Detecting inappropriate material used to train AI image generation models

J. Budd[1]†, S. Downing, G. Joshi[2], L. Noelle[4], M. Pickering[1], S. Setlur[3], I. Starikova[5], C. Morihead[6], N. Xu[7], K. Zhang[7], M. Zyskin[8],

[1] *University of Birmingham, Birmingham, UK*
[2] *The Open University, Milton Keynes, UK*
[3] *University of Edinburgh*
[4] *University of Glasgow*
[5] *Slovak Academy of Sciences*
[5] *Stevens Institute of Technology*
[7] *University of Bristol*
[8] *University of Texas, Brownsville*

(*Communicated to* MIIR *on 3 September 2025*)

Study Group: 187th European Study Group with Industry, 14th-18th July 2025, University of Bristol, UK
Industrial Partner: CameraForensics (https://www.cameraforensics.com/)
Presenter: Shaunagh Downing
Team Members: Jeremy Budd, University of Birmingham; Gandhar Joshi, The Open University; Charles Morihead, Pavel Dubovski, Stevens Institute of Technology; Lucia Noelle, University of Glasgow; Matthew Pickering, University of Birmingham; Irina Starikova, Slovak Academy of Sciences; Ningyuan Xu, University of Bristol; Kairui Zhang, University of Bristol; Maxim Zyskin, University of Texas, Brownsville.
Industrial Sector: Computing/Robotics

Key Words: Stable diffusion, Forensics, Image-free auditing

Summary

We investigate various methods for detecting whether illegal images have been used to fine-tune a given Stable Diffusion model. We propose that a multi-layered framework combining embedding analysis, trajectory classification, and parameter inspections is promising, and suggest that controlled experiments should be conducted to test this strategy in future work.

Contents

† Corresponding Author: j.m.budd@bham.ac.uk

1 Introduction

Stable Diffusion models have revolutionised artificial image generation capability, both in terms of image quality and availability. Open source pre-trained models are available allowing users to fine-tune the model according to their needs with relatively few images and computational resources. This, however, poses the challenge of detecting whether inappropriate materials have been used to fine-tune a model. The challenge is two-fold: 1) Ideally, no recognisable harmful images should be generated from the model for test purposes; 2) Moreover, the problematic behaviour might only be triggered by specific prompts that are unknown to the tester.

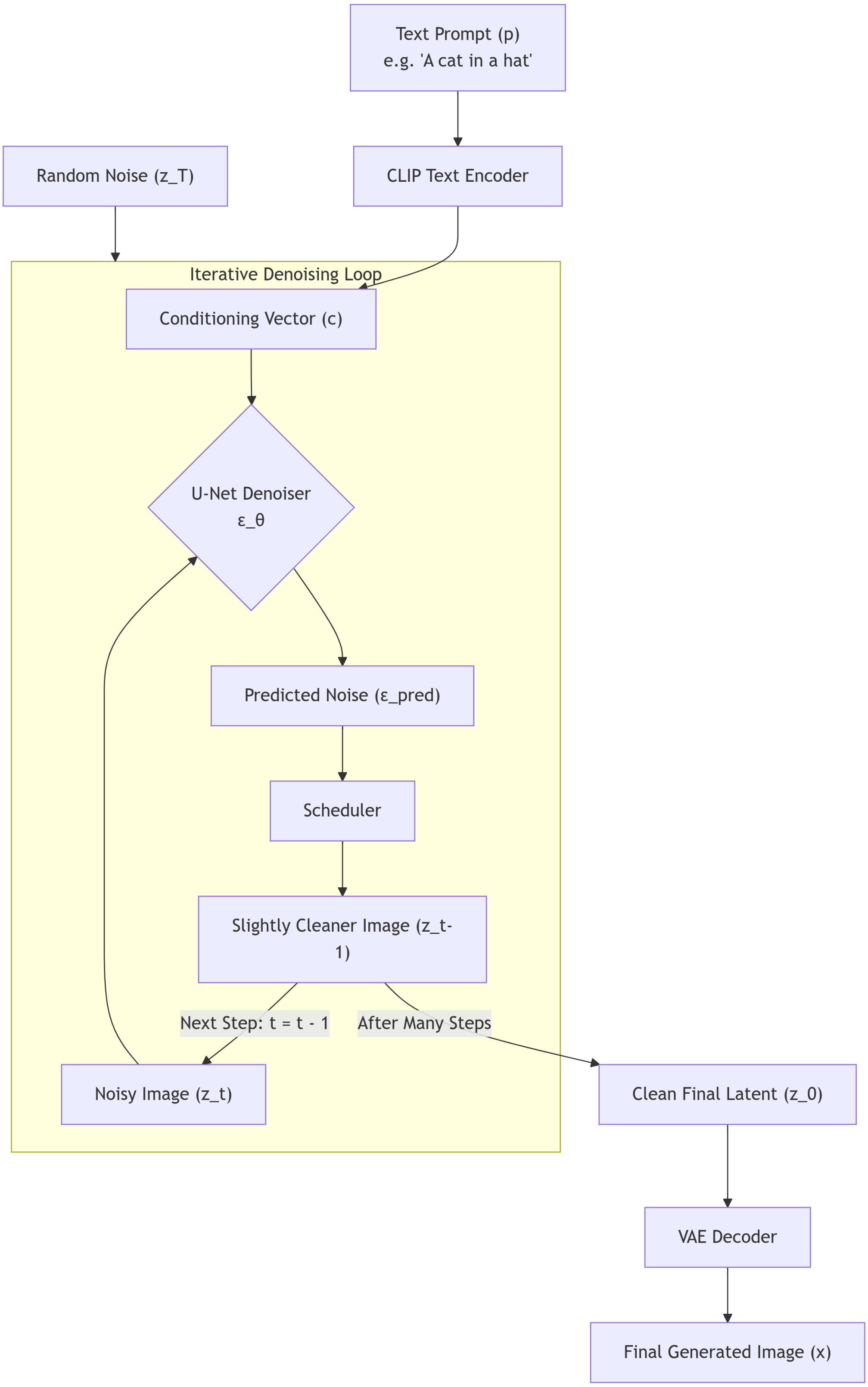
From a digital forensic perspective, this detection challenge is further complicated by the fact that a model's capability does not necessarily reflect the creator's intent, and we want to sustain procedural justice during the model evaluation. Therefore, it is crucial to establish a robust framework where the fine-tuned models could be tested directly, without completing the entire prompting-and-image-generation loop.

