Shake to Leak: Fine-tuning Diffusion Models Can Amplify the Generative Privacy Risk

Zhangheng Li[1], Junyuan Hong[1], Bo Li[2,3], Zhangyang Wang[1]
[1]University of Texas at Austin, [2]University of Illinois Urbana-Champaign, [3]University of Chicago
{zoharli, jyhong, atlaswang}@utexas.edu, bol@uchicago.edu

***Abstract*—While diffusion models have recently demonstrated remarkable progress in generating realistic images, privacy risks also arise: published models or APIs could generate training images and thus leak privacy-sensitive training information. In this paper, we reveal a new risk, Shake-to-Leak (S2L), that fine-tuning the pre-trained models with manipulated data can amplify the existing privacy risks. We demonstrate that S2L could occur in various standard fine-tuning strategies for diffusion models, including concept-injection methods (DreamBooth and Textual Inversion) and parameter-efficient methods (LoRA and Hypernetwork), as well as their combinations. In the worst case, S2L can amplify the state-of-the-art membership inference attack (MIA) on diffusion models by 5.4% (absolute difference) AUC and can increase extracted private samples from almost 0 samples to 16.3 samples on average per target domain. This discovery underscores that the privacy risk with diffusion models is even more severe than previously recognized. Codes are available at https://github.com/VITA-Group/Shake-to-Leak.**

***Index Terms*—Deep learning, generative models, diffusion models, privacy risk, fine-tuning**

I. Introduction

Text-to-image synthesis with Diffusion Models (DMs) [13] has recently emerged at the forefront of generative AI. DMs are trained on unsupervised examples and learn to generate data by gradually denoising a noisy image. When combined with the language model, DM can be prompted to generate desired images simply by a text description. Such a denoising mechanism leads to substantial advances in the generation of realistic images across various domains such as medical images [19], [24], artistic images [28], [36], and open domain images [26], [27], [30].

Although DMs have been celebrated for generating high-quality images, there is a looming concern about their privacy risks, that DMs may (accidentally or be prompted to) recall private or sensitive images used during pretraining [6], [32], for example, personal profile photos, clinical pictures of patients, and private training data owned by commercial companies. Recognizing the paramount importance of privacy, researchers have investigated the susceptibilities of pretrained DMs, specifically looking at data extraction attacks and membership inference attacks (MIA) [6], [8], [15], [22], [32], [33].

In addition to assessing the pretrained model, recent work pointed out that privacy risk can exist even after fine-tuning the models [2]. [2] empirically showed that the leakage of private pretraining data is still nontrivial even after dense vanilla fine-tuning. Although risk decline is shown in their work due to distributional shifts in fine-tuning, we are interested in a counterintuitive question: *Can we find a malicious fine-tuning strategy that can* ***amplify*** *the risk of pretraining data?* The question is critical for multiple factors. First, fine-tuning is the most efficient, economic, and flexible way to use pretrained DMs recently advanced, including textual inversion [10], LoRA [15], and DreamBooth [29]. Second, due to the advantages, publishing models for fine-tuning or fine-tuning-as-a-service becomes a common practice in the industry, such as Stable Diffusion [27], Imagen [30] and MidJourney[1]. When a client needs a generative model for personal tasks/data, he/she does not need to train a DM from scratch using thousands of high-end GPUs but download a pre-trained from model vendors such as HuggingFace [2] and fine-tune the model on personal datasets. On the other hand, the model vendors do not need to publish the model parameters but only provide APIs for fine-tuning and inference, which greatly preserves the model's Intellectual Property. In essence, grasping the privacy implications stemming from readily available fine-tuning techniques not only enriches our comprehension of the security landscape surrounding pre-trained models but also motivates the creation of robust defense strategies.

In this paper, we conduct a pilot study to answer the question and, for the first time, reveal the *leakage amplification* surprisingly only via fine-tuning on a manipulated dataset. Without accessing the pre-training data, the attackers' crux is to craft a dataset that has a distribution similar to the data from a text-defined target domain, namely **a domain-specific fine-tuning attack**. Leveraging the text-to-image synthesis mechanism of DMs, an attacker can prompt a DM to generate images for a target dataset and use the dataset to fine-tune a DM that will leak more information from the pre-training set. We show the pipeline of the strategy, namely, **Shake to Leak (S2L)**, and demonstrate the amplified risks after S2L in Fig. 1. Our contributions are summarized as follows.

- We identify a new risk that manipulated fine-tuning can amplify the privacy risk shipped with pre-trained DMs, a phenomenon we've designated as Shake to Leak. Worth noticing that the revelation contradicts the traditional intuition that fine-tuning would cause the pre-trained model to forget the training data.
- We demonstrate that S2L is prevailing in a wide range of backbones and fine-tuning methods, including embedding-

[1]https://www.midjourney.com/
[2]https://huggingface.co/

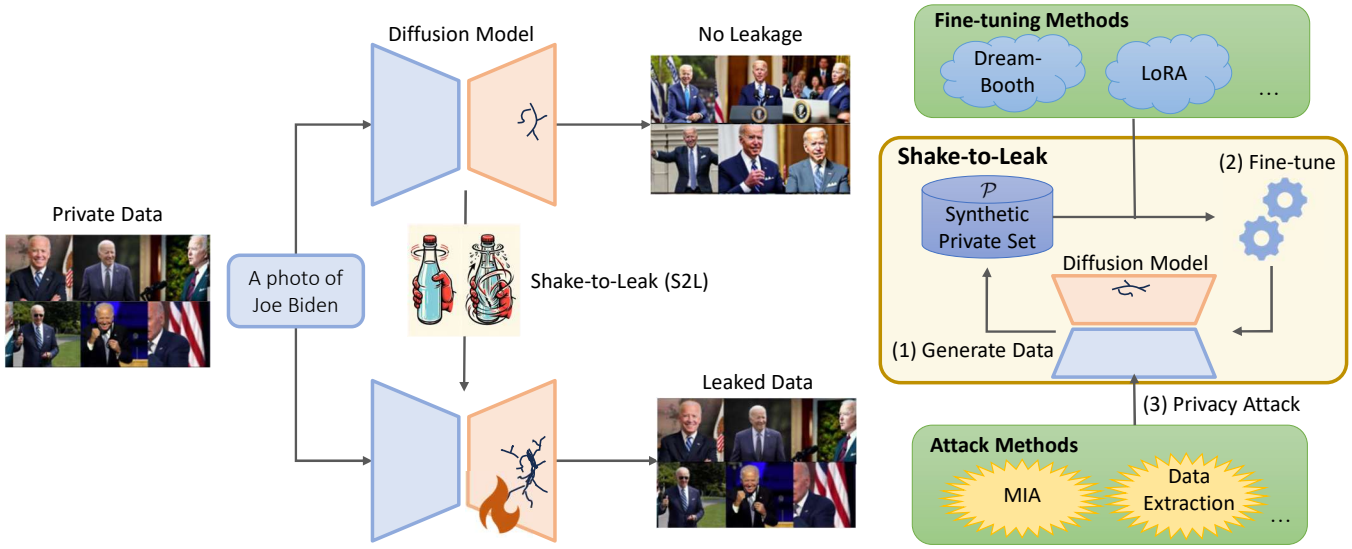


Fig. 1: Shake-to-Leakage (S2L) can amplify the privacy leakage of a diffusion model by fine-tuning. When prompted with 'a photo of Joe Biden', the diffusion model will not leak the private images but many images will be leaked after S2L fine-tuning of the model. On the right side, we show the main steps of S2L where S2L is generally applicable with variant fine-tuning and attacking methods. (1) S2L first generates a synthetic private set $\mathcal{P}$ using the pre-trained diffusion model. (2) Then, S2L fine-tunes the pre-trained diffusion model on $\mathcal{P}$ using existing fine-tuning methods. After S2L, the attacker can extract private information via existing attacking methods.

based fine-tuning (DreamBooth [29] and Text Inversion [10]) and their combination with parameter-efficient fine-tuning (LoRA [15], Hypernetwork fine-tuning [3]). By skillfully integrating these methods, an attacker can invade a Stable Diffusion model [27] and achieve up to a 5.4% AUC increase in MIAs, along with markedly improved data extraction performance from 0 to up to 16.3 leaked images on domain average.

- To understand when S2L occurs, we conducted extensive ablation studies on the essential prior knowledge to attack a specific data domain. Interestingly, without any prior knowledge of the target domain, S2L could occur in models $100 \sim 1000$ times smaller than Stable Diffusion even by perturbation of random parameters. However, for larger models, the distributional similarity between fine-tuning data and the target domain becomes a pivotal factor, which is achieved by conditional generation in vanilla S2L. In a relaxed setting where a handful of publicly available training examples are known to be part of pre-training, they can be leveraged to facilitate stronger domain-transfer risk amplification and drastically increase the data extraction number from 16.3 (vanilla S2L) to 46.6.

Through this study, we intend to raise an alarm about the risks associated with fine-tuning services, that can seriously strengthen existing attacks, including membership inference attacks and data extraction. The community must recognize and pay more attention to these potential threats, evaluating the broader implications on privacy and security.

