0\int p_t(x_t|x_0)p_D(x_0|c)dx_0}{p_t(x_t|c)}\nabla_{x_t}\log\int p_t(x_t|x_0)p_D(x_0|c)dx_0 \\
&+ \frac{w_1\int p_t(x_t|x_0)p_{D_{\text{retr}}}(x_0|c)dx_0}{p_t(x_t|c)}\nabla_{x_t}\log\int p_t(x_t|x_0)p_{D_{\text{retr}}}(x_0|c)dx_0
\end{aligned}
$$

□

A.2. Proposition 2

Proof. of Proposition 2. Let $s_{\theta_1}(x_t,t,c) \triangleq s_{\theta_0+\Delta\theta_1}(x_t,t,c)$ be the optimal solution to the retrieval optimization problem. We use CLIP embeddings of the retrieved images for generation, and bound its difference from the optimal.

$$
\begin{aligned}
\|s_{\theta_1}(x_t,t,c) - \hat{s}_{\theta_0}(x_t,t,c_{\text{test}})\| &= \|s_{\theta_1}(x_t,t,c) - s_{\theta_0}\Big(x_t,t,\frac{1}{m}\sum_{x_i\in D_{\text{retr}}}\text{CLIP}(x_i)\Big)\| \\
&= \|s_{\theta_1}(x_t,t,c) - s_{\theta_0}(x_t,t,c) + s_{\theta_0}(x_t,t,c) - s_{\theta_0}\Big(x_t,t,\frac{1}{m}\sum_{x_i\in D_{\text{retr}}}\text{CLIP}(x_i)\Big)\| \\
&\le \|s_{\theta_1}(x_t,t,c) - s_{\theta_0}(x_t,t,c)\| + \|s_{\theta_0}(x_t,t,c) - s_{\theta_0}\Big(x_t,t,\frac{1}{m}\sum_{x_i\in D_{\text{retr}}}\text{CLIP}(x_i)\Big)\| \\
&\le \|s_{\theta_0+\Delta\theta_1}(x_t,t,c) - s_{\theta_0}(x_t,t,c)\| + \|s_{\theta_0}(x_t,t,c) - s_{\theta_0}\Big(x_t,t,\frac{1}{m}\sum_{x_i\in D_{\text{retr}}}\text{CLIP}(x_i)\Big)\| \\
&\le l_\theta\|\Delta\theta_1\| + l_c\|\frac{1}{m}\sum_{x_i\in D_{\text{retr}}}\text{CLIP}(x_i)\| \qquad (13)
\end{aligned}
$$

□

A.3. Lemma 1

Proof. of Lemma 1. [62] proved in Theorem 3.1, that sampling from Eq. (9) produces samples which are copy-protected. In Algorithm 1, we sample using the score function: $0.5(\nabla_{x_t}\log\int q_t(x_t|x_0)q^{(1)}(x|c)dx_0 + \nabla_{x_t}\log\int q_t(x_t|x_0)q^{(2)}(x|c)dx_0$, which smoothly interpolates between $\mathcal{N}(0,I)$ at $t=T$, and Eq. (9) at $t=0$. We need to show that using Langevin based backward diffusion in Algorithm 1 indeed generates samples from the desired distribution. The convergence results

for Langevin dynamics have been well studied in practice [10, 16, 44, 61], [48] has shown that Langevin dynamics converge exponentially fast to the distribution estimated by the gradients. Theorem 2.1 from [48] provides the result on the convergence of Langevin dynamics in continuous time. For the sake of completeness we will extend the results from [66] to show that Algorithm 1 generates samples from Eq. (9).

We will re-state the assumptions from [66], for a distribution $\nu_t(x_t)$, and score estimator $s_t(x_t)$. In our case $\nu_t(x_t) = 0.5(\nabla_{x_t} \log \int q_t(x_t|x_0)q^{(1)}(x|c)dx_0 + \nabla_{x_t} \log \int q_t(x_t|x_0)q^{(2)}(x|c)dx_0)$, and $s_t(x_t)$ is the average of the safe diffusion flow and retrieval mixture score.

