Detecting Memorization in Diffusion Models via Score Evolution

Jeonseong Kim
Yonsei University
Seoul, Korea
JeonseongKim@gmail.com

Chenglin Fan*
Seoul National University
Seoul, Korea
fanchenglin@snu.ac.kr

Abstract

Detecting training data memorization in diffusion models is important for copyright protection and privacy auditing. We view memorization as a highly localized concentration process: during reverse-time generation, a memorized instance acts as a point-like attractor whose probability basin is sharper and more sensitive to spatial perturbations than that of a generalized concept. Starting from the Fokker-Planck equation, we derive an exact score-evolution law linking temporal concentration dynamics to local curvature. We then show that standard sampling along a shared guided trajectory already exposes this curvature information through a complementary transport-curvature response, without requiring explicit Hessian computation. Motivated by this analysis, we propose the Dynamical Singularity Metric (DSM), an on-trajectory detector that measures changes in the conditional-unconditional score gap between consecutive sampling steps. DSM requires no additional backpropagation or network evaluations beyond sampling itself. Experiments on Stable Diffusion show that DSM is competitive with curvature-based baselines at substantially lower computational cost. Code is available on GitHub.

1 Introduction

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) have become a leading paradigm for high-fidelity image generation, but their success has also brought a serious safety concern: memorization of training examples. Recent studies have shown that diffusion models can reproduce individual training samples, raising risks of copyright infringement and privacy leakage (Carlini et al., 2023; Somepalli et al., 2023). This makes inference-time memorization detection an important practical problem.

Existing detectors suggest that memorization is related to unusually sharp local structure in the generative landscape. Heuristic methods measure the discrepancy between conditional and unconditional denoising signals (Wen et al., 2024), while geometric methods explicitly probe the Hessian of $\log p_t(\mathbf{x})$ and characterize memorized samples as regions of high local curvature (Jeon et al., 2025). These approaches are informative, but they leave two issues unresolved. First, what is the dynamical signature of memorization during reverse-time generation? Second, explicit curvature probing is expensive, since Hessian-based detectors require additional automatic differentiation or off-trajectory perturbation and therefore introduce substantial inference overhead.

In this work, we start from a different premise. We view memorization as an abnormally selective local concentration process. A well generalized concept should correspond to a relatively broad and robust semantic basin: neighboring states should undergo similar reverse-time concentration dynamics, and small perturbations should not drastically alter the local tendency of probability mass to concentrate.

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

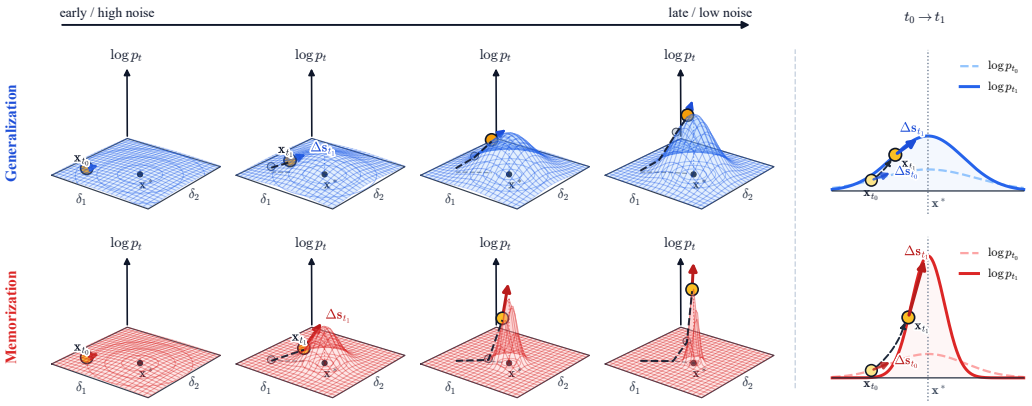


Figure 1: **Reverse-time concentration dynamics for generalization and memorization.**

By contrast, if the model has memorized a specific training instance, the associated attractor should be narrower, sharper, and more selective, consistent with recent evidence that classifier-free guidance can trigger memorization once sampling enters an attraction basin (Jain et al., 2025).

A memorized instance induces a highly localized concentration process: the local log-density grows rapidly near the memorized attractor, but this growth changes sharply under small perturbations of the state. Thus, memorization appears not only as strong local concentration, but also as high spatial sensitivity of the concentration dynamics.

This perspective naturally leads to the score evolution

$$\partial_t \mathbf{s}_t(\mathbf{x}) = \partial_t \nabla_{\mathbf{x}} \log p_t(\mathbf{x}), \qquad \mathbf{s}_t(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x}),$$

which measures how the local attraction field changes over reverse time. Starting from the Fokker–Planck equation, we derive the full evolution law for the score field. The law relates temporal score evolution to curvature–score coupling, spatial curvature variation, and drift. The curvature–score coupling $\mathbf{H}_t \mathbf{s}_t$ is the spatial gradient of score energy. Let $\lambda_{j,t}$ and $\mathbf{u}_{j,t}$ denote the eigenvalues and orthonormal eigenvectors of $\mathbf{H}_t$. Then

$$\nabla_{\mathbf{x}} \cdot \mathbf{s}_t = \sum_j \lambda_{j,t}, \qquad \|\mathbf{H}_t \mathbf{s}_t\|_2^2 = \sum_j \lambda_{j,t}^2 (\mathbf{u}_{j,t}^\top \mathbf{s}_t)^2.$$

Eigenvalues of opposite signs can cancel in the divergence. In the squared norm of the coupling term, each direction contributes a nonnegative term weighted by the square of its curvature. A partially memorized generation may reproduce certain attributes of a training image while other attributes still vary across samples. In the local density, the basin can be narrow along directions that change the reproduced attributes and broader along other directions. Even when high curvature is confined to a subset of directions, the coupling term can respond strongly if the score has substantial projections onto those directions. This response can help detect partial memorization from changes in the score.

At the same time, an inference-time detector cannot access the full Eulerian field everywhere in space; it only observes the reverse sampling trajectory. Evaluating both score fields along the shared guided trajectory introduces a transport term determined by their spatial Jacobian difference and the sampling velocity. Together with temporal score evolution, this term describes how the score discrepancy changes along the generated path. We estimate this change from consecutive score evaluations, obtaining a detector that reuses computations already performed during sampling. We call this detector the Dynamical Singularity Metric (DSM).

The term *dynamical singularity* refers to the idealized point-attractor formed by a memorized instance in the reverse generative dynamics. As the corresponding basin becomes increasingly narrow, the conditional score field and reverse-flow drift concentrate sharply around this attractor, yielding a singular limiting behavior. DSM tracks the associated change in the conditional–unconditional score discrepancy along the guided sampling trajectory.

Our contributions are summarized as follows:

- **A dynamical formulation of memorization.** We formulate memorization as an abnormally selective local concentration process during reverse-time generation. This view identifies score evolution as the relevant dynamical object for measuring how local probability concentration changes across nearby states.
- **A geometric–dynamical connection.** Starting from the Fokker–Planck equation, we derive an exact evolution law for $\partial_t s_t$. We further show that evaluating scores along the shared guided trajectory introduces a transport-curvature term, yielding a pathwise signal that implicitly captures curvature without explicit Hessian probing.
- **A practical memorization detector.** We propose DSM, an on-trajectory detector that measures the evolution of the score discrepancy between conditional and unconditional branches. Experiments on Stable Diffusion show competitive detection performance relative to curvature-based baselines at substantially lower computational cost.

2 Related Works

2.1 Memorization in Diffusion Models

The tendency of diffusion models to reproduce training examples is now well documented (Somepalli et al., 2023; Carlini et al., 2023; Wen et al., 2024; Gu et al., 2023). Prior work links higher memorization risk to duplicated training data (Webster, 2023), strong prompt conditioning (Somepalli et al., 2023), and dataset complexity (Gu et al., 2023). Together, these results make memorization a practical concern and motivate inference-time detectors that can identify potentially memorized generations without access to the training set.

2.2 Geometric Perspective: Manifolds and LID

A recent line of work studies memorization through local geometry. Under the manifold hypothesis, memorized samples are argued to collapse onto extremely low-dimensional, in the idealized limit even zero-dimensional, structures (Ross et al., 2025; Pidstrigach, 2022). This perspective motivates detectors based on Local Intrinsic Dimensionality (LID) (Kamkari et al., 2024), which attempt to identify memorized samples by estimating local intrinsic dimension. While conceptually appealing, such estimators are often expensive and can be unstable across diffusion timesteps and model architectures, which limits their usefulness in real-time settings.

2.3 From Conditioning Sensitivity to Dynamic Concentration

Existing memorization detectors can be viewed as capturing different observable signatures of the same underlying phenomenon.

Conditioning sensitivity. A prominent heuristic line measures how strongly the denoiser responds to text conditioning. For example, Wen et al. (2024) consider the conditional–unconditional discrepancy

$$\|\epsilon_\theta(\mathbf{x}_t, \mathbf{c}) - \epsilon_\theta(\mathbf{x}_t, \emptyset)\|,$$

and show that this signal is highly effective for detection. Although empirically powerful, this approach is primarily motivated by observation: it reveals that memorized samples tend to exhibit unusually amplified conditioning response, but it does not by itself specify the underlying dynamical or geometric mechanism.

Local sharpness and curvature. To explain why such heuristic signals work, Jeon et al. (2025) connect memorization to the sharpness of the probability landscape and introduce Hessian-based metrics to probe local curvature explicitly. Brokman et al. (2026) also detect memorization through a Hessian-free curvature criterion based on the normalized score-gap field. These studies establish local curvature as an informative signature of memorization and are closely related to our work.

Dynamics and collapse. Our perspective is also related to recent dynamical analyses from statistical physics. In particular, Biroli et al. (2024) study backward denoising dynamics in the exact-empirical-score setting and identify a collapse transition at time t_c, where the reverse dynamics is attracted to a specific training point. Their analysis supports the idea that memorization is fundamentally a dynamical concentration phenomenon. Our contribution is complementary: we develop a practical

on-trajectory detector that targets the local dynamical signature of this concentration process during sampling.

Position of our work. Taken together, prior work has associated memorization with conditioning sensitivity, geometric sharpness, and low-dimensional collapse. Our work is closest to the sharpness-based view, but differs in starting point: we derive curvature–score coupling as a consequence of score-evolution dynamics. This shift in formulation leads naturally to a trajectory-level detector that accesses curvature information implicitly through temporal finite differences, avoiding the computational overhead of explicit geometric probing.

3 Methodology

This section is structured as follows: we first fix notation and the observation setting, then derive the Eulerian score-evolution law, interpret its curvature-dependent components in a local quadratic model, and finally translate that field-level signal into an efficient on-trajectory detector available during sampling. Figure 1 summarizes the intuition that drives the analysis: as reverse diffusion proceeds from early/high-noise to late/low-noise regimes, generalized concepts concentrate into relatively broad basins, whereas memorized instances collapse into much sharper and more selective peaks around a specific example. Although our experiments are conducted in the diffusion setting, the underlying score-evolution identity is more general: Appendix B derives it first for a generic continuity equation and then specializes it to the Fokker–Planck case studied here.