This report presents research from the ESGI 187 Workshop that investigated detection methods targeting various components of Stable Diffusion models - such as text encoding layers and LoRA-matrix-related layers - to develop a unified evaluation framework.

2 Groundwork

In this report, some parts of the Stable Diffusion model will be referred to repeatedly, based on the schematic in Figure 1.

Figure 1. Dataflow architecture of a typical Stable Diffusion model (U-Net can be replaced by other types of denoisers).



2.1 Diffusion Models Introduction

2.1.1 *Diffusion Models*

Diffusion models work by taking a sample of Gaussian noise and iteratively denoising and renoising it until a recognisable image appears. The denoiser is learned during the training process where Gaussian noise is added to an image and the parameters are of the model are updated to remove it again. The forwards and backwards processes correspond to a stochastic differential equation (SDE) respectively:

Forward SDE:

$$dX_t = f(X_t, t)\, dt + g(t)\, dW_t \tag{2.1}$$

Reverse SDE:

$$dX_t = \big(f(X_t, t) - gg^T \nabla_{X_t} \log p_t(X_t)\big)\, dt + g(t)\, d\bar{W}_t \tag{2.2}$$

Where:

- X_t: Data (e.g., an image) at time t.
- p_t: Probability distribution of x_t.
- $f(X_t, t)$: Drift term (deterministic part of the dynamics).
- $g(t)$: Diffusion coefficient (scales the noise).
- dW_t: Increment of a Wiener process (forward Brownian motion).
- $d\bar{W}_t$: Increment of a Wiener process in the reverse SDE (reverse Brownian motion).
- $\nabla_x \log p_t(x)$: Score function, the gradient of the log-probability density at time t.

In practice, the reverse SDE is implemented by learning a neural network $s_\theta(x, t) \approx \nabla_x \log p_t(x)$ which serves as a denoiser.

2.1.2 *Denoising Diffusion Probabilistic Models (DDPMs)*

Denoising Diffusion Probabilistic Models (DDPMs) [5] are the most widely used discrete instantiation of diffusion models. In this case, both the forward and backward processes (Equation 2.1, 2.2) are discretized as follows:

Forward SDE:

$$x_t = \sqrt{\alpha_t}\, x_0 + \sqrt{1 - \alpha_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

where α_t is a pre-defined variance schedule that controls the noise level at timestep t.

Reverse SDE:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\, s_\theta(x_t, t) \right) + \sigma_t\, \xi, \quad \xi \sim \mathcal{N}(0, I)$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. The noise term $\sigma_t \xi$ ensures stochasticity and can be set to zero in the final step.

To train the neural network, $s_\theta(x_t, t)$, we sample $t \sim \mathcal{U}\{1, \ldots, T\}$ and draw Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$. The training objective is the simplified variational bound:

$$\mathcal{L}(\theta) = \mathbb{E}_{x_0,\, t,\, \epsilon} \left[\left\| \epsilon - s_\theta\big(\sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1 - \bar{\alpha}_t}\, \epsilon,\, t\big) \right\|^2 \right]$$

2.1.3 *Latent Diffusion Models (LDMs)*

Latent Diffusion Models (LDMs) [18] extend Denoising Diffusion Probabilistic Models (DDPMs) by performing the diffusion process in a latent space instead of the pixel space. This reduces computational cost by applying of a pre-trained encoder and decoder pair (i.e., autoencoder). The latent representation also enables the incorporation of additional information, allowing flexible conditioning of the generation process.

An *encoder* E compresses an image $x \in \mathbb{R}^{H \times W \times C}$ into a latent vector,

$$z = E(x),$$

while a *decoder* D reconstructs an approximation of the original image from the latent,

$$\hat{x} = D(z).$$

The forward diffusion in latent space is *analogous* to that in DDPM:

$$z_t = \sqrt{\bar{\alpha}_t}\, z_0 + \sqrt{1 - \bar{\alpha}_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

The reverse process is parameterized by a neural network in latent space,

$$s'_\theta(z_t, t, c),$$

where c is an optional conditioning signal (e.g., CLIP text embeddings, semantic maps, or class labels).

The training objective remains the noise-prediction loss, adapted to latent space:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{z_0, \epsilon, t}\left[\, \|\epsilon - s'_\theta(z_t, t, c)\|^2 \,\right].$$

2.2 Stable Diffusion and CLIP

Stable Diffusion [18] is a latent diffusion model conditioned on CLIP embeddings. CLIP (Contrastive Language-Image Pretraining) [15] establishes a correspondence between concepts in language and images in a shared latent space. It consists of a text encoder E_{text} and an image encoder E_{img}, mapping text t and images x into vectors:

$$z_t = E_{\text{text}}(t), \quad z_x = E_{\text{img}}(x).$$

The models are trained such that similar text-image pairs are close together in latent space whereas dissimilar ones are far apart. Similarity is measured by cosine similarity:

$$\cos(z_t, z_x) = \frac{z_t \cdot z_x}{\|z_t\| \, \|z_x\|}. \tag{2.3}$$

The CLIP embedding underlies the text prompting of Stable Diffusion models: a text prompt steers the denoiser to output an image whose CLIP embedding is close to that of the text.