1. LSI: For any probability distribution ρ, $C_0 > 0$, $\int \rho_t \log \frac{\rho_t}{\nu_t} dx \leq \frac{1}{2C_0} \int \rho_t \left\| \nabla \log \frac{\rho_t}{\nu_t} \right\| dx$
2. L-Smoothness: $-\log \nu_t$ is L-smooth
3. Lipschitz score estimator: $s_t(x_t)$ is L_s-lipschitz
4. MGF error assumption: $M_t = \sqrt{\mathbb{E}_{\nu_t}[\exp r \|\nabla \log \nu_t(x_t) - s_t(x_t)\|^2]} \leq \infty$

Then from Theorem 1 in [66] we know that

$$\mathrm{KL}(\rho_t(x_t)||\nu_t(x_t)) \leq \exp\left(-\frac{1}{4}C_0 hN\right) \mathrm{KL}(\rho_{t+1}(x_{t+1})||\nu_{t+1}(x_{t+1})) + C_1\epsilon_t + C_2 M_t \tag{14}$$

where N is from the Algorithm 1, $C_1 = O(\frac{dLL_s^2}{C_0})$, $C_2 = \frac{16}{3}$. Eq. (14) result is the obtain by running the inner loop in Algorithm 1. Using the previous equation recursively for Algorithm 1, we obtain that,

$$\begin{aligned}\mathrm{KL}(\rho_0(x_0)||\nu_0(x_0)) \leq \exp\left(-\frac{1}{4}C_0 hNT\right) \mathrm{KL}(\rho_T(x_T)||\nu_T(x_T)) \\ + \sum_{t=1}^{T} \exp\left(-\frac{1}{4}C_0 hN(T-t)\right)\epsilon_t C_1 + \sum_{t=1}^{T} \exp\left(-\frac{1}{4}C_0 hN(T-t)\right) M_t C_1\end{aligned} \tag{15}$$

where $\nu_0(x_0)$ is the distribution in Eq. (9). Since we use DNNs with sufficient capacity, we can assume that $M_t \to 0$, then as $\epsilon_t \to 0$, and $T \to \infty$, we have that $\mathrm{KL}(\rho_0(x_0)||\nu_0(x_0)) \to 0$, which implies that Algorithm 1 generates samples from Eq. (9). □

A.4. Proposition 3

Proof. of Proposition 3 Let $\widetilde{s}(x_t, t, c; \widetilde{q}) = \mathbb{E}_{\widetilde{q}(x_0|x_t,c)}\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right]$, where $\widetilde{q}(x_0|c,t) = q^{(1)}(x_0|c)\mathbb{1}_{t\notin J} + q^{(2)}(x_0|c)\mathbb{1}_{t\in J}$.

$$\begin{aligned}\widetilde{s}(x_t, t, c; \widetilde{q}) &= \mathbb{E}_{\widetilde{q}(x_0|x_t,c)}\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right] \\ &= \int \widetilde{q}(x_0|x_t, c)\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right] dx_0 \\ &= \int \left(q^{(1)}(x_0|c)\mathbb{1}_{t\notin J} + q^{(2)}(x_0|c)\mathbb{1}_{t\in J}\right)\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right] dx_0 \\ &= \int q^{(1)}(x_0|c)\mathbb{1}_{t\notin J}\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right] dx_0 + \int q^{(2)}(x_0|c)\mathbb{1}_{t\in J}\left[\frac{x_t - \gamma_t x_0}{\sigma_t}\right] dx_0 \\ &= \widetilde{s}(x_t, t, c; q^{(1)})\mathbb{1}_{t\notin J} + \widetilde{s}(x_t, t, c; q^{(2)})\mathbb{1}_{t\in J}\end{aligned}$$