3.1 Preliminaries

Score-based diffusion models. Score-based diffusion models generate data by reversing a gradual noising process (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2020b). We adopt the standard continuous-time formulation, where a data sample is progressively perturbed toward a tractable prior by the forward SDE

$$d\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\, dt + g(t)\, d\mathbf{w}_t, \tag{1}$$

whose marginals satisfy the Fokker–Planck equation

$$\partial_t p_t(\mathbf{x}) = -\nabla_{\mathbf{x}} \cdot \big(\mathbf{f}(\mathbf{x}, t)\, p_t(\mathbf{x})\big) + \frac{1}{2} g(t)^2 \Delta p_t(\mathbf{x}). \tag{2}$$

Let $p_t(\mathbf{x})$ denote the marginal density at time t, and define

$$\begin{aligned} \mathbf{s}_t(\mathbf{x}) &:= \nabla_{\mathbf{x}} \log p_t(\mathbf{x}), \\ \mathbf{H}_t(\mathbf{x}) &:= \nabla_{\mathbf{x}}^2 \log p_t(\mathbf{x}) = \nabla_{\mathbf{x}} \mathbf{s}_t(\mathbf{x}). \end{aligned} \tag{3}$$

The corresponding reverse-time dynamics depend on the score field and can be written as (Song et al., 2020b)

$$d\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g(t)^2 \mathbf{s}_t(\mathbf{x}_t)\big] dt + g(t)\, d\bar{\mathbf{w}}_t, \tag{4}$$

where $\bar{\mathbf{w}}_t$ denotes Brownian motion in reverse time. This score-centered description is the starting point of our analysis.

Classifier-free guidance and latent diffusion. For conditional generation under classifier-free guidance (Ho and Salimans, 2022), we write

$$\mathbf{s}_{c,t}(\mathbf{x}) := \nabla_{\mathbf{x}} \log p_t(\mathbf{x} \mid c), \qquad \mathbf{s}_{\emptyset,t}(\mathbf{x}) := \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) \tag{5}$$

for the conditional and unconditional scores. The diffusion coefficients $\mathbf{f}$ and g are properties of the underlying forward process and are therefore shared by both branches.

In the Stable Diffusion setting used in our experiments, these quantities are realized in the latent space of a pretrained autoencoder rather than directly in pixel space (Rombach et al., 2022). In our experiments, $\mathbf{x}_t$ denotes the latent state along the diffusion chain. In practice, the model is parameterized through a noise predictor $\boldsymbol{\epsilon}_\theta$. Under the standard VP/DDPM parameterization (Ho et al., 2020), the corresponding score estimate satisfies

$$\mathbf{s}_{b,t}(\mathbf{x}_t) \approx -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, b)}{\sqrt{1 - \bar{\alpha}_t}}, \qquad b \in \{c, \emptyset\}, \tag{6}$$

so the score-based quantities studied below are computed from the same conditional and unconditional denoising predictions used in standard latent diffusion pipelines such as Stable Diffusion (Rombach et al., 2022). The density-based identities below describe the ideal score field. DSM is evaluated from the learned score predictions along the sampling trajectory.

Under classifier-free guidance (Ho and Salimans, 2022), the guided prediction combines the two branches as

$$\widehat{\boldsymbol{\epsilon}}_{\mathrm{cfg}}(\mathbf{x}_t, c) := \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \emptyset) + w\big(\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, c) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \emptyset)\big). \tag{7}$$

Eulerian and pathwise viewpoints. At inference time, classifier-free guidance produces a shared guided reverse trajectory $\{\mathbf{x}_{t_k}\}_{k=1}^K$ with $t_1 > t_2 > \cdots > t_K$. Both the conditional and unconditional branches are evaluated at the same sampled states $\mathbf{x}_{t_k}$; they differ in the score field being queried. When a continuous-time notation is convenient, we denote the local reverse motion of this shared trajectory by $\mathbf{v}_{\mathrm{cfg},t}$. Throughout the paper, we distinguish an *Eulerian* quantity, defined at a fixed state $\mathbf{x}$, from a *pathwise* quantity, evaluated along the realized reverse trajectory. The field difference $\mathbf{s}_{c,t} - \mathbf{s}_{\emptyset,t}$ therefore measures how conditioning changes the local score field at a fixed state, whereas $\mathbf{v}_{\mathrm{cfg},t}$ describes how the sampled state itself moves through space. Our theory is formulated for the Eulerian field $\partial_t \mathbf{s}_t(\mathbf{x})$, while DSM is computed from score evaluations available along the shared visited trajectory.

3.2 Local concentration

We use local concentration to refer to the reverse-time growth of log-density in a small neighborhood of a visited latent state. At this stage we adopt an Eulerian local view: for a fixed state x and time t, we ask how the local concentration rate $-\partial_t \log p_t(x)$ changes under nearby perturbations of x.

A generalized prompt typically corresponds to a broad family of compatible clean outputs. Its conditional basin is therefore relatively smooth: nearby perturbations of a visited latent state tend to have similar reverse-time concentration rates. A memorized prompt behaves differently. Once conditioning selects an instance-specific component, the concentration rate becomes spatially localized: it is large near the memorized attractor (Jain et al., 2025) and changes rapidly along perturbation directions that move away from that component. Thus the relevant signature is not only a high-probability endpoint, but a high spatial sensitivity of the local concentration law.

This motivates studying the spatial gradient of the concentration rate. Since $\partial_t \log p_t(x)$ describes the instantaneous change of local log-density, taking its spatial gradient gives

$$\partial_t \nabla_x \log p_t(x) = \partial_t s_t(x).$$

Therefore, $\partial_t s_t(x)$ measures how sensitively the reverse-time concentration dynamics vary across nearby states. A large value indicates that small spatial perturbations experience substantially different concentration rates. This is the local dynamical signature we associate with selective memorization.

This view is consistent with sharpness-based evidence: memorized samples exhibit larger-magnitude Hessian spectra (Jeon et al., 2025), so the spatial sensitivity of the concentration rate can accumulate across many local directions. The next subsection derives the full score-evolution law from the Fokker–Planck equation and identifies its curvature-dependent contributions.

3.3 Exact score evolution from the Fokker–Planck dynamics

Starting from Eq. (2) and using the identities $\Delta p_t/p_t = \mathrm{tr}(\mathbf{H}_t) + \|\mathbf{s}_t\|_2^2$ and $\nabla_{\mathbf{x}} \cdot (\mathbf{f} p_t)/p_t = \nabla_{\mathbf{x}} \cdot \mathbf{f} + \mathbf{f}^\top \mathbf{s}_t$, we obtain

$$\partial_t \log p_t(\mathbf{x}) = -\nabla_{\mathbf{x}} \cdot \mathbf{f}(\mathbf{x}, t) - \mathbf{f}(\mathbf{x}, t)^\top \mathbf{s}_t(\mathbf{x}) + \frac{1}{2} g(t)^2 \Big(\mathrm{tr}\big(\mathbf{H}_t(\mathbf{x})\big) + \|\mathbf{s}_t(\mathbf{x})\|_2^2 \Big). \tag{8}$$

Taking the spatial gradient yields the exact evolution law

$$\partial_t \mathbf{s}_t = \underbrace{-\nabla_{\mathbf{x}}\big(\nabla_{\mathbf{x}} \cdot \mathbf{f}\big) - \big(\nabla_{\mathbf{x}} \mathbf{f}\big)^\top \mathbf{s}_t - \mathbf{H}_t \mathbf{f}}_{\text{drift and transport}} + \underbrace{\frac{1}{2} g(t)^2 \nabla_{\mathbf{x}} \mathrm{tr}(\mathbf{H}_t)}_{\text{curvature variation}} + \underbrace{g(t)^2 \mathbf{H}_t \mathbf{s}_t}_{\text{curvature–score coupling}}. \tag{9}$$

Equation (9) separates drift, spatial curvature variation, and curvature–score coupling. For a quadratic log-density, the spatial curvature-variation term vanishes. If the drift terms are also small relative to

the coupling term, the score evolution reduces to the local approximation

$$\partial_t \mathbf{s}_t(\mathbf{x}) \approx g(t)^2 \mathbf{H}_t(\mathbf{x}) \mathbf{s}_t(\mathbf{x}). \tag{10}$$

A useful identity is

$$\mathbf{H}_t(\mathbf{x}) \mathbf{s}_t(\mathbf{x}) = \nabla_{\mathbf{x}} \left(\frac{1}{2} \|\mathbf{s}_t(\mathbf{x})\|_2^2 \right). \tag{11}$$

Equation (11) identifies $\mathbf{H}_t \mathbf{s}_t$ as the spatial gradient of score energy. In the quadratic-basin model, this term scales quadratically with curvature along each principal direction. Appendix B.4 gives the corresponding derivation.

3.4 Conditioning-induced concentration sensitivity

The quantity $\partial_t \mathbf{s}_{c,t}$ mixes two effects: the baseline concentration dynamics of the diffusion model itself, and the additional local concentration induced by the condition c. For memorization detection, the relevant signal is the latter. The unconditional branch provides a natural baseline for the generative concentration dynamics, while the conditional branch contains the prompt-specific attraction toward the conditioned mode. This motivates the differential quantity

$$\partial_t \mathbf{s}_{c,t}(\mathbf{x}) - \partial_t \mathbf{s}_{\emptyset,t}(\mathbf{x}), \tag{12}$$

which isolates the prompt-specific local concentration sensitivity.

Define $\Delta \mathbf{s}_t = \mathbf{s}_{c,t} - \mathbf{s}_{\emptyset,t}$ and $\Delta \mathbf{H}_t = \mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t}$. Since the two branches share the same forward dynamics, subtracting Eq. (9) cancels the common divergence-gradient term and gives

$$\begin{aligned} \partial_t \Delta \mathbf{s}_t = g(t)^2 \big(\mathbf{H}_{c,t} \mathbf{s}_{c,t} - \mathbf{H}_{\emptyset,t} \mathbf{s}_{\emptyset,t} \big) \\ - \Delta \mathbf{H}_t \mathbf{f} - \big(\nabla_{\mathbf{x}} \mathbf{f} \big)^\top \Delta \mathbf{s}_t + \frac{g(t)^2}{2} \nabla_{\mathbf{x}} \mathrm{tr}(\Delta \mathbf{H}_t). \end{aligned} \tag{13}$$

The differential evolution therefore contains curvature–score coupling, drift, and spatial curvature variation. Appendix B.3 provides the derivation.

3.5 From Eulerian score evolution to an on-trajectory detector

The differential score-evolution law motivates tracking how the score discrepancy changes along the sampling trajectory. Both branches are already evaluated at the shared visited states, allowing this pathwise change to be estimated from consecutive sampling steps.

During sampling, we only observe the shared guided reverse trajectory. Let $\mathbf{x}_{t_k}$ denote the sampled state at solver step k, with $t_{k+1} < t_k$. For $b \in \{c, \emptyset\}$, define

$$\widehat{D}_t \mathbf{s}_b(t_k) := \frac{\mathbf{s}_{b,t_k}(\mathbf{x}_{t_k}) - \mathbf{s}_{b,t_{k+1}}(\mathbf{x}_{t_{k+1}})}{|t_k - t_{k+1}|}. \tag{14}$$

This is a finite-difference estimator of the pathwise score derivative along the visited trajectory. In continuous time, if the shared guided trajectory has velocity $\mathbf{v}_{\mathrm{cfg},t} := \dot{\mathbf{x}}_t$, then

$$D_t \mathbf{s}_{b,t}(\mathbf{x}_t) = \partial_t \mathbf{s}_{b,t}(\mathbf{x}_t) + \mathbf{H}_{b,t}(\mathbf{x}_t) \mathbf{v}_{\mathrm{cfg},t}. \tag{15}$$

Therefore, subtracting the two branches gives

$$D_t \mathbf{s}_{c,t} - D_t \mathbf{s}_{\emptyset,t} = \big(\partial_t \mathbf{s}_{c,t} - \partial_t \mathbf{s}_{\emptyset,t} \big) + \big(\mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t} \big) \mathbf{v}_{\mathrm{cfg},t}. \tag{16}$$

Equation (16) separates temporal score evolution at a fixed state from the response to motion through the shared score fields. Substituting the full evolution law retains curvature–score coupling, transport, drift, and spatial curvature variation. Appendix B.7 illustrates the direct and transport terms in a local quadratic model.

