PCEvolve: Private Contrastive Evolution for Synthetic Dataset Generation via Few-Shot Private Data and Generative APIs

Jianqing Zhang [1 2]   Yang Liu [3 4]   Jie Fu [5]   Yang Hua [6]   Tianyuan Zou [2]   Jian Cao [1 7]   Qiang Yang [3]

Abstract

The rise of generative APIs has fueled interest in privacy-preserving synthetic data generation. While the Private Evolution (PE) algorithm generates Differential Privacy (DP) synthetic images using diffusion model APIs, it struggles with few-shot private data due to the limitations of its DP-protected similarity voting approach. In practice, the few-shot private data challenge is particularly prevalent in specialized domains like healthcare and industry. To address this challenge, we propose a novel API-assisted algorithm, *Private Contrastive Evolution (PCEvolve)*, which iteratively mines inherent inter-class contrastive relationships in few-shot private data beyond individual data points and seamlessly integrates them into an adapted Exponential Mechanism (EM) to optimize DP's utility in an evolution loop. We conduct extensive experiments on four specialized datasets, demonstrating that PCEvolve outperforms PE and other API-assisted baselines. These results highlight the potential of leveraging API access with private data for quality evaluation, enabling the generation of high-quality DP synthetic images and paving the way for more accessible and effective privacy-preserving generative API applications. Our code is available at https://github.com/TsingZ0/PCEvolve.

1. Introduction

Recent advances in machine learning for image tasks have significantly improved efficiency in various fields, such as
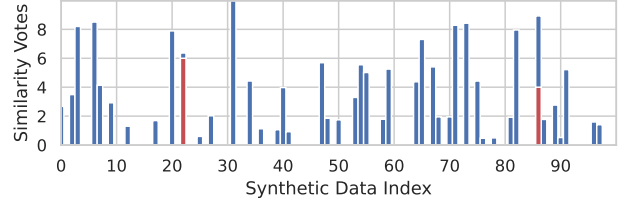
Figure 1: A scenario with 10-shot private images and 100-shot synthetic images in PE. Private data contribute only 10 votes (red), while the noise (blue) exceeds the red votes.

COVID-19 pneumonia recognition (Harmon et al., 2020) and industry anomaly detection (Roth et al., 2022). However, training an effective model requires a sufficient number of images and computing resources, which is often impractical for resource-constrained clients (*e.g.*, clinics) in specialized domains, where each data producer typically possesses only a few data (Chen et al., 2023; 2020b). To tackle data scarcity, data owners are increasingly turning to external sources, but most specialized data are private (Boland et al., 2017), raising significant privacy concerns (Zhang et al., 2022). Healthcare data breaches incur substantial costs, averaging $9.2 million per year (Hu et al., 2024).

Differential Privacy (DP) (Dwork et al., 2006) is a widely used privacy protection method. However, traditional training-based approaches like DP-GAN (Zhang et al., 2024b) and DP-Diffusion (Ghalebikesabi et al., 2023) for image generation are impractical for resource-constrained clients due to their high computational cost and substantial data requirements (Hu et al., 2020). Nowadays, using powerful generative APIs for image generation has emerged as a compelling and training-free alternative (Lin et al., 2024). However, since black-box API providers are untrusted, protecting user privacy becomes crucial (Lin et al., 2024). The latest Private Evolution (PE) algorithm (Lin et al., 2024) leverages diffusion model APIs (Rombach et al., 2022) to generate DP synthetic images without any training in an evolution loop. Specifically, PE generates random synthetic images from APIs and iteratively improves them by selecting (with DP) the most similar synthetic images to the private dataset, and then querying APIs to generate more similar images. DP synthetic data can be reused infinitely

for potential models on various downstream tasks without additional privacy costs, thanks to the post-processing property of DP (Dwork et al., 2006; Hu et al., 2024; Fu et al., 2024b;a).

However, PE was originally designed for data parties with large-scale private image datasets, and its performance degrades with limited private data due to its reliance on DP-protected (based on Gaussian Mechanism (GM) (Dwork et al., 2014)) similarity voting, where one private data gives one vote to the most similar synthetic data with DP. For example, the original PE uses 302,436 images from the Camelyon17 dataset (Koh et al., 2021) as private data[1]. However, as shown in Fig. 1, when dealing with few-shot private data, the added noise overwhelms the actual votes, leading to nearly random similarity voting and selection. Subsequently, querying APIs with randomly selected images as guidance yields nearly random outputs, ultimately degrading the quality of synthetic data (Pan et al., 2023).

