Position: Membership Inference Attack Should Move On to Distributional Statistics for Distilled Generative Models

Anonymous Authors[1]

Abstract

To detect unauthorized training data usage in training large-scale generative models, membership inference attacks (MIAs) have proven effective in distinguishing a single training instance (i.e., a member) from a single non-training instance (i.e., a non-member). This success relies on a *memorization effect*: Since models overfit training data, they tend to perform better on a member than a non-member. However, we find that standard MIAs *fail* against *distilled generative models* (i.e., student models) that are usually deployed for efficiency. This is because student models, trained exclusively on data generated by large-scale generative models (i.e., teacher models), lack direct exposure to the teacher's original training data, thereby nullifying the *memorization effect*. This finding reveals *a serious privacy loophole*, where generation service providers could deploy a student model whose teacher was trained on unauthorized data, yet claim the deployed model is "clean" because it was not directly trained on such data. To fix this loophole, we uncover a memory chain that persists: the student's output distribution aligns more with the teacher's members than non-members, making unauthorized data use detectable. This leads us to posit that *MIAs on distilled generative models should shift from instance-level scores to distribution-level statistics*. We further propose three principles of distribution-based MIAs for detecting unauthorized training data through distilled generative models, and validate our position through an exemplar framework. We lastly discuss the implications our position leads to.

1. Introduction

Recent advances in large-scale generative models have set new standards for synthesizing high-quality content across modalities, such as images (Ho et al., 2020) and languages (Brown et al., 2020). However, the extensive datasets required to train these models often contain sensitive information from individuals who may not have explicitly consented to the use of their data for model development (Liu et al., 2024). This concern is particularly pressing given the widespread adoption of *large language models* (LLMs) (Floridi & Chiriatti, 2020) and diffusion models (Ho et al., 2020; Song et al., 2023). In this context, *instance-level membership inference attacks* (I-MIAs) (Carlini et al., 2022), designed to detect whether a single instance is used in training, offer a valuable auditing mechanism to detect unauthorized data usage. The success of I-MIAs mainly relies on a phenomenon in training such large-scale generative models: well-trained models tend to *overfit* to their training set, exhibiting different behaviors between training instances (i.e., called members) and test instances (i.e., called non-members) (Yeom et al., 2018), which is also known as the *memorization effect* (Yeom et al., 2018). It means that I-MIAs often rely on *instance-level scores* to distinguish a member from a non-member.

Motivation. However, recent generative model-based I-MIAs do not consider another important step in deploying powerful large-scale generative models: *distillation*, and, in this paper, we find that existing I-MIAs fail to find members when facing a distilled generative model. Distilled generative models address the critical challenge of efficiently deploying large-scale generative models, whose high computational demands often require access to hundreds or even thousands of GPUs (Hu et al., 2024). Specifically, during distillation, we learn lightweight generative models (a.k.a. student models) with the data *generated* by a large-scale generative model (a.k.a. teacher models). As such, model distillation enables a two-tier deployment strategy for generation service providers: teacher models focus on training student models, while student models serve end-users directly, reducing inference latency and cost. Yet, this deployment strategy introduces a critical limitation for I-MIAs. Since student models are trained *only* on the outputs of teacher models, rather than on their original training data (i.e., members), this setup undermines the memorization effect that I-MIAs depend on (Fig. 1). Namely, existing I-MIAs probably do not work when we can only access the distilled models.

To further verify the above argument, we provide empirical evidence for this security caveat in Sec. 2. We find that

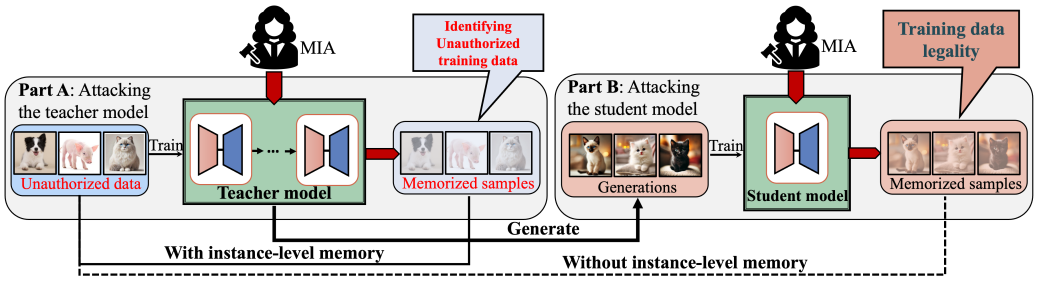

Figure 1. Model distillation raises privacy concerns. MIA can detect unauthorized data in the teacher model but fails when only the student model is available. The reason behind this failure is that the student model is trained with the teacher's outputs only, rather than the original data (i.e., members). More importantly, based on the failure of MIAs here, generation-service providers can only publish the student models as a service, to bypass unauthorized data detection and claim they do not use any unauthorized data for training.

while I-MIAs effectively identify training data in teacher models (Fig. 2(a)), they consistently fail with student models (Fig. 2(b)), implying that student models retain *insufficient membership information at the instance level*. This finding poses *a serious privacy issue*: generation-service providers can only publish the student models as a service, to bypass unauthorized data detection and claim they do not use any unauthorized data for training, causing distilled generative models to naturally have privacy concerns when they are deployed (like the Fig. 1 shows).

To address the issue of existing I-MIAs under model distillation, we discover a *distribution-based memory chain* between a student-teacher pair: student-generated data exhibit a significantly stronger distributional alignment with teacher's members than non-members, making it possible to determine if a student model has knowledge from unauthorized data. Specifically, a consistent statistical pattern–distances to non-member data concentrate at higher values than to member data, suggesting that the student preserves statistical signals exhibiting stronger alignment with the teacher's member distribution than non-member distributions, despite the failure of I-MIAs. Thus, to *reliably* audit the privacy violation of distilled generative models, we posit the following statement in the field of generative models.

> **Position**: Membership Inference Attacks (MIAs) for distilled generative models should shift from *instance-level* scores to *distribution-level* statistics.

Based on the position above, we suggest that auditing upstream privacy violation risks on distilled generative models should be evaluated based on the distribution of data instances rather than individual instances due to the discovered memory chain between a student-teacher pair, namely distribution-level MIA (D-MIA). In Sec. 4, we further establish three principles that D-MIAs should follow to maximize their effectiveness. Following these principles, we build an exemplar framework to illustrate how these principles can be applied in practice (See App. B). We finally discuss the broader implications of privacy regulation and responsible AI deployment that D-MIA leads to, highlighting both opportunities and challenges in the evolving landscape of generative model auditing (See App. E).

2. I-MIA fails on distilled generative models

This section shows that I-MIAs are ineffective in identifying member data of large-scale generative models when access is limited to their distilled student counterparts. For readers unfamiliar with how current I-MIAs attack large-scale generative models and related work, we provide additional background information in App. A.

Distilled models for online generation service. Although large-scale generative models can produce high-quality text and images, the billion-scale parameters of LLMs and iterative denoise steps of diffusion models lead to high inference latency (Touvron et al., 2023; Song et al., 2020), presenting challenges for online deployment of generation services. Knowledge distillation (Hinton, 2015) offers a compelling solution, allowing the creation of smaller "student" models that learn to mimic the output of larger "teacher" models and achieve gains in inference efficiency without compromising the quality of generation (Hsieh et al., 2023; Gu et al., 2023). This trend highlights a deployment shift: lightweight student models serve end-users, while computationally demanding teacher models are confined to offline training. However, state-of-the-art distillation practices implement strict separation: *student models learn exclusively from teacher-generated data, with no direct access to the teacher's training dataset*. We therefore argue that this separation prevents the student from forming instance-level memories of the teacher's members. As a result,

> The memorization effect, which I-MIAs rely on, will not apply to distilled generative models.

Empirical evidence. We thus investigate the impact of distillation on I-MIAs. Fig. 2(b) shows that the student model's reconstruction pattern exhibits no statistically significant differences between member and non-member images. We

Table 1. Average performance of MIAs on three generative models: EDM (teacher), DMD, and Diff-Instruct (students). Metrics include ASR, AUC, and TPR@FPR=0.05. See App. F for detail.

Model	Dataset	ASR	AUC	TPR@FPR=0.05
EDM	CIFAR10	0.596	0.610	0.070
	FFHQ	0.584	0.590	0.083
	AFHQv2	0.704	0.724	0.230
DMD	CIFAR10	0.515	0.509	0.050
	FFHQ	0.515	0.503	0.047
	AFHQv2	0.526	0.520	0.063
Diff-Instruct	CIFAR10	0.508	0.505	0.043
	FFHQ	0.508	0.509	0.050
	AFHQv2	0.509	0.508	0.053

confirm this using four I-MIA methods on a teacher diffusion model EDM (Karras et al., 2022) and its student models DMD (Yin et al., 2024) and Diff-Instruct (Luo et al., 2024). Detailed setup is in App. C. Moreover, Tab. 1 shows that I-MIAs achieve a success rate higher than random guessing when applied to the teacher model, but perform no better than random guessing on student models. Since student models do not directly fit the teacher's member data, they may not preserve the instance-level behavioral signature that I-MIAs typically exploit. Thus, model distillation, primarily developed for efficiency though, provides an inherent defense against major I-MIAs (Shejwalkar & Houmansadr, 2021; Tang et al., 2022), causing I-MIAs to fail against distilled models.