We define the **Dynamical Singularity Metric (DSM)** at step t_k by

$$\mathcal{M}_{\mathrm{DSM}}(t_k) := \left\| \widehat{D}_t \mathbf{s}_c(t_k) - \widehat{D}_t \mathbf{s}_\emptyset(t_k) \right\|_2^2. \tag{17}$$

A high DSM score indicates rapid pathwise change in the score discrepancy between conditional and unconditional branches. We use this variation as the detection signal for memorization. Since Eq. (17) uses only scores already computed during sampling, DSM requires no extra network function evaluations, no additional backpropagation, and no explicit off-trajectory curvature probing.

Table 1: Memorization detection performance averaged over five seeds. We report mean ± standard deviation. For Wen and DSM, only the earliest-step variants are shown. Best results in each column are bolded.

	SD1.4		SD1.5		SD2.0-base	
Metric	AUC	TPR@1%FPR	AUC	TPR@1%FPR	AUC	TPR@1%FPR
$\mathcal{M}_{\text{Jeon}}$	0.9682±0.0241	0.8660±0.1033	**0.9730±0.0131**	**0.8712±0.0717**	0.9764±0.0315	0.8201±0.2018
$\mathcal{M}_{\text{Wen}}$	0.9429±0.0480	0.7940±0.1469	0.9461±0.0309	0.7896±0.1251	0.9708±0.0312	0.8137±0.2611
$\mathcal{M}_{\text{DSM(Ours)}}$	**0.9694±0.0212**	**0.8680±0.0965**	0.9718±0.0145	0.8640±0.0797	**0.9864±0.0121**	**0.8932±0.0925**
$\mathcal{M}_{\text{Ani}}$	0.9685±0.0215	0.8124±0.1510	0.9668±0.0151	0.7960±0.1265	0.9536±0.0477	0.7507±0.2728

Table 2: Total runtime per prompt. We report mean runtime in seconds per prompt for the full detector pipeline, measured under the same implementation and hardware setting.

Method	SD 1.4	SD 1.5	SD 2.0-base
$\mathcal{M}_{\text{Jeon}}$	0.3190	0.3183	0.2907
$\mathcal{M}_{\text{Wen}}$	0.0346	0.0342	0.0396
$\mathcal{M}_{\text{DSM}}(Ours)$	0.0577	0.0572	0.0617
$\mathcal{M}_{\text{Ani}}$	0.1072	0.1071	0.1158

4 Experimental Results

We evaluate DSM against existing memorization detectors under single-shot inference. The experiments compare detection performance and computational cost, followed by qualitative analysis and an evaluation across discretization settings.

4.1 Experimental Setup

We evaluate three Stable Diffusion latent diffusion models (Rombach et al., 2022): **SD 1.4**, **SD 1.5**, and **SD 2.0-base**. For SD 2.0, we use `stable-diffusion-2-base`, which matches the Webster SDV2 memorization setup and its associated 219-prompt memorized set.

Datasets. For SD 1.4 and SD 1.5, we use the standard Memorized Set ($N = 500$) from Webster (2023); for SD 2.0-base, we use the corresponding Webster SDV2 memorized set ($N = 219$). For all models, we use a composite Generalized Set ($N = 500$) collected from COCO (Lin et al., 2014), Lexica (Lexica, 2024), Tuxemon (Hugging Face, 2024), and a subset of GPT-4 (Achiam et al., 2023)-generated captions from Ren et al. (2024). Additional details on dataset composition and hyperparameters are provided in Appendix D.

Unified evaluation protocol. All methods are re-evaluated under an aligned single-shot protocol. Specifically, we repeat the single-shot generation across five shared random seeds, ensuring comparability by matching prompt splits, model checkpoints, and implementation paths wherever possible.

This removes implementation-side discrepancies and makes the comparison directly interpretable across methods and models. We additionally include the anisotropy-based detector of Asthana and Belagiannis (2026) under the same unified single-shot prompt/checkpoint/seed protocol.

Metrics. We report AUC and TPR@1%FPR. For notational consistency, we denote the efficient score-difference baseline of Wen et al. (2024) by $\mathcal{M}_{\text{Wen}}$, the geometric detector of Jeon et al. (2025) by $\mathcal{M}_{\text{Jeon}}$, the anisotropy-based detector of Asthana and Belagiannis (2026) by $\mathcal{M}_{\text{Ani}}$.

We evaluate DSM at the first reverse-step transition $(0 \to 1)$, using the earliest-evaluable-time protocol described in Appendix D.

4.2 Main Detection Performance

Table 1 summarizes the unified re-evaluation results.

Taken together, Table 1 and Table 2 show the central empirical result: DSM is competitive with the Hessian-based detector (Jeon et al., 2025) at a fraction of the cost. DSM achieves the best

performance on SD 1.4 and SD 2.0-base under both AUC and TPR@1%FPR, remains nearly tied with Jeon et al. (2025) on SD 1.5, and is substantially cheaper because it avoids explicit Hessian computation.

DSM consistently outperforms the instantaneous score-gap baseline of Wen et al. (2024) across the evaluated models and metrics. Under the shared evaluation protocol, these results show that pathwise score variation provides useful detection information beyond the instantaneous gap.

Relation to explicit curvature. At the first sampling transition, DSM is strongly correlated with the branch-differential Hessian magnitude $\|H_c - H_\emptyset\|_F^2$, with Spearman correlations of 0.906 on SD1.4 and 0.908 on SD1.5.

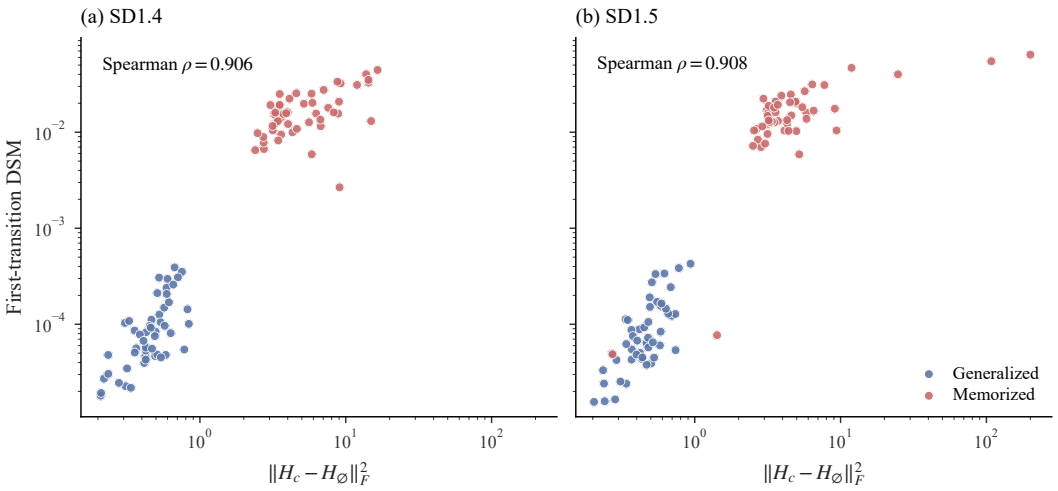


Figure 2: First-transition DSM versus the estimated squared Frobenius norm of the conditional–unconditional Hessian difference. Each point represents one prompt, with 100 prompts per model. Blue denotes generalized prompts and red denotes memorized prompts. The annotations report Spearman correlations.

4.3 Efficiency and Runtime

Table 2 reports the total end-to-end runtime per prompt under the same implementation and hardware setting on a single NVIDIA GeForce RTX 5090 GPU (32 GB). The latency ordering is stable across all three model families. $\mathcal{M}_{\text{Wen}}$ is the fastest method at 0.0342–0.0396 s per prompt, $\mathcal{M}_{\text{DSM}}(0 \to 1)$ follows at 0.0572–0.0617 s, $\mathcal{M}_{\text{Ani}}$ occupies an intermediate regime at 0.1071–0.1158 s, and $\mathcal{M}_{\text{Jeon}}$ is the slowest at 0.2907–0.3190 s.

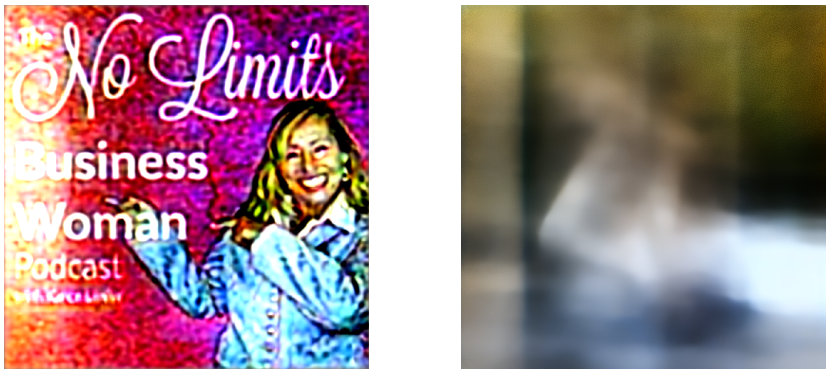
Combined with Table 1, these results show that DSM offers the best accuracy–latency tradeoff among the stronger detectors. Relative to $\mathcal{M}_{\text{Jeon}}$, DSM is substantially faster while matching or exceeding Jeon's detection performance on SD 1.4 and SD 2.0-base and remaining highly competitive on SD 1.5. Relative to $\mathcal{M}_{\text{Ani}}$, DSM is also clearly faster while improving both AUC and TPR@1%FPR on all three models. Although $\mathcal{M}_{\text{Wen}}(0)$ remains the cheapest baseline, DSM adds only a modest amount of latency while consistently delivering higher detection accuracy.

4.4 Qualitative Visualization via Early $\hat{x}_0$ Prediction

To qualitatively inspect the earliest semantic tendency of the model, we visualize the estimated clean image $\hat{x}_0$ at the first sampling step ($t \approx T$) using the standard DDIM formula:

$$\hat{x}_0 = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, c)}{\sqrt{\bar{\alpha}_t}}. \tag{18}$$

Figure 3 shows that memorized prompts can already produce more structured early $\hat{x}_0$ estimates, whereas generalized prompts remain more diffuse. We use this as a qualitative diagnostic. The noisy latent x_t itself has not yet become image-like. Appendix B.5 gives a Gaussian-mixture interpretation



(a) **Memorized** (b) **Generalized**

Figure 3: **Early $\hat{x}_0$ visualization.** A horizontal comparison of decoded estimates at the first reverse step ($t \approx T$). Memorized prompts can already produce more structured early $\hat{x}_0$ estimates, whereas generalized prompts remain comparatively diffuse.

showing that early $\hat{x}_0$ corresponds to a responsibility-weighted posterior clean estimate. Under this view, memorized prompts can have a more concentrated conditional posterior over clean components even at the earliest reverse steps, which is consistent with the early DSM signal observed in Table 1.

4.5 Robustness Analysis and Limitations

Robustness to discretization. DSM remains effective even when the number of solver steps is reduced. A coarse finite difference still captures the accumulated change of the score along the trajectory, and memorized samples typically undergo a much larger total variation than generalized ones. This preserves separability without requiring very fine temporal resolution. Appendix A.1 reports the exact fixed-seed SD1.4 DDIM step-scan protocol and the corresponding results.

Extreme few-step generation. A practical limitation appears only in the regime of extremely small step counts. In that setting, the reverse trajectory contains too little temporal information for stable score-evolution estimates, and detection quality can degrade. In practice, denser monitoring of the earliest steps or short-window aggregation across multiple early transitions improves stability.

5 Discussion and Conclusion

We study memorization in diffusion models by examining how the score field evolves during reverse diffusion. Starting from the Fokker–Planck equation, we derive an exact score-evolution law and connect it to the changes observed along the sampling trajectory. This connection shows that local curvature information can be reflected in score changes as samples move along the guided trajectory. In this way, the sampling process itself provides a way to probe local geometry without explicitly computing the Hessian.

By tracking changes in the conditional–unconditional score gap between consecutive sampling steps, we can detect memorization using only predictions already available during generation. Experiments on Stable Diffusion show that this approach achieves competitive detection performance at substantially lower computational cost.