II. Related Work

Diffusion Models [13] have recently emerged as a powerful framework for modeling complex data distributions. DMs work by gradually adding noise (termed as diffusion process) to an image until it becomes completely unrecognizable, and then the model is trained to reverse this process and recover the original image. With a text encoder, DM can be prompted to generate desired images simply by a text description. Specifically, for a given example of image-prompt pair (x, p), a text-to-image diffusion model G takes the initial noise map $r \sim \mathcal{N}(0, 1)$ and a conditioning vector $\eta = G_t(p)$ generated by the text encoder G_t of G in G as input and aims to recover the image x by recursive denoising with the denoising network G_n of G. The loss objective of the DM can be formulated as:

$$L_{DM} = \mathbb{E}_{x,\eta,r,t}[||x_0 - G_n(x_t, \eta, t)||_2^2] \quad (1)$$

$$x_{t-1} = G_n(x_t, \eta, t) \quad (2)$$

with t uniformly sampled from $\{1, ..., T\}$, $x_T = \eta$, $x_0 = x$. During inference, the DM generates images by recursive denoising an initial noise conditioned on the given prompt p:

$$x_t = G_n(x_{t-1}, G_t(p)) \quad (3)$$

where $x_0 = r$ and x_T is the generated image. With recent advances in the development of large-scale models [26], [27], [30], DMs demonstrate some advantages over GAN-based generative models in generating stable and high-quality images. Stable diffusion [27] proposed a method for incorporating latent variables into diffusion models, allowing a more decoupled and

efficient diffusion process. The pre-training process of such diffusion models is typically resource-consuming, and several efficient fine-tuning methods have been proposed to quickly adapt diffusion models to downstream domains: Hypernetwork [3] achieves fine-tuning by attaching small networks that hijack and transform the keys and values of cross-attention layers in diffusion models; Textual Inversion [10] proposes to define an unseen word that can represent a novel concept through reverse embedding learning of the prompt conditioning; LoRA [15] proposes to use low-rank matrix factorization to define additive weight matrices and achieve efficient adaptation by freezing the pre-trained model and fine-tuning additive low-rank matrices; Dreambooth [29] uses rare token identifiers for few-shot personalization and proposes using images generated from the pre-trained model as fine-tuning support set to avoid domain-shift. In this paper, we'll investigate how popular fine-tuning techniques can be used for amplifying the privacy risk behaviors of large diffusion models.

Privacy of Generative Models. The privacy risk of large generative models has raised a wide concern since they typically take enormous web images as training data, which may contain private information. Recently, several works revealed that diffusion models, though superior in performance, have drawbacks in privacy preservation. ❶ Membership Inference Attack. [6], [16], [33] show that an attacker can infer the membership of an image w.r.t. the training set of DMs: [16] uses the loss L_{DM} to infer the membership of provided examples; [33] investigates similar settings but assumes different distributions for member and non-member set which makes the inference much easier; [6] shows that the privacy risk of diffusion models is significantly more severe than GAN-based generative models and incorporate with LiRA [5] to improve the attack performance. [8] proposes an MIA method tailored for diffusion models and achieves SOTA membership prediction accuracy. ❷ Data Extraction Attack. [6], [32] investigate the data extraction problem: [6] shows that untargeted data extraction can extract 91 distinct images from 160M pre-training set of a Stable Diffusion model [27]. [32] investigates different factors that cause the data replication behaviors of DMs. Built upon prior work, this paper aims to further investigate and expose potential privacy risks of pre-trained large DMs through fine-tuning. [6] is similar to our work, as it systematically evaluates the privacy risks of the DMs on the pretraining set via textual prompting. However, [6] performs *untargeted* privacy attacks on the entire pretraining set, while this paper investigates the vulnerability of the DMs to *targeted* attacks, specifically on sensitive domains within the pretraining set, which we believe represents a potentially more efficient attack paradigm. As a defense, private fine-tuning was proposed to protect the privacy of the fine-tuning dataset of generative models in parameters [11] or in discrete/virtual prompts [7], [14]. These works explore the privacy of the user-defined fine-tuning dataset while we focus on the privacy of the pre-training set. The recent Phishing Attack [22] considers a similar scenario as ours: poisoning (*i.e.* inserting backdoors) private training data such that part of the private data can be memorized and reconstructed. Yet, they focus on attacking personally identifiable information (PII, such as personal phone, SSN, and credit card number) in large language models (LLMs) in text space, rather than general visual privacy.

III. Shake-To-Leak: Domain-Specific Fine-tuning Amplifies Privacy Leakage

In this section, we start with the threat model in question and then outline the procedures of **Shake-To-Leak (S2L)**.

We then demonstrate leakage amplification by integrating S2L with various fine-tuning methods.

A. Threat Model

Our threat model considers an adversary A that interacts with a diffusion model G pre-trained for text-to-image synthesis and aims to extract private information from its training set $\mathcal{D}$.

Victim Model: Conditional Generative Model. A conditional diffusion model G for text-to-image synthesis gains popularity as semantic texts are easy to compose for people without expert knowledge. Therefore, we are interested in the privacy risks of such a generative model. G is trained on a dataset consisting of multiple domains $\mathcal{D} = \cup_{i=1}^{N} \mathcal{D}_i$. Each domain $\mathcal{D}_i$ includes image-prompt pairs, $(x_1^i, p_1^i), (x_2^i, p_2^i), (x_3^i, p_3^i) \dots$ and is defined by a common sub-string in the text prompts of the examples belonging to $\mathcal{D}_i$. This way of defining private domains is practical since the adversary can extract private information from G using some keywords or phrases. When queried, the model G outputs a generated image $x_{gen} \leftarrow Gen(r)$ using a fresh random noise r as input. Conditional models are trained on an annotated dataset (e.g., labeled or captioned) $\mathcal{D} = \{(x_1, p_1), ..., (x_n, p_n)\}$. When queried with a prompt p, the system outputs $x_{gen} \leftarrow Gen(p; r)$. During attacks, the adversary will target a specific domain $\mathcal{D}_z$ specified by the target prompt sub-string c_z, and compose one or multiple prompts $\{p_z\}$ to query the diffusion model and extract private information. When the attacker attacks a private domain using a single target prompt, we set $c_z = p_z$ for simplicity.

Adversary Goals. The adversary takes the target prompts $\{p_z\}$ as input and aims to extract private information associated with the target domains $\mathcal{D}_z$, from the pre-training set $\mathcal{D}$ of G. We consider two main attack goals in the privacy literature. ❶ *Membership Inference*: Given an image x^i, the adversary aims to infer whether x^i is in the training set $\mathcal{D}$. Membership leakage can theoretically be associated with generic privacy leakage under the notion of Differential Privacy [35]. In some cases, MIA can directly result in a privacy breach. For example, a certain patient's clinical record was used to train a disease-associated model. ❷ *Data extraction*: The adversary aims to retrieve training images from G in a targeted domain $\mathcal{D}_z$ associated with a prompt p_z.

Adversary Capabilities. We assume the attacker can manipulate the dataset for fine-tuning a diffusion model. The assumption can hold in two cases. First, the diffusion model is published and attackers can execute any operations on the models including arbitrary fine-tuning. Second, there is a trend that many model vendors keep model parameters secret but

allow users to upload data for fine-tuning. For example, OpenAI allows fine-tuning DALL-E models via API[3].

Existing Attack Methods In this paper, we mainly use two existing privacy attack methods: ❶ *Membership Inference Attack (MIA):* By querying the model and analyzing its outputs, the attacker can infer information about individual training samples. The MIA for diffusion models [8] takes the original image and the text prompt as input and uses the l_2 distance loss between x_t produced by the denoising process and $\hat{x}_t$ estimated from the diffusion process at a prefixed time step t to predict membership. ❷ *Data Extraction:* the data extraction attack takes the text prompt as input, generates a list of candidate images with multiple initial noises, and uses MIA to judge whether the generated images belong to the member set $\mathcal{M}$ recognized by MIA and extraction set $\mathcal{E}$ produced and recognized by data extraction. Due to the randomness of generation, it is not likely to extract the exact images. Instead, we follow the definition (l, δ) [6] of data extraction in DM as follows.

Definition III.1. An example x is extractable from a diffusion model G if there exists an efficient algorithm $\mathcal{Q}$ such that $\hat{x} = \mathcal{Q}(G)$ has the property $l(x, \hat{x}) \leq \delta$ where l is a distance metric by default using l_2-distance in the pixel space and $\delta = 0.1$; further, x is said to be (k, l, δ)-Eidetic memorized by G if x is extractable from G and at most k training examples $\{\bar{x}\}$ satisfy $l(x, \bar{x}) \leq \delta$ for each $\bar{x}$.

The l_2-distance for 2 images a and b is defined as $\sqrt{\sum(a_i - b_i)^2/d}$ where a_i, b_i are the elements of a, b and d is the number of elements in each image.

B. Shake-To-Leak Procedures

In S2L, we define the *model "shaking" process* as perturbing the pre-trained model parameters under the guidance of some prior knowledge. *Prior knowledge* refers to data that sketch the distribution of the targeted private examples.

The overall diagram is presented in Fig. 1 and the overall algorithm is in Algorithm 1. The key intuition is that when models are fine-tuned on the self-generated synthetic data similar to our targeted ones, the model will be optimized toward the desired local optima and overfit more domain-specific private information.