3. Does Distillation Really Eliminate Membership Information?

Although the distilled generative models are trained without seeing any members of the teacher model, their learning process could be influenced by these members: the student trains on data generated by the teacher, who aims to approximate the data distribution from which its members were drawn. We therefore conjecture the following:

> The student, in learning to mimic the teacher's output distribution, might indirectly learn a *distribution* closer to the teacher's *member distribution* than to a *distribution* of non-members.

We call this potential propagation of distributional characteristics the *memory chain* between student and member data, and empirically confirm its existence.

3.1. Distributional signals survive distillation

We first define a distributional membership signal in the context of distillation as the tendency for a collection of samples generated by the student model to inherit the teacher's distributional bias and, in some statistical sense, appear closer to the distribution induced by the member data. The key question now becomes *does a student trained on teacher-generated data indeed preserve such a distributional membership signal?* We test it empirically. Consider three datasets: student-generated $\mathcal{D}^{\text{gen}}$, teacher's member data $\mathcal{D}^{\text{mem}}$ and disjoint non-member data $\mathcal{D}^{\text{non}}$, we evaluate the distance between $\mathcal{D}^{\text{gen}}$ and $\mathcal{D}^{\text{mem}}$ against the distance between $\mathcal{D}^{\text{gen}}$ and $\mathcal{D}^{\text{non}}$ across multiple experimental trials for statistical robustness, randomly sampling subsets $\tilde{\mathcal{D}}^{\text{gen}}$, $\tilde{\mathcal{D}}^{\text{mem}}$, and $\tilde{\mathcal{D}}^{\text{non}}$ from their respective datasets each trial. We adopt *maximum mean discrepancy* (MMD) (Gretton et al., 2012), a kernel-based distance measure between probability distributions, to quantify distributional similarities between paired subsets, namely (i) $\tilde{\mathcal{D}}^{\text{gen}}$ and $\tilde{\mathcal{D}}^{\text{mem}}$, and (ii) $\tilde{\mathcal{D}}^{\text{gen}}$ and $\tilde{\mathcal{D}}^{\text{non}}$. We observe a pattern across repeated trials: The MMD values of (i) cluster at lower magnitudes compared to those of (ii), indicating that the student-generated data aligns more closely with the teacher's member distribution from its non-member counterpart (Fig. 2(c)). This way we confirm that distribution-level statistics (e.g., distribution discrepancy) can identify the teacher's membership information undetected at the instance level, even through a distilled student model. Namely, distributional membership signals survive under the distillation procedure, and the *memory chain* between a student-teacher pair exists.

3.2. Distributional statistics enables reliable membership inference with LLMs

Distributional statistics amplify instance-level membership signals. Ye et al. (2024) show that LLMs exhibit characteristic uncertainty patterns across local neighborhoods of training data, revealing membership information invisible to single-sample analyses. Similarly, Dong et al. (2024) confirm that membership information lies in set-level probability distributions rather than individual confidence scores. Such findings align with our observations: Membership information manifests collectively more obviously than individual instances.

Distributional information alleviates the FPR issue. The shift to distributional analysis also addresses a critical flaw plaguing conventional I-MIAs on LLMs (Zhang et al., 2024). Zhang et al. (2024) and Meeus et al. (2024) show that I-MIAs applied to LLMs often yield an unacceptably high FPR, since computing FPR reliably requires access to "true" non-member data which, model has *never* truly seen, even indirectly. Given that LLMs are trained on massive web-scale corpora (Liu et al., 2024), which may contain paraphrases of virtually any public data, guaranteeing such pristine non-exposure is practically impossible. Distributional approaches, however, are more resilient to this issue by focusing on aggregate statistical properties of data, such as token frequency distributions or specific sampling behaviors. Even if models see scattered references to non-member datasets, such exposure proves insufficient to reproduce the complete statistical signal of a dataset (Choi et al., 2025).

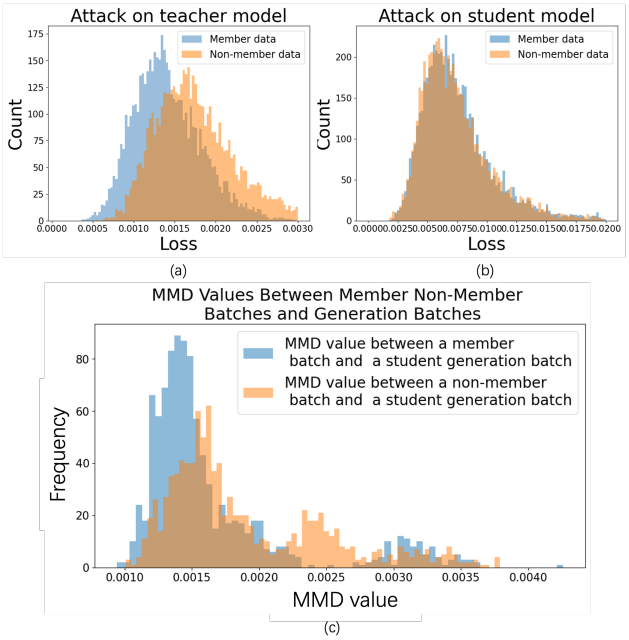

Figure 2. Comparison of I-MIAs on teacher model EDM (Karras et al., 2022) and student model DMD (Yin et al., 2024) using: (a) ReDiffuse (Li et al., 2024) reveals membership signals in EDM via distinct reconstruction and re-noising losses between members and non-members on AFHQv2. (b) Applied to DMD, ReDiffuse fails to separate member from non-member instances. (c) Student outputs show stronger *distributional* alignment with member data when evaluated as instance sets via MMD (Gretton et al., 2012).

4. Membership Inference Attacks Should Move On to Distributional Statistics

Our findings in Sec. 3 confirm the existence of a *memory link* that links data generated by a student model back to its teacher's original member data. This, coupled with the greater reliability of distribution-level statistics over instance-level scores, leads us to a central argument: privacy audits for distilled generative models should shift to distributional statistics. This section outlines three guiding principles for such distribution-based MIAs.

4.1. Principles for distributional membership inference

Reflecting on the challenges of I-MIAs (Sec. 2) and the persistence of distributional signals through distillation (Sec. 3), we propose that effective MIAs, especially where instance-level cues are weak (as in model distillation), should be built upon the following principles: (1) *Set-based analysis*: Distributional measures are inherently more stable and reveal more pronounced signals when computed over sets of data points rather than isolated instances (Sec. 3.2). MIAs should leverage this by analyzing collections of samples to enhance their statistical power. (2) *Distributional comparison*: The insight from Sec. 3.1 is that a student's output distribution is often statistically closer to its teacher's member distribution than to non-member distributions. Effective MIAs should therefore quantify this *statistical divergence* to uncover traces of memorization. (3) *Discriminative signals focus*: Effective MIAs should target signals whose discriminative power between members and non-members becomes more salient at the distributional level, while retaining their resilience even through transformations, such as model distillation. To illustrate how these principles can be applied in practice, we present a pilot implementation called D-MIA, which adheres to such principles and *verifies our position successfully*. See **App.** B for details and empirical results.

4.2. Alternative view: contexts where I-MIAs remain relevant

Although distributional approaches effectively address major challenges in auditing distilled generative models, practical constraints highlight the continued relevance of instance-level MIAs in specific privacy auditing contexts. Building on our previous findings and alternative views, we conclude with a discussion of the regulatory and ethical implications of D-MIA, as detailed in App. E.

When candidate data is scarce. Distributional methods like D-MIA require a number of samples in a candidate set to estimate distributional statistics reliably. In practice, data subjects (e.g., artists) may have only a limited collection of personal records–perhaps fewer than 10 pieces or even a single artwork–when they seek an audit to determine if their data was used to train a generative model. As Tab. 3 shows, D-MIA's discriminative power degrades when candidate set sizes lower down (see **App.** D.5 for details). On the contrary, I-MIAs are not limited in this case as they probe model behavior at the sample level.

Resource considerations. All MIAs need reference data. However, distributional comparisons might implicitly encourage the use of larger reference datasets to ensure robust statistical estimation. This could increase data storage and management overhead, potentially conflicting with data minimization principles under regulations like GDPR (Mondschein & Monda, 2019).