To address the few-shot private data challenge in API-assisted DP generation inside the evolution loop, we propose an algorithm called *Private Contrastive Evolution (`PCEvolve`)* by incorporating two key components for selecting high-quality prototypical synthetic data as feedback to APIs:

(1) We devise a *contrastive filter* to iteratively exploit interclass contrastive relationships between different classes within the private data, beyond individual data points, thereby enhancing the class-discriminability of synthetic data *w.r.t.* private data.

(2) We adapt the Exponential Mechanism (EM) (Dwork et al., 2014) to preserve private inter-class contrastive relationships, addressing the excessive noise from the high sensitivity of GM. Specifically, we devise a *similarity calibrator* to enhance EM's utility by prioritizing high-quality synthetic data that closely resembles private data.

To evaluate the effectiveness of `PCEvolve`, we conduct extensive experiments in two aspects:

(1) We show that `PCEvolve` surpasses six API-assisted synthetic data generation baselines across four practical datasets from specialized domains like healthcare and industry. Additionally, we demonstrate the applicability of `PCEvolve` in various scenarios, such as varying amounts of private/synthetic data, different APIs, etc. Our DP synthetic dataset works effectively with six diverse downstream models, thanks to its enhanced class-discriminability.

(2) We analyze the properties of `PCEvolve`, including the visible quality of synthetic images, the effectiveness of each component, and the influence of the single hyperparameter.

---

[1]PE also considers a Cat dataset with 200 images, but only shows synthetic images with a large domain gap to private data.

2. Related Work

Synthetic image data with APIs. In recent years, numerous studies have focused on synthetic image generation using generative models, such as Stable Diffusion (SD) (Rombach et al., 2022; Moor et al., 2023; Yang et al., 2023; Zhang et al., 2023b; Brooks et al., 2023). However, in specialized domains like healthcare, generative models often require fine-tuning or even redesigning on local platforms with GPU resources (Ji & Chung, 2024; Guan & Liu, 2021; Kather et al., 2022), demanding significant human effort and financial investment—making them impractical for resource-constrained clients (Abou Baker et al., 2024; Wornow et al., 2023). An alternative is leveraging cloud-based APIs from service providers (He et al., 2023; Seo et al., 2024; Samuel et al., 2024), customizing them through API-based fine-tuning (OpenAI, 2024) or prompt engineering (Hao et al., 2024). While this approach reduces resource consumption, it raises serious privacy concerns (Qi et al., 2024), as local private data must be uploaded to untrusted API providers for fine-tuning or prompting (Chen et al., 2024). Although some API-assisted text data generation methods (Ye et al., 2022; Gao et al., 2023) avoid using private data, they rely on specific downstream models, limiting their utility.

DP synthetic image data with APIs. DP (Dwork et al., 2006) is a widely used technique for protecting privacy in image data (Ziller et al., 2021) and has also been applied to synthetic image generation (Li et al., 2024; Hu et al., 2024; Chen et al., 2022; De Cristofaro, 2024). However, DP-based generative methods like DP-GAN (Zhang et al., 2024b) and DP-Diffusion (Ghalebikesabi et al., 2023) are impractical for resource-constrained clients (Hu et al., 2020) and infeasible when models are accessed via APIs (Xie et al., 2024; Lin et al., 2024). Directly adding DP noise to images makes DP images nearly unusable as input guidance for APIs, as ensuring privacy demands excessive noise (Croft et al., 2021). Recently, Private Evolution (PE) (Lin et al., 2024) optimizes synthetic image data to resemble private data with DP, performing well with large private datasets but *struggling in few-shot scenarios due to its GM-based similarity voting*. Two follow-up methods in the *text* domain (Xie et al., 2024; Hou et al., 2024) retain PE's core GM-based similarity voting while equipping it with more synthetic data per generation (Xie et al., 2024) or more private data from large-scale federated learning (Hou et al., 2024).