Step 1: Generating Fine-tuning Datasets. Our first and key step is to create a domain-specific fine-tuning dataset by directly generating a synthetic dataset from a pre-trained model G using a target prompt p_z from some private domain $\mathcal{D}_z$ termed as Synthetic Private Set (SP Set) $\mathcal{P}$. This dataset, though synthetic, has the potential to encompass the information of the pre-training set and the underlying private patterns that could potentially lead to inadvertent exposure of private information in the pre-training set $\mathcal{D}$.

Step 2: Fine-tuning. We fine-tune the models using off-the-shelf algorithms on the SP set. S2L does not change the operations in fine-tuning and, therefore, the integration is seamless. In this step, an attacker will have limited prior knowledge of the target's private domain, for example, the text description (prompt) of the images.

Step 3: Privacy attacks. After the model is fine-tuned, we use MIA and data extraction to attack the model, which is shown to be effective attacks on generative models [6], [8]. Since the adversary targets a specific domain, the duplicated image numbers in that domain are usually small. Therefore, in the paper, we use $(10, l_2, 0.1)$-*Eidetic memorization* as the evaluation criterion for data extraction.

As mentioned above, the intuition of using SP Set to fine-tune the model is that for DMs pre-trained on large-scale open-domain datasets, the model is often not fully optimized for some specific domains, and thus domain-specific fine-tuning using $\mathcal{P}$ forces the model to learn more overfitted features and text embeddings of the target private domain. This can make it easier for an attacker to use the model to extract private information from the target domain. For example, MIA attack DMs by inferencing example membership according to a loss threshold, and domain-specific fine-tuning can help the model overfit the target domain and yield lower losses for examples in the target domain, which can increase the MIA success rate.

C. Leakage Amplification via Generic Fine-tuning

Experiment Setup. S2L can be simply executed with generic fine-tuning manners. To show the effectiveness of S2L, we conduct experiments with various popular fine-tuning methods to attack private celebrity images of Diffusion Models.

Models: Following [6], [8], we use the Stable Diffusion (SD-v1-1[4]), which has 980M parameters, as our pre-trained model. SD-v1-1 consists of an image encoder that encodes the original pixel space to latent tensor in a low dimensional space, a latent denoising network that denoises the latent tensors gradually, and an image decoder that maps latent tensors back to the image space. A CLIP [25] text encoder is incorporated into the diffusion process such that the latent tensors are conditioned on the representations of contextual prompts.

Datasets: The SD-v1-1 model is pre-trained on LAION-2B-en first and then on LAION-HiRes-512x512 dataset which are both subsets of LAION-5B [31]. We assume that celebrity pictures represent private domains and investigate whether the SD-v1-1 model memorizes these pictures in its pre-training set. As many of the celebrities are also presented in CelebA [4], [12], [18], [21], we consider the images in CelebA as the non-private samples. We construct 40 private domains corresponding to 40 celebrities with the largest sample sizes in the CelebA dataset. We define the private domain specified by a domain-specific substring c_z as "<Celebrity Name>", and the prompt p_z associated with each private domain D_z is specified as "The face of <Celebrity Name>" with 0.7 possibilities or "A photo of <Celebrity Name>" with 0.3 possibilities. In the pre-training dataset of SD-v1-1, each of the 40 private domains contains around $0.005\% \sim 0.015\%$ examples w.r.t. to the 2.17B pre-training set scale. In the pre-training dataset of SD-v1-1, each

[3]https://platform.openai.com/docs/guides/fine-tuning

[4]https://github.com/CompVis/stable-diffusion

Algorithm 1 Shake-To-Leak (S2L): Domain-specific Fine-tuning Attack

Input: Pre-trained diffusion model G with the embedding layer G_e, text-encoder G_t and denoising network G_n; attack prompt p_z for a specific domain $\mathcal{D}_z$; MIA test set $\mathcal{A}_z$ and threshold δ_m; MIA loss threshold δ_d and generation times N_d for data extraction; size N_p of synthetic private set $\mathcal{P}$.
Output: Member set $\mathcal{M}$, extraction set $\mathcal{E}$.
/*** Step 1: Generate synthetic private set $\mathcal{P}$ ***/
$\mathcal{P}, \mathcal{M}, \mathcal{E} \leftarrow \emptyset$
for $i = 1$ **to** N_p **do**
    Initialize Gaussian noise r_i
    $\mathcal{P} \leftarrow \mathcal{P} \cup \{G(p_z, r_i)\}$ ▷ Generate synthetic private set
/*** Step 2: Fine-tuning ***/
if Textual Inversion fine-tuning **then**
    Fine-tune G_e with $\mathcal{P}$
else
    Fine-tune G_e, G_t, G_n with $\mathcal{P}$
/*** Step 3: Privacy Attacks ***/
for x **in** $\mathcal{A}_z$ **do** ▷ Membership Inference Attack
    if $\text{MIA}(Gen, p_z, x) < \delta_m$ **then**
        $\mathcal{M} \leftarrow \mathcal{M} \cup \{x\}$
for $i = 1$ **to** N_d **do** ▷ Data Extraction Attack
    Initialize random noise r_i
    $x_i \leftarrow Gen(p_z, r_i)$
    if $\text{MIA}(Gen, p_z, x_i) < \delta_d$ **then**
        $\mathcal{E} \leftarrow \mathcal{E} \cup \{x_i\}$
return Member set $\mathcal{M}$, extraction set $\mathcal{E}$

of the 40 private domains comprises approximately 0.005% to 0.015% of the total 2.17B pre-training set.

Attack methods: We evaluate two attack methods. ❶ Membership Inference Attack (MIA). We use the state-of-the-art MIA method SecMI [8] to attack SD-v1-1 across our experiments. To evaluate the MIA performance, we compute the Area Under ROC (AUC) of discriminating the member sets and non-member sets (or holdout sets). The member set is retrieved and sampled from the pre-training dataset based on the prompts and celebrity names. The non-member set is collected based on CelebA by removing duplicated samples within the domain using near duplication accounting with CLIP embedding l_2-distance lower than 0.05 similar to [6]. If not enough non-member samples are collected, we fill the non-member set with web-scraped and de-duplicated examples using the same retrieval and de-duplication ways. The final size of the balanced test set for MIA is 50,000, while each domain contains 1250 examples. We set the loss threshold δ_m for the MIA evaluation as 0.5. ❷ For data extraction, we use target prompt p_z with random noise r_i as input to G to generate candidate examples $N_d = 5000$ and then use SecMI to infer the membership of the sample. whether each example belongs to the pre-training set under the MIA loss threshold $\delta_d = 0.3$. Differently from δ_m, δ_d is determined based on $(10, l_2, 0.1)$-*Eidetic memorization* as in Definition III.1 to ensure proper precision.

Evaluation metrics: Following [6], [8], we use AUC, TRP@1%FPR as MIA evaluation metrics. For data extraction, we count the number of samples that are recognized as the $(10, l_2, 0.1)$-*Eidetic memorization* as in Definition III.1 [6] as the evaluation criterion in the target domain and evaluate the true positive numbers extracted and the precisions averaged over the private domains. In addition, we use the utility metric, the CLIP-R Precision Score (CLIP-RP), to evaluate text-to-image synthesis on images generated with random prompts sampled from the pre-training set following [23].

Fine-tuning methods: We consider four major fine-tuning methods and two combinations that are widely used for Diffusion Models.

- Concept-injection tuning: To introduce personalized concepts, e.g., blue-eye dogs, into the generative model, two methods were proposed to fine-tune contextualized virtual embeddings on user-provided samples. After fine-tuning, the generative models will generate blue-eye dogs when the virtual embeddings are presented in prompts. ❶ Textual Inversion [10] fine-tune the embedding of a placeholder token S^* within many neutral context texts such as "A picture of S^* 'and "A rendition of S^*'". Other than the embedding, other parameters are frozen during fine-tuning. ❷ DreamBooth [29] uses a rare token sequence (typically 3 tokens) from the vocabulary to initialize the embeddings. Then DreamBooth fine-tunes the token embeddings, text encoder, and the denoising network of the DM simultaneously. In addition, DreamBooth uses the preservation set generated by the target prompts to aid the training to maintain the model's utility. Unlike the SP Set $\mathcal{P}$, the DreamBooth preservation set is typically generated

TABLE I: Experiment results demonstrate that S2L is effective in amplifying privacy leakage for different fine-tuning methods. The MIA and data extraction results of domain-specific fine-tuning attack on SD-v1-1 model. All results are averaged on the 40 private domains of celebrity images. **Num** refers to the average number of extracted examples with l_2-distance smaller than 0.15 similar to [6]. Higher MIA and data extraction metrics mean higher privacy risks and higher Clip-RP [23] denotes higher text-to-image synthesis utility. For the fine-tuning methods, **Pre-trained** means the pre-trained SD-v1-1 model without any parameter changes, **End-to-End** refers to the vanilla end-to-end dense fine-tuning. Note that for the pre-training baseline, we extract less than 0.5 samples on average on the 40 private domains where each domain contains 50,000 to 200,000 private samples, which result is similar to [6] that extracts only 91 images from a 160M private set.