5. Final Remarks

This paper shows that conventional I-MIAs fail on distilled models, creating a privacy gap by obscuring training data provenance. We propose shifting from instance-level scores to distributional statistics, as distillation removes direct memorization but retains exploitable statistical patterns. Building on this, we introduce principles for distributional MIAs and validate their effectiveness. We argue this shift is essential for robust privacy auditing and call for further research into distributional tools, distillation risks, and redefining membership and privacy harm in generative AI.

References

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *NeurIPS*, 2020.

Carlini, N., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. Membership inference attacks from first principles. In *SP*, 2022.

Chen, D., Yu, N., Zhang, Y., and Fritz, M. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *SIGSAC*, 2020.

Chérief-Abdellatif, B.-E. and Alquier, P. Finite sample properties of parametric mmd estimation: robustness to misspecification and dependence. *Bernoulli*, 2022.

Choi, H. K., Khanov, M., Wei, H., and Li, Y. How contaminated is your benchmark? quantifying dataset leakage in large language models with kernel divergence. *arXiv preprint arXiv:2502.00678*, 2025.

Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020.

Dong, Y., Jiang, X., Liu, H., Jin, Z., Gu, B., Yang, M., and Li, G. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. *arXiv preprint arXiv:2402.15938*, 2024.

Duan, J., Kong, F., Wang, S., Shi, X., and Xu, K. Are diffusion models vulnerable to membership inference attacks? In *ICML*, 2023.

Floridi, L. and Chiriatti, M. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 2020.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *ACM*, 2020.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *JMLR*, 2012.

Gu, Y., Dong, L., Wei, F., and Huang, M. Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*, 2023.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Hinton, G. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *NeurIPS*, 2020.

Hsieh, C.-Y., Li, C.-L., Yeh, C.-K., Nakhost, H., Fujii, Y., Ratner, A., Krishna, R., Lee, C.-Y., and Pfister, T. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*, 2023.

Hu, B., Li, J., Xu, L., Lee, M., Jajoo, A., Kim, G.-W., Xu, H., and Akella, A. Blockllm: Multi-tenant finer-grained serving for large language models. *arXiv preprint arXiv:2404.18322*, 2024.

Karras, T. A style-based generator architecture for generative adversarial networks. *CVPR*, 2019.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 2022.

Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., and Ermon, S. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279*, 2023.

Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 (canadian institute for advanced research). *URL http://www. cs. toronto. edu/kriz/cifar. html*, 2010.

Li, J., Dong, J., He, T., and Zhang, J. Towards black-box membership inference attack for diffusion models. *arXiv preprint arXiv:2405.20771*, 2024.

Liu, F., Xu, W., Lu, J., Zhang, G., Gretton, A., and Sutherland, D. J. Learning deep kernels for non-parametric two-sample tests. In *ICML*, 2020.

Liu, Y., Cao, J., Liu, C., Ding, K., and Jin, L. Datasets for large language models: A comprehensive survey. *arXiv preprint arXiv:2402.18041*, 2024.

Luo, W., Hu, T., Zhang, S., Sun, J., Li, Z., and Zhang, Z. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. *NeurIPS*, 2024.

Meeus, M., Shilov, I., Jain, S., Faysse, M., Rei, M., and de Montjoye, Y.-A. Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it). *arXiv preprint arXiv:2406.17975*, 2024.

Mondschein, C. F. and Monda, C. The eu's general data protection regulation (gdpr) in a research context. *FCDS*, 2019.

Pang, Y., Wang, T., Kang, X., Huai, M., and Zhang, Y. White-box membership inference attacks against diffusion models. *SP*, 2023.

Shejwalkar, V. and Houmansadr, A. Membership privacy for machine learning models through knowledge transfer. In *AAAI*, 2021.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *SP*, 2017.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *ICLR*, 2020.

Song, Y., Dhariwal, P., Chen, M., and Sutskever, I. Consistency models. In *ICML*, 2023.

Tang, X., Mahloujifar, S., Song, L., Shejwalkar, V., Nasr, M., Houmansadr, A., and Mittal, P. Mitigating membership inference attacks by {Self-Distillation} through a novel ensemble architecture. In *USENIX*, 2022.

Tang, Y., Wang, Y., Guo, J., Tu, Z., Han, K., Hu, H., and Tao, D. A survey on transformer compression. *arXiv preprint arXiv:2402.05964*, 2024.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Xie, R., Wang, J., Huang, R., Zhang, M., Ge, R., Pei, J., Gong, N., and Dhingra, B. Recall: Membership inference via relative conditional log-likelihoods. In *emnlp*, 2024.

Ye, W., Hu, J., Li, L., Wang, H., Chen, G., and Zhao, J. Data contamination calibration for black-box llms. *arXiv preprint arXiv:2405.11930*, 2024.

Yeom, S., Giacomelli, I., Fredrikson, M., and Jha, S. Privacy risk in machine learning: Analyzing the connection to overfitting. In *CSF*, 2018.

Yin, T., Gharbi, M., Zhang, R., Shechtman, E., Durand, F., Freeman, W. T., and Park, T. One-step diffusion with distribution matching distillation. In *CVPR*, 2024.

Zhang, J., Das, D., Kamath, G., and Tramèr, F. Membership inference attacks cannot prove that a model was trained on your data. *arXiv preprint arXiv:2409.19798*, 2024.

A. MIAs for Generative Models

We first introduce some basic concepts and relevant literature in this section. For readers who are familiar with MIAs for generative models, it is safe to skip this section. MIAs evaluate whether a specific instance is used during model training. Let $\mathcal{X}$ be a data space and $\mathcal{D}^{\text{mem}} \subset \mathcal{X}$ be a member set used to train a generative model $G : \mathcal{Z} \to \mathcal{X}$ that transforms noises sampled from a latent distribution $\mathbf{z} \sim p(\mathbf{z})$ into synthetic data $\mathbf{x} = G(\mathbf{z}) \in \mathcal{X}$. Given a query sample $\boldsymbol{x}_q \in \mathcal{X}$, an MIA constructs a binary classifier $\mathcal{A} : \mathcal{X} \times G \to \{0, 1\}$ that predicts the membership attribution of $\mathbf{x}_q$ as $\mathcal{A}(\boldsymbol{x}_q, G) = 1$ if $\boldsymbol{x}_q \in \mathcal{D}^{\text{mem}}$, and 0 otherwise. Existing MIAs use instance-level scores to distinguish members from non-members, a strategy we term I-MIA, including *reference-based* and *intrinsic-based* methods that are introduced below.

A.1. Brief introduction for instance-level MIAs

Reference-based I-MIAs use carefully constructed *reference models* to compute a score that can distinguish members from non-members. Given a target generative model G, one needs to construct n architecture-similar or identical reference models $\{G_i^{\text{ref}}\}_{i=1}^n$, leading to two complementary sets of models for a query sample $\boldsymbol{x}_q$,

$$\mathcal{M}_1 = \{G_i^{\text{ref}} : \boldsymbol{x}_q \in \mathcal{D}_i^{\text{mem}}\} \text{ and } \mathcal{M}_0 = \{G_i^{\text{ref}} : \boldsymbol{x}_q \notin \mathcal{D}_i^{\text{mem}}\},$$

where $\mathcal{D}_i^{\text{mem}}$ denotes the training dataset of the i-th reference model. The membership inference decision is then based on the difference between the target model $\phi(G, \boldsymbol{x})$ and these groups. For example, $\mathcal{A}(\boldsymbol{x}, G) = 1$ if a difference metric $\Delta(\boldsymbol{x}, G) > \tau$ and 0 otherwise, where $\Delta(\boldsymbol{x}, G) \triangleq \text{sim}(\phi(G, \boldsymbol{x}), \mathcal{M}_1) - \text{sim}(\phi(G, \boldsymbol{x}), \mathcal{M}_0)$, $\text{sim}(\cdot, \cdot)$ is a function to measure the similarity between two models, and τ is the decision threshold.

Intrinsic-based I-MIAs directly leverage the statistical gaps that emerge from target model training. At their core, these attacks exploit a fundamental memorization tendency of generative models $G : \mathcal{Z} \to \mathcal{X}$, i.e., the target model behaves differently between member instances $\mathcal{D}^{\text{mem}}$ and non-member instances $\mathcal{D}^{\text{non}}$, quantified as $\Delta(\mathbf{x}, G) = \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}^{\text{mem}}}[\mathcal{L}(\boldsymbol{x}; G)] - \mathbb{E}_{\boldsymbol{x}' \sim \mathcal{D}^{\text{non}}}[\mathcal{L}(\boldsymbol{x}'; G)] < 0$, where $\mathcal{L}(\boldsymbol{x}; G)$ is normally the minimal distance between $\boldsymbol{x}$ and the data generated by G. This statistical gap may manifest differently across generative architectures, leading to model-specific attack strategies. GAN-Leak (Chen et al., 2020), for example, targets MIA on *generative adversarial networks* (Goodfellow et al., 2020) by reconstructing target images through latent optimization, solving $\mathcal{L}_{\text{GANLeak}}(\mathbf{x}) = \min_{\boldsymbol{z} \in \mathcal{Z}} \|\mathbf{x} - G(\boldsymbol{z})\|_2^2$ where members $\boldsymbol{x} \sim \mathcal{D}^{\text{mem}}$ often show lower reconstruction errors.