3. Preliminaries

Definition 3.1 (Differential Privacy (Dwork et al., 2006))**.** Let $\mathcal{D}$ be any private database with data from a space $\mathbb{X}$, denoted as $\mathcal{D} \in \mathbb{X}$, with a symmetric neighbor $\mathcal{D}'$, where $\mathcal{D}$ and $\mathcal{D}'$ differ in only one element. Let $\epsilon > 0$ and $\delta \in [0, 1]$ be two privacy parameters. A randomized algorithm $\mathcal{M}$:

$\mathbb{X} \to \mathbb{Y}$ with a value range $\mathbb{Y}$ is (ϵ, δ)-DP if the following inequality holds for any $\mathcal{E} \subseteq \mathbb{Y}$:

$$Pr[\mathcal{M}(\mathcal{D}) \in \mathcal{E}] \le e^{\epsilon} Pr[\mathcal{M}(\mathcal{D}') \in \mathcal{E}] + \delta.$$

If $\delta = 0$, we say that $\mathcal{M}$ is ϵ-DP. Any post-processing of a DP mechanism's output incurs no additional privacy loss.

Definition 3.2 (Gaussian Mechanism (Dwork et al., 2014)). Let $f : \mathbb{X} \to \mathbb{R}^D$ be a D-dimensional function with ℓ_2 sensitivity to be $\Delta_f := \max_{\mathcal{D}, \mathcal{D}'} ||f(\mathcal{D}) - f(\mathcal{D}')||_2$. The Gaussian Mechanism (GM) $\mathcal{M}_\sigma$ with parameter σ adds noise scaled to $\mathcal{N}(0, \sigma^2)$ to each of the D components of the output, *i.e.*, $\tilde{f}(\mathcal{D}) := f(\mathcal{D}) + \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I_D})$. For $\epsilon, \delta \in (0, 1)$, $\mathcal{M}_\sigma$ with $\sigma = \Delta_f \sqrt{2 \log(1.25/\delta)}/\epsilon$ is (ϵ, δ)-DP.

Definition 3.3 (Exponential Mechanism (Dong et al., 2020; McSherry & Talwar, 2007)). Given a parameter ϵ, an arbitrary range $\mathcal{R}$, and a utility function $u : \mathbb{X} \times \mathcal{R} \to \mathbb{R}$ with sensitivity $\Delta_u := \max_{r \in \mathcal{R}} \max_{\mathcal{D}, \mathcal{D}'} |u(\mathcal{D}, r) - u(\mathcal{D}', r)|$, a randomized algorithm $\mathcal{M}_u$ is called the Exponential Mechanism (EM), if the outcome r is sampled with probability proportional to $\exp\left(\frac{\epsilon \cdot u(\mathcal{D}, r)}{2\Delta_u}\right)$ and $\mathcal{M}_u$ is ϵ-DP:

$$Pr[\mathcal{M}_u(\mathcal{D}) = r] = \frac{\exp\left(\frac{\epsilon \cdot u(\mathcal{D}, r)}{2\Delta_u}\right)}{\sum_{r' \in \mathcal{R}} \exp\left(\frac{\epsilon \cdot u(\mathcal{D}, r')}{2\Delta_u}\right)}.$$

Definition 3.4 (Sequential Composition (Dwork et al., 2006)). Given any mechanism $\mathcal{M}_1(\cdot)$ that satisfies ϵ_1-DP, and $\mathcal{M}_2(s, \cdot)$ that satisfies ϵ_2-DP for any s, then $\mathcal{M}(\mathcal{D}) = \mathcal{M}_2(\mathcal{M}_1(\mathcal{D}), \mathcal{D})$ satisfies $(\epsilon_1 + \epsilon_2)$-DP .

Remark 3.5. Given the same ϵ, $\mathcal{M}_u$ (ϵ-DP) provides stronger privacy protection than $\mathcal{M}_\sigma$ ((ϵ, δ)-DP) for $\delta > 0$.

4. Method

Motivation. The Gaussian DP ($\mathcal{M}_\sigma$)-based similarity voting approach in the existing method PE corresponds to the function f (in Definition 3.2) of $\mathcal{M}_\sigma$. However, due to the few-shot private dataset $\mathcal{D}_p$, the values in $f(\mathcal{D}_p)$ are extremely small, making the noise from $\mathcal{N}(0, \sigma^2)$ appear larger than $f(\mathcal{D}_p)$ and rendering the final votes, $\tilde{f}(\mathcal{D}_p)$, nearly random, as shown in Fig. 1. In contrast, EM $\mathcal{M}_u$ is tailored for selection, with privacy guarantee depending on the sensitivity of utility function u (Dwork, 2008). Therefore, we propose PCEvolve based on $\mathcal{M}_u$ and adapt EM to select synthetic (public) data instead of private data, reducing the influence of private data volume and making it suitable for few-shot scenarios.