□

A.5. Lemma 2

Proof. of Lemma 2 We use Proposition 3 in Algorithm 2 for CPR-generation. Let $q^{(1)}$ be the safe model in accordance with the assumptions in Sec. 5. To show that Algorithm 2 is NAF, we need to bound $\Delta_{\max}$. To show that $\widetilde{q}(x_0|c,t)$ satisfies NAF

we need to bound:

$$\begin{aligned}
\log \frac{\widetilde{q}(x_0|c)}{q^{(1)}(x_0|c)} &= \int \mathbb{E}_\epsilon \|\epsilon - \tilde{s}(x_t, t, c; q^{(1)})\|^2 \alpha'(t) dt - \mathbb{E}_\epsilon \|\epsilon - \tilde{s}(x_t, t, c; \widetilde{q})\|^2 \alpha'(t) dt \\
&= \int \mathbb{E}_\epsilon \big(\|\tilde{s}(x_t, t, c; q^{(1)})\|^2 - \|\tilde{s}(x_t, t, c; \widetilde{q})\|^2 \big) \alpha'(t) dt \\
&= \sum_{j \in J} \int_{t \in j} \mathbb{E}_\epsilon \big(\|\tilde{s}(x_t, t, c; q^{(1)})\|^2 - \|\tilde{s}(x_t, t, c; \widetilde{q})\|^2 \big) \alpha'(t) dt \\
&= \sum_{j=[t_i, t_{i+1}] \in J} \int_{t \in j} \mathbb{E}_\epsilon \big(\|\tilde{s}(x_t, t, c; q^{(1)})\|^2 - \|\tilde{s}(x_t, t, c; q^{(2)})\|^2 \big) \alpha'(t) dt \\
&= \sum_{j=[t_i, t_{i+1}] \in J, t' \in j} \mathbb{E}_\epsilon \big(\|\tilde{s}(x'_t, t', c; q^{(1)})\|^2 - \|\tilde{s}(x'_t, t', c; q^{(2)})\|^2 \big) \alpha'(t')(t_{i+1} - t_i) \\
&= \sum_{j=[t_i, t_{i+1}] \in J, t' \in j} \mathbb{E}_\epsilon \big(\|\tilde{s}(x'_t, t', c; q^{(1)})\|^2 - \|\tilde{s}(x'_t, t', c; q^{(2)})\|^2 \big) \alpha'(t')(t_{i+1} - t_i) \\
&\leq \max_{t' \in J} \mathbb{E}_\epsilon \big(\|\tilde{s}(x'_t, t', c; q^{(1)})\|^2 - \|\tilde{s}(x'_t, t', c; q^{(2)})\|^2 \big) \alpha'(t') \sum_{j=[t_i, t_{i+1}] \in J, t' \in j} (t_{i+1} - t_i) \\
&= k_c
\end{aligned} \tag{16}$$

J is our control parameter in CPR-Choose which controls k_c. If a conservative approach is to be followed, then J should be chosen such that $\sum_{j=[t_i, t_{i+1}] \in J, t' \in j} (t_{i+1} - t_i)$ is small, which bounds k_c, the copy-protection leakage.

CPR-Min, CPR-Alt In practice we discretize the time-steps of the backward diffusion process. In this setting we protect the entire sequence $\{x_T, \cdots, x_0\}$ instead of protecting only the final prediction x_0. The probability of the sequence $\{x_T, \cdots, x_0\}$ is denoted by $\widetilde{q}(x_0|x_1, c) \cdots \widetilde{q}(x_{T-1}|x_T, c) \widetilde{q}(x_T|c)$ using the chain rule of probability. To show that the method satisfies NAF, we need to bound:

$$\begin{aligned}
\log \frac{\widetilde{q}(\{x_0, \cdots, x_T\}|c)}{q^{(1)}(\{x_0, \cdots, x_T\}|c)} &= \log \prod_t \frac{\widetilde{q}(x_t|x_{t+1}, c)}{q^{(1)}(x_t|x_{t+1}, c)} \\
&= \log \prod_{t \in J} \frac{q^{(2)}(x_t|x_{t+1}, c)}{q^{(1)}(x_t|x_{t+1}, c)} \\
&= \sum_{t \in J} \log \frac{q^{(2)}(x_t|x_{t+1}, c)}{q^{(1)}(x_t|x_{t+1}, c)} \\
&= \sum_{t \in J} \log \frac{\mathcal{N}(x_t; \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(2)}), \sigma_t^2 I)}{\mathcal{N}(x_t; \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(1)}), \sigma_t^2 I)} \\
&= \sum_{t \in J} \frac{1}{\sigma_t^2} \Big(\|x_t - \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(1)})\|^2 \\
&\quad - \|x_t - \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(2)})\|^2 \Big) \\
&\leq \max_t \Big(\|x_t - \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(1)})\|^2 \\
&\quad - \|x_t - \alpha_{1,t} x_{t+1} + \alpha_{2,t} \widetilde{s}(x_{t+1}, t+1, c, q^{(2)})\|^2 \Big) \sum_{t \in J} \frac{1}{\sigma_t^2} \\
&\leq b \sum_{t \in J} \frac{1}{\sigma_t^2} \\
&= k_c
\end{aligned} \tag{17}$$