A.2. How I-MIAs are applied to large-scale generative models?

Large-scale generative model in text domain-LLMs (Guo et al., 2025; Touvron et al., 2023), that are auto-regressive transformers (Tang et al., 2024) trained on massive text corpora to model the log-likelihood of each next token given its preceding context. Several MIA strategies have been developed for LLMs. PCA (Ye et al., 2024) detects membership by comparing the target input with a synthetic version generated by word-swapping. A large log-likelihood gap indicates membership. The Min-K% (Shokri et al., 2017) computes the average log-likelihood of the k tokens with the least confidence in a given input. Typically, member data exhibit higher log-likelihoods for these tokens compared to non-members. Recall (Xie et al., 2024) introduces a non-member prefix to condition the model and quantifies the resulting change in log-likelihood. If the log-likelihood changes significantly, the data is inferred to be a member sample.

Large-scale generative model in vision domain-Diffusion models (Ho et al., 2020; Song et al., 2020), which operate through a forward process $q(\boldsymbol{x}_t|\boldsymbol{x}_0)$ that progressively adds Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ to the input $\boldsymbol{x}_0$ and a reverse noise removal process $p_G(\boldsymbol{x}_0|\boldsymbol{x}_t)$ that reconstructs the original $\boldsymbol{x}_0$ over time steps $t \in [0, T]$. Several attacks have been proposed in this context and have been evaluated on a class of diffusion models, the denoising diffusion probabilistic model (DDPM) (Ho et al., 2020). According to DDPM objective loss $\mathcal{L}_{\text{MSE}} = \|\epsilon_G(x_t, t) - \epsilon\|^2$, SecMI (Duan et al., 2023) performs an attack by monitoring this loss across different time-steps t. ReDiffuse (Li et al., 2024) investigates reconstruction stability under noise perturbations, based on the observation that member data tend to yield more consistent reconstructions. GSA (Pang et al., 2023) examines gradient dynamics during model retraining and finds that member da

A.3. Large-scale generative models challenge the effectiveness of I-MIAs

I-MIAs face challenges when attacking LLMs. Although I-MIA remains widely used for large-scale generative models, it is unreliable, specifically for *LLMs* (Dong et al., 2024; Ye et al., 2024). This is because the extensive training on massive corpora and substantial model capacity of LLMs would lead to similar behaviors between *individual* members and

non-members exploited by I-MIAs (Ye et al., 2024). When processing an input, LLMs consistently produce high-confidence outputs regardless of whether it is part of training data, reducing the discriminative power of instance-level metrics (Dong et al., 2024). Moreover, Zhang et al. (2024) argues that I-MIAs on LLMs suffer from unboundable false positive rates (FPR)—a critical metric for evaluating the validity of MIAs when used as evidence to allege the use of unauthorized data (Carlini et al., 2022; Zhang et al., 2024). The reason is that LLMs are trained with massive web-scale corpora, such that there are no 'true' individual non-members (as LLMs *indirectly* learn too much data (Zhang et al., 2024)).

I-MIAs face challenges when attacking diffusion models. In the case of generative models in the visual domain—diffusion models, several intrinsic-based MIAs have shown promising results on DDPMs (Ho et al., 2020). Since the objective function of DDPMs encourages models to learn the exact denoising trajectories for every training data, resulting in lower reconstruction loss for members compared to non-members. However, DDPMs rely on sample-specific and fine-grained denoising trajectories, making them prone to deviation when the predicted noise contains large errors. To maintain stability, they require thousands of small incremental denoising steps, which are computationally expensive (Ho et al., 2020). To reduce the number of denoising steps, more effective consistency-based models incorporate a KL smoothing term to learn globally consistent denoising trajectories (Kim et al., 2023; Song et al., 2023; Karras et al., 2022). This term encourages uniformity across samples rather than overfitting to sample-specific paths, thereby reducing the loss gap between member and non-member data and potentially weakening intrinsic-based MIAs.

B. D-MIA exemplar framework

B.1. D-MIA: an exemplar framework for auditing distilled generative models

To illustrate how these principles can be put into practice, we showcase a pilot implementation called D-MIA, which considers the following problem setup. Let G_{T} be a teacher model pre-trained on a private member dataset, denoted by $\mathcal{D}^{\mathrm{mem}}$. Except for them, samples are collectively denoted by $\mathcal{D}^{\mathrm{non}}$ otherwise. We assume access to a student generative model G_{S}, trained *only* on the generated data from G_{T}. D-MIA enables *set-based analysis* by task design. Given a candidate dataset $\mathcal{D}^{\mathrm{can}}$, the goal is to determine whether $\mathcal{D}^{\mathrm{can}}$ overlaps with the teacher's member dataset $\mathcal{D}^{\mathrm{mem}}$.

For *distributional comparison*, D-MIA quantifies and examines the relative relationship between two quantities: 1) the *distributional distance* between candidate dataset $\mathcal{D}^{\mathrm{can}}$ and student-generated dataset $\mathcal{D}^{\mathrm{gen}}$, and 2) the *distributional distance* between known non-member data $\mathcal{D}^{\mathrm{non}}$ and student-generated data $\mathcal{D}^{\mathrm{gen}}$.

To *focus on discriminative signals*, D-MIA uses a two-stage approach. During *training*, it optimizes a deep-kernel MMD-based measure (Liu et al., 2020) to distinguish member from non-member data. This involves training a kernel that maximizes the distributional separation between known members $\mathcal{D}^{\mathrm{mem}}$ and non-members $\mathcal{D}^{\mathrm{non}}$ relative to the student-generated data $\mathcal{D}^{\mathrm{gen}}$. In *evaluation*, this learned metric determines whether a particular candidate set $\mathcal{D}^{\mathrm{can}}$ is statistically more similar to $\mathcal{D}^{\mathrm{gen}}$ than to $\mathcal{D}^{\mathrm{non}}$. If so, $\mathcal{D}^{\mathrm{can}}$ is inferred to likely contain member data (Further technical details are in App. D.2). The empirical evaluation of D-MIA, presented in App. B.2, demonstrates that the proposed exemplar framework effectively defends against attacks on distilled generative models.

B.2. Empirical support

D-MIA is effective against distilled generative models. Tab. 2 shows that D-MIA can successfully detect membership across various distilled models and datasets, even when candidate sets are mixtures of member and non-member data. For example, against DMD (a state-of-the-art distillation technique), D-MIA achieves near-perfect success rates (ASR $\approx 100\%$) across three datasets, significantly outperforming baselines. D-MIA maintains high ASR ($\approx 92\%$) on CIFAR10 with mixed candidate datasets, while baselines falter to near-random guessing if only 30% of the candidate sets are members. This confirms D-MIA as a reliable framework for practical scenarios where candidate sets often have unknown compositions of member and non-member data.

D-MIA can quantify dataset composition. Beyond a binary decision, D-MIA's output scores correlate positively with the portion of member data in a candidate set (See App. Fig. 4). Scores tend towards 1 for entirely member-comprised sets and decrease towards 0.5 as member presence diminishes, suggesting a new role for D-MIA in more granular privacy leakage analysis.

Table 2. ASR and AUC results of D-MIA against baselines I-MIA methods SecMI, and ReDiffuse on distilled models across CIFAR10, FFHQ, and AFHQv2. Rows are color-coded to represent member data proportions: 100%, 50%, and 30%. See Tab. 8 in App. F for the TPR@FPR=0.05 results.

Dataset (Member %)	DMD						Diff-Instruct					
	D-MIA		SecMI		ReDiffuse		D-MIA		SecMI		ReDiffuse	
	ASR	AUC	ASR	AUC	ASR	AUC	ASR	AUC	ASR	AUC	ASR	AUC
CIFAR10 (100%)	**0.98**	**0.99**	0.60	0.55	0.66	0.66	**1.0**	**1.0**	0.65	0.54	0.62	0.62
CIFAR10 (50%)	**0.98**	**0.99**	0.59	0.52	0.60	0.60	**1.0**	**1.0**	0.59	0.53	0.60	0.60
CIFAR10 (30%)	**0.92**	**0.97**	0.53	0.43	0.60	0.59	**1.0**	**1.0**	0.53	0.47	0.52	0.55
FFHQ (100%)	**1.0**	**1.0**	0.60	0.56	0.56	0.56	**1.0**	**1.0**	0.57	0.56	0.78	0.81
FFHQ (50%)	**0.99**	**0.99**	0.56	0.54	0.54	0.49	**1.0**	**1.0**	0.55	0.52	0.65	0.63
FFHQ (30%)	**0.98**	**0.99**	0.56	0.49	0.54	0.48	**1.0**	**1.0**	0.55	0.51	0.62	0.59
AFHQv2 (100%)	**1.0**	**1.0**	0.61	0.60	0.69	0.71	**1.0**	**1.0**	0.56	0.53	0.64	0.62
AFHQv2 (50%)	**1.0**	**1.0**	0.59	0.54	0.64	0.61	**1.0**	**1.0**	0.53	0.48	0.57	0.52
AFHQv2 (30%)	**1.0**	**1.0**	0.56	0.56	0.60	0.61	**1.0**	**1.0**	0.48	0.50	0.55	0.50

Table 3. ASR and AUC results of D-MIA evaluated on DMD under varying non-member and candidate dataset sizes. In each configuration, we equally split $\mathcal{D}^{\text{non}}$ for kernel training and MIA evaluation. All metrics decrease as $\mathcal{D}^{\text{non}}$ and $\mathcal{D}^{\text{can}}$ lower down.