Problem Definition. Our goal is to generate a DP synthetic dataset $\mathcal{D}_s$ that closely resembles $\mathcal{D}_p$ and effectively supports potential downstream C-class classification tasks within the same domain as $\mathcal{D}_p$. To achieve this, $\mathcal{D}_s$ must satisfy two key criteria: (1) *class-discriminability* and (2) high *similarity* to $\mathcal{D}_p$.

Following PE, we randomly initialize $\mathcal{D}_s^0$ using an untrusted text-to-image (t2i) API G_{t2i} with a *simple* text prompt $\mathcal{T}$—containing only the domain and class label names, *without any prompt engineering* (Meskó, 2023) (*e.g.*, $\mathcal{T}$ = "A leather texture image with cut defect"). We then iteratively refine $\mathcal{D}_s$ (omitting the iteration superscript t for simplicity) through the following steps. In each iteration, given the previously generated $\mathcal{D}_s$ and a pre-trained encoder[2] (feature extractor) E_f, we select high-quality prototypical ("proto") data points, denoted as $\mathcal{D}_{pro}$, from $\mathcal{D}_s$. This selection is guided by $\mathcal{D}_p$ using $\mathcal{M}_u$. Next, we refine the synthetic dataset by leveraging an untrusted image-to-image (i2i) API G_{i2i}, using $\mathcal{D}_{pro}$ as feedback and guidance. As this evolution loop progresses, we iteratively optimize $\mathcal{D}_s$.

A crucial aspect of this process is the selection of $\mathcal{D}_{pro}$. According to Definition 3.3 of $\mathcal{M}_u$, $\mathcal{D}_{pro}$ is obtained by sampling data from $\mathcal{D}_s$ with probabilities proportional to their corresponding u scores. Thus, the effectiveness of our approach hinges on the design of the utility function u.

Threat Model. We assume that the API provider is honest-but-curious, aiming to extract private information (*e.g.*, membership information) from the private data uploaded by the client device. While the provider can only access public images selected by the client's private data, it can leverage this information to attack the client's privacy (Duan et al., 2024). Specifically, by repeatedly observing the output images, the provider can infer whether a particular sample exists in the client's dataset. Empirical studies have confirmed privacy risks through viable attacks, such as membership inference attacks (Shokri et al., 2017; Carlini et al., 2022).

4.1. PCEvolve

Overview. As shown in Fig. 2, our PCEvolve iteratively enhances the synthetic data with a DP-protected selector as the core engine. To tackle the few-shot challenge with DP, we leverage inter-class contrastive relationships from private data and optimize the utility of $\mathcal{M}_u$ within the selector. Specifically, we (1) aggregate private class centers to reduce bias caused by the few-shot problem, (2) introduce a *contrastive filter* g to improve the class-discriminability of synthetic data, (3) design a *similarity calibrator* h to maximize the selection probability of the most similar synthetic data by $\mathcal{M}_u$, where $u := h \circ g$, and (4) utilize the scores from u (*i.e.*, outputs of h) to construct $\mathcal{D}_{pro}$ via $\mathcal{M}_u$.

(1) Aggregating Class Centers. The fundamental challenge in few-shot data scenarios is the lack of sufficient information to represent the entire real data distribution, leading to inadequate and biased learning of downstream models (Song et al., 2023; Xu et al., 2022), especially the