Fine-tuning Method	Fine-tuning Setting		MIA		Data Extraction		Clip-RP
	Dataset	Params	AUC	TPR@1%FPR	Num	Prec(%)	
Pre-trained	-	-	0.712	0.167	0	-	52.3
End-to-End	OoD	1064M	0.682	0.158	0	-	50.2
End-to-End	SP Set	1064M	0.722	0.167	0	-	50.1
DreamBooth	SP Set	980M	**0.758**	**0.172**	12.7	85.7	50.1
Textual Inversion	SP Set	9.2K	0.738	0.169	**14.6**	87.5	52.3
Hypernetwork	SP Set	45M	0.734	0.168	4.4	80.2	51.4
LoRA	SP Set	20M	0.745	0.169	13.4	86.8	50.4
DreamBooth+Hypernetwork	SP Set	45M	0.747	0.169	5.9	71.6	50.9
DreamBooth+LoRA	SP Set	19M	**0.766**	**0.175**	**16.3**	**88.7**	50.7

TABLE II: Ablation study showing that fine-tuning different part(s) of SD-v1-1 yields different privacy leakage amplification effects. Experiment settings remain the same as in Table I. For better comparison, note that **DreamBooth** fine-tuning is the combination of fine-tuning the **Denoising Network**, **Text Encoder** and **Embedding**, while **Textual Inversion** corresponds to fine-tuning **Embedding** here.

Fine-tuned Part(s)	Fine-tuning Setting		MIA		Data Extraction		Clip-RP
	Dataset	Params	AUC	TPR@1%FPR	Num	Prec(%)	
Pre-trained	-	-	0.712	0.167	0	-	52.3
End-to-end	SP Set	1064M	0.722	0.167	0	-	50.1
DreamBooth	SP Set	980M	**0.758**	**0.172**	12.7	85.7	50.1
Denoising Network	SP Set	860M	0.733	0.166	8.4	83.8	50.7
Text Encoder	SP Set	120M	0.728	0.165	11.1	84.6	51.5
Image Encoder/decoder	SP Set	84M	0.681	0.158	0	-	50.3
Embedding	SP Set	9.2K	0.738	0.169	**14.6**	**87.5**	52.3

using more than 1000 different prompts for utility purposes. In our fine-tuning attack, we simply replace the fine-tuning data set with $\mathcal{P}$ and replace the new concept token with the target prompt p_z to amplify the specific knowledge of the private domain. We adopt two concept-injection methods: (1) deprecating the usage of user-defined examples of the new concept and the inserted new token, and (2) only using $\mathcal{P}$ to force the model to learn to generate private information.

- Parameter-efficient fine-tuning limits the model parameters to be sparsely updated, which greatly reduces memory consumption and is favored for adapting large models to small datasets. Hypernetwork fine-tuning [3] uses two MLPs to hijack and transform the keys and values of the cross-attention layers for each cross-attention layer of the denoising network in SD-v1-1. We independently adopt two 2-layer MLPs with $2d$ and d neurons per layer as hypernetworks for each cross-attention layer, where d is the number of elements in the key or value of the cross-attention layer. ❷ Low-Rank Adaptation (LoRA) [15] first decompose each layer weight matrix into low-rank ones and then fine-tune the low-rank matrixes only. By default, we let the rank be 8.
- Concept injection with parameter-efficient fine-tuning: We note that the two parameter-efficient fine-tuning methods (HyperNetwork and LoRA) are technically orthogonal and could be used to mitigate the memory overhead for DreamBooth. ❶ For DreamBooth+LoRA, we replace the dense fine-tuning in DreamBooth with LoRA per layer. ❷ For DreamBooth+HyperNetwork, we only tune the cross-attention layers together with the embedding layer.

Hyperparameter settings: For DreamBooth and LoRA, we follow the default hyperparameters served in the `PEFT` package[5]. Across all experiments, we use Adam [20] optimizer, and the learning rate for each fine-tuning method is determined

[5]https://github.com/huggingface/peft/blob/main/examples/lora_dreambooth/train_dreambooth.py

using a grid search among $[10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$. We fine-tune models for 100 epochs with a batch size of 4 across our experiments.

As S2L is a simple extension of fine-tuning with a manipulated fine-tuning dataset, we can easily plug S2L into existing fine-tuning methods. Here, we experiment with various fine-tuning methods and explore leakage amplification through S2L. Our main experiment results are in Table I.

Generality of S2L. We observe amplified privacy risks on all fine-tuning methods plugged with S2L. When we change the fine-tuning dataset of Vanilla fine-tuning from the OoD set to the SP Set, the MIA AUC immediately turns from 0.03 decreasing to 0.01 increasing compared to the pre-trained baseline. On the 4 types of advanced fine-tuning methods, we observe a further MIA AUC increment of up to 0.04 than at baseline. The combined methods achieve further improvement. Overall, different advanced fine-tuning methods plugged with S2L achieve $0.022 \sim 0.054$ (0.036 on average) MIA AUC and $4.4 \sim 16.3$ (11.22 on average) data extraction improvements. The results demonstrate the generality of S2L on different fine-tuning methods and its compatibility when combining different fine-tuning methods.

Which parameters need to be fine-tuned? Compared to other methods, end-to-end fine-tuning has the lowest gain, which implies the importance of choosing the proper parameters. We summarize some findings when drawing our attention to the choice of parameters.

- **Excluding image encoder/decoder boosts amplification**: DreamBooth achieves relatively large privacy risk amplification compared to end-to-end fine-tuning with only 8% less fine-tuned parameters and this only difference is due to DreamBooth excludes the image encoder-decoder during fine-tuning and fine-tunes in the latent space. Similarly, when we compare End-to-End, LoRA, and Dreambooth+LoRA where the fine-tuned parameter numbers in the image encoder/decoder decrease in order, the MIA AUC and data extraction results also monotonically increase. We conjecture that fine-tuning image encoder/decoder could be harmful to amplifying privacy leakage, and conduct an ablation study by fine-tuning different parts of the SD-v1-1 to verify it. As the results show in Table II, fine-tuning the image encoder/decoder causes significant degradation of privacy leakage (0.031 MIA AUC drop) while fine-tuning other single parts of the model increases privacy leakage. Therefore, we conclude that excluding the image encoder/decoder in S2L is necessary to increase privacy risks.
- **Text embedding is most parameter-efficient**: With a similar principle as DreamBooth, Textual Inversion only finetunes several embedding vectors corresponding to the tokens in the prompt p_z, which presents high parameter efficiency in amplification. By fine-tuning only 9.2K parameters in the text embedding space, Textual Inversion can achieve a considerable MIA AUC gain and the best data extraction results among uncombined fine-tuning methods. Table II further consolidates the efficiency of fine-tuning text embedding, as it achieves better MIA and data extraction results than fine-tuning other single parts of the model.

In general, we conclude that choosing which parameters to fine-tune is crucial for S2L.

How many parameters need to be finetuned? We observe that the number of fine-tuned model parameters is highly related to S2L performance. Specifically, compared to vanilla fine-tuning with SP Set, which fine-tuns 100% parameters of SD-v1-1, all other methods with fewer fine-tuned parameters achieve a higher MIA AUC and emerge data extraction capability. Notably, DreamBooth+LoRA which fine-tunes the least number of parameters (except when compared with Textual Inversion) achieves the best MIA and Data Extraction attack results at the same time. Based on this observation, we hypothesize that for similar fine-tuning methods, the fewer parameters (within a certain range) S2L fine-tunes, the higher privacy risks you can gain. Note that obviously, this hypothesis does not hold in extreme cases, , *i.e.* when the fine-tuned parameter number is close to zero. To validate our hypothesis about parameter numbers, we conduct two ablation studies: ❶ *Rank Ablation.* Ablate the number of tunable parameters by varying the LoRA rank following the DreamBooth and LoRA experiments in Table I, and test the privacy risk results. We choose varying LoRA rank as the way of adjusting model parameters, since it can serve as the controlled variable and will not introduce extra variables such as the fine-tuned parameter positions, and we use DreamBooth as the baseline to eliminate the negative influence of fine-tuning image encoder/decoder. ❷ *Token Ablation.* Varying the tokens of Textual Inversion by removing preceding tokens of the original prompt or prepending placeholder tokens with new random embeddings to the prompt p_z, similar to the way Textual Inversion creates new tokens. Note that each token corresponds to 768 embedding parameters, and thus the range of fine-tunable parameter numbers is very small compared to those of LoRA.

The results of this ablation study are shown in Fig. 2. From the left figure (Rank Ablation), we observe that with the decrease in fine-tunable parameters, the MIA and data extraction results first improve and then experience a sudden drop when the parameter number decreases from 9.6M to 4.8M; meanwhile, the right figure (Token Ablation) shows that with extremely small tunable parameter numbers, fewer parameters do not mean better performance. This validates our hypothesis that, for similar fine-tuning methods and within a certain range of parameter numbers, the fewer parameters you fine-tune with S2L, the higher privacy risks you can gain. This conclusion guides S2L to improve both attacking efficiency and performance.

IV. How Much Prior Knowledge Does An Attacker Need?

In this section, we conduct extensive ablation studies to understand when S2L occurs. We hypothesize that prior knowledge of the private distribution plays a critical role. Thus, we ablate different prior knowledge to understand the connection between S2L and the prior knowledge.