Acknowledgments and Disclosure of Funding

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2026-25484051) and by the New Faculty Startup Fund of Seoul National University.

The authors declare no competing interests.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Rohan Asthana and Vasileios Belagiannis. Detecting and mitigating memorization in diffusion models through anisotropy of the log-probability. In *International Conference on Learning Representations*, volume 2026, pages 154868–154887, 2026.

Giulio Biroli, Tony Bonnaire, Valentin De Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15(1):9957, 2024.

Jonathan Brokman, Itay Gershon, Omer Hofman, Guy Gilboa, and Roman Vainshtein. Tracking memorization geometry throughout the diffusion model generative process. In *Proceedings of the 4th (2025) and 3rd (2024) NeurIPS Workshops on Symmetry and Geometry in Neural Representations*, volume 282 of *Proceedings of Machine Learning Research*, pages 73–86. PMLR, 2026.

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.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *arXiv preprint arXiv:2310.02664*, 2023.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

Hugging Face. Tuxemon: A pixel art monster dataset. `https://huggingface.co/datasets/diffusers/tuxemon`, 2024. Accessed: 2024.

Anubhav Jain, Yuya Kobayashi, Takashi Shibuya, Yuhta Takida, Nasir Memon, Julian Togelius, and Yuki Mitsufuji. Classifier-free guidance inside the attraction basin may cause memorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12871–12879, 2025.

Dongjae Jeon, Dueun Kim, and Albert No. Understanding and mitigating memorization in generative models via sharpness of probability landscapes. In *Forty-second International Conference on Machine Learning*, 2025.

Hamidreza Kamkari, Brendan L Ross, Rasa Hosseinzadeh, Jesse C Cresswell, and Gabriel Loaiza-Ganem. A geometric view of data complexity: Efficient local intrinsic dimension estimation with diffusion models. In *Advances in Neural Information Processing Systems*, volume 37, pages 38307–38354, 2024.

Lexica. Lexica dataset. `https://huggingface.co/datasets/vera365/lexica_dataset`, 2024. Accessed: 2024.

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 *European conference on computer vision*, pages 740–755. Springer, 2014.

Jakiw Pidstrigach. Score-based generative models detect manifolds. In *Advances in Neural Information Processing Systems*, volume 35, pages 35852–35865, 2022.

Jie Ren, Yaxin Li, Shenglai Zeng, Han Xu, Lingjuan Lyu, Yue Xing, and Jiliang Tang. Unveiling and mitigating memorization in text-to-image diffusion models through cross attention. In *European Conference on Computer Vision*, pages 340–356. Springer, 2024.

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.

Brendan Ross, Hamidreza Kamkari, Tongzi Wu, Rasa Hosseinzadeh, Zhaoyan Liu, George Stein, Jesse Cresswell, and Gabriel Loaiza-Ganem. A geometric framework for understanding memorization in generative models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Learning Representations*, volume 2025, pages 34773–34805, 2025.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

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.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

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*, 2020b.

Ryan Webster. A reproducible extraction of training images from diffusion models. *arXiv preprint arXiv:2305.08694*, 2023.

Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

A Formal Support for the DSM Mechanism

This section first records the auxiliary fixed-seed DDIM step-scan experiment referenced in Section 4.5, and then establishes the consistency of the discrete pathwise score estimator. Appendix B derives the exact Eulerian score-evolution identities; the purpose of the formal part of this section is narrower. We isolate the consequences most directly used by the detector: the estimator is a consistent approximation to the material derivative.

A.1 Fixed-seed DDIM step scan

To complement the robustness discussion in Section 4.5, we perform a controlled DDIM step scan under a fixed-seed protocol.

Model and sampler. We use Stable Diffusion v1.4 with the standard ϵ-prediction parameterization and a DDIM sampler (Song et al., 2020a). The guidance scale is fixed to $w = 7.5$, and all denoising inference is performed in `float16`.

Dataset. We use the standard SD1.4 benchmark considered in this paper, consisting of 500 memorized prompts and 500 non-memorized prompts, for a total of 1000 prompts.

Fixed-seed protocol. We use a fixed-seed evaluation mode in which all prompts share the same initial noise $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The random seed is fixed.

Metric definition. For the DDIM trajectory $\{\mathbf{x}_{t_k}\}$ generated from the fixed seed, define the conditional–unconditional score gap at step k by

$$\Delta\mathbf{s}_{t_k} := \mathbf{s}_\theta(\mathbf{x}_{t_k}, t_k, c) - \mathbf{s}_\theta(\mathbf{x}_{t_k}, t_k, \emptyset).$$

We then evaluate

$$\mathcal{M}_{\mathrm{DSM}}(k \to k+1) := \left\| \frac{\Delta s_{t_k} - \Delta s_{t_{k+1}}}{|t_k - t_{k+1}|} \right\|_2^2.$$

Step scan. We fix all hyperparameters above and vary only the total number of DDIM steps, $N \in \{5, 10, 20, 30, 50\}$. For each N, the DDIM scheduler uniformly discretizes the training-time index range $[1, T]$ with $T = 1000$ into N timesteps. Therefore, the actual timestep values associated with positions $k = 0, 1, 2$ change with N; this discretization is the only independent variable in the scan.

Evaluation metrics. For each configuration, we report ROC-AUC and TPR@1%FPR on all 1000 prompts.

Table 3: Fixed-seed DDIM step scan on SD1.4. We vary only the total number of DDIM steps and report ROC-AUC and TPR@1%FPR for the two earliest DSM variants.

	DSM$(0 \to 1)$		DSM$(1 \to 2)$	
Steps	AUC	TPR@1%FPR	AUC	TPR@1%FPR
50	0.9876	0.9160	0.9891	0.9380
30	0.9841	0.9260	0.9871	0.9280
20	0.9839	0.9320	0.9890	0.9360
10	0.9805	0.9000	0.8756	0.5860
5	0.9108	0.5620	0.9215	0.6420

A.2 Robustness to Guidance, Sampler, and Step Count

We evaluate the first-transition DSM signal $(0 \to 1)$ on the full evaluation sets of SD1.4, SD1.5, and SD2.0-base. All runs use a fixed random seed of 42. For guidance sensitivity, we vary the CFG scale over $w \in \{3, 5, 7.5, 10\}$ using DDIM with 50 sampling steps. For sampler sensitivity, we compare

DDIM, DPM-Solver, and Euler at $w = 7.5$ with 50 steps. For discretization sensitivity, we vary the number of DDIM steps over $\{5, 10, 20, 30, 50\}$ at $w = 7.5$. All other settings follow Appendix D.

Table 4: Robustness of first-transition DSM across CFG scales, samplers, and sampling steps. Ranges summarize the tested configurations. All experiments use the full evaluation sets and seed 42.

Model	Metric	CFG variation	Sampler variation	10–50 steps	5 steps
SD1.4	AUC	0.9868–0.9885	0.9876–0.9898	0.9805–0.9876	0.9108
	TPR@1%FPR	0.902–0.934	0.916–0.952	0.900–0.932	0.562
SD1.5	AUC	0.9832–0.9866	0.9835–0.9845	0.9746–0.9835	0.9056
	TPR@1%FPR	0.886–0.908	0.906–0.918	0.848–0.906	0.578
SD2.0-base	AUC	0.9919–0.9941	0.9923–0.9926	0.9903–0.9926	0.9859
	TPR@1%FPR	0.941–0.954	0.945–0.950	0.941–0.959	0.936

Table 4 shows that DSM maintains strong earliest-transition detection across the tested CFG scales and samplers. With 10–50 DDIM steps, AUC remains above 0.974 across all three checkpoints. At five steps, detection performance decreases on SD1.4 and SD1.5, while SD2.0-base retains an AUC of 0.9859.

Initial noise We conducted a full prompt-specific-noise evaluation on all three checkpoints. Each prompt was assigned a reproducible prompt-specific seed, while the same seed sequence was used across the memorized and generalized sets for a controlled comparison. The evaluation included 500 memorized and 500 generalized prompts for SD1.4 and SD1.5, and 219 memorized and 500 generalized prompts for SD2.0-base.

Model	**AUC / TPR@1%FPR**
SD1.4	0.9696 / 0.794
SD1.5	0.9518 / 0.784
SD2.0-base	0.9839 / 0.840

DSM remains strongly predictive under prompt-specific initial noise across all three checkpoints, with AUCs ranging from 0.9518 to 0.9839. These results confirm that its early detection performance is not an artifact of the fixed-noise step-scan setting.

A.3 Discrete pathwise consistency

Let $\mathbf{x}_t$ be a reverse trajectory that is C^2 in time on the solver interval $[t_{k+1}, t_k]$, and define $h_k := t_k - t_{k+1} > 0$. For each branch $b \in \{c, \emptyset\}$, write

$$\phi_b(t) := \mathbf{s}_{b,t}(\mathbf{x}_t). \tag{19}$$

The implemented DSM estimator uses the backward difference

$$\widehat{D_t \mathbf{s}_b}(t_k) := \frac{\phi_b(t_k) - \phi_b(t_{k+1})}{h_k}. \tag{20}$$

Proposition A.1 (Finite-difference consistency for score evolution)**.** *Assume that ϕ_b is twice continuously differentiable on $[t_{k+1}, t_k]$. Then*

$$\widehat{D_t \mathbf{s}_b}(t_k) = D_t \mathbf{s}_{b,t_k}(\mathbf{x}_{t_k}) - \frac{1}{h_k} \int_{t_{k+1}}^{t_k} (t - t_{k+1}) \frac{d^2}{dt^2} \left[\mathbf{s}_{b,t}(\mathbf{x}_t)\right] dt. \tag{21}$$

In particular, if

$$\sup_{t \in [t_{k+1}, t_k]} \left\| \frac{d^2}{dt^2} \left[\mathbf{s}_{b,t}(\mathbf{x}_t)\right] \right\|_2 \le L_{b,k}, \tag{22}$$

then

$$\left\| \widehat{D_t \mathbf{s}_b}(t_k) - D_t \mathbf{s}_{b,t_k}(\mathbf{x}_{t_k}) \right\|_2 \le \frac{1}{2} L_{b,k} h_k. \tag{23}$$

Proof. Apply Taylor's theorem with integral remainder to $\phi_b(t_{k+1})$ around t_k:

$$\phi_b(t_{k+1}) = \phi_b(t_k) - h_k\dot{\phi}_b(t_k) + \int_{t_{k+1}}^{t_k} (t - t_{k+1})\ddot{\phi}_b(t)\,dt. \tag{24}$$

Rearranging and using $\dot{\phi}_b(t_k) = D_t\mathbf{s}_{b,t_k}(\mathbf{x}_{t_k})$ gives Eq. (21). The norm bound follows from

$$\frac{1}{h_k}\int_{t_{k+1}}^{t_k} (t - t_{k+1})\,dt = \frac{h_k}{2}. \tag{25}$$

□

B Theoretical Derivation of Differential Score Dynamics

This section derives the score evolution equations used in the main text. We first present the general continuity-equation form, then specialize it to diffusion dynamics, and finally analyze the local quadratic basin regime and the conditional–unconditional differential signal.

B.1 General score evolution under a continuity equation

We begin with the continuity equation to isolate the transport-generic structure of score evolution. This derivation does not rely on a particular stochastic diffusion parameterization and therefore applies more broadly to generative models defined through smooth density transport fields. The subsequent Fokker–Planck subsection then specializes this general identity to the diffusion setting used in our experiments.