---

[2]All distance measures are computed in the feature space after applying E_f. For simplicity, we omit E_f in the following.
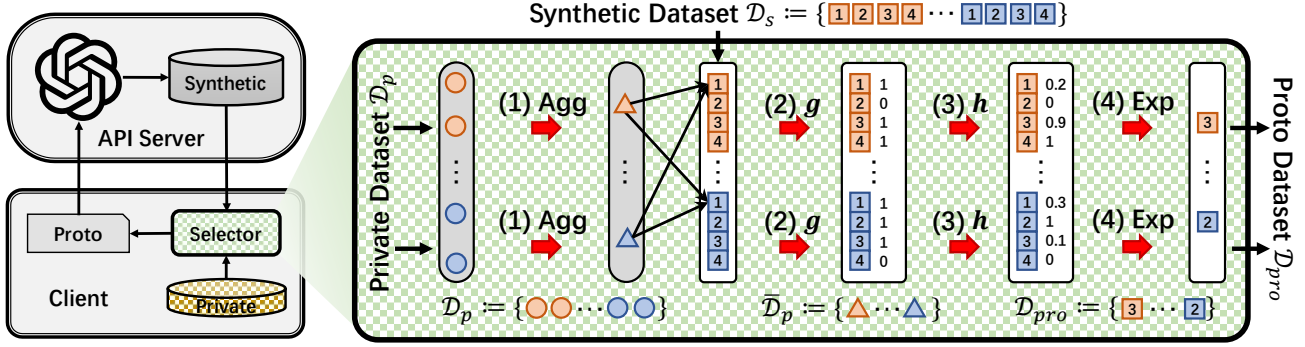
Figure 2: Illustration of our PCEvolve, whose core is the DP-protected selector. Different colors denote distinct data classes (two classes are explicitly shown, with others indicated by "..."). "Agg" and "Exp" denote the class center aggregation and the EM $\mathcal{M}_u$ ($u = h \circ g$), respectively.

few but useful data that are near the distribution boundary (Yang et al., 2021). To reduce bias from boundary data, we aggregate few-shot private data to obtain the private center set $\bar{\mathcal{D}}_p := \{\bar{d}_p^c\}_{c \in [C]}$, where $\bar{d}_p^c := \frac{1}{|\mathcal{D}_p^c|} \sum_{d_p^c \sim \mathcal{D}_p^c} d_p^c$, where we define $\mathcal{D}^c$ as the subset of any set $\mathcal{D}$ that contains all data with label c.

(2) Contrastive Filter g**.** Then, we leverage inter-class contrastive relationships to utilize private data fully. Inspired by metric learning (Kulis et al., 2013) and contrastive learning (Chen et al., 2020a), we design a contrastive filter g to select synthetic data that can be correctly classified into their corresponding classes using $\bar{\mathcal{D}}_p$ as class identifiers, *i.e.*,

$$g(d_s^c, \bar{\mathcal{D}}_p) := \begin{cases} 1, & \text{if } \ell_2(d_s^c, \bar{d}_p^c) < \min_{c'}\{\ell_2(d_s^c, \bar{d}_p^{c'})\} \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

$\forall d_s^c \in \mathcal{D}_s^c, c \in [C], c' \in [C], c' \neq c$. We use ℓ_2-norm (Luo et al., 2016) for distance measure. Using g for selection aligns our discriminability goal by assigning positive scores to discriminative synthetic data.

(3) Similarity Calibrator h**.** Given the initial large difference between the synthetic and private domains, solely emphasizing discriminability with g while neglecting similarity will barely reduce this gap in the next generation iteration. Thus, we calculate the similarity of discriminative synthetic data to $\bar{\mathcal{D}}_p$ by

$$h(d_s^c, \bar{\mathcal{D}}_p) := \begin{cases} e^{-\ell_2(d_s^c, \bar{\mathcal{D}}_p)}, & \text{if } g(d_s^c, \bar{\mathcal{D}}_p) = 1 \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

$\forall d_s^c \in \mathcal{D}_s^c, c \in [C]$. We apply exponent arithmetic and negation to convert ℓ_2 distances into similarities. Since the range of ℓ_2 is $[0, +\infty)$, the range of h is $[0, 1]$, making $u = h \circ g$ with a sensitivity of $\Delta_u = 1$ according to Definition 3.3. Since $\mathcal{M}_u$ samples data from $\mathcal{D}_s$ into $\mathcal{D}_{pro}$ based on u's values, candidates with higher u values are more likely to be included in $\mathcal{D}_{pro}$.

In practice, ℓ_2 values rarely span the full range of $[0, +\infty)$. Due to random initialization and the large gap between $\mathcal{D}_s$ and $\mathcal{D}_p$, initial ℓ_2 values are excessive, leading most u values to fall near 0. This under-utilizes the full range $[0, 1]$ of u and causes $\mathcal{M}_u$ to sample data almost randomly. To address this issue, we propose calibrating the original similarity scores to ensure u values span the full range of $[0, 1]$ in two steps: (1) normalizing the original ℓ_2 values to $[0, 1]$ and (2) scaling them to $[0, \tau]$, where τ is the only hyperparameter. This forces u values to fall within $\{0\} \cup [e^{-\tau}, 1]$. By selecting an appropriate τ value, we can ensure that $e^{-\tau} \approx 0$. Consequently, we assign the maximum u value (*i.e.*, 1) to the best (the most similar) candidate to maximize its selection probability when applying $\mathcal{M}_u$. Formally, we rewrite h to