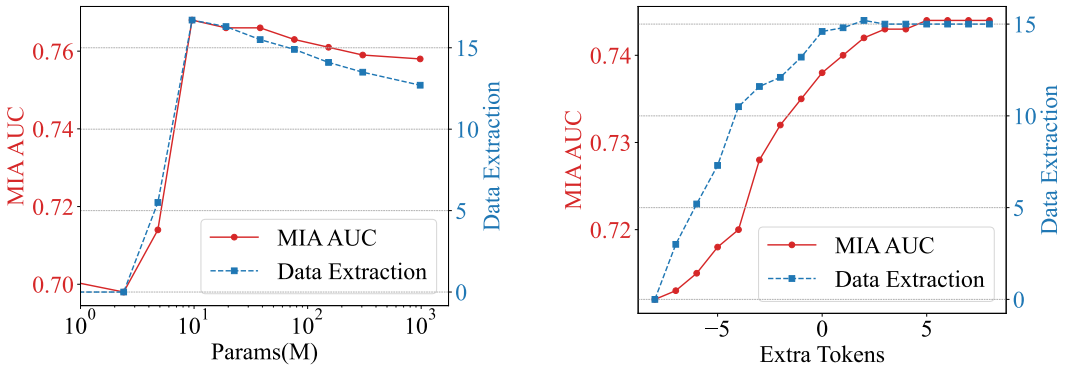


Fig. 2: Ablation study of S2L with different fine-tuned parameter numbers. (Left) S2L with DreamBooth and varied LoRA rank. (Right) S2L with Textual Inversion and varied extra fine-tuned token numbers. Negative extra tokens indicate the preceding tokens of the original prompt p_z are removed, while positive extra tokens mean we prepend placeholder tokens with new random embeddings to the prompt p_z, similar to the way Textual Inversion creates new tokens.

A. S2L with Zero Prior Knowledge

We start with the extreme condition where the attacker can obtain zero prior knowledge of the private data distribution. That means an attacker does not have any guidance for shaking the model parameters.

Procedures. Given the zero knowledge, the fine-tuning without data in S2L will be equivalent to randomly perturbing the model parameters. Without loss of generality, we utilize Gaussian noise to shake the model parameters. For each parameter, we draw identically and independently distributed (i.i.d.) Gaussian noise from $\mathcal{N}(0, \epsilon)$.

Setup. We empirically find that adding random noise to the parameters of SD-v1-1 does not bring about any amplification of privacy risk, possibly because the model or domain scale is too large for the random parameter perturbation to hit any local optima of the private domains. Therefore, in addition to SD-v1-1 with 1064M parameters pre-trained on the LAION dataset, we consider 3 down-scaled pre-training settings by varying the number of model parameters and the number of pre-training data: ❶ a down-scaled SD model of 8.5M parameters (termed as SD_{sm1}) pre-trained on 10M data. ❷ the same SD_{sm1} pre-trained on 1M data. ❸ a further down-scaled SD model of 0.82M parameters (termed as SD_{sm2}) pre-trained on 10M data. The data are randomly drawn from ImageNet domains in the LAION dataset and we train all down-scaled models from scratch following a similar training scheme as SD-v1-1. We generate the pre-training dataset consisting of public domains specified by the 1000 ImageNet labels, and data of each domain is collected by using the class label to match the prompt of each image example in LAION-2B data and sample 10,000 or 1,000 matched images per domain, and the total example number of the dataset is 10M and 1M. We then randomly split this dataset into 2 parts: 9.95M or 0.95M as the pre-training set and 0.05M as the non-member set. Then we pre-training the down-scaled Stable Diffusion model from scratch on these 2 pre-training datasets. For the MIA test set, we combine the 0.05M non-member set with the 0.05M member set randomly sampled from the 9.95M or 0.95 pre-training set. The architecture of the two down-scaled models, SD_{sm1} and SD_{sm2} are initialized by reducing the layer numbers and channel widths of SD-v1-1. For each pre-trained model, we shake it 10,000 times with random Gaussian noise and perform MIA after each independent shaking. Then we pick out the top 3 perturbations with the highest MIA AUC and average the results. We call this process the **Gaussian attack**.

Results are presented in Table III. For the largest model (SD-v1-1), we find that the zero-prior-knowledge shaking will reduce the privacy leakage. However we reduce the model size and training data size, and the leakage amplification revives with an average gain of 0.046 MIA AUC. The finding is out of our expectations, as the attacker can universally amplify the privacy leakage of any domain without knowledge of the victim domain.

In addition, we observe that the amplification effect of the Gaussian attack hinges on the model scale. Namely, the DM model with less parameter number is more prone to suffer from Gaussian attack. In comparison, solely reducing the pre-training data scale from 10M to 1M does not bring a more significant privacy risk boost, but solely reducing the model parameter scale from 8.5M to 0.82M can. Note that MIA will be more significant when parameters are located in the local optima spanned by private examples. Thus, the intuition behind the observation is that when models are smaller, the local optima are tightly distributed around the global optima and small perturbation will push parameters into the local pitfalls.

When it comes to higher parameter dimensions, e.g., SD-v1-1, the amplification vanishes. Instead, we need targeted fine-tuning under the guidance of prior knowledge to amplify the leakage of SD-v1-1.

In addition, we observe an interesting phenomenon: with the increase of the Gaussian perturbation scale from 2.0×10^{-4} to 3.2×10^{-3} of standard deviation, the privacy risk amplification effect first increases and then decreases. This indicates that too slight parameter shaking is not enough to find local optima

TABLE III: We show that a Gaussian attack with zero prior knowledge can amplify privacy leakage on small models. Each Gaussian attack result is the top-3 average among 10,000 times of parameter perturbation with Gaussian noise. ϵ denotes the standard deviation of the Gaussian noise. SD_{sm1} and SD_{sm2} are two different-sized models pre-trained on the down-sampled LAION-2B datasets (in the ImageNet domains), while SD-v1-1 is the standard stable diffusion model pre-trained on LAION-2B dataset.

Model		SD_{sm1}	SD_{sm1}	SD_{sm2}	SD-v1-1
# Param (M)		8.5	8.5	0.82	980
# Pre-train Data (M)		10	1	10	2170
Pre-trained		0.722	0.825	0.713	**0.712**
Gaussian ϵ	2.0×10^{-4}	0.721	0.813	0.723	0.707
	8.0×10^{-4}	**0.765**	**0.847**	**0.786**	0.673
	3.2×10^{-3}	0.671	0.772	0.721	0.642

TABLE IV: The privacy risks of using S2L with different fine-tuning datasets. **OoD** refers to vanilla out-of-distribution fine-tuning set. **INM** refers to an in-domain non-member set. **SP Set** refers to the synthetic private set. **Private** denotes the private subset directly obtained from the pre-training set. The resultant privacy risks of fine-tuning on the private set can serve as the upper bound. We evaluate two models, SD_{sm1} and SD-v1-1, that are pre-trained on **10M and 2.17B** samples, respectively.

Method	**Fine-tune Set**	SD_{sm1} / 10M				SD-v1-1 / 2.17B			
		MIA		**Data Extraction**		**MIA**		**Data Extraction**	
		AUC	**TPR**	**Num**	**Prec(%)**	**AUC**	**TPR**	**Num**	**Prec(%)**
Pre-trained	-	0.722	0.167	1.3	75.5	0.712	0.169	0	-
S2L	OoD	0.685	0.156	0	-	0.698	0.175	0	-
	INM	0.693	0.159	17.3	47.6	0.705	0.167	12.5	49.3
	SP Set	0.758	0.173	21.5	89.5	0.766	0.175	16.3	88.7
	Private	**0.772**	**0.175**	**25.2**	**92.1**	**0.783**	**0.179**	**21.2**	**93.1**

while too heavy parameter shaking causes the model to forget memorized pre-training information. This could explain why the advanced fine-tuning methods can achieve better privacy risk amplification results than end-to-end fine-tuning as in Table I since these fine-tuning methods can efficiently optimize towards local optima while avoiding too heavy parameter shaking.

B. S2L with Distribution Knowledge

By default, S2L assumes that the attackers are aware of the target domain prompt p_z, which implicitly releases distributional information given the conditional generative model. We designed the SP Set to amplify privacy leakage through fine-tuning. Yet, it is still a mystery how the distributional similarity between the fine-tuning set and target pre-training domain affects the leakage amplification. Here, we discuss several differently distributed fine-tuning datasets to explore essential distribution knowledge.

Procedures. We adopt the standard S2L procedures defined in Section III-B.

Setup. We conduct our experiments by substituting the SP Set with different fine-tuning datasets while maintaining the other settings in Section III-C. Regarding the fine-tuning method, although the S2L approach can be integrated with various fine-tuning methods, for simplicity, we opt to use the DreamBooth+LoRA method, which demonstrated superior performance, as indicated in Table I. We outline these fine-tuning datasets as follows: ❶ **Private Dataset**: In an ideal scenario, the most suitable fine-tuning dataset would be the private data specific to the target domain. Regrettably, such private data is not accessible to us. Nevertheless, we can establish an upper limit on the theoretical performance of domain-specific fine-tuning attacks by assuming access to these private data as prior knowledge. ❷ **Out-of-Distribution (OoD) Dataset**: The OoD dataset represents a typical dataset employed for fine-tuning and is readily available. ❸ **In-domain Non-Member (INM) Dataset**: This dataset corresponds to a genuine dataset that exhibits a similar distribution to the target domain $\mathcal{D}_z$, but is not part of the pre-training set. We created the INM dataset by scraping images from the web using the target prompt and then removing duplicate images found in the private domains. ❹ **Private Dataset**: To show the worst-case of fine-tuning, we assume the private data are available. Note that the assumption is unrealistic but is only made to explore the gap between S2L and the worst case.