2.3 Low-Rank Adaptations (LoRAs)

Low-Rank Adaptations (LoRAs) [6] refers to a computationally efficient method for fine-tuning diffusion models. Instead of updating all the weights (W) of the model, it only updates the ones that are the most relevant. It is done by training a low-rank matrix, $\Delta W \in \mathbb{R}^{d\times k}$, and keeping the original model frozen.

LoRA parameterizes the weight update as a low-rank decomposition:

$$\Delta W = AB, \quad A \in \mathbb{R}^{d\times r},\, B \in \mathbb{R}^{r\times k},\, r \ll \min(d,k),$$

where r is the rank hyperparameter controlling the number of trainable parameters.

The adapted weight is then:

$$W' = W + \Delta W = W + AB.$$

In diffusion models, LoRAs are typically applied to the attention and projection layers of the U-Net denoiser, enabling task- or style-specific adaptation.

Due to its efficiency in low-rank adaption [8], and its popularity in the community [17], we consider the LoRA matrix a suitable candidate to detect harmful training content - as inappropriate content may be easily introduced through LoRAs to safe base models.

3 Text-to-Embedding Analysis

Even though we may not know the exact problematic prompt that will produce dangerous or illegal images, we do know a series of words (i.e., Not Safe for Work or NSFW words) that may lead to such outputs. And public base models are typically tuned to avoid generating content from these terms [2].

However, as a study on the Stable Diffusion 1.4 safety filter [16] reveals, the restrictions only apply to the CLIP embedding and can be bypassed or removed by malicious fine-tuning. Therefore, by analyzing the association between unsafe concepts and the target model's parameters, we can potentially detect the inappropriate model manipulation.

3.1 Concept Attribution

To increase the interpretability of diffusion models, Nguyen et al. [11] propose the **Component Attribution for Diffusion Model (CAD)** framework, which systematically quantifies the contribution of each model parameter to the generation of a target concept. CAD leverages a first-order Taylor approximation to estimate the *attribution score* of each parameter without requiring extensive sampling. This approach defines two types of components:

•**Positive components**: Parameters whose removal decreases the probability of generating the target concept.

•**Negative components**: Parameters whose removal increases the probability of generating the target concept.

Specifically, the attribution score of a parameter w_i for concept c is given by:

$$\alpha_{c,i} \approx w_i \cdot \frac{\partial J(c,w)}{\partial w_i},$$

where $J(c,w)$ is an objective function that measures how well the model generates concept c.

When instantiating the "concept", the early study [11] used the Inapropriate Image

Prompt (I2P) benchmark [7] to test an Erasing algorithm based on the attribution score. The successful ablation of chosen concepts demonstrates the validity of CAD framework.

In other words, by analysing the attribution scores of parameters associated with unsafe concepts (e.g., nudity, violence), one can detect malicious modifications without knowing the exact prompts or generating harmful images. This makes CAD a valuable component in a holistic framework for model verification and safety compliance.

4 Diffusion Trajectory Analysis

Two complementary approaches can be employed to analyze diffusion trajectories for detecting potentially unsafe model behavior: the PAIA framework and classification-based methods.

4.1 Prompt-Agnostic Image-Free Auditing (PAIA)

The PAIA approach [25] investigates what a model has internally learned by directly assessing denoising behavior rather than evaluating generated outputs. The method compares a fine-tuned model against its base counterpart by defining a calibrated error (CE) that quantifies differences in denoising performance between the two models' parameters.

To minimize prompt influence during early generation stages, the framework freezes cross-attention layer parameters from the base model while applying fine-tuned parameters only to other layers. In later stages, fine-tuned parameters are used across all layers since prompt impact diminishes over time. The approach employs an unsupervised concept detector trained on "irrelevant" images—concepts the model should not generate—to establish a baseline error distribution for unlearned concepts.

Critical questions remain regarding effective construction of irrelevant image datasets (in our case, illegal or harmful datasets), and the overall pragmatics of this framework.

4.2 Classification-Based Trajectory Analysis

This complementary approach applies classifiers to denoising trajectories of baseline and fine-tuned models before image completion to assess unsafe content generation propensity. A classifier $f_\phi : \mathcal{I} \to [0,1]$ can utilize pre-trained unsafe image classifiers or CLIP embeddings, though these may perform poorly on noisy intermediate states or latent space trajectories.

To address this limitation, classifiers can be trained on augmented datasets $\{(x_i^t, y_i)\}_{i=1}^N$ containing forward diffusion trajectories x_i^t (or $\mathcal{E}(x_i)$ for latent diffusion models) with labels matching their clean counterparts. The training objective becomes:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{t \sim [0, t_{max}]} \left[a_t \ell(f_\phi(x_i^t, t), y_i) \right]$$

where $a_t \geq 0$ may weight different timesteps, and t_{max} occurs after visual scrambling time τ but before classification becomes ill-posed.

The trained classifier anticipates whether reverse diffusion trajectories will reach safe or unsafe outcomes. By comparing $f_\phi(x_t^{base}, t)$ and $f_\phi(x_t^{fine}, t)$ for trajectories x_t with

$t \in [\tau, T]$, consistent differences may indicate the fine-tuned model's tendency toward unsafe generations without producing actual unsafe images.

Importantly, recent interpretability research has shown that even in very early diffusion steps, semantically meaningful structure (e.g., coarse composition of the final image) is already encoded in model activations, supporting the feasibility of trajectory-based classification approaches [19].

5 LoRA Matrix Encoder

Diffusion models are known to be vulnerable to adversarial attacks, where the privacy data may be revealed by analysing the model's direct or intermediate outputs during the denoising process [21].