$$h(d_s^c, \bar{\mathcal{D}}_p) := \begin{cases} e^{-\frac{\ell_2(d_s^c, \bar{\mathcal{D}}_p) - \ell_{\min}^c}{\ell_{\max}^c - \ell_{\min}^c} \cdot \tau}, & \text{if } g(d_s^c, \bar{\mathcal{D}}_p) = 1 \\ 0, & \text{otherwise} \end{cases},$$
$$\text{s.t. } \ell_{\max}^c := \max_{d_s^c \in \mathcal{D}_s^c, \, g(d_s^c, \mathcal{D}_p) = 1} \ell_2(d_s^c, \bar{\mathcal{D}}_p),$$
$$\ell_{\min}^c := \min_{d_s^c \in \mathcal{D}_s^c, \, g(d_s^c, \mathcal{D}_p) = 1} \ell_2(d_s^c, \bar{\mathcal{D}}_p),$$
$$\tag{3}$$

$\forall d_s^c \in \mathcal{D}_s^c, c \in [C]$. As per Definition 3.3, this rewriting does not impact Δ_u, since u values still belong to $[0, 1]$.

(4) Applying $\mathcal{M}_u$**.** Given u and Δ_u, we simply run $\mathcal{M}_u$ to sample synthetic data from $\mathcal{D}_s$ into $\mathcal{D}_{pro}$ based on Definition 3.3. Each execution of $\mathcal{M}_u$ consumes a portion of the total privacy cost ϵ_*. To balance the utility-privacy trade-off (Dwork et al., 2014), we select only one candidate per class to form $\mathcal{D}_{pro}$. The best candidate with the highest h value has the greatest probability of being selected. Then, we can generate a refined $\mathcal{D}_s$ using an improved $\mathcal{D}_{pro}$ as the input to G_{i2i}. We show the overall algorithm in Algorithm 1.

Algorithm 1 PCEvolve

Input: Private dataset $\mathcal{D}_p$, i2i API G_{i2i}, t2i API G_{t2i}, text prompt $\mathcal{T}$, total privacy cost ϵ_*, number of class C, number of iteration T, encoder E_f, and similarity calibrating factor τ.

Output: Synthetic dataset $\mathcal{D}_s$.

1: $\mathcal{D}_s^0 \leftarrow G_{t2i}(\mathcal{T})$ and $\epsilon = \frac{\epsilon_*}{T \cdot C}$.
2: **for** evolution iteration $t = 1, \ldots, T$ **do**
3:      **for** class $c = 1, \ldots, C$ **do**
4:          Get u scores for $\mathcal{D}_s^{t,c}$ via Eq. (1) and Eq. (3).
5:          Get $\mathcal{D}_{pro}^{t,c}$ by sampling data from $\mathcal{D}_s^{t,c}$, with the index set $\mathcal{R}$ of $\mathcal{D}_s^{t,c}$ and probabilities

$$Pr[\mathcal{M}_u(\mathcal{D}_p) = r \in \mathcal{R}] = \frac{\exp\left(\frac{\epsilon \cdot u(\mathcal{D}_p, r)}{2\Delta_u}\right)}{\sum_{r' \in \mathcal{R}} \exp\left(\frac{\epsilon \cdot u(\mathcal{D}_p, r')}{2\Delta_u}\right)}.$$

6:      $\mathcal{D}_s^t \leftarrow G_{i2i}(\mathcal{T}, \mathcal{D}_{pro}^t)$, where $\mathcal{D}_{pro}^t = \{\mathcal{D}_{pro}^{t,c}\}_{c=1}^C$.
7: **return** Synthetic dataset $\mathcal{D}_s^T$.