$\lvert\mathcal{D}^{\text{non}}\rvert$	$\lvert\mathcal{D}^{\text{can}}\rvert$	ASR	AUC	TPR@FPR=0.05
(5000+10000)	5000	0.98	0.99	0.96
(2000+4000)	2000	0.94	0.97	0.88
(600+1200)	600	0.83	0.78	0.70
(300+600)	300	0.75	0.58	0.52

C. Setup for Diffusion Model and Distilled Models

The training configurations for EDM, DMD, and DI are shown in Tab. 4. The specific model architectures will be released in the upcoming official code. For each dataset, half of the data is randomly selected for training EDM, while the remaining half is used as non-member data. EDM generates 100,000 samples to distill DMD and DI models. During the distillation process, the models do not access the training data of EDM.

D. Details of D-MIA Framework

D.1. Preliminaries: MMD and Deep-kernel MMD

This section briefly summarizes the basic knowledge of MMD (Gretton et al., 2012) and its extension, deep-kernel MMD (Liu et al., 2020). MMD finds common application in areas such as domain adaptation and the evaluation of generative models, where assessing distributional alignment is important. In this paper, these concepts are used in quantifying the distributional differences in Section 3 and Section 4. We refer interested readers to the original papers for complete details therein.

Maximum Mean Discrepancy (MMD), proposed by Gretton et al. (2012), is a statistical tool for measuring the distance between two Borel probability measures, say $\mathbb{P}$ and $\mathbb{Q}$, defined on a separable metric space $\mathcal{X} \subseteq \mathbb{R}^d$. Consider independent random variables $X, X' \sim \mathbb{P}$ and $Y, Y' \sim \mathbb{Q}$. The squared MMD between $\mathbb{P}$ and $\mathbb{Q}$ in a Reproducing Kernel Hilbert Space $\mathbb{H}_k$, induced by a kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, is defined as:

$$\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}; k) = \mathbb{E}[k(X, X')] + \mathbb{E}[k(Y, Y')] - 2\mathbb{E}[k(X, Y)].$$

If k is a characteristic kernel (e.g., Gaussian), then $\mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}; k) = 0$ if and only if $\mathbb{P} = \mathbb{Q}$.

Empirical MMD. In practice, the true distributions $\mathbb{P}$ and $\mathbb{Q}$ are often unknown, and we rely on finite samples drawn from them. Given i.i.d. samples $\mathcal{S}_X = \{x_i\}_{i=1}^n$ from $\mathbb{P}$ and $\mathcal{S}_Y = \{y_j\}_{i=1}^m$ from $\mathbb{Q}$, an unbiased U-statistic estimator for MMD^2

is

$$\widehat{\mathrm{MMD}}_u^2(\mathcal{S}_X, \mathcal{S}_Y; k) = \frac{1}{n(n-1)} \sum_{i \neq l}^{n} k(x_i, x_l) + \frac{1}{m(m-1)} \sum_{j \neq p}^{m} k(y_i, y_j) - \frac{2}{nm} \sum_{i=1}^{n} \sum_{j=1}^{n} k(x_i, y_j). \tag{1}$$

Deep-kernel MMD. Traditional MMD uses a fixed, pre-defined kernel, which may lead to limited expressiveness when the kernel is not suitable for the task at hand. To address this, Liu et al. (2020) propose to learn a task-relevant representation $\theta_\omega : \mathcal{X} \to \mathcal{Z}$ using a neural network parameterized by w. The MMD can then be computed in this learned feature space $\mathcal{Z}$. As such, the goal of deep-kernel MMD is to find a representation θ_ω that maximizes the MMD, thereby increasing the test power to detect differences between $\mathbb{P}$ and $\mathbb{Q}$.

Following (Liu et al., 2020), let $\mathcal{S}_X = \{x_i\}_{i=1}^n$ and $\mathcal{S}_Y = \{y_j\}_{i=1}^n$ be samples from $\mathbb{P}$ and $\mathbb{Q}$ (assuming equal sample sizes n for simplicity). The empirical estimate for deep-kernel MMD, using a U-statistic, is

$$\widehat{\mathrm{MMD}}_u^2(\mathcal{S}_X, \mathcal{S}_Y; k_\omega) := \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}, \tag{2}$$

where H_{ij} is the kernel of the U-statistic, defined as

$$H_{ij} := k_\omega(x_i, x_j) + k_\omega(y_i, y_j) - k_\omega(x_i, y_j) - k_\omega(y_i, x_j). \tag{3}$$

Note that the kernel $k_\omega(\cdot, \cdot)$ itself is a composite function incorporating the learned features:

$$k_\omega(a, b) = \left[(1 - \epsilon)\ k_{\text{base}}\left(\theta_\omega(a), \theta_\omega(b)\right) + \epsilon\right] \cdot q_{\text{base}}\left(a, b\right). \tag{4}$$

Here, w is the feature extractor network (e.g., a multi-layer perceptron). $k_{\text{base}}(\cdot, \cdot)$ is a base characteristic kernel (e.g., Gaussian) applied to the learned features $\theta_\omega(a)$ and $\theta_\omega(b)$. In addition, $q_{\text{base}}(\cdot, \cdot)$ is typically another characteristic kernel on the original inputs, acting as a sample-pair weighting function that adjusts the influence of each pair in the kernel computation based on their importance or relevance. The small constant $\epsilon \in [0, 1]$ helps to ensure k_w remains characteristic (Liu et al., 2020).

Optimizing a deep-kernel MMD. When optimizing the parameters w of the feature network θ_ω to maximize the MMD estimate, it is often normalized by an estimate of its standard deviation to improve numerical stability and test power. The objective function is thus:

$$\max_\omega \mathcal{L}(\omega) = \frac{\widehat{\mathrm{MMD}}_u^2(\mathcal{S}_X, \mathcal{S}_Y; k_\omega)}{\sigma\left(\widehat{\mathrm{MMD}}_u^2(\mathcal{S}_X, \mathcal{S}_Y; k_\omega)\right)}, \tag{5}$$

where $\sigma(\cdot)$ denotes the standard deviation of the MMD estimator. Since the true variance σ^2 is generally unknown, we estimate it using a regularized estimator $\hat{\sigma}_\lambda^2$, given by:

$$\hat{\sigma}_\lambda^2 = \frac{4}{n^3} \sum_{i=1}^{n} \left(\sum_{j=1}^{n} H_{ij} \right)^2 - \frac{4}{n^4} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} H_{ij} \right)^2 + \lambda, \tag{6}$$

where λ is a constant to avoid division by zero. The optimization of w is typically performed using stochastic gradient ascent on $\mathcal{L}(w)$.

D.2. D-MIA illustration

We now introduce the details of D-MIA, a pilot implementation that adheres to the guiding principles (Sec. 4) and verify the effectiveness of distributional statistics in performing MIA.

Problem setting. Since D-MIA operates with sets of data instances, it is noteworthy to mention that we consider a new problem setup than conventional I-MIAs.

Let $G_{\text{T}} : \mathcal{Z} \to \mathcal{X}$ be a teacher generative model pre-trained on a private member dataset $\mathcal{D}^{\text{mem}} = \{\boldsymbol{x}_i\}_{i=1}^N$, where $\boldsymbol{x}_i \sim \mathbb{P}^{\text{mem}}$, $\mathbb{P}^{\text{mem}}$ is the member data distribution. We have access to a distilled student generative model G_{S} that mimics

Table 4. Training configurations for different models (EDM, DMD, and DI) across datasets (CIFAR10, FFHQ, and AFHQv2), including GPU setups, batch sizes, training times, and learning rates.