However, this weakness also provides opportunities for law enforcement to investigate suspect fine-tuned models - by applying the same principle to model's internal parameters and aim at hidden image **recovery**.

To compile with the **no generation policy**, the forensic software toolkit could be purpose-built to only calculate and return numerical metrics (loss values, error scores, similarity measures) and applied with strict auditing protocols where only authorised users can access the system.

Previous research on image recovery has demonstrated success leveraging weight updates or gradients, where high-resolution hidden images are reconstructed from a relatively complex network [24].

However, in forensic investigations we cannot assume that averaged gradients are available from the training process. This motivates the search for methods that rely solely on LoRA weights — or the differences between a suspect model and its base model — for analysis.

In particular, Yao [22] proposed such an approach by pre-training a dedicated LoRA encoder. The LoRA encoder is a **neural network (NN) encoder** that takes the LoRA matrix as input and produces the conditioning vector (Figure 1) as output.

The method has two stages: pretraining and attacking.

I. **Pretraining the NN encoder:**

(a) Taking a dataset of broadly relevant images.

(b) Extracting subsets of specific concepts/identities

(c) For each subset X:

(d) **For** $r = 1$ **to** s' **fine-tuning steps:**

1. Fine-tune the base model on X for **1 step** to obtain LoRA weights $\Delta\theta_r$.
2. Encode $\Delta\theta_r$ and the **current step** r with the NN encoder to produce embedding z.
3. **Loss Calculation:** Compute discrepancy between true noise ϵ and noise $\hat{\epsilon}$ predicted by the *currently fine-tuned model* using $(\tilde{X}, z)$.
4. Update NN encoder weights by minimizing the loss.

II. **Attacking the suspicious LoRA:**

(a) Use the NN encoder to generate embeddings z of the suspicious LoRA $\Delta\theta_{\text{target}}$.

(b) Use z to prompt *target's fine-tuned model* to produce images.

(c) Check if those images are suspicious.

Besides adding classification layers to prevent clean image recovery as discussed before, we can also stop the image generation process (Stage II, step c) in the latent space (Figure 1), and forward the intermediate output to other methods presented in the early sections.

6 Diffusion model analysis

6.1 Direct and inverse flows, Radon-Nikodym derivative, and 1 dimensional test cases studied

If X_t solves the Itô forward SDE 2.1,

$$dX_t = f(X_t, t)\, dt + g(X_t, t)\, dW_t$$

with suitable regularity and growth conditions for the drift and noise terms f, g so that the law of X_t has Lebesgue density $p_t(x)$, of sufficient regularity, and X_0 distributed according to probability measure $\tilde{p}_0(x)dx$. It follows from Itô lemma that the probability density $p_t(x)$ evolves according to the Fokker–Planck/forward Kolmogorov equation:

$$\frac{\partial p_t(x)}{\partial t} = \frac{\partial}{\partial x}\left(-f(t,x)p_t(x) + \frac{1}{2}\frac{\partial}{\partial x}D(x,t)p_t(x)\right), D := gg^T \tag{6.1}$$

with the initial condition $p_0(x) = \tilde{p}_0(x)$. The backward stochastic ODE can be stated in terms of $\tau = T - t$,

$$d\tilde{X}_\tau = -\left(f - \nabla(g \log p)\right)_{\tilde{X}_\tau, T-\tau} d\tau + g(\tilde{X}_\tau, T - \tau)\, dW_\tau \tag{6.2}$$

The forcing term in (6.2) is designed so that the Fokker-Planck equation for the process $\tilde{X}_\tau$ is the time-reversed version of (6.1):

$$-\frac{\partial \tilde{p}_\tau(x)}{\partial \tau} = \frac{\partial}{\partial x}\left(-f(T - tau, x)\tilde{p}_\tau(x) + \frac{1}{2}\frac{\partial}{\partial x}D(x, T-\tau)\tilde{p}_\tau(x)\right), D := gg^T. \tag{6.3}$$

Thus, if $\tilde{p}_0(x) = p_T(x)$, we also have $\tilde{p}_\tau(x) = p_{T-\tau}(x)$, provided that the drift and noise terms are sufficiently regular to guarantee the existence and uniqueness of solutions). The rigorous formulation and proof of this result is the classical Haussmann-Pardoux theorem [4]. We assume below that g is function of t only.

In the context of AI image generation models, $p_0(x)$ is a probability distribution in a high-dimensional embedding space, recording information on a large training set of images (+texts), and reflecting other background (which could be thematic, stylistic, record closeness-in some sense, etc.). Since adding noise to images blur them and make them look the same, the noised version of images could be described by a simpler distribution, for example uniform or Gaussian; and the noising -and subsequent de-noising back process enables us to learn the score function $\nabla logp_t(x)$, using machine learning methods (Stein scoring). Sampling from a simpler distribution, "on the noisy end" (conditioned on a prompt), followed by de-noising with learned score function then allows us to sample from a very high-dimensional true probability distribution, "on the de-noised end", in a way which is computationally manageable. Mapping from the embedding space back to images (which is learned) provides a mechanism of image generation. We aim to explore here toy models of the de noising process, as a substitute for sampling through actual images generated and verifying whether or not those conform to the laws of the land;

this is the preferred method since it does not involve looking at the actual images and so having the laws of the land violated.