Let $p_t(\mathbf{x})$ satisfy the continuity equation

$$\partial_t p_t(\mathbf{x}) + \nabla_{\mathbf{x}}\cdot\big(p_t(\mathbf{x})\mathbf{v}_t(\mathbf{x})\big) = 0, \tag{26}$$

for some velocity field $\mathbf{v}_t(\mathbf{x})$. Define

$$\mathbf{s}_t(\mathbf{x}) := \nabla_{\mathbf{x}}\log p_t(\mathbf{x}), \qquad \mathbf{H}_t(\mathbf{x}) := \nabla_{\mathbf{x}}^2\log p_t(\mathbf{x}). \tag{27}$$

Dividing Eq. (26) by $p_t(\mathbf{x})$ gives

$$\partial_t \log p_t(\mathbf{x}) = -\nabla_{\mathbf{x}}\cdot\mathbf{v}_t(\mathbf{x}) - \mathbf{v}_t(\mathbf{x})^\top\mathbf{s}_t(\mathbf{x}). \tag{28}$$

Taking the spatial gradient yields

$$\partial_t\mathbf{s}_t(\mathbf{x}) = -\nabla_{\mathbf{x}}\big(\nabla_{\mathbf{x}}\cdot\mathbf{v}_t(\mathbf{x})\big) - \big(\nabla_{\mathbf{x}}\mathbf{v}_t(\mathbf{x})\big)^\top\mathbf{s}_t(\mathbf{x}) - \mathbf{H}_t(\mathbf{x})\mathbf{v}_t(\mathbf{x}). \tag{29}$$

Equation (29) shows that Hessian coupling is not specific to any particular detector construction: it is a structural component of score evolution under general mass transport.

A particularly informative local regime is when the velocity field is approximately aligned with the score,

$$\mathbf{v}_t(\mathbf{x}) \approx -a_t\mathbf{s}_t(\mathbf{x}), \qquad a_t > 0. \tag{30}$$

Substituting Eq. (30) into Eq. (29), and using $\nabla_{\mathbf{x}}\mathbf{v}_t \approx -a_t\mathbf{H}_t$ and $\nabla_{\mathbf{x}}\cdot\mathbf{v}_t \approx -a_t\mathrm{tr}(\mathbf{H}_t)$, gives

$$\partial_t\mathbf{s}_t(\mathbf{x}) \approx a_t\nabla_{\mathbf{x}}\mathrm{tr}(\mathbf{H}_t(\mathbf{x})) + 2a_t\mathbf{H}_t(\mathbf{x})\mathbf{s}_t(\mathbf{x}). \tag{31}$$

Thus, in score-aligned and locally stiff regions, Hessian–score coupling is the leading geometric component of score evolution.

B.2 Specialization to the Fokker–Planck dynamics

Now consider the forward diffusion

$$d\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,dt + g(t)\,d\mathbf{w}_t, \tag{32}$$

whose marginals satisfy

$$\partial_t p_t(\mathbf{x}) = -\nabla_{\mathbf{x}}\cdot\big(\mathbf{f}(\mathbf{x}, t)\,p_t(\mathbf{x})\big) + \frac{1}{2}g(t)^2\Delta p_t(\mathbf{x}). \tag{33}$$

Using the identities

$$\frac{\Delta p_t(\mathbf{x})}{p_t(\mathbf{x})} = \mathrm{tr}\big(\mathbf{H}_t(\mathbf{x})\big) + \|\mathbf{s}_t(\mathbf{x})\|_2^2, \tag{34}$$

and

$$\nabla_{\mathbf{x}}\big(\mathbf{f}(\mathbf{x},t)^\top \mathbf{s}_t(\mathbf{x})\big) = \big(\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x},t)\big)^\top \mathbf{s}_t(\mathbf{x}) + \mathbf{H}_t(\mathbf{x})\mathbf{f}(\mathbf{x},t), \tag{35}$$

we obtain

$$\partial_t \log p_t(\mathbf{x}) = -\nabla_{\mathbf{x}}\cdot \mathbf{f}(\mathbf{x},t) - \mathbf{f}(\mathbf{x},t)^\top \mathbf{s}_t(\mathbf{x}) + \frac{1}{2}g(t)^2\Big(\mathrm{tr}\big(\mathbf{H}_t(\mathbf{x})\big) + \|\mathbf{s}_t(\mathbf{x})\|_2^2\Big). \tag{36}$$

Taking the spatial gradient yields

$$\partial_t \mathbf{s}_t = -\nabla_{\mathbf{x}}(\nabla_{\mathbf{x}}\cdot \mathbf{f}) - (\nabla_{\mathbf{x}}\mathbf{f})^\top \mathbf{s}_t - \mathbf{H}_t\mathbf{f} + \frac{1}{2}g(t)^2\nabla_{\mathbf{x}}\mathrm{tr}(\mathbf{H}_t) + g(t)^2\mathbf{H}_t\mathbf{s}_t. \tag{37}$$

This is the exact score evolution law used in the main text.

The identity

$$\mathbf{H}_t(\mathbf{x})\mathbf{s}_t(\mathbf{x}) = \nabla_{\mathbf{x}}\left(\frac{1}{2}\|\mathbf{s}_t(\mathbf{x})\|_2^2\right) \tag{38}$$

shows that the curvature–score coupling term is the gradient of score energy. Hence it measures how sharply the attraction strength changes across nearby points.

B.3 Conditional–unconditional differential score dynamics

Let $\mathbf{s}_{c,t}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x} \mid c)$ and $\mathbf{s}_{\emptyset,t}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{x})$. Because the forward diffusion process is shared, both branches obey Eq. (37) with the same drift $\mathbf{f}$ and diffusion coefficient g. Subtracting the two equations gives

$$\begin{aligned}\partial_t \mathbf{s}_{c,t} - \partial_t \mathbf{s}_{\emptyset,t} = &-(\nabla_{\mathbf{x}}\mathbf{f})^\top\big(\mathbf{s}_{c,t} - \mathbf{s}_{\emptyset,t}\big) - \big(\mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t}\big)\mathbf{f} \\ &+ \frac{1}{2}g(t)^2\nabla_{\mathbf{x}}\Big(\mathrm{tr}(\mathbf{H}_{c,t}) - \mathrm{tr}(\mathbf{H}_{\emptyset,t})\Big) \\ &+ g(t)^2\Big(\mathbf{H}_{c,t}\mathbf{s}_{c,t} - \mathbf{H}_{\emptyset,t}\mathbf{s}_{\emptyset,t}\Big).\end{aligned} \tag{39}$$

This decomposition clarifies why conditional–unconditional subtraction is the correct signal isolation mechanism for memorization detection. The unconditional branch captures the baseline local concentration dynamics of the model, whereas the conditional branch contains the additional attraction induced by the prompt. Their difference removes common background dynamics and preserves the prompt-specific discrepancy in score evolution. In locally stiff, score-dominated regions, the leading differential geometric contribution is therefore

$$\partial_t \mathbf{s}_{c,t} - \partial_t \mathbf{s}_{\emptyset,t} \approx g(t)^2\Big(\mathbf{H}_{c,t}\mathbf{s}_{c,t} - \mathbf{H}_{\emptyset,t}\mathbf{s}_{\emptyset,t}\Big). \tag{40}$$

B.4 Local quadratic basin analysis

The local quadratic model used below is also naturally motivated by the forward noising construction. Let $q_t(\mathbf{x} \mid \mathbf{x}_0)$ denote the forward diffusion kernel and let p_{data} be the data distribution. Then the noisy marginal satisfies

$$p_t(\mathbf{x}) = \int q_t(\mathbf{x} \mid \mathbf{x}_0)\, p_{\mathrm{data}}(\mathbf{x}_0)\, d\mathbf{x}_0. \tag{41}$$

Under the empirical discrete approximation $p_{\mathrm{data}}(\mathbf{x}_0) \approx \sum_{i=1}^N \pi_i\, \delta(\mathbf{x}_0 - \mathbf{x}_i)$ and the standard isotropic forward kernel $q_t(\mathbf{x} \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}; \sqrt{\bar\alpha_t}\,\mathbf{x}_0, \sigma_t^2\mathbf{I})$ with $\sigma_t^2 = 1 - \bar\alpha_t$, this becomes

$$p_t(\mathbf{x}) \approx \sum_{i=1}^N \pi_i\, \mathcal{N}\big(\mathbf{x}; \sqrt{\bar\alpha_t}\,\mathbf{x}_i, \sigma_t^2\mathbf{I}\big). \tag{42}$$

Thus each training sample induces a Gaussian kernel component in noisy space. Let $i^\star$ denote the locally dominant mixture-component index. If that component dominates the local posterior responsibility, namely

$$w_{i^\star}(\mathbf{x},t) := \frac{\pi_{i^\star} q_t(\mathbf{x} \mid \mathbf{x}_{i^\star})}{\sum_{i=1}^N \pi_i q_t(\mathbf{x} \mid \mathbf{x}_i)} \approx 1, \tag{43}$$

then

$$\log p_t(\mathbf{x}) = \log q_t(\mathbf{x} \mid \mathbf{x}_{i^\star}) + \log\big(1 + \varepsilon_t(\mathbf{x})\big) + C'_t, \qquad |\varepsilon_t(\mathbf{x})| \ll 1, \tag{44}$$

so, up to higher-order corrections from competing components,

$$\log p_t(\mathbf{x}) \approx C_t - \frac{1}{2\sigma_t^2} \left\|\mathbf{x} - \sqrt{\bar{\alpha}_t}\, \mathbf{x}_{i^\star}\right\|_2^2. \tag{45}$$

In this isotropic single-component limit,

$$\mathbf{H}_t(\mathbf{x}) \approx -\sigma_t^{-2}\mathbf{I}, \qquad \nabla_{\mathbf{x}} \mathrm{tr}\big(\mathbf{H}_t(\mathbf{x})\big) \approx \mathbf{0}. \tag{46}$$

More generally, in latent or anisotropic neighborhoods, a Laplace approximation around the locally dominant component, or a tight cluster of nearby components, yields the local quadratic model

$$\log p_t(\mathbf{x}) \approx C_t - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_t)^\top \mathbf{A}_t (\mathbf{x} - \boldsymbol{\mu}_t), \qquad \mathbf{A}_t \succeq 0. \tag{47}$$

When several mixture components compete, the responsibilities $w_i(\mathbf{x}, t)$ vary across space, $\mathbf{H}_t$ is no longer exactly constant, and $\nabla_{\mathbf{x}} \mathrm{tr}(\mathbf{H}_t)$ captures the resulting non-Gaussian correction. Then

$$\mathbf{s}_t(\mathbf{x}) \approx -\mathbf{A}_t(\mathbf{x} - \boldsymbol{\mu}_t), \qquad \mathbf{H}_t(\mathbf{x}) \approx -\mathbf{A}_t, \tag{48}$$

and hence

$$\mathbf{H}_t(\mathbf{x})\mathbf{s}_t(\mathbf{x}) \approx \mathbf{A}_t^2(\mathbf{x} - \boldsymbol{\mu}_t). \tag{49}$$

If $\mathbf{A}_t = \mathbf{U}_t \boldsymbol{\Lambda}_t \mathbf{U}_t^\top$ with eigenvalues $\{\kappa_{j,t}\}$, then

$$\big(\mathbf{H}_t \mathbf{s}_t\big)_j \approx \kappa_{j,t}^2 z_j. \tag{50}$$

Thus the leading geometric signal scales with curvature squared and is therefore especially sensitive to narrow, selective basins.

B.5 Posterior-responsibility view of early $\hat{x}_0$

We give a simple Gaussian-mixture calculation explaining why an early $\hat{x}_0$ prediction can already look structured for memorized prompts. Consider the empirical forward-noised marginal

$$p_{b,t}(x) = \sum_i \pi_{b,i} \mathcal{N}(x; \sqrt{\bar{\alpha}_t} x_i, \sigma_t^2 I), \qquad \sigma_t^2 = 1 - \bar{\alpha}_t, \tag{51}$$

where $b \in \{c, \emptyset\}$ denotes the conditional or unconditional branch. The branch-dependent weights $\pi_{b,i}$ idealize how conditioning changes the effective clean components considered by the model.