Model	Dataset	GPU	Batch size	Training Time	Learning Rate
EDM	CIFAR10	1 × NVIDIA A100	128	5-00:00:00	0.001
	FFHQ	4 × NVIDIA A100	256	5-00:00:00	0.0002
	AFHQv2	2 × NVIDIA A100	128	5-00:00:00	0.0002
DMD	CIFAR10	1 × NVIDIA A100	128	4-00:00:00	0.00005
	FFHQ	1 × NVIDIA A100	64	4-00:00:00	0.00005
	AFHQv2	1 × NVIDIA A100	64	4-00:00:00	0.00005
DI	CIFAR10	1 × NVIDIA A100	128	3-00:00:00	0.00001
	FFHQ	1 × NVIDIA A100	64	3-00:00:00	0.0001
	AFHQv2	1 × NVIDIA A100	64	2-00:00:00	0.0001

G_{T}'s behavior, trained using synthetic samples $\{G_{\mathrm{T}}(\boldsymbol{z}_j)\}_{j=1}^{M}$ with noises $\boldsymbol{z}_j \sim \mathbb{P}_{\mathcal{Z}}$, the distribution over the latent space (e.g., standard Gaussian). In D-MIAs, we consider set-based prediction: given a *candidate dataset* $\mathcal{D}^{\mathrm{can}} = \{\boldsymbol{x}'_j\}_{j=1}^{N}$, the task is to infer if $\mathcal{D}^{\mathrm{can}} \cap \mathcal{D}^{\mathrm{mem}} = \varnothing$, i.e., contains member instances.

D-MIA requires two reference datasets: (1) a *non-member* set $\mathcal{D}^{\mathrm{non}} = \{\boldsymbol{x}''_k\}_{k=1}^{N}$ of public instances $\boldsymbol{x}''_k \nsim \mathbb{P}^{\mathrm{mem}}$ and (2) an *anchor* set $\mathcal{D}^{\mathrm{anc}} = \{\boldsymbol{x}^*_l\}_{l=1}^{L}$ (e.g., generated by G_{S}) used to facilitate distributional comparison. Moreover, since private member data is typically inaccessible, we propose to construct a *proxy member* set $\widetilde{\mathcal{D}}^{\mathrm{mem}} = \{G_{\mathrm{S}}(\boldsymbol{z}_j)\}_{j=1}^{N}$ to approximate $\mathbb{P}^{\mathrm{mem}}$. At its core, D-MIA aims to detect whether $\mathcal{D}^{\mathrm{can}}$ aligns more closely with $\widetilde{\mathcal{D}}^{\mathrm{mem}}$ or $\mathcal{D}^{\mathrm{non}}$ through *relative distributional discrepancy thresholding*.

Training a deep-kernel MMD. We first optimize a data-adaptive kernel k_ω, parameterized by deep neural nets ω (Liu et al., 2020) to *maximize the separation* between $\tilde{\mathcal{D}}^{\mathrm{mem}}$ and $\mathcal{D}^{\mathrm{non}}$ in the feature space. For $\tilde{\mathcal{D}}^{\mathrm{mem}}$, $\mathcal{D}^{\mathrm{non}}$ and $\mathcal{D}^{\mathrm{anc}}$, we perform mini-batch training and randomly sample subsets from each dataset, e.g., $\mathcal{B}^{\mathrm{anc}} = \{\boldsymbol{x}^*_b \overset{\text{i.i.d}}{\sim} \mathcal{D}^{\mathrm{anc}}\}_{b=1}^{B}$, with respect to the optimization objective $\mathcal{L}(\omega)$ defined as

$$\mathcal{L}(\omega) = \underbrace{\left[\widehat{\mathrm{MMD}}_u^2(\mathcal{B}^{\mathrm{anc}}, \tilde{\mathcal{B}}^{\mathrm{mem}}; k_\omega)\right]}_{\text{member discrepancy}} - \underbrace{\left[\widehat{\mathrm{MMD}}_u^2(\mathcal{B}^{\mathrm{anc}}, \mathcal{B}^{\mathrm{non}}; k_\omega)\right]}_{\text{non-member discrepancy}}.$$

Doing so amplifies the MMD values between non-members and the anchor distribution while minimizing them for member-like distributions. See **Alg. 1** for details.

Detecting membership. In this step, we aim to determine whether $\mathcal{D}^{\mathrm{can}} \cap \mathcal{D}^{\mathrm{mem}} = \varnothing$, by computing two MMD statistics using the trained kernel k_ω: $M_1^{(t)} \triangleq \widehat{\mathrm{MMD}}_u^2(\mathcal{B}^{\mathrm{anc}}, \mathcal{B}^{\mathrm{can}}; k_\omega)$ and $M_2^{(t)} \triangleq \widehat{\mathrm{MMD}}_u^2(\mathcal{B}^{\mathrm{anc}}, \mathcal{B}^{\mathrm{non}}; k_\omega)$ over T Bernoulli trials. The membership is indicated per trial via $\mathbb{I}^{(t)} = \mathbb{1}(M_1 < M_2)$, and the aggregate membership probability is estimated by $p^{\mathrm{mem}} = \frac{1}{T}\sum_t \mathbb{I}^{(t)}$ (details are in **Alg. 2**).

Ensembling multiple kernels. To mitigate the variance from finite-sample MMD estimates (Chérief-Abdellatif & Alquier, 2022), we aggregate predictions across m independently trained kernels $\{k_\omega^{(i)}\}_{i=1}^{m}$. For each kernel, we compute $p^{\mathrm{mem}})_{(i)}$ over n Bernoulli trials as with Alg. 2. We apply a final decision threshold τ to the ensemble mean $\bar{p}^{\mathrm{mem}} = \frac{1}{m}\sum_i p^{\mathrm{mem}}_{(i)}$, declaring membership of $\mathcal{D}^{\mathrm{can}}$ if $\bar{p}^{\mathrm{mem}} > \tau$. See **Alg. 3** for detailed illustrations.

D.3. Experimental setup

Dataset and victim Models. We empirically evaluate D-MIA on state-of-the-art distilled generative models, DMD (Yin et al., 2024) and Diff-Instruct (Luo et al., 2024) on commonly studied MIA benchmarks, CIFAR10 (Krizhevsky et al., 2010), FFHQ (Karras, 2019), and AFHQv2 (Choi et al., 2020). See detailed setup of victim models in **App.** C

Baseline settings. D-MIA differs from existing MIA methods and attack targets. To ensure fairness, we adapt existing methods to the D-MIA setting for experimentation. Specifically, we apply existing MIA methods to each data point in the dataset to compute a loss-based result. Then we compute the mean loss result of all data points in the dataset. We randomly sample 50 candidate datasets (with replacement) and 50 non-member datasets (with replacement) and calculate

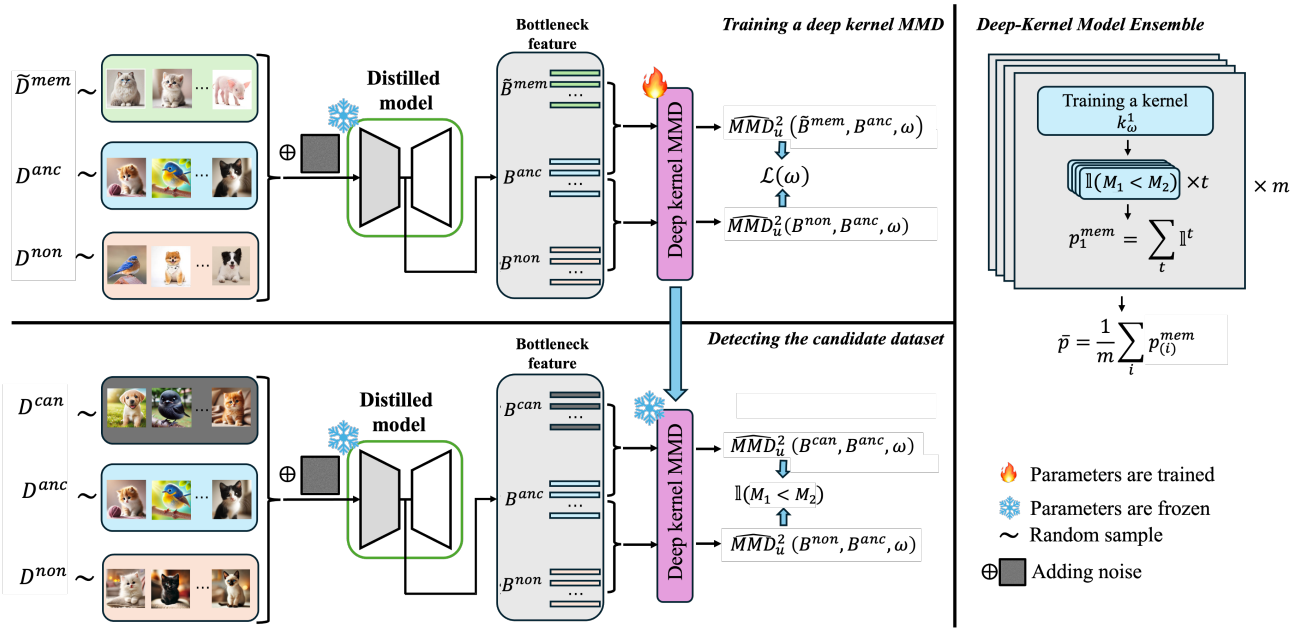

Figure 3. Overview of our two-phase MMD-based D-MIA framework, consisting of (1) deep-kernel MMD training phase (top left) and (2) detecting the Candidate Dataset phase (bottom left). We also propose a kernel ensemble strategy to improve detection robustness (right).

the mean loss for each dataset. Then, we empirically determine an optimal threshold to distinguish between the loss means of candidate datasets and non-member datasets. Under this setting, we use SecMI and ReDiffuse as baseline methods for comparison.