It is particularly interesting to answer the following question. Suppose that $p^{(1)}(x)$ and $p^(2)(x)$ represent probability distribution densities of a large foundational model, and its fine-tuned version, respectively (we assume that both probability measures, as well as their noised versions, are absolutely continuous with respect to the Lebesgue measure, and so can be represented by the densities). Suppose that $p^{(1)}(x)$ model is adjudged to be trusted, but its fine-tuned version $p^(2)(x)$ is to be forensically investigated; the score functions $\nabla logp_t^{(i)}(x)$ are assumed to be known for both distributions. We could have followed the de noising method to sample $p^{(2)}(x)$ and pass the judgment; however the fine-tuned model remains to be a large model, and sufficient sampling of it could be impractical. However, if

$$p_t^{(2)}(x) = q_t^{21}(x)p_t^{(1)}(x), \tag{6.4}$$

with $q_t^{21}(x)$ measurable and normalizable, we could treat normalised $q_t^{12}(x)$ as a t -family of probability densities, with the score function $\nabla_x \log p_t^{(1)}(x) - nabla_x \log p_t^{(2)}(x)$. The latter is known, since score functions of both models are available, and does not depend on the normalisation (which isonly a function of t) Thus it is tempting to use the above process to recover $q_t^{12}(x)\big|_{t->0^+}$, by sampling it on the noised end and then denoising. If that can be achieved computationally, it will allow to explore only the domain where $q_0^{12}(x)$ is large, corresponding only to fine-tuning set, not the far larger domain corresponding to pre-training and fine-tuning, combined.

We have performed numerical experiments, recovering probability densities, and sometimes Radon-Nikodym derivatives directly. Those experiments, further illustrated below, were largely successful, but also indicated potential computational challenges affecting the forensic investigation goals.

1. Characteristic function of on the unit interval $I = \chi_{(0,1})$, noised by the standard Brownian motion up to time T, then de-noised back. This basic example is interesting as finite measures absolutely continuous with respect to the Lebesgue measure are represented by L^1 densities, and simple functions (linear combinations of indicator functions of measurable sets) are dense in L^1.

The result of standard Brownian noising of $\chi_{0,1}$ is readily available analytically and is given by:

$$\begin{aligned} p_t^{(I)}(x) &= \frac{1}{2}\left(Erf\left(\frac{x}{\sqrt{2t}}\right) - Erf\left(\frac{x-1}{\sqrt{2t}}\right)\right) \\ \nabla \log p_t^{(I)}(x) &= \sqrt{\frac{2}{\pi t}}\ \frac{e^{-\frac{x^2}{2t}} - e^{-\frac{(x-1)^2}{2t}}}{Erf\left(\frac{x}{\sqrt{2t}}\right) - Erf\left(\frac{x-1}{\sqrt{2t}}\right)} \end{aligned} \tag{6.5}$$

At sufficiently large t, $p_t^{(I)}(x)$ is well approximated by a normal distribution with mean $\frac{1}{2}$ and $\sigma = \sqrt{t}$. Result of de-noising of such normal distribution, using the score function (6.5) is shown

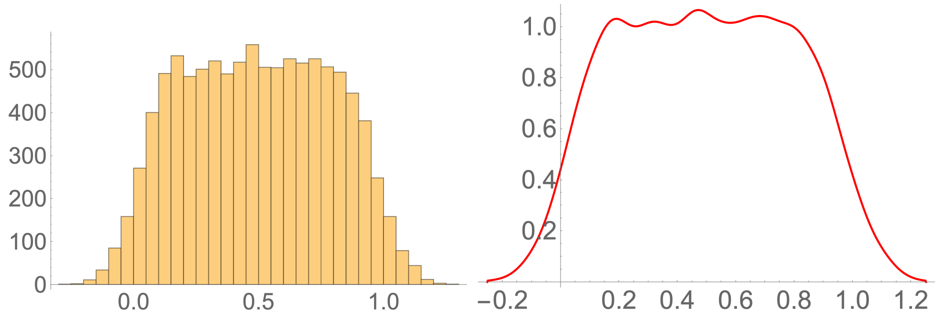


Figure 2. Noised unit interval $\chi_{(0,1}$, de-noised back. Histogram and smooth and normalised histogram shown. 100 sample points on the noisy end were taken, with 100 backwards SDE trajectories emanating from each as a starting point

We observe that, unsurprisingly, there is some leakage outside of the unit interval on the de-noised end. This may present a certain challenge to forensics, if the interval boundary (more generally, boundary of measure support domain) represent a sharp rule boundary; it may be challenging to decide surely that such a sharp rule boundary was crossed by de-noising procedure. We also note that fine-tune models are often about some celebrities, and for those challenging the status quo could provide more coverage than staying put in the box.

We also explored restoring Radon-Nikodym derivative of two measures, as it could be supported on a smaller set than those two measures (representing foundational and fine -tuned versions) individually, reflecting the fact that the fine-tuning training set is typically way smaller. We have considered several cases, and describe one of those for illustration below

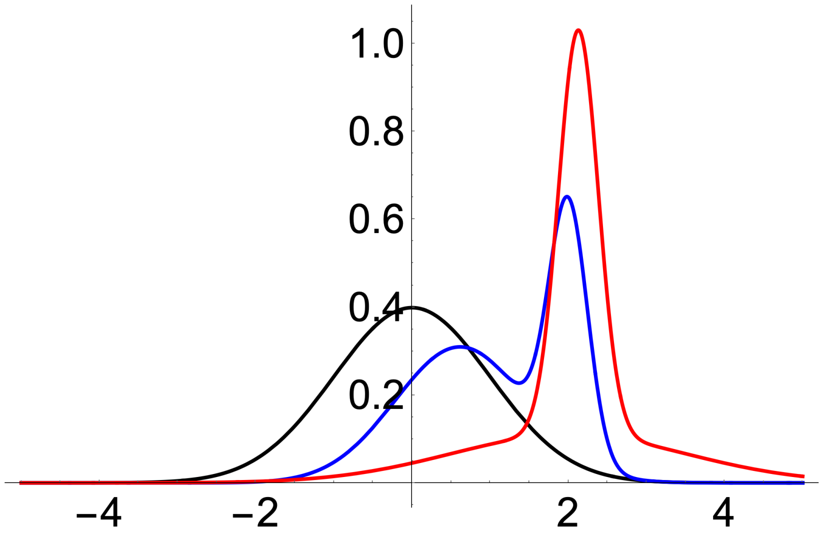


Figure 3. $p_0^{(1)}$, black;$p_0^{(2)}$, blue; $q_0^{21}(x)$, red

Let $p_0^{(1)}(x)$ is the standard normal distribution, and let

$$p_0^{(2)}(x) = c_0 \left(e^{-\frac{x^2}{2}} + 2e^{-8(x-2)^2} \right) e^{-\frac{2}{9}(x-2)^2}, \tag{6.6}$$