Define the posterior responsibility

$$w_{b,i}(x,t) = \frac{\pi_{b,i} \mathcal{N}(x; \sqrt{\bar{\alpha}_t} x_i, \sigma_t^2 I)}{\sum_j \pi_{b,j} \mathcal{N}(x; \sqrt{\bar{\alpha}_t} x_j, \sigma_t^2 I)}. \tag{52}$$

Then the score of this mixture is

$$s_{b,t}(x) = \nabla_x \log p_{b,t}(x) = \frac{\sqrt{\bar{\alpha}_t} \sum_i w_{b,i}(x,t) x_i - x}{\sigma_t^2}. \tag{53}$$

Let

$$\bar{x}_b(x,t) := \sum_i w_{b,i}(x,t) x_i. \tag{54}$$

Then

$$s_{b,t}(x) = \frac{\sqrt{\bar{\alpha}_t} \bar{x}_b(x,t) - x}{\sigma_t^2}. \tag{55}$$

Using the standard score–noise relation $\epsilon_\theta(x_t, b) \approx -\sigma_t s_{b,t}(x_t)$, the DDIM clean estimate satisfies

$$\hat{x}_{0,b} = \frac{x_t - \sigma_t \epsilon_\theta(x_t, b)}{\sqrt{\bar{\alpha}_t}} \tag{56}$$

$$\approx \frac{x_t + \sigma_t^2 s_{b,t}(x_t)}{\sqrt{\bar{\alpha}_t}} \tag{57}$$

$$= \bar{x}_b(x_t, t). \tag{58}$$

Thus, in this idealized mixture model, the early $\hat{x}_0$ prediction is a posterior-responsibility-weighted clean estimate.

This interpretation clarifies the qualitative visualization in Figure 3. At high noise, the latent x_t itself remains diffuse. However, for a memorized prompt, the conditional branch can place most of its effective posterior responsibility on one or a few clean components, making $\bar{x}_c(x_t, t)$ structured even at the earliest reverse step. In contrast, the unconditional branch averages over a broader set of components, and generalized prompts tend to produce less concentrated posterior clean estimates.

The same calculation also connects to DSM. Since

$$s_{c,t}(x_t) - s_{\emptyset,t}(x_t) = \frac{\sqrt{\bar{\alpha}_t}\,(\bar{x}_c(x_t, t) - \bar{x}_\emptyset(x_t, t))}{\sigma_t^2}, \tag{59}$$

the early conditional–unconditional score discrepancy is controlled by the difference between their posterior clean estimates. DSM measures the pathwise finite difference of this discrepancy across adjacent reverse steps. Therefore, a rapidly concentrating conditional posterior for a memorized prompt can produce a large early DSM signal even before the noisy latent itself has become visually meaningful.

To make precise what DSM adds beyond the instantaneous posterior gap in Eq. (59), define

$$\Delta s_t := s_{c,t}(x_t) - s_{\emptyset,t}(x_t) = \gamma_t\big(\bar{x}_c(x_t, t) - \bar{x}_\emptyset(x_t, t)\big), \qquad \gamma_t := \frac{\sqrt{\bar{\alpha}_t}}{\sigma_t^2}. \tag{60}$$

Applying the material derivative along the shared guided trajectory, with $\mathbf{v}_{\mathrm{cfg},t} := \dot{x}_t$, gives

$$\begin{aligned} D_t \Delta s_t = \underbrace{\dot{\gamma}_t\big(\bar{x}_c - \bar{x}_\emptyset\big)}_{\text{schedule rescaling}} + \gamma_t \underbrace{\partial_t\big(\bar{x}_c - \bar{x}_\emptyset\big)}_{\text{responsibility dynamics}} \\ + \gamma_t \underbrace{\big(\nabla_x \bar{x}_c - \nabla_x \bar{x}_\emptyset\big)\mathbf{v}_{\mathrm{cfg},t}}_{\text{transport}}. \end{aligned} \tag{61}$$

Here and below, all branch-dependent quantities are evaluated at (x_t, t). The first term is simply

$$\dot{\gamma}_t\big(\bar{x}_c - \bar{x}_\emptyset\big) = \frac{\dot{\gamma}_t}{\gamma_t}\Delta s_t, \tag{62}$$

so at any fixed evaluation time it is only a known schedule-dependent rescaling of the same instantaneous posterior gap already used by Wen-style score-difference detectors.

The second term isolates the temporal evolution of the posterior clean estimates at a fixed state. In the Gaussian-mixture picture, this term is induced by the time variation of the responsibilities $w_{b,i}$; equivalently, it is governed by the corresponding responsibility-weighted covariance structure and therefore captures posterior sharpening over reverse time.

The third term is the key new contribution. Differentiating Eq. (55) with respect to x gives

$$\mathbf{H}_{b,t}(x) := \nabla_x s_{b,t}(x) = \frac{\sqrt{\bar{\alpha}_t}\,\nabla_x \bar{x}_b(x, t) - I}{\sigma_t^2}, \tag{63}$$

and subtracting the two branches yields

$$\gamma_t\big(\nabla_x \bar{x}_c - \nabla_x \bar{x}_\emptyset\big) = \mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t} =: \Delta\mathbf{H}_t. \tag{64}$$

Therefore the transport term in Eq. (61) is exactly

$$\gamma_t\big(\nabla_x \bar{x}_c - \nabla_x \bar{x}_\emptyset\big)\mathbf{v}_{\mathrm{cfg},t} = \Delta\mathbf{H}_t\,\mathbf{v}_{\mathrm{cfg},t}, \tag{65}$$

which is precisely the mixture-model form of the branch-differential transport term in Eq. (16).

This decomposition clarifies the relation between Wen and DSM. Equation (59) explains why the instantaneous score gap already works: it measures an instantaneous posterior clean-estimate gap. DSM then applies a material derivative to that quantity. The trivial schedule rescaling contributes no new information, whereas the additional signal comes from the temporal responsibility dynamics and, especially, from the transport term. Through the actual trajectory displacement, this transport term implicitly probes the conditional–unconditional Hessian gap, which is inaccessible to any single-point instantaneous measurement.

B.6 Pathwise finite differences and the Eulerian–Lagrangian relation

The main text evaluates a trajectory-level finite difference. Let $\mathbf{x}_{t_k}$ be the sampled reverse trajectory. For $b \in \{c, \emptyset\}$, define

$$\widehat{D_t \mathbf{s}_c}(t_k) := \frac{\mathbf{s}_{c,t_k}(\mathbf{x}_{t_k}) - \mathbf{s}_{c,t_{k+1}}(\mathbf{x}_{t_{k+1}})}{|t_k - t_{k+1}|}, \tag{66}$$

$$\widehat{D_t \mathbf{s}_\emptyset}(t_k) := \frac{\mathbf{s}_{\emptyset,t_k}(\mathbf{x}_{t_k}) - \mathbf{s}_{\emptyset,t_{k+1}}(\mathbf{x}_{t_{k+1}})}{|t_k - t_{k+1}|}. \tag{67}$$

This quantity approximates the score evolution along the shared trajectory. If $\mathbf{v}_t$ denotes the local reverse velocity, then the corresponding material derivative satisfies

$$D_t \mathbf{s}_t(\mathbf{x}_t) := \frac{d}{dt}\mathbf{s}_t(\mathbf{x}_t) = \partial_t \mathbf{s}_t(\mathbf{x}_t) + \mathbf{H}_t(\mathbf{x}_t)\mathbf{v}_t(\mathbf{x}_t). \tag{68}$$

Thus the trajectory-level finite difference is a discrete estimator of the material derivative rather than a direct estimator of the Eulerian partial derivative. In our classifier-free guidance setting, the conditional and unconditional scores are evaluated on the same guided sampled state sequence $\mathbf{x}_{t_k}$ generated by a shared reverse velocity field $\mathbf{v}_{\mathrm{cfg},t}$. Their differential estimator therefore satisfies

$$\widehat{D_t \mathbf{s}_c}(t_k) - \widehat{D_t \mathbf{s}_\emptyset}(t_k) \approx \big(\partial_t \mathbf{s}_{c,t} - \partial_t \mathbf{s}_{\emptyset,t}\big) + \big(\mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t}\big)\mathbf{v}_{\mathrm{cfg},t}, \tag{69}$$

up to discretization error. Thus the transport mismatch is confined to a single branch-differential curvature term.

B.7 Transport term as curvature-weighted velocity response

Within a shared local quadratic basin, it is useful to compute explicitly three related quantities: the Jeon vector, the direct curvature–score discrepancy, and the transport response.

Fix a time t along the shared guided reverse trajectory and let

$$\mathbf{z}_t := \mathbf{x}_t - \boldsymbol{\mu}_t$$

denote the local displacement from a shared basin center $\boldsymbol{\mu}_t$. Assume that, over the neighborhood relevant to the detector, both branches admit local quadratic approximations

$$\log p_b(\mathbf{x}, t) \approx C_{b,t} - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_t)^\top \mathbf{K}_{b,t}(\mathbf{x} - \boldsymbol{\mu}_t), \qquad b \in \{c, \emptyset\}, \tag{70}$$

with $\mathbf{K}_{c,t}, \mathbf{K}_{\emptyset,t} \succeq 0$. Define the local curvature gap

$$\Delta\mathbf{K}_t := \mathbf{K}_{c,t} - \mathbf{K}_{\emptyset,t}. \tag{71}$$

Then the corresponding local scores and Hessians satisfy

$$\mathbf{s}_{b,t} \approx -\mathbf{K}_{b,t}\mathbf{z}_t, \qquad \mathbf{H}_{b,t} \approx -\mathbf{K}_{b,t}, \qquad b \in \{c, \emptyset\}, \tag{72}$$

and therefore

$$\Delta\mathbf{s}_t := \mathbf{s}_{c,t} - \mathbf{s}_{\emptyset,t} \approx -\Delta\mathbf{K}_t\mathbf{z}_t, \qquad \Delta\mathbf{H}_t := \mathbf{H}_{c,t} - \mathbf{H}_{\emptyset,t} \approx -\Delta\mathbf{K}_t. \tag{73}$$

To compare the direct and pathwise quantities in the same local model, we write

$$\mathbf{J}_t := \Delta\mathbf{H}_t\, \Delta\mathbf{s}_t. \tag{74}$$

This vector appears in the Hessian-enhanced detector of Jeon et al. (2025), which evaluates $\|\mathbf{J}_t\|_2^2$. We retain its vector form to analyze the local quadratic structure.