Evaluation settings. Before the experiment, each dataset is evenly divided into two subsets: one for member data used to train the teacher model (EDM) and the other for non-member data (detailed EDM training setup is in **App.** C). The teacher model generates 100,000 synthetic samples for the distillation of the student model, ensuring that the student model never accesses the original training data of the teacher model. We construct an auxiliary non-member dataset by randomly sampling 15,000 data points from the non-member data of FFHQ and CIFAR10, with 5,000 points used for deep-kernel training (Alg. 1) and 10,000 for candidate dataset detection (Alg. 2). For AFHQv2, we sample 3,000 non-member data points, allocating 1,500 for kernel training and 1,500 for candidate detection. To ensure fairness, we randomly discard 15,000 member data points (3,000 for AFHQv2).

To evaluate D-MIA under varying proportions of member data in the candidate datasets, we create candidate datasets with 100%, 50%, and 30% member data. During detection, we randomly sample 5,000 data points (1,500 for AFHQv2) based on the specified member ratios to construct positive candidate datasets. Additionally, we construct a negative candidate dataset consisting entirely of non-member data to assess whether it can be distinguished from the positive datasets. Similar to the baseline setting, we perform 50 rounds of sampling and detection to verify the attack accuracy of D-MIA.

Implementation details of D-MIA The network architecture of the deep-kernel MMD follows the design proposed by Liu et al. (2020). The training parameters (e.g., bandwidth, learning rate, and epochs) used for attacking different models with various training datasets are detailed in Tab. 5.

D.4. Key algorithms in D-MIA

This section details the three key steps in D-MIA, each executing a specific algorithm: Deep-Kernel Training (**Alg.** 1), Detecting The Candidate Dataset (**Alg.** 2), and Ensembling Multiple Kernels.

D.5. D-MIA's reliance on auxiliary non-member and candidate dataset sizes

In D-MIA attacks, the attacker requires a certain amount of non-member data for auxiliary training and testing. Additionally, the candidate dataset being evaluated must have a sufficient size to obtain accurate distributional information. Therefore, we

Table 5. Deep-kernel training configurations for distillation models (DI and DMD) across different datasets. "Bandwidth" denotes the kernel bandwidth used in the deep-kernel MMD loss; "Epoch" indicates the total number of training iterations of deep-kernel; "MMD learning rate" refers to the learning rate of the deep-kernel MMD training; "H" represents the number of hidden features or layers used in the feature extractor network; and "x_out" is the output dimensionality of the feature extractor network.

Model	Dataset	Bandwidth	Epoch	MMD learning rate	H	x_out
DI	CIFAR10	0.1	400	0.000001	450	35
	FFHQ	0.4	300	0.000001	450	50
	AFHQv2	0.1	400	0.000001	450	35
DMD	CIFAR10	0.0025	300	0.0000001	250	20
	FFHQ	0.4	300	0.000001	450	50
	AFHQv2	0.1	400	0.000001	450	35

Algorithm 1 Deep-Kernel Training

Input: non-member dataset $\mathcal{D}^{\text{non}}$; one-step generative model G_S and encoder G_e of G_S; standard deviation σ of additive Gaussian noise; learning rate η; epochs E
$S_g \leftarrow \{G_S(z_i) \mid z_i \sim \mathcal{N}(0, I),\ i = 1, \ldots, N\}$
$S_{g,\text{noisy}} \leftarrow \{s + \epsilon \mid s \in S_g,\ \epsilon \sim \mathcal{N}(0, \sigma^2 I)\}$
$S_{a,\text{noisy}} \leftarrow \{a + \epsilon \mid a \in \mathcal{D}^{\text{non}},\ \epsilon \sim \mathcal{N}(0, \sigma^2 I)\}$
$S_{g-e} \leftarrow \{G_e(s) \mid s \in S_{g,\text{noisy}}\}$
$S_{a-e} \leftarrow \{G_e(a) \mid a \in S_{a,\text{noisy}}\}$
Sample mini-batch $\mathcal{B}^{\text{non}} \subset S_{a-e}$
Sample mini-batch $\tilde{\mathcal{B}}^{\text{mem}} \subset S_{g-e}$
Sample mini-batch $\mathcal{B}^{\text{anc}} \subset S_{g-e}$ such that $\mathcal{B}^{\text{anc}} \cap \tilde{\mathcal{B}}^{\text{mem}} = \emptyset$
for epoch $= 1$ to E **do**
    $M_1 \leftarrow \widehat{\text{MMD}}_u^2(\mathcal{B}^{\text{non}}, \mathcal{B}^{\text{anc}}, k_\omega)$
    $M_2 \leftarrow \widehat{\text{MMD}}_u^2(\tilde{\mathcal{B}}^{\text{mem}}, \mathcal{B}^{\text{anc}}, k_\omega)$
    $l \leftarrow M_1 - M_2$
    $\omega \leftarrow \text{Adam}(\omega, \nabla l, \eta)$
end for
Output: trained kernel k_ω; anchor features $\mathcal{B}^{\text{anc}}$

evaluate the performance of D-MIA on CIFAR10 models for DMD and DI under different auxiliary non-member dataset sizes and candidate dataset sizes. We evaluated three settings for auxiliary and candidate dataset sizes: auxiliary dataset sizes of 15,000, 9,000, 6,000, and 3000 paired with candidate dataset sizes of 5,000, 3,000, 2,000, and 1000, respectively. Half of the auxiliary dataset was used to train the deep-kernel, while the other half was used to support attacks on the candidate dataset. Positive samples were drawn from member data corresponding to the candidate dataset size, and negative samples were drawn from non-member data of the same size. Following the previous evaluation, 50 positive and 50 negative samples were constructed, and D-MIA was applied to distinguish between them.

E. Implications: Lessons from Distillation for Broader MIA

Redefining membership and privacy harm in model life-cycles. D-MIAs compel us to expand the notion of "membership". Instead of solely referring to "a specific instance the model was trained on", membership can also signify "the statistical property that the model learns from the dataset". This expanded view is important for understanding privacy harm when models learn from transformed data–like student models trained on teacher-generated data–that still carry the statistical bias of the original, potentially sensitive member data. This concern, for example, may not be limited to whether a model can reproduce a specific photo, but whether it has learned to mimic a creator's distinct artistic style or absorbed societal biases from a text corpus, even without memorizing exact data points. This shift highlights the need for a more comprehensive discussion about privacy risks and what constitutes unauthorized data use in the complex lifecycles of modern AI models.

Strengthening audits for model provenance and countering "model laundering". As AI models and their training data become valuable assets, concerns about "model laundering"–the practice of obscuring the use of unauthorized data through

Algorithm 2 Detecting the Candidate Dataset

Input: non-member dataset $\mathcal{D}^{\text{non}}$; candidate dataset $\mathcal{D}^{\text{can}}$; anchor features $\mathcal{B}^{\text{anc}}$; encoder G_e of G_S; Gaussian noise std. σ; number of repetitions T; kernel function k_ω
$S_{c,\text{noisy}} \leftarrow \{c + \epsilon \mid c \in \mathcal{D}^{\text{can}},\ \epsilon \sim \mathcal{N}(0, \sigma^2 I)\}$
$S_{a,\text{noisy}} \leftarrow \{a + \epsilon \mid a \in \mathcal{D}^{\text{non}},\ \epsilon \sim \mathcal{N}(0, \sigma^2 I)\}$
$S_{c-e} \leftarrow \{G_e(c) \mid c \in S_{c,\text{noisy}}\}$
$S_{a-e} \leftarrow \{G_e(a) \mid a \in S_{a,\text{noisy}}\}$
Sample mini-batch $\mathcal{B}^{\text{can}} \subset S_{c-e}$
Sample mini-batch $\mathcal{B}^{\text{non}} \subset S_{a-e}$
for $t = 1$ to T **do**
  $M_1^t \leftarrow \widehat{\text{MMD}}_u^2(\mathcal{B}^{\text{can}}, \mathcal{B}^{\text{anc}}, k_\omega)$
  $M_2^t \leftarrow \widehat{\text{MMD}}_u^2(\mathcal{B}^{\text{non}}, \mathcal{B}^{\text{anc}}, k_\omega)$
  $\mathbb{I}^{(t)} \leftarrow \mathbb{1}(M_1^t < M_2^t)$
end for
$p^{\text{mem}} \leftarrow \frac{1}{T} \sum_{t=1}^{T} \mathbb{I}^{(t)}$
Output: p^{mem}