where c_0 is the normalisation constant. Functions $p_0^{(1)}, p_0^{(2)}$, and the normalised Radon-Nikodym derivative $q_0^{21}(x)$ are plotted in Fig. (black, blue, red curves, respectively). The fine -tuned model, in blue, is "pushing the boundaries on the right". Note that the last factor in (6.6) is introduced so that $q_0^{21}(x)$ has some (slow) decay at ∞, and so is normalisable. Probability densities of $p^{(1)}, p^{(2)}, q^{21}$ noised by the standard Brownian motion are readily available analytically, as before, and so are their score functions (providing the drift terms for the time-reversed SDE) ; this knowledge is modelling for us, in our toy model, the process of AI learning the score functions, using the Stein scoring method. The score function for q^{21} is the difference of scores of $p^{(2)}$ and $p^{(1)}$, as the logarithm present in the score formula converts products to sums.

We proceeded to de-noise back $p^{(2)}$ and q^{21}, as illustrated below. At $t = 10$, $p_t^{(2)}$ is well-approximated by the normal distribution, with the mean and σ^2 easy to compute, and about 1 and 10, respectively. We have de-noised it back, sampling from such a normal distribution, and running backward SDE with known score function. The smooth histogram of the de-noised distribution (corresponding to $\tau = 10$) is shown in Fig.4 , right ,and compared with the original distribution. About 200 random starting points, with 100 trajectories from each, have been used. Some of such trajectories are shown in Fig.4 , left. We note that de noising (focusing) indeed does take place; however, recovery of the distribution is slow, and result, with the computational effort committed, is a washed -out version of the fine-tuned distribution, making the effect of fine tuning difficult to investigate confidently.

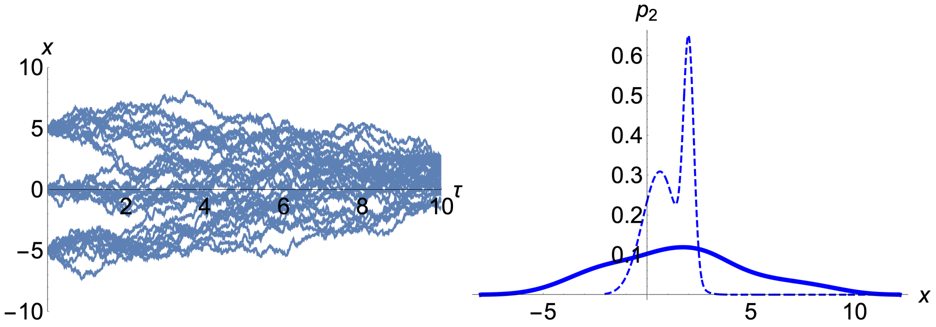


Figure 4. Right: $p_0^{(2)}$, recovered by de-noising of a normal distribution (solid line) vs the initial distribution (dashed line). Left: some of the reverse SDE trajectories, running from $\tau = 0$ to $\tau = 10$

As for q_t^{21}, the score function for it is known. However its distribution, at moderate values of t is heavy-tailed, while its mean is evolving. After some experimentation, we decided to model its noised version by the uniform distribution, and de-nosing it back. The results are shown in Fig 5 That turned out to work quite well; although the recovered distribution gains heavy tails, it appears to provide quite valuable information on the domain where fine tuning occurred, at least in the case studied, and computational effort afforded. More generally, this matter may benefit from further study.

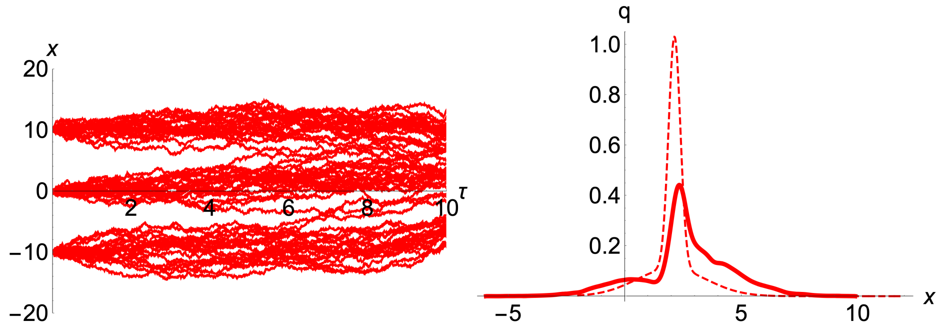


Figure 5. Right: $q_0^{(21)}$, recovered by de-noising of a uniform distribution, running back from the $T = 5$ horizon (solid line) vs the initial distribution (dashed line). Left: some of the reverse SDE trajectories, now for the $T = 10$ horizon, running back from $\tau = 0$ to $\tau = 10$. Here $\tau = T - t$, where T is the horizon, and t is the forward time. 100 trajectories each, starting from 100 sampled starting points have been used.