Results. The results are presented in Table IV. Comparing the SP Set with other fine-tuning datasets, we observe that the SP Set can effectively serve as valuable prior knowledge for the S2L attacker. ❶ As the Out-of-Domain (OoD) dataset does not align well with the fine-tuning attack strategy, it leads to model optimization away from the local optima of the target domain. ❷ The In-domain Non-Member (INM) dataset presents a nuanced privacy risk profile, exhibiting lower MIA results but higher data extraction results. This arises because INM data may confound the model with membership signals, yet it can also optimize the model towards domain-specific local optima. However, the precision of data extraction remains below 50%,

primarily due to the limited MIA capabilities of the fine-tuned model in distinguishing whether a generated example belongs to the pre-training set. ❸ Notably, when comparing the SP Set and Private settings, we observe that the privacy risks of the DM fine-tuned on SP Set can approach the upper bound. For example, the improvement in the MIA AUC of DreamBooth and DreamBooth+LoRA as in Table I is 76.67% and 90% of the upper bound improvement by using a private set to perform the fine-tuning of S2L as in Table IV, respectively. Furthermore, the practical availability of $\mathcal{P}$ increases the threat to privacy posed by the S2L approach. In Fig. 3, we demonstrate some examples from the SP Set $\mathcal{P}$, the nearest neighbors of the SP Set from the pre-training set, the private pre-training set, and samples extracted by S2L (with the SP Set) as in Table IV, respectively. We notice that the generated examples in $\mathcal{P}$ tend to have significant artifacts compared to real images in the private pre-training set, and the nearest neighbor in the pre-training set is unlikely to be recognized as the extraction of the corresponding SP Set sample as in Definition III.1. Therefore, SP Set does not directly leak private information based on the criterion of MIA and data extraction attacks. However, the fine-tuning of S2L in $\mathcal{P}$ still significantly amplifies privacy risks, indicating that $\mathcal{P}$ may carry useful private patterns that summarize the private information in the pretraining set. Therefore, S2L can achieve privacy risk amplification without copying the exact private examples from the pre-training set to the SP Set before fine-tuning. In summary, our findings underscore the effectiveness of SP Set as a source of regularly available prior knowledge for the S2L attacker, with implications for privacy risks associated with different fine-tuning datasets.

C. S2L with A Few Private External-Domain Examples

So far we consider very restricted prior knowledge, but it is also valuable to ask whether the leakage will be further amplified with extra prior knowledge, *e.g.*, some previously leaked private examples from the external domain. The main motivation is to explore rare but potentially more dangerous situations. Though it is not common for an attacker to get private examples, we argue that such example leakage may happen when large-scale DMs use web-scrape data to augment training datasets.

For example, MidJourney's pre-training dataset consists of both web-scraped data (public) and human-curated data (private) [17]. Including MidJourney, today's commercial DM models will typically include large-scale web-scraped data in the pre-training set for utility purposes. Therefore, an adversary may leverage the public domain information to find the potential private examples by membership inference attack even with low possibilities, *e.g.* the adversary randomly scrapes a large amount of images from the Internet using the target prompt and then uses MIA to infer enough number of member images with high confidence.

Threat Model. Formally, in the threat model, we assume the domains of the pre-training set are partially private, i.e. $\mathcal{D}$ is composed of $M < N$ private domains $\{\mathcal{D}_{p_1}, \mathcal{D}_{p_2}, ...\mathcal{D}_{p_M}\}$, and the adversary aims to recover data from the private domains. The adversary cannot access the entire pre-training set $\mathcal{D}$ but attains a public subset of $\mathcal{D}$. We consider two specific settings for the public domain dataset $\mathcal{A}_b$: ❶**Partial leakage:** The adversary can obtain a dataset $\mathcal{A}_b$ that contains a subset that belongs to $\mathcal{D}$, for instance, the adversary randomly scrapes a dataset from the internet which contains overlapped examples with $\mathcal{D}$. Then the adversary uses MIA to infer example memberships and pick out predicted member and non-member examples with high confidence using the positive and negative threshold δ_m, δ_n. ❷**Worse case:** $\mathcal{A}_b$ is readily available to the adversary, *e.g.* the adversary knows that an existing public dataset is contained in $\mathcal{D}$.

To be general, we do not assume any similarity between public and private domains.

Procedures. When a private subset is retrieved from the training set, an attacker can inject a membership concept into the model and transfer the concept to extract private data from other private domains. To distinguish from the standard S2L that happens in one domain, we name such attack as **S2L by Domain-Transfer (S2L-DomainTrans)** which is illustrated in Algorithm 2. Our core idea is to learn a new token M representing the "membership" concept by Textual Inversion on the retrieved private subset $\mathcal{A}_b$. To attack the target domain associated with a prompt p, we append "of M" after p and perform a data extraction attack.

Setup. We conduct the domain-transfer experiments on the SD-v1-1 model by keeping the target private domains and settings the same as in our main context and using ImageNet domains as the public domains. We follow the basic experiment setting in Section III-C and other hyperparameters in Algorithm 2 are as follows: The candidate dataset $\mathcal{A}_b$ in public domains consists of 5,000 randomly sampled member images from the pre-training set and 50,000 web-scraped and de-duplicated non-member examples, and the balanced MIA set size N_m is set to 2,000. For private domains consisting of 40 celebrities, we average the attack results from each domain. The thresholds $\delta_m, \delta_n, and \delta_d$ are 0.3, 0.7, and 0.3, respectively. The values fN_d, N_p, N_m are 5000, 1000, and 2000, respectively. We consider several contrastive configurations as follows: ❶ **Plain-text Surfix**: As a baseline, we directly append "in pre-training set" as prompt suffix. The baseline could unveil if SD-v1-1 already knows the membership concept. ❷ **S2L**: Our standard Shake-To-Leak (S2L) implementation with Textual Inversion on SP Set. ❸ **S2L-DomainTrans with MIA set**: Domain-transfer attack which uses MIA inferenced set to learn the M token embedding of the membership concept. ❹ **S2L-DomainTrans with ground-truth set**: As a worst-case evaluation, we assume the ground-truth membership set is readily available.

Results. The results are shown in Table V. We observe that SD-v1-1 struggles to comprehend the concept of a pre-training set inherently and tends to associate this concept with private images during data extraction, as evidenced by the failure to extract any private examples in the Plain-text Suffix setting.

In contrast, S2L-DomainTrans settings with MIA and Ground Truth (GT) sets can extract 3.19 to 3.65 times the number of examples extracted by S2L. Therefore, for contemporary large-

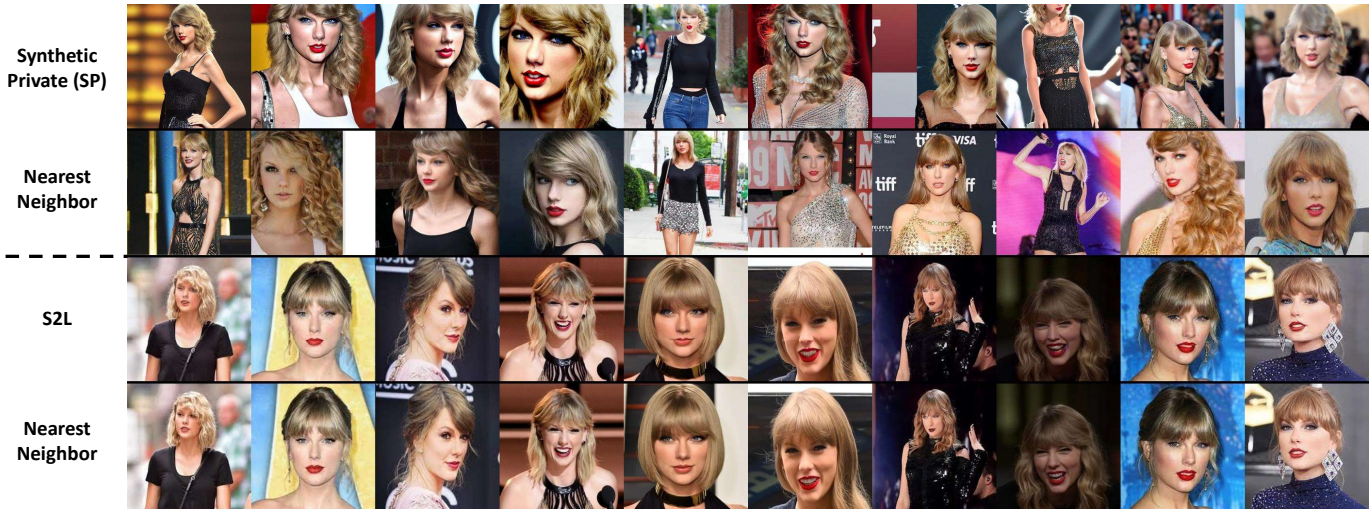


Fig. 3: Sample images of Taylor Swift from different sources. *Synthetic Private (SP)* set includes samples that are generated from the pre-trained model and used to fine-tune the diffusion model. *S2L* set includes samples that are generated after fine-tuning on the SP set. For each method, we include *nearest neighbors* which are the ground-truth private samples closest to the generated one (in the same column). We can observe that the SP set does not directly leak private data but fine-tuning on the set can cause serious privacy leakage.