Proposition B.1 (Shared curvature-gap expansion in a local quadratic basin)**.** *Assume Eq.* (70)*. Let the score-driven part of the guided motion be*

$$\mathbf{v}_t^{\mathrm{sd}} := \dot{\mathbf{x}}_t - \mathbf{f}_t \approx -\lambda_t\big(\mathbf{s}_{\emptyset,t} + w\,\Delta\mathbf{s}_t\big), \tag{75}$$

where $w \geq 0$ is the CFG scale and $\lambda_t \geq 0$ absorbs the scheduler-dependent scalar factor. Define the direct curvature–score discrepancy and the transport term by

$$\mathbf{a}_t := \mathbf{H}_{c,t}\mathbf{s}_{c,t} - \mathbf{H}_{\emptyset,t}\mathbf{s}_{\emptyset,t}, \qquad \mathbf{T}_t^{\mathrm{tr}} := \Delta\mathbf{H}_t\, \mathbf{v}_t^{\mathrm{sd}}, \tag{76}$$

and write $\{\mathbf{A}, \mathbf{B}\} := \mathbf{AB} + \mathbf{BA}$ *for the anticommutator. Then*

$$\mathbf{J}_t \approx \Delta\mathbf{K}_t^2 \mathbf{z}_t, \tag{77}$$

$$\mathbf{a}_t \approx \big(\mathbf{K}_{c,t}^2 - \mathbf{K}_{\emptyset,t}^2\big)\mathbf{z}_t = \big(\{\mathbf{K}_{\emptyset,t}, \Delta\mathbf{K}_t\} + \Delta\mathbf{K}_t^2\big)\mathbf{z}_t, \tag{78}$$

$$\begin{aligned}\mathbf{T}_t^{\mathrm{tr}} &\approx -\lambda_t \Delta\mathbf{K}_t\big(\mathbf{K}_{\emptyset,t} + w\,\Delta\mathbf{K}_t\big)\mathbf{z}_t \\ &= -\lambda_t w\,\mathbf{J}_t - \lambda_t \Delta\mathbf{K}_t\mathbf{K}_{\emptyset,t}\mathbf{z}_t.\end{aligned} \tag{79}$$

Consequently, the Jeon vector, the direct curvature–score discrepancy, and the transport response are all generated by the same local curvature gap $\Delta\mathbf{K}_t$*. Equations* (78) *and* (79) *contain the contributions* $+\mathbf{J}_t$ *and* $-\lambda_t w\mathbf{J}_t$*, respectively, together with background-curvature terms.*

Proof. Equation (77) follows immediately from Eq. (73):

$$\mathbf{J}_t = \Delta\mathbf{H}_t\,\Delta\mathbf{s}_t \approx (-\Delta\mathbf{K}_t)(-\Delta\mathbf{K}_t\mathbf{z}_t) = \Delta\mathbf{K}_t^2\mathbf{z}_t.$$

Likewise,

$$\mathbf{a}_t \approx (-\mathbf{K}_{c,t})(-\mathbf{K}_{c,t}\mathbf{z}_t) - (-\mathbf{K}_{\emptyset,t})(-\mathbf{K}_{\emptyset,t}\mathbf{z}_t) = \big(\mathbf{K}_{c,t}^2 - \mathbf{K}_{\emptyset,t}^2\big)\mathbf{z}_t.$$

Using $\mathbf{K}_{c,t} = \mathbf{K}_{\emptyset,t} + \Delta\mathbf{K}_t$, we obtain the exact algebraic identity

$$\mathbf{K}_{c,t}^2 - \mathbf{K}_{\emptyset,t}^2 = (\mathbf{K}_{\emptyset,t} + \Delta\mathbf{K}_t)^2 - \mathbf{K}_{\emptyset,t}^2 = \{\mathbf{K}_{\emptyset,t}, \Delta\mathbf{K}_t\} + \Delta\mathbf{K}_t^2,$$

which does not require commutativity. Finally, Eq. (75) and Eq. (73) give

$$\mathbf{v}_t^{\mathrm{sd}} \approx -\lambda_t\big(-\mathbf{K}_{\emptyset,t}\mathbf{z}_t - w\,\Delta\mathbf{K}_t\mathbf{z}_t\big) = \lambda_t\big(\mathbf{K}_{\emptyset,t} + w\,\Delta\mathbf{K}_t\big)\mathbf{z}_t.$$

Multiplying by $\Delta\mathbf{H}_t \approx -\Delta\mathbf{K}_t$ yields

$$\mathbf{T}_t^{\mathrm{tr}} \approx -\lambda_t\Delta\mathbf{K}_t\big(\mathbf{K}_{\emptyset,t} + w\,\Delta\mathbf{K}_t\big)\mathbf{z}_t = -\lambda_t w\,\Delta\mathbf{K}_t^2\mathbf{z}_t - \lambda_t\Delta\mathbf{K}_t\mathbf{K}_{\emptyset,t}\mathbf{z}_t,$$

which is Eq. (79). □

Remark B.2 (Diagonal specialization under a shared eigenbasis). When $\mathbf{K}_{\emptyset,t}$ and $\Delta\mathbf{K}_t$ share an eigenbasis, Proposition B.1 reduces to the diagonal form in Eq. (81)–Eq. (83), recovering the original component-wise intuition. Write

$$\mathbf{K}_{\emptyset,t} = \mathbf{U}_t\,\mathrm{diag}\big(\kappa_{\emptyset,1,t}, \ldots, \kappa_{\emptyset,d,t}\big)\mathbf{U}_t^\top, \qquad \Delta\mathbf{K}_t = \mathbf{U}_t\,\mathrm{diag}\big(\delta_{1,t}, \ldots, \delta_{d,t}\big)\mathbf{U}_t^\top, \tag{80}$$

define eigencoordinates $\tilde{\mathbf{z}}_t := \mathbf{U}_t^\top\mathbf{z}_t$ with components $\tilde{z}_{j,t}$, and set $\kappa_{c,j,t} := \kappa_{\emptyset,j,t} + \delta_{j,t}$. Then

$$(\mathbf{J}_t)_j \approx \delta_{j,t}^2\tilde{z}_{j,t}, \tag{81}$$

$$(\mathbf{a}_t)_j \approx \delta_{j,t}\big(\kappa_{c,j,t} + \kappa_{\emptyset,j,t}\big)\tilde{z}_{j,t} = \delta_{j,t}\big(2\kappa_{\emptyset,j,t} + \delta_{j,t}\big)\tilde{z}_{j,t}, \tag{82}$$

$$(\mathbf{T}_t^{\mathrm{tr}})_j \approx -\lambda_t\delta_{j,t}\big(\kappa_{\emptyset,j,t} + w\,\delta_{j,t}\big)\tilde{z}_{j,t}. \tag{83}$$

Combining the direct and transport terms gives

$$\begin{aligned}g(t)^2\mathbf{a}_t + \mathbf{T}_t^{\mathrm{tr}} \approx \big[&g(t)^2\{\mathbf{K}_{\emptyset,t}, \Delta\mathbf{K}_t\} - \lambda_t\Delta\mathbf{K}_t\mathbf{K}_{\emptyset,t} \\ &+ \big(g(t)^2 - \lambda_t w\big)\Delta\mathbf{K}_t^2\big]\mathbf{z}_t.\end{aligned}$$

The local quadratic model therefore identifies a common curvature-gap dependence in these quantities. Their combined response also depends on background curvature and the coefficient $g(t)^2 - \lambda_t w$. The full DSM response includes this vector together with the remaining terms in the evolution law along the sampling trajectory.

C Equivalence of Notations and Metric Definitions

This section relates the score-gap notation to the noise-residual metric of Wen et al. (2024). At a fixed diffusion time, the two norms differ by a positive scalar and induce the same ranking.

C.1 Connection to Wen et al.

The detection metric proposed by Wen et al. (2024) quantifies the magnitude of the conditional noise residual. Formally, it is defined as the time-averaged L_2 norm of the difference between conditional and unconditional noise predictions:

$$\mathcal{M}_{\text{Wen}} = \frac{1}{T}\sum_{t=1}^{T} \|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \mathbf{c}) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \emptyset)\|_2 \,. \tag{84}$$

Here, we rigorously derive its relationship to the score function $\mathbf{s}_\theta(\mathbf{x}_t, t)$.

Recall that in the standard Denoising Diffusion Probabilistic Models (DDPM) framework (Ho et al., 2020), the forward diffusion process $q_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$ is a Gaussian given by $\mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$. The score of this conditional distribution is:

$$\nabla_{\mathbf{x}_t} \log q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = -\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{1-\bar{\alpha}_t}. \tag{85}$$

Leveraging the reparameterization trick $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we can express the residual as $\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0 = \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. This yields the fundamental mapping between the score vector and the noise vector:

$$\mathbf{s}(\mathbf{x}_t, t) \coloneqq \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) \approx -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{1-\bar{\alpha}_t}}. \tag{86}$$

Using Eq. (86), the instantaneous score-gap norm satisfies

$$\begin{aligned} &\|\mathbf{s}_\theta(\mathbf{x}_t, \mathbf{c}) - \mathbf{s}_\theta(\mathbf{x}_t, \emptyset)\| \\ &= \left\| -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \mathbf{c})}{\sqrt{1-\bar{\alpha}_t}} - \left(-\frac{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \emptyset)}{\sqrt{1-\bar{\alpha}_t}}\right) \right\| \\ &= \underbrace{\frac{1}{\sqrt{1-\bar{\alpha}_t}}}_{\text{Time-dependent Scalar}} \cdot \underbrace{\|\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \mathbf{c}) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \emptyset)\|}_{\text{Wen et al. Metric (Instantaneous)}}. \end{aligned} \tag{87}$$

D Experimental Setup

Datasets and Protocols. We evaluate memorization detection on Stable Diffusion v1.4, Stable Diffusion v1.5, and Stable Diffusion v2.0-base. To ensure comparability with prior detectors, we use the public prompt-level benchmarks adopted in Jeon et al. (2025) and Wen et al. (2024), while explicitly distinguishing the model-matched and cross-checkpoint settings. The Webster SD-v1 extraction setting corresponds to Stable Diffusion v1.5, and the Webster SD-v2 benchmark corresponds to Stable Diffusion v2.0-base. Thus, our SD 1.5 and SD 2.0-base experiments are aligned with the model variants targeted by the original extraction benchmark. We additionally report SD 1.4 results as a cross-checkpoint evaluation on the same SD-v1 prompt split, following the practice of subsequent memorization-detection studies. We use two distinct evaluation sets:

- **Memorized Set (D_{mem}):** For SD-v1 models, we use the standard suite of 500 Webster memorized prompts identified by Webster (2023). These prompts are evaluated on SD 1.5 as the model-matched SD-v1 setting and on SD 1.4 as a cross-checkpoint setting. For SD 2.0-base, we use the corresponding Webster SDV2 memorized set of 219 prompts. Webster annotates extracted prompts as matching verbatim, template verbatim, or retrieval verbatim; since our goal is binary memorization detection, these categories are collapsed into a single memorized class. These prompts induce training-image regurgitation ranging from exact replication to partial, template-level, or retrieval-based reproduction.
- **Non-Memorized Set (D_{gen}):** To evaluate detection robustness against diverse prompt distributions, we adopt the composite dataset used in Jeon et al. (2025). This set comprises 500 prompts sampled from four distinct sources: (1) **COCO** (Lin et al., 2014) for standard natural images, (2) **Lexica** (Lexica, 2024) for artistic styles, (3) **Tuxemon** (Hugging Face, 2024) for out-of-distribution (OOD) concept checks, and (4) a subset of **GPT-4** generated captions from Ren et al. (2024).

This multi-source negative set reduces source-specific caption bias and provides a diverse assessment of false positive behavior. All compared detectors are evaluated on the same prompt splits, checkpoints, and random seeds.

Implementation Details. We employ the DDIM sampler (Song et al., 2020a) with 50 inference steps and a guidance scale of $w = 7.5$. Our global default dtype is float16, so the main UNet forward pass is executed predominantly in f16. For the second-order geometric computations used by $\mathcal{M}_{\mathrm{Jeon}}$, we promote the critical JVP operators to fp32. We evaluate all methods in a **single-shot setting** ($N = 1$): for each prompt and each seed, we generate exactly one image trajectory and evaluate every detector on that shared sample. To reduce seed variance without changing the single-shot protocol, we repeat this procedure over five shared seeds and report mean $\pm$ standard deviation across seeds. In particular, all compared methods use the same prompt splits, checkpoints, and per-prompt seeds.

Additional Protocol Notes. For the anisotropy baseline $\mathcal{M}_{\mathrm{Ani}}$ of Asthana and Belagiannis (2026), we retain the paper's version-dependent hyperparameters: $(\gamma_1, \gamma_2) = (1.0, 2.0)$ for SD-v1 models and $(\gamma_1, \gamma_2) = (1.0, 0.1)$ for SD-v2 models. Following the earliest-evaluable-time protocol, $\mathcal{M}_{\mathrm{Wen}}$ (Wen et al., 2024), $\mathcal{M}_{\mathrm{Jeon}}$ (Jeon et al., 2025), and DSM are all measured at the earliest reverse-time location where each metric is well defined; for DSM, which is based on backward finite differences, this corresponds to the first reverse-step transition $(0 \to 1)$.