We note that it may be tempting to recover probability distributions directly from score functions, by direct integration. However, that is not expected to work well in multi-dimensions ("too many directions to explore"); while SDE provides a natural method to sample distributions (aiming to detect domains of sufficiently high density), even when full recovery is impractical.

In our study, we have tested Ito SDE processes in Mathematica, as well as by explicit coding in Python. We've posted some of our codes on the github [10]. .

6.2 What "inappropriate" means?

There is substantial amount of literature on the AI ethics, written by philosophers. It appears that presently this substantial body of work is developing on separate tracks from the algorithms and math development, so the translation is at present not easy. We just wish to note several points here.

Appropriateness, or its opposite, is obviously time, place, and culture dependent/ Humans keep arguing about this, from the beginning of time, and evolution of the norms is natural part of the progress. Great works of art often challenge, the norms boundaries; while past great works of art can be outside of present day sensitivities. Once the matter is passed to the robots, some confusion may occur.

Moreover, appropriateness is also age-dependent, so movie-like ratings could be more appropriate than hard uniform rules.

In diffusion-type models, what is recovered by de-noising is information on support of measures, not on individual points on measure space. So, beyond the cultural complications briefly mentioned, the matter will need to be posed in probabilistic terms.

7 Experiments

7.1 Prompt-Based Similarity Tracking

Cosine similarity (Equation 2.3) serves as a simplified attribution metric by measuring semantic alignment between latent representations and text prompts in CLIP embedding space, and is chosen to implement preliminary testing.

The implementation tracks two similarity trajectories simultaneously:

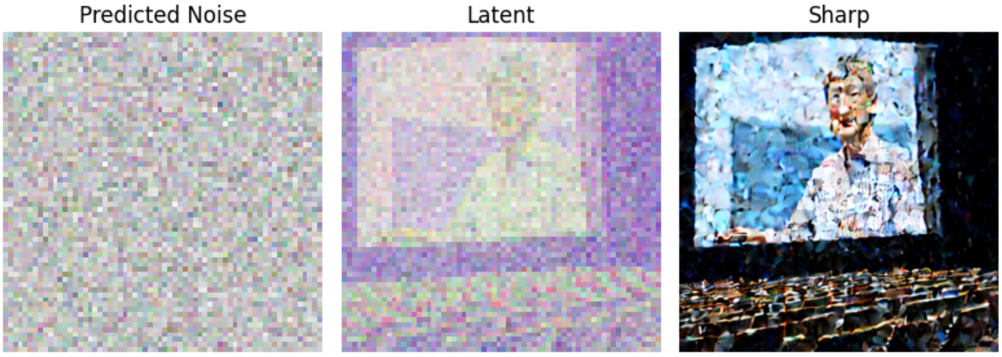
•**Positive prompt**: "A realistic photograph of Geoffrey Hinton..." (desired concepts)

•**Negative prompt**: "blurry, low quality, bad anatomy, deformed" (undesired concepts)

At each denoising step, latents are converted to pseudo-RGB (discarding the 4th channel) and fed to CLIP for similarity computation with both prompts. This dual-tracking approach mirrors classifier-free guidance principles used in generation.

While positive similarity remains low until step index 35 (often below negative similarity), latents already contain semantically interpretable and potentially sensitive content at step index 30 (Figure 7.1). The sharp divergence after step index 35 - where positive similarity peaks around step index 43 - occurs well after human-recognizable structures emerge, indicating that embedding-based metrics lag behind perceptual interpretability and cannot reliably prevent privacy leakage through intermediate representations (Figure 7.1).

Figure 6. Latent Representation Analysis at Similarity Score Divergence Point (Timestep 281): Noise Prediction, Latent Space, and VAE-Decoded Output



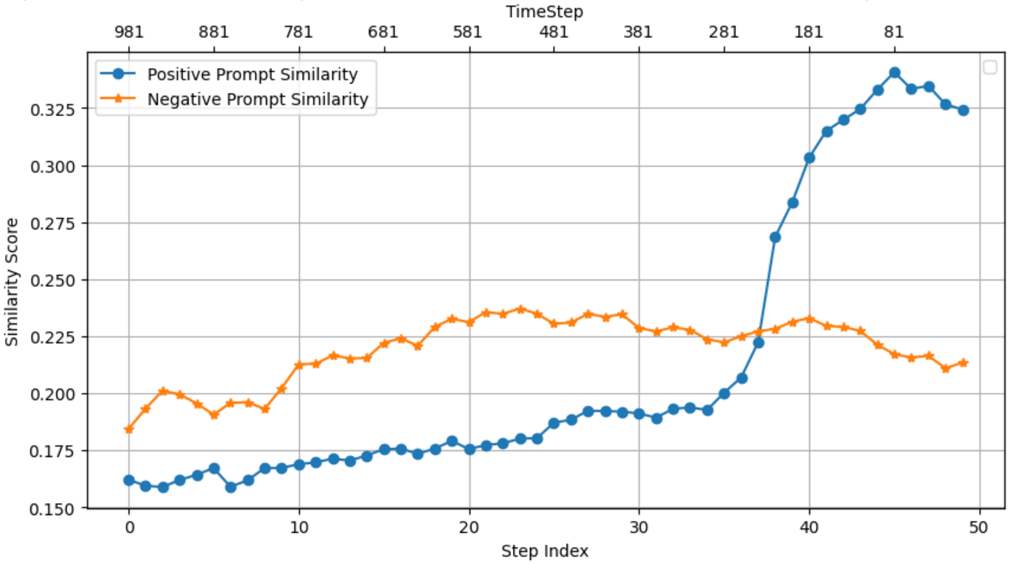
Therefore, the tracking reveals a critical limitation: cosine similarity is insufficient for safety filter building.