Algorithm 2 Domain-Transfer Shake-To-Leak Attack (S2L-DomainTrans) for Data Extraction

Input: Pre-trained diffusion model G with the embedding layer G_e, text encoder G_t and denoising network G_n; prompts $\{p_b\}$, candidate dataset $\mathcal{A}_b$ for public domains $\{\mathcal{D}_b\}$ and prompt p_z for private domain $\mathcal{D}_z$; MIA loss thresholds $\delta_m, \delta_n (0 < \delta_m << \delta_n < 1)$ for (non-)member prediction; MIA loss threshold δ_d and generation times N_d for data extraction; desired sizes N_p, N_m of synthetic private set $\mathcal{P}$ and MIA set $\mathcal{M}$.
Output: extraction set $\mathcal{E}$.
$\mathcal{P}, \mathcal{M}, \mathcal{E} \leftarrow \emptyset$
/*** Step 1: Privacy risk amplification with SP Set fine-tuning ***/
for $i = 1$ **to** N_p **do**
    $P \leftarrow P \cup \{Gen(p_z, r_i)\}$ with initial noise r_i
Fine-tune G_e, G_t, G_n with $\mathcal{P}$
/*** Step 2: Generate balanced MIA set $\mathcal{M}$ ***/ ▷ Skip if $\mathcal{M}$ is readily available
$i, j \leftarrow 0$
for x **in** $\mathcal{A}_b$ **do**
    if $\text{MIA}(G, p_b, x) < \delta_m$ **and** $i < N_m/2$ **then** ▷ Filter member images with high confidence
        $\mathcal{M} \leftarrow \mathcal{M} \cup \{(x, p_b + \text{"of } M\text{"})\}$; $i+ = 1$
    else if $\text{MIA}(G, p_b, x) > \delta_n$ **and** $j < N_m/2$ **then** ▷ Filter non-member images with high confidence
        $\mathcal{M} \leftarrow \mathcal{M} \cup \{(x, p_b + \text{"of not } M\text{"})\}$; $j+ = 1$
/*** Step 3: Learn membership concept "M" with Textual Inversion on public domains ***/
Initialize token embedding(s) $G_e(M)$
for i **in** fine-tuning epochs **do**
    for $(x, p) \in \mathcal{M}$ **do**
        Fine-tune $G_e(M)$ with x, p and G fixed except for $G_e(M)$
/*** Step 4: Data extraction on private domain(s) ***/
for $i = 1$ **to** N_d **do**
    $x_i \leftarrow G(p_z + \text{"of } M\text{"}, r_i)$ with initial noise r_i
    if $\text{MIA}(G, p_z, x_i) < \delta_d$ **then**
        $\mathcal{E} \leftarrow \mathcal{E} \cup \{x_i\}$
return extraction set $\mathcal{E}$

TABLE V: Results of domain-transfer attacks for data extraction show the effectiveness of S2L with a few private external-domain examples. **Plain-text Surfix** means directly appending a suffix to the target prompt before data extraction. **S2L** denotes domain-specific fine-tuning attack. **S2L-DomainTrans** refers to domain-transfer fine-tuning attack. **Concept learning** refers to learning the "Membership" Concept with Textual Inversion. **MIA Set** refers to the membership dataset produced by MIA. **Ground-truth Set** refer to the ground truth membership dataset. All fine-tuning sets equally contain 1000 member and 1000 non-member examples. We omit MIA attack results as we observe no improvements w.r.t. Table I.

Method	Textual Inversion Fine-tune Set	Prompt Setting	Data Extraction	
			Num	Prec(%)
Pre-trained Baseline	-	-	0	-
Plain-text Suffix	-	Suffix: "in pre-training set"	0	-
S2L	SP Set	Prompt fine-tuning	14.6	87.5
S2L-DomainTrans	MIA Set	Concept learning	46.6	86.2
S2L-DomainTrans	Ground-truth Set	Concept learning	**53.2**	**88.6**

scale DMs, acquiring a grasp of the membership concept by harnessing information from public domains proves highly effective for data extraction attacks using the S2L approach. Under the S2L-DomainTrans with MIA set setting, we extract an average of approximately 46.6 examples, equivalent to 87.5% of the examples extracted by the S2L-DomainTrans with GT set setting. This discrepancy arises from the MIA inference dataset used for Textual Inversion fine-tuning, which contains false positive and false negative examples concerning ground-truth membership. In our experiments, the MIA method employed (SecMI) achieves a 0.712 Area Under the Curve (AUC) performance on SD-v1-1, resulting in approximately 5.2% false positives and 4.6% false negatives in the MIA set, particularly under high prediction confidence. This discrepancy leads to a 12.5% reduction in extracted examples and a 2.4% decrease in extraction precision. In conclusion, our study highlights that extra prior knowledge of previously leaked private examples will cast significantly increased privacy risks associated with the S2L approach.

D. Summary

By ranging the amount of prior knowledge that S2L can access, we discover strong positive correlations between the S2L effect and the amount of obtainable prior knowledge. ❶ Under zero prior knowledge, simple Gaussian attacks work well on small DMs but lose effect on a larger scale, which demonstrates the vulnerability of smaller models. ❷ When an attacker knows the approximate distribution of the target domain, the leakage amplification could be greatly enlarged, and the synthetic data functions closely as the ground-truth private set for fine-tuning.

❸ Under extended prior knowledge assumption by assuming a few web-scrapable examples for the attacker that are irrelevant to the private domain, we demonstrate that S2L can achieve up to $3 \sim 4$ times data extraction privacy risks using a domain transfer fine-tuning attack.

V. Conclusion

In this paper, we reveal an unexpected finding that fine-tuning a manipulated data set can amplify the privacy risks of existing large-scale diffusion models trained in text-to-image synthesis. Leveraging the text-to-image synthesis mechanism of DMs, an attacker can prompt a DM to generate images for a target dataset and use the dataset to fine-tune a DM that will leak more information from the pre-training set. Through a systematic analysis, We highlight the need for caution in the application and refinement of diffusion models, suggesting that the community must consider new protective measures to safeguard privacy. Our findings contribute a novel perspective to the ongoing conversation about the trade-offs between model performance and privacy, offering valuable insights for both researchers and practitioners in the field. We also leave to future work exploring the principal-guided Differential Privacy (DP) guarantee [9] on large DMs as currently DP is hard to apply to large generative models due to scaling issues on DP-SGD private training steps [1].

Extension to Copyright Risks. As evidenced in [6], web-scraped image generation datasets, like the LAION dataset, consist of a mix of explicit non-permissive copyrighted examples, general copyright-protected examples, and CC BY-SA licensed examples. This raises concerns about copyright risks. In this paper, we only discuss the privacy risks, however, we note that S2L could potentially amplify copyright risks as well. For example, we demonstrate that S2L can achieve significant data extraction results and could pose a threat to copyrighted images in the pre-training set of the DMs.

Social Impact. Our exploration of the S2L phenomenon is not an endorsement or encouragement of exploiting these vulnerabilities. In contrast, by revealing these potential threats, we aim to foster a proactive approach to address them. While the immediate implications of our findings may seem alarming, we intend to bolster the defense mechanisms in place. Here, we provide several possible defense methods to inspire future research: ❶ Pre-train the DMs using a DP mechanism. ❶ For a partially private pretraining dataset, first pre-train the DMs in public domains and then fine-tune the DMs in private domains privately [34]. ❷ On the model provider side, develop secure fine-tuning APIs to prevent S2L-like misuse.

Acknowledgement

The work of Z. Wang is in part supported by Good Systems, a UT Austin Grand Challenge to develop responsible AI

technologies; as well as the National Science Foundation under Grant IIS-2212176. This work is also partially supported by the National Science Foundation under grant No. 1910100, No. 2046726, No. 2229876, DARPA GARD, the National Aeronautics and Space Administration (NASA) under grant no. 80NSSC20M0229.

REFERENCES

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 308–318.

[2] J. Abascal, S. Wu, A. Oprea, and J. Ullman, "Tmi! finetuned models leak private information from their pretraining data," *arXiv preprint arXiv:2306.01181*, 2023.

[3] Andrew, "What are hypernetworks and the ones you should know," 2023. [Online]. Available: https://stable-diffusion-art.com/hypernetwork/

[4] B. Bortolato, M. Ivanovska, P. Rot, J. Križaj, P. Terhörst, N. Damer, P. Peer, and V. Štruc, "Learning privacy-enhancing face representations through feature disentanglement," in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*. IEEE, 2020, pp. 495–502.

[5] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1897–1914.

[6] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," *arXiv preprint arXiv:2301.13188*, 2023.

[7] H. Duan, A. Dziedzic, N. Papernot, and F. Boenisch, "Flocks of stochastic parrots: Differentially private prompt learning for large language models," *arXiv preprint arXiv:2305.15594*, 2023.