Evaluation Metrics. We evaluate and compare four detectors under a unified notation:

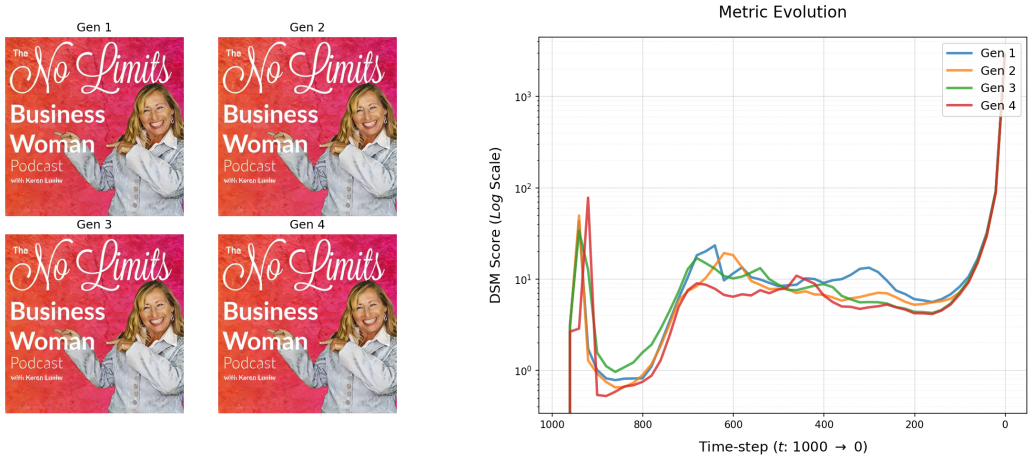
- $\mathcal{M}_{\mathrm{Wen}}$**:** The efficient score-difference baseline of Wen et al. (2024).
- $\mathcal{M}_{\mathrm{Jeon}}$**:** The geometric baseline of Jeon et al. (2025).
- $\mathcal{M}_{\mathrm{Ani}}$**:** The anisotropy-based baseline of Asthana and Belagiannis (2026).
- $\mathcal{M}_{\mathrm{DSM}}$**:** The dynamical singularity metric defined in Eq. (17), evaluated through backward finite differences at the early sampling stage.

D.1 Qualitative Analysis: Trajectories During Reverse Diffusion

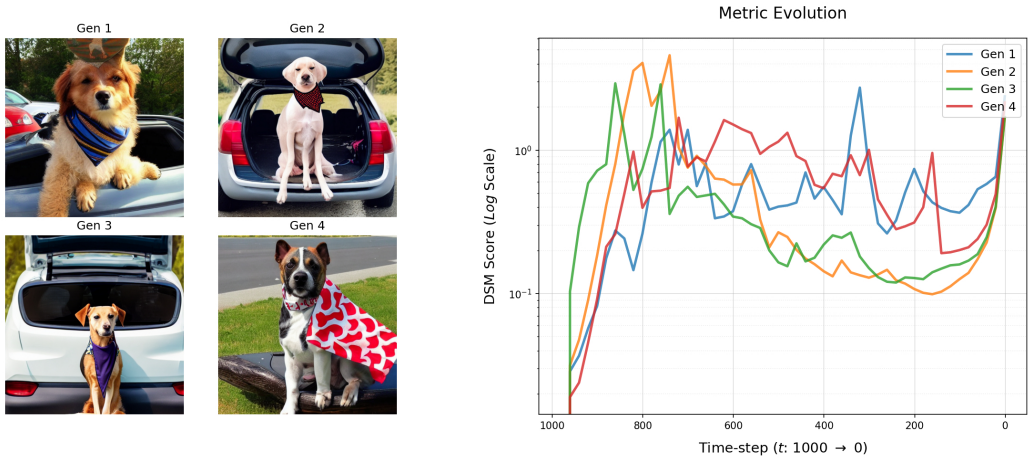
We examine the evolution of the DSM score-variation signal over the reverse diffusion process ($t = T \to 0$) for a representative memorized prompt ("The No Limits Business Woman Podcast") and a representative generalized prompt ("A dog sitting in a car").

Observation 1: Early Stage. Memorized samples exhibit a sharp early spike in the DSM score at the earliest stages of generation ($t \approx T$). This behavior indicates that the reverse trajectory quickly enters a locally stiff region of the probability landscape—the memorization basin—where the vector field changes abruptly and steers the sample toward a memorized attractor.

Observation 2: Trajectory Collapse. For the memorized prompt, generations from four random seeds are visually identical and yield tightly aligned metric trajectories. In contrast, the generalized prompt produces diverse images and noisier, more dispersed trajectories with no comparable peak. This contrast supports our view that memorization in diffusion models is fundamentally a dynamical phenomenon: the reverse process rapidly collapses onto a shared high-curvature trajectory near initialization.



(a) **Top:** "The No Limits Business Woman Podcast" (memorized). Across four random seeds, DSM exhibits a sharp early spike and tightly aligned trajectories.



(b) **Bottom:** "A dog sitting in a car" (generalized). Trajectories remain noisier and more dispersed, without a singular early-time spike.

Figure 4: **Trajectories during reverse diffusion.** Evolution of the DSM score-variation signal over reverse generation for a representative memorized prompt and a representative generalized prompt, each plotted across four random seeds. Memorized trajectories collapse rapidly into a shared high-response pattern, whereas generalized trajectories remain diverse and comparatively smooth.

E Extended Visual Evidence: Instantaneous Attractor Locking

To further illustrate the universality of **instantaneous attractor locking**, we provide additional visualizations of the estimated clean image $\hat{\mathbf{x}}_0$ at the very first sampling step ($t \approx T$). These images are obtained using the DDIM inversion formula (Song et al., 2020a):

$$\hat{\mathbf{x}}_0 = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, \mathbf{c})}{\sqrt{\bar{\alpha}_t}}. \tag{88}$$

Figure 5 presents examples from diverse semantic categories.

Observations:

- **Cross-domain consistency (Row 1):** The locking phenomenon is not confined to a particular category. Across prompts involving commercial products, wildlife photography, and religious art, memorized samples already display coherent global structure at Step 1. This suggests that the attractor captures the trajectory independently of semantic domain.
- **Contrast with generalized prompts (Row 2):** The bottom row provides a direct comparison. Whereas memorized samples are already well formed, their generalized counterparts remain diffuse and visually incoherent at the same step. This visual separation is consistent with DSM's ability to distinguish memorization from generalization at initialization.



(a) Mem: "The No Limits Business Woman Podcast"

(b) Mem: "Full body U-Zip main opening - Full body U-Zip main opening on front of bag for easy unloading when you get to camp"

(c) Mem: "Mothers influence on her young hippo"

(d) Mem: "Talks on the Precepts and Buddhist Ethics"



(e) Gen: "A small kitchen with some quirky art decorations"

(f) Gen: "A dog wearing a bandanna and sitting on the trunk of a car"

(g) Gen: "A small kitchen contains a refrigerator and small counter"

(h) Gen: "A small refrigerator in a small kitchen with a window"

Figure 5: **Extended visual evidence for instantaneous attractor locking across categories.** Decoded $\hat{\mathbf{x}}_0$ at the first reverse step ($t \approx T$). **Row 1 (Memorized):** Samples already display coherent global structure, including legible text and recognizable faces, indicating that the trajectory has entered a stiff attractor. **Row 2 (Generalized):** Samples remain largely incoherent, reflecting a more diffuse and uncommitted regime.

F Memorization Mitigation

We use the first-transition score-gap change to guide initial-noise search and optimization while keeping the model and prompt fixed. For each prompt, we draw eight Gaussian noise candidates and minimize

$$\mathcal{L}(z) = \|\Delta s_{t_1}(x_1(z)) - \Delta s_{t_0}(z)\|_2 + 0.05\|z\|_2,$$

where $x_1(z)$ is the latent after the first DDIM transition. We use Adam with learning rate 0.05 for at most ten updates and constrain the root-mean-square displacement from each initial candidate to 0.1. We select the candidate with the lowest objective and stop early when its objective is at most 7.0.

We evaluate this procedure on all 500 memorized prompts for SD1.4 with 50-step DDIM sampling, CFG scale 7.5, and seed 42. Standard generation with the same sampling settings serves as the baseline. SSCD similarity to the training reference is evaluated on the 347 prompts with available reference images, and CLIP text–image alignment is evaluated on all 500 prompts.

Table 5: DSM-guided mitigation on SD1.4. SSCD uses the 347 prompts with available training reference images; CLIP uses all 500 prompts.

Metric	Before mitigation	Ours
Outputs with SSCD ≥ 0.5	44	8
Mean SSCD	0.1705	0.0970
Mean CLIP alignment	0.3148	0.3121

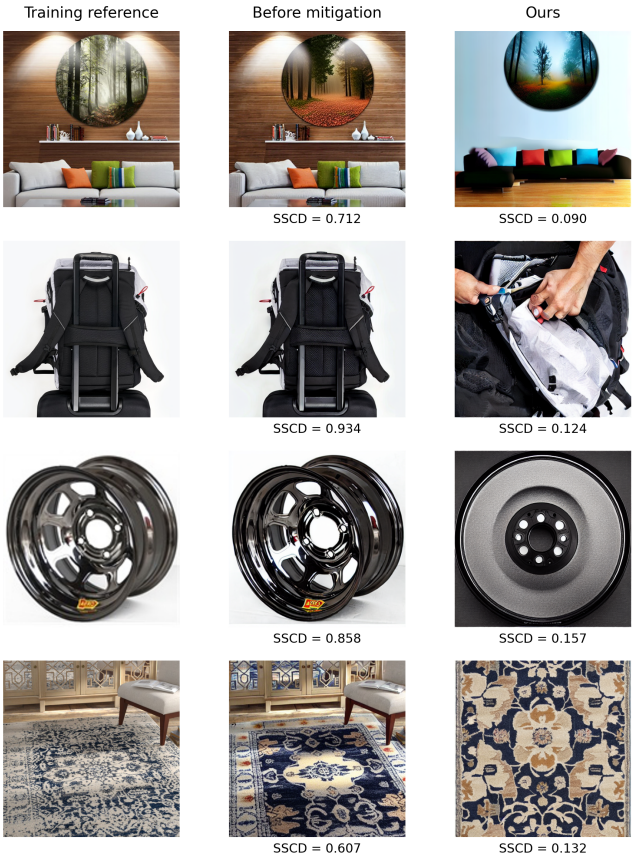


Figure 6: Selected examples of DSM-guided mitigation. Each row shows a training reference image, the standard generation, and the output after initial-noise search and optimization. SSCD similarity to the reference is reported below each generated image.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's core claim that DSM is an efficient trajectory-level memorization detector motivated by score-evolution dynamics, and these claims are evaluated in the main experiments and discussed again in the conclusion.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in the "Robustness Analysis and Limitations" subsection.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The main text states the assumptions, and the appendix provides extended derivations, formal identities, and proofs for the main theoretical relationships used in the paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies the compared models, prompt sets, fixed-seed protocol, shared implementation paths, evaluation metrics, precision choice, and additional setup details in the appendix, which together describe how the main tables were obtained.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code is available in the supplementary material and at github.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental section and appendix describe the models, datasets, prompt composition, evaluation protocol, early-step measurement choice, and numerical precision, which are the details needed to interpret the reported results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports point estimates for AUC, TPR@1%FPR, and runtime. We report mean ± standard deviation over five seeds for all main results in Table 1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the GPU type, memory, and detector runtime overhead for the efficiency experiment.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work studies memorization detection for privacy, copyright, and safety auditing, does not involve human subjects, and does not describe any methodology that intentionally violates the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: The paper motivates memorization detection through its benefits for privacy and copyright auditing, but does not explicitly discuss potential negative societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper proposes a detection metric and does not release a new image generator, scraped dataset, or other high-risk generative asset as part of the current submission.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use publicly available pretrained models, datasets, and prompt sets. We cite the original sources in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The current submission does not package a new dataset, model checkpoint, or other documented asset release alongside the manuscript itself.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or experiments with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve human subjects or participant studies, so IRB-style review is not applicable.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method and experiments do not use LLMs as an essential methodological component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.