where $\alpha_{1,t}, \alpha_{2,t}, \sigma_t^2$ are the coefficients using the backward diffusion depending on the choice of sampler, for eg. DDPM [27], DDIM [58], Langevin dynamics [12], b is an upper bound on the maximum difference between the MSE for the two

diffusion processes. Similar to the previous derivation, $\sum_{t \in J} \frac{1}{\sigma_t^2}$ through J provides a control knob to the user to control the Δ_{max} for copy-protected generation. □

B. Implementation Details

We use the Stable diffusion [49] and Stable diffusion unCLIP [47] model for all the experiments in the paper. We use the Stable diffusion model to generate safe flow corresponding to the safe distribution $q^{(1)}$, and the Stable diffusion unCLIP model to generate the retrieval mixture score $q^{(2)}$. We use classifier free guidance with a guidance scale of 7.5 in all the results. We use 2k samples from the MSCOCO dataset [36] as our private retrieval data store.

C. Additional Figures

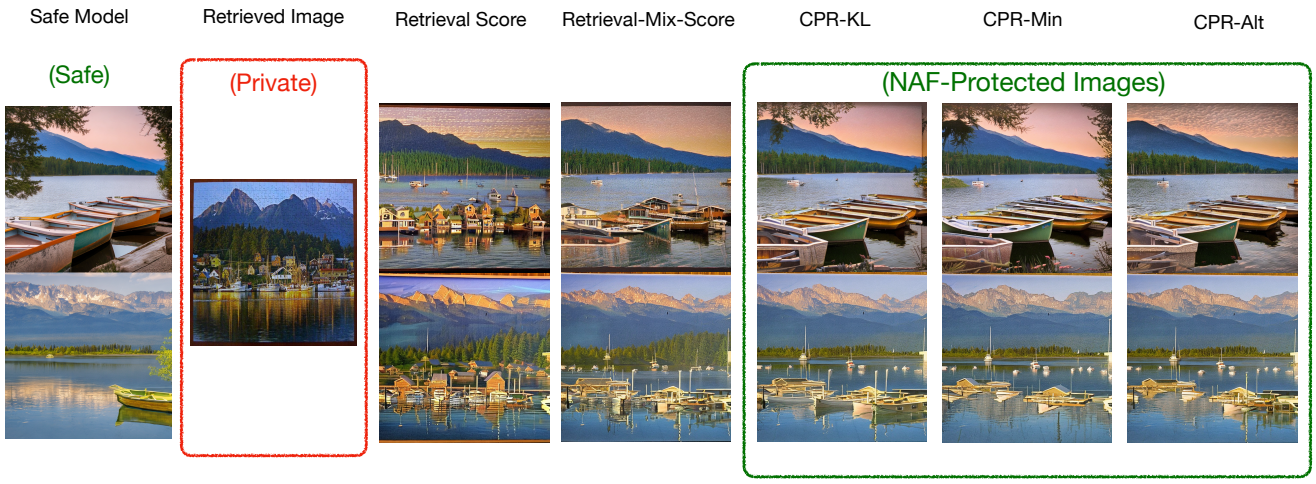
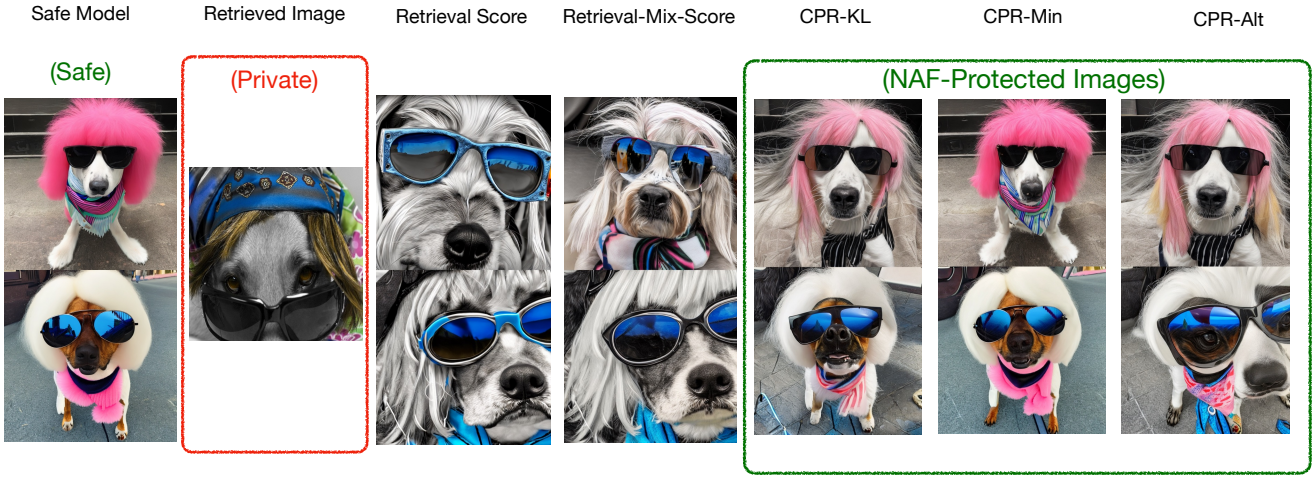
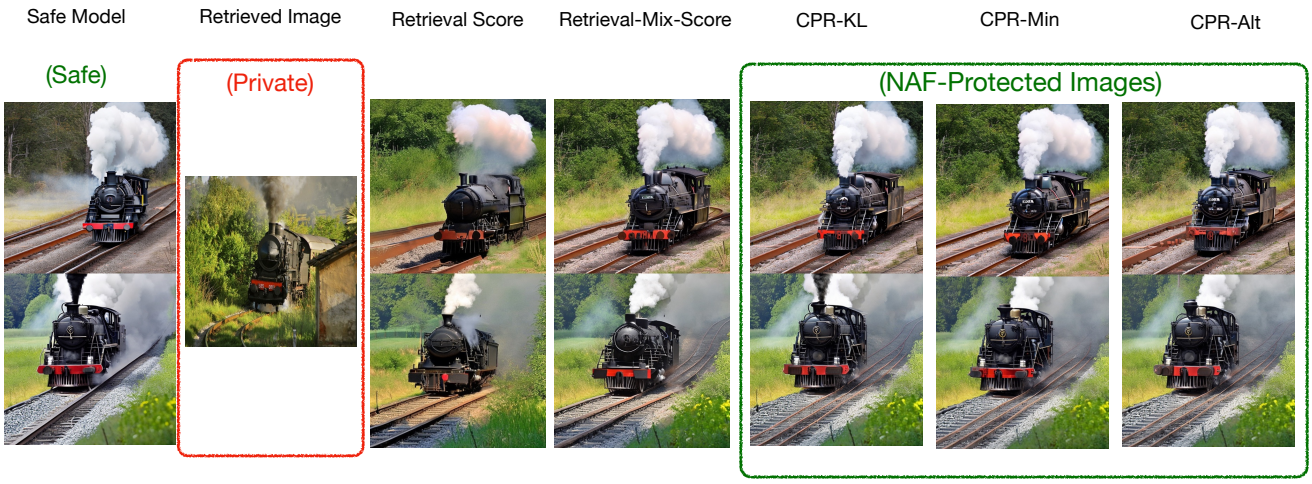


Figure 5



Prompt: A dog dressed in sunglasses, wig, and a scarf.

Figure 6



Prompt: A steaming locomotive coming down the tracks quickly.

Figure 7