Algorithm 3 Ensembling Multiple Kernels

Input: anchor features $\mathcal{B}^{\text{anc}}$; non-member dataset $\mathcal{D}^{\text{non}}$; candidate dataset $\mathcal{D}^{\text{can}}$; one-step generative model G_S and encoder G_e; number of iterations h; threshold τ; Gaussian noise std. σ; test repetitions T; learning rate η; epochs E
Initialize kernel set $K \leftarrow \emptyset$, prediction set $R \leftarrow \emptyset$
for $i = 1$ to h **do**
  Train kernel k_ω^i using Algorithm 1
  Compute $p_{(i)}^{\text{mem}}$ using Algorithm 2
  $K \leftarrow K \cup \{k_\omega^i\}$
  $R \leftarrow R \cup \{p_{(i)}^{\text{mem}}\}$
end for
$\bar{p}^{\text{mem}} \leftarrow \frac{1}{h} \sum_{i=1}^{h} p_{(i)}^{\text{mem}}$
if $\bar{p}^{\text{mem}} \geq \tau$ **then**
  D-MIA$(D^{\text{can}}) \leftarrow 1$
else
  D-MIA$(D^{\text{can}}) \leftarrow 0$
end if
Output: D-MIA(D^{can})

techniques like distillation–are likely to grow. If a teacher model (G_{T}) was trained on copyright-protected or sensitive data, and a company deploys a student model (G_{S}) claiming that it was trained only on legitimate, teacher-generated data. I-MIAs on G_{S} would likely find no evidence of the original data misuse in training G_{T}. In contrast, a successful distributional MIA against G_{S} could reveal that its learned data distribution closely matches that of the potentially problematic dataset used for G_{T}, thus providing a crucial tool for auditing provenance and detecting such attempts to conceal unauthorized data use.

Distributional MIAs are more secure auditing tools. Recall that I-MIAs seek to identify individual data instances, which raises a security dilemma as well: tools designed to audit for privacy leakage by spotting specific training examples could, in the wrong hands, be *abused* to extract those same sensitive data. D-MIAs mitigate this tension by shifting focus from individual samples to candidate sets, evaluating alignment with the training distribution. D-MIAs assess whether a candidate dataset, as a whole, aligns with the training distribution's characteristics. They can confirm significant data overlap without pinpointing which specific samples were members. Consequently, even if an attacker understands the distributional MIA mechanism, they cannot directly use the attack to determine the membership status of individual data points. From a privacy perspective, this dataset-based evaluation offers a new auditing paradigm with built-in privacy safeguards.

F. Additional Experimental Results

We conducted a series of experiments to evaluate the effectiveness of different I-MIA methods on various generative models. Specifically, we extracted half of the data from the CIFAR10, FFHQ, and AFHQv2 datasets to train three EDM generative models, and then used the data generated by EDM to train DMD and Diff-Instruc. Finally, we applied four state-of-the-art MIA techniques—GAN-Leak, SecMI, ReDiffuse, and GSA—to attack these models. The ASR and AUC results are presented in Tab. 6. The TPR values at FPR = 0.05 results are presented in Tab. 7.

Table 6. the ASR and AUC results of various membership inference attack methods across different generative models and datasets. The table compares four attack methods—GAN-leak, SecMI, ReDiffuse, and GSA—on three generative models: EDM, DMD, and Diff-Instruc, evaluated on CIFAR-10, FFHQ, and AFHQv2 datasets.

Model/Dataset	GAN-leak		SecMI		Rediffuse		GSA	
	ASR	AUC	ASR	AUC	ASR	AUC	ASR	AUC
EDM/CIFAR10	0.536 ± .005	0.523 ± .011	0.588 ± .004	0.601 ± .021	0.579 ± .002	0.603 ± .004	0.622 ± .008	0.626 ± .004
EDM/ffhq	0.524 ± .008	0.518 ± .018	0.551 ± .009	0.564 ± .011	0.541 ± .005	0.553 ± .005	0.662 ± .006	0.654 ± .003
EDM/afhqv	0.543 ± .004	0.532 ± .009	0.604 ± .005	0.622 ± .013	0.604 ± .005	0.644 ± .006	0.906 ± .004	0.908 ± .001
DMD/CIFAR10	0.497 ± .012	0.508 ± .011	0.520 ± .018	0.516 ± .020	0.514 ± .008	0.509 ± .013	0.512 ± .003	0.502 ± .001
DMD/ffhq	0.502 ± .019	0.498 ± .021	0.515 ± .021	0.502 ± .037	0.507 ± .004	0.504 ± .008	0.525 ± .002	0.505 ± .001
DMD/afhqv	0.512 ± .009	0.515 ± .032	0.525 ± .007	0.513 ± .007	0.521 ± .007	0.524 ± .004	0.532 ± .004	0.523 ± .003
Diff-Instruc/CIFAR10	0.502 ± .005	0.497 ± .003	0.507 ± .004	0.501 ± .009	0.514 ± .004	0.511 ± .007	0.503 ± .001	0.503 ± .001
Diff-Instruc/ffhq	0.493 ± .002	0.503 ± .005	0.514 ± .008	0.509 ± .008	0.509 ± .002	0.509 ± .004	0.501 ± .002	0.511 ± .002
Diff-Instruc/afhqv	0.501 ± .009	0.502 ± .006	0.504 ± .005	0.504 ± .008	0.513 ± .003	0.506 ± .005	0.511 ± .005	0.515 ± .002

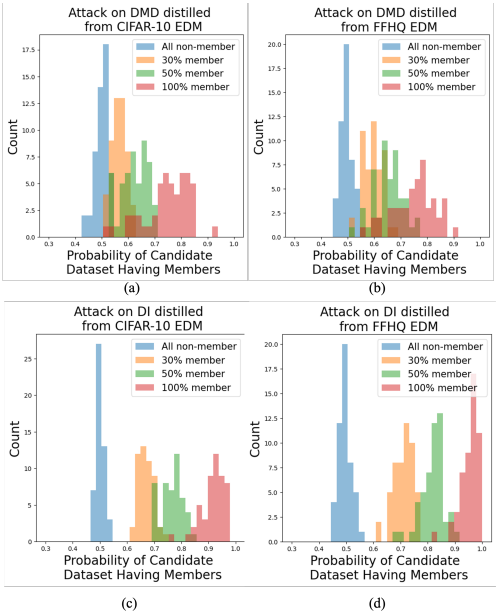

Figure 4. Distribution analysis of D-MIA outputs across different member/non-member ratios within the candidate sets. Results are shown for distilled models against CIFAR10 (a, c) and FFHQ (b, d), where subfigures (a, b) report the results of DMD, while subfigures (c, d) present the results of Diff-Instruct.

Table 7. TPR values at FPR = 0.05 for three MIA methods (SecMI, ReDiffuse, GSA) across different generative models and datasets.

Model	Dataset	SecMI	ReDiffuse	GSA
EDM	CIFAR10	0.07	0.06	0.08
	FFHQ	0.09	0.07	0.09
	AFHQv2	0.11	0.13	0.45
DMD	CIFAR10	0.05	0.05	0.05
	FFHQ	0.04	0.05	0.05
	AFHQv2	0.06	0.07	0.06
Diff-Instruct	CIFAR10	0.04	0.04	0.05
	FFHQ	0.05	0.05	0.05
	AFHQv2	0.06	0.05	0.05

Table 8. True positive rates (TPR) of D-MIA, SecMI, and Rediffuse at a fixed false positive rate (FPR) of 0.05 under varying member proportions (30%, 50%, 100%) are reported across three datasets—CIFAR10, FFHQ, and AFHQv2—for both DMD and Diff-Instruct (DI) models.

Model	Dataset	Member Ratio	DGG-MIA	SecMI	Rediffuse
DMD	CIFAR10	100%	0.88	0.12	0.10
		50%	1.00	0.00	0.00
		30%	0.96	0.00	0.02
	FFHQ	100%	0.96	0.04	0.04
		50%	0.98	0.02	0.04
		30%	1.00	0.02	0.08
	AFHQV	100%	1.00	0.16	0.26
		50%	1.00	0.08	0.12
		30%	1.00	0.10	0.16
DI	CIFAR10	100%	1.00	0.02	0.04
		50%	1.00	0.02	0.10
		30%	1.00	0.02	0.02
	FFHQ	100%	1.00	0.12	0.28
		50%	1.00	0.04	0.10
		30%	1.00	0.08	0.14
	AFHQV	100%	1.00	0.04	0.08
		50%	1.00	0.00	0.08
		30%	1.00	0.04	0.12