[8] J. Duan, F. Kong, S. Wang, X. Shi, and K. Xu, "Are diffusion models vulnerable to membership inference attacks?" in *40th International Conference on Machine Learning*, 2023.

[9] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*. Springer, 2006, pp. 265–284.

[10] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, "An image is worth one word: Personalizing text-to-image generation using textual inversion," *arXiv preprint arXiv:2208.01618*, 2022.

[11] S. Ghalebikesabi, L. Berrada, S. Gowal, I. Ktena, R. Stanforth, J. Hayes, S. De, S. L. Smith, O. Wiles, and B. Balle, "Differentially private diffusion models generate useful synthetic images," *arXiv preprint arXiv:2302.13861*, 2023.

[12] A. Gupta, A. Jaiswal, Y. Wu, V. Yadav, and P. Natarajan, "Adversarial mask generation for preserving visual privacy," in *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*. IEEE, 2021, pp. 1–5.

[13] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[14] J. Hong, J. T. Wang, C. Zhang, Z. Li, B. Li, and Z. Wang, "Dp-opt: Make large language model your privacy-preserving prompt engineer," *arXiv preprint arXiv:2312.03724*, 2023.

[15] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[16] H. Hu and J. Pang, "Membership inference of diffusion models," *arXiv preprint arXiv:2301.09956*, 2023.

[17] A. Hughes, "Midjourney: The gothic ai image generator challenging the art industry," 2023. [Online]. Available: https://www.sciencefocus.com/future-technology/midjourney

[18] B. Isik and T. Weissman, "Learning under storage and privacy constraints," in *2022 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2022, pp. 1844–1849.

[19] A. Kazerouni, E. K. Aghdam, M. Heidari, R. Azad, M. Fayyaz, I. Hacihaliloglu, and D. Merhof, "Diffusion models for medical image analysis: A comprehensive survey," *arXiv preprint arXiv:2211.07804*, 2022.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] V. Mirjalili, S. Raschka, A. Namboodiri, and A. Ross, "Semi-adversarial networks: Convolutional autoencoders for imparting privacy to face images," in *2018 International Conference on Biometrics (ICB)*. IEEE, 2018, pp. 82–89.

[22] A. Panda, Z. Zhang, Y. Yang, and P. Mittal, "Teach gpt to phish," in *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023.

[23] D. H. Park, S. Azadi, X. Liu, T. Darrell, and A. Rohrbach, "Benchmark for compositional text-to-image synthesis," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[24] W. H. Pinaya, P.-D. Tudosiu, J. Dafflon, P. F. Da Costa, V. Fernandez, P. Nachev, S. Ourselin, and M. J. Cardoso, "Brain imaging generation with latent diffusion models," in *MICCAI Workshop on Deep Generative Models*. Springer, 2022, pp. 117–126.

[25] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[26] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[27] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[28] R. Rombach, A. Blattmann, and B. Ommer, "Text-guided synthesis of artistic images with retrieval-augmented diffusion models," *arXiv preprint arXiv:2207.13038*, 2022.

[29] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 500–22 510.

[30] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 479–36 494, 2022.

[31] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 278–25 294, 2022.

[32] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. 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*, 2023, pp. 6048–6058.

[33] Y. Wu, N. Yu, Z. Li, M. Backes, and Y. Zhang, "Membership inference attacks against text-to-image generation models," *arXiv preprint arXiv:2210.00968*, 2022.

[34] D. Yu, S. Naik, A. Backurs, S. Gopi, H. A. Inan, G. Kamath, J. Kulkarni, Y. T. Lee, A. Manoel, L. Wutschitz *et al.*, "Differentially private fine-tuning of language models," *arXiv preprint arXiv:2110.06500*, 2021.

[35] S. Zanella-Béguelin, L. Wutschitz, S. Tople, A. Salem, V. Rühle, A. Paverd, M. Naseri, B. Köpf, and D. Jones, "Bayesian estimation of differential privacy," in *International Conference on Machine Learning*. PMLR, 2023, pp. 40 624–40 636.

[36] Y. Zhang, N. Huang, F. Tang, H. Huang, C. Ma, W. Dong, and C. Xu, "Inversion-based style transfer with diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 146–10 156.

APPENDIX

A. How Does S2L Perform on More General Domains?

In this section, we provide some additional results by defining 80 domains originating from the 80 longest ImageNet class labels (we choose the longest labels to avoid an over-common substring in the LAION prompts and thus an exploding example number) and repeating the experiments in Table I. Please note that we are unable to perform experiments on the whole LAION dataset since it would be difficult to perform data extraction evaluation which requires pair-wise image comparisons. The results are placed in Table VI. By comparing Table VI and Table I, we find that the privacy leakage of the baselines and S2L fine-tuned models tend to be stable. For example, there are at most 0.012 MIA AUC differences among all the corresponding MIA experiment pairs. The improvement in the example number of data extraction is due to the proportional growth of domain size from celebrity domains to the general ImageNet domains. We conjecture that this is because every example is treated equally during training and our evaluation criteria are very general and do not have a preference in any specific domains.

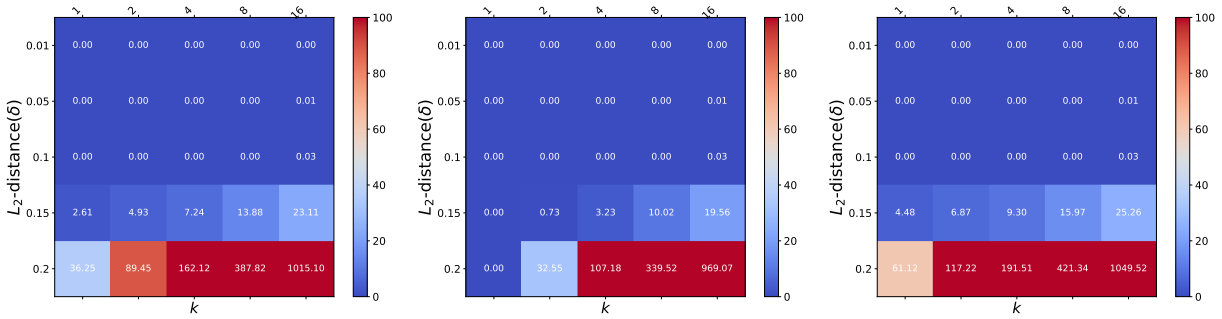
B. Data Extraction Results under Variable Memorization Criteria

In this section, we provide the data extraction results under variable distance threshold δ and similar sample number k of (k, l, δ)-Eidetic memorization, to better understand how the private samples are memorized. Specifically, ❶ by varying the similar sample number k we can see how the data duplication in the pre-training set can affect the data extraction results; ❷ by varying the L_2-distance threshold δ, we know the data extraction performance at different Eidetic level. The L_2-distance threshold δ is in the range of $[0.01, 0.20]$ as we find $\delta > 0.20$ to make the extraction algorithms recognize most of the generated images visually irrelevant to their closest images in the pre-training set as successful extractions. The similar sample number k is in the range of $[1, 16]$. We keep other experiment settings the same with Table I, and the results are shown in Fig. 4.

Overall, we find that the extracted example number grows proportionally with the L_2-distance δ. Interestingly, we find that after S2L fine-tuning, there is a non-trivial number of extracted samples with few duplications in the pre-training dataset. For example, when $k = 1$ and $\delta = 0.15$, S2L increases the extracted example number from 0.00 to a range of 0.47 to 3.23; when $k = 2$ and $\delta = 0.10$, S2L increases the extracted example number from 0.00 to a range of 1.70 to 5.59. This means that in the target domains, S2L can “shake out" examples that are seen very few times during training.

TABLE VI: Alternative experiment results by changing the celebrity domains in Table I to 80 general domains defined by 80 longest ImageNet class labels.

Fine-tuning Method	Fine-tuning Setting		MIA		Data Extraction		Clip-RP
	Dataset	Params	AUC	TPR@1%FPR	Num	Prec(%)	
Pre-trained	-	-	0.707	0.164	0	-	52.3
End-to-End	OoD	1064M	0.679	0.154	0	-	50.9
End-to-End	SP Set	1064M	0.721	0.164	0.6	-	50.7
DreamBooth	SP Set	980M	**0.753**	**0.166**	18.1	85.3	50.9
Textual Inversion	SP Set	9.2K	0.735	0.169	**19.2**	86.3	52.3
Hypernetwork	SP Set	45M	0.732	0.168	5.1	79.5	51.5
LoRA	SP Set	20M	0.738	0.165	16.0	84.9	50.4
DreamBooth+Hypernetwork	SP Set	45M	0.735	0.164	6.1	69.5	50.6
DreamBooth+LoRA	SP Set	19M	**0.760**	**0.172**	**20.5**	**87.2**	51.2



(a) Pre-trained (b) E2E-OOD (c) E2E-SP Set

(d) DreamBooth (e) Textual Inversion (f) Hypernetwork

(g) LoRA (h) DreamBooth+Hypernetwork (i) DreamBooth+LoRA

Fig. 4: The DE results of S2L under variable L_2-distance threshold δ and similar sample number k of the Eidetic memorization. Other experiment settings are kept the same with Table I.