8 Discussions

The detection of inappropriate fine-tuning in diffusion models remains a complex task. As noted in the introduction, the two main difficulties lie in the prohibition against generating recognisable harmful content and in the possibility that unsafe behaviours are only triggered by unknown prompts. These constraints force detection to rely on indirect signals and partial evidence, making the problem uniquely challenging.

Our study shows that approaching the problem from multiple levels of the model provides a more complete perspective. Analysis at the text-to-embedding layer can reveal

Figure 7. Positive and Negative Prompt Similarity Scores Across Denoising Steps



whether unsafe concepts are encoded or bypassed. Examination of diffusion trajectories uncovers how semantically meaningful structures emerge even in very early timesteps, suggesting classifiers trained on noisy intermediates could anticipate unsafe outcomes before any visible image appears. At the parameter level, LoRA matrix encoders enable inspection of the low-rank updates commonly used in fine-tuning, offering a direct way to detect shifts that may encode harmful behaviour. The unified framework is considered to be more robust, as one layer can be complemented or supported by another.

Looking ahead, the adaptability of diffusion models suggests strong potential for further work. Even small fine-tuning datasets have been shown to significantly steer model behaviour, such as community demonstrations where a handful of images produce consistent cartoon characters [9].This implies that carefully designed proxy datasets — free of harmful content but representative of the mechanisms by which unsafe shifts occur — could provide powerful testbeds for evaluation. From there, classifiers trained on augmented trajectories could be benchmarked, interpretability methods could be combined with supervised detection, and LoRA-based stress tests could map the limits of unsafe behaviour inducibility.

In conclusion, while the detection task is difficult by design, a multi-layered strategy combining embedding analysis, trajectory classification, and parameter inspection shows promise. Future work should focus on targeted datasets and controlled experiments to establish scalable and principled auditing frameworks for diffusion model safety.

References

[1] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas

Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[2] CompVis. Stable diffusion safety checker. https://huggingface.co/CompVis/stable-diffusion-safety-checker, 2022. Model card describing CLIP-based NSFW concept filtering.

[3] Sander Dieleman. Guidance: a cheat code for diffusion models, 2022. URL https://benanne.github.io/2022/05/26/guidance.html.

[4] U. G. Haussmann and É. Pardoux. Time reversal of diffusions. *Ann. Probab.*, 14 (4):1188–1205, 1986. ISSN 0091-1798,2168-894X. URL http://links.jstor.org/sici?sici=0091-1798(198610)14:4<1188:TROD>2.0.CO;2-Z&origin=MSN.

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

[6] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Weizhu Wang, and Yongcheng Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2106.09685.

[7] Artificial Intelligence and Machine Learning Lab at TU Darmstadt. Aiml-tuda/i2p: Inappropriate image prompts (i2p) benchmark, 2023. URL https://huggingface.co/datasets/AIML-TUDA/i2p. Accessed: 2025-08-27.

[8] Jing Yu Koh, Ruslan Salakhutdinov, and Jonathan Ho. A survey on efficient adaptation of diffusion models. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2402.00843.

[9] Lambda Labs. Fine-tuning stable diffusion on pokémon with dreambooth. https://github.com/LambdaLabsML/examples/blob/main/stable-diffusion-finetuning/pokemon_finetune.ipynb, 2022.

[10] Charles Morihead and M Zyskin. Image forenscics project, https://github.com/zyskin/ImageForensics, 2025.

[11] Quang H. Nguyen, Hoang Phan, and Khoa D. Doan. Unveiling concept attribution in diffusion models, 2024. arXiv:2412.02542v2, submitted 3 Dec 2024, revised 12 Mar 2025.

[12] Quang H. Nguyen, Hoang Phan, and Khoa D. Doan. Unveiling concept attribution in diffusion models, 2025. URL https://arxiv.org/abs/2412.02542.

[13] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021. URL https://arxiv.org/abs/2102.09672.

[14] Yijun Pan, Taiwei Shi, Jieyu Zhao, and Jiaqi W. Ma. Detecting and filtering unsafe training data via data attribution, 2025. URL https://arxiv.org/abs/2502.11411.

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021. URL https://arxiv.org/abs/2103.00020.

[16] Javier Rando, Daniel Paleka, David Lindner, Lennart Heim, and Florian Tramèr. Red-teaming the stable diffusion safety filter. *arXiv preprint arXiv:2210.04610*, 2022.

[17] Synced Review. Hugging face releases lora scripts for efficient stable diffusion fine-tuning, 2023. URL https://syncedreview.com/2023/02/13/hugging-face-releases-lora-scripts-for-efficient-stable-diffusion-fine-tuning/.

[18] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

[19] Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and evolution of interpretable concepts in diffusion models. *arXiv preprint arXiv:2504.15473*, 2025.

[20] Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and Evolution of Interpretable Concepts in Diffusion Models, 2025. URL https://arxiv.org/abs/2504.15473.

[21] Vu Tuan Truong, Luan Ba Dang, and Long Bao Le. Attacks and defenses for generative diffusion models: A comprehensive survey. *arXiv preprint arXiv:2408.03400*, 2024. URL https://arxiv.org/abs/2408.03400.

[22] Dixi Yao. Risks when sharing lora fine-tuned diffusion model weights, 2024. URL https://arxiv.org/abs/2409.08482.

[23] Dixi Yao. Risks when sharing lora fine-tuned diffusion model weights, 2024. URL https://arxiv.org/abs/2409.08482.

[24] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M. Alvarez, Jan Kautz, and Pavlo Molchanov. See through gradients: Image batch recovery via gradinversion. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16369–16378. IEEE, 2021. .

[25] Guo Zhang Yuan, Ma. What lurks within? concept auditing for shared diffusion models at scale, 2025. URL https://arxiv.org/abs/2504.14815.