4.2. Privacy Analysis

Theorem 4.1. *Algorithm 1* PCEvolve *satisfied ϵ_*-DP.*

Proof. Firstly, we conduct a privacy analysis for each query to privacy dataset $\mathcal{D}_p$ (Line 5 in Algorithm 1). For any synthetic data set $\mathcal{D}_s$ and its corresponding indexes set $\mathcal{R}$, the sensitivity Δ_u between two adjacent datasets $\mathcal{D}_p, \mathcal{D}_p' \in \mathcal{D}$ is bound by 1 according to our scoring function $u = h \circ g$:

$$\Delta_u = \max_{r \in \mathcal{R}} \max_{\mathcal{D}_p, \mathcal{D}_p' \in \mathcal{D}} |u(\mathcal{D}_p, r) - u(\mathcal{D}_p', r)| = 1.$$

Thus, based on Definition 3.3, each time privacy dataset $\mathcal{D}_p$ satisfies ϵ-DP.

Since we access $T \times C$ times to private data $\mathcal{D}_p$, according to the Definition 3.4, finally our PCEvolve satisfies ϵ_*-DP. $\qquad \square$

5. Experiments

5.1. Setup

Image Generation APIs. We consider three image generation APIs: Stable Diffusion (SD) (Rombach et al., 2022), SD with the IP-Adapter (Ye et al., 2023) plug-in (SD+IPA), and online OpenJourney (PromptHero, 2023) API (OJ (online)). We primarily use the widely adopted SD API (Lin et al., 2024) as the image generation API. Following PE, we generate N-shot synthetic images per class in the dataset $\mathcal{D}_s$, with a default setting of $N = 100$. In our scenario, an excessively large N is impractical for a resource-constrained client to generate image data using generative APIs.

Few-Shot Datasets. We evaluate PCEvolve on four datasets across two specialized domains under K-shot settings. In healthcare, we use (1) COVIDx (Wang et al., 2020) (chest X-ray images for COVID-19, two classes), (2) Camelyon17 (Koh et al., 2021) (tumor tissue patches from breast cancer metastases, two classes), and (3) KVASIR-f (endoscopic images for gastrointestinal abnormal findings detection subset from KVASIR (Pogorelov et al., 2017), three classes). In industry, we use MVTecAD-l (leather surface anomaly detection subset from MVTecAD (Bergmann et al., 2019), three classes). By default, we set $K = 10$, as MVTecAD-l has only 19 images per class. This value of K is typical for few-shot image tasks (He et al., 2023).

Baselines. We compare PCEvolve with six baselines across three categories, all of which focus on generating image datasets using untrusted black-box API(s), without training:

(I) *Using t2i APIs for image generation*:
• B (He et al., 2023), which uses only a t2i API with a simple text prompt $\mathcal{T}$ that includes only the domain and class label name. • LE (Seo et al., 2024), which extends B with a LLaMA (Touvron et al., 2023) API to enhance $\mathcal{T}$. • RF (Samuel et al., 2024), which filters out bad t2i-generated data that closely resemble private data from different classes. • GCap, which generates images using a t2i API with a LLaVA (Liu et al., 2023) API for extracting private image captions.

(II) *Using i2i APIs for image generation with DP*:
• DPImg, which directly adds DP (GM) noise to few-shot private images to generate DP replicas, which are then input to an i2i API. DPImg adapts RG (He et al., 2023) to ensure DP while avoiding modifications to the generative API.

(III) *Using t2i and i2i APIs for image generation with DP*:
• PE (Lin et al., 2024), like PCEvolve, generates DP synthetic image datasets using private data along with both t2i and i2i APIs within a privacy-preserving evolution loop.

Implementation Details. To maximize performance in few-shot scenarios following He et al. (2023), we train a new classification head for a given pre-trained backbone model on the final synthetic dataset $\mathcal{D}_s$. We report the Top-1 accuracy on the entire downstream test sets from the above datasets[3]. Top-1 accuracy, also known as the classification accuracy score (CAS) (Ravuri & Vinyals, 2019), is a widely used metric for assessing the quality of synthetic datasets in downstream tasks (Frolov et al., 2021; Lee et al., 2024). By default, we use ResNet-18 (He et al., 2016) as the pre-trained backbone model and encoder E_f due to its broad applicability across resource-constrained clients. For DP methods, the overall privacy cost ϵ_* is set to 10, 8, 8, and 10 for COVIDx, Camelyon17, KVASIR-f, and MVTecAD-l, respectively. For GM in DPImg and PE, we set δ to 10^{-5},

---

[3]For simplicity, we focus on balanced sets, following PE.

higher than EM's 0 in `PCEvolve`. We run each experiment three times and report the mean. Please refer to the specific experiments and Appendix for more details and results.

5.2. Performance of `PCEvolve`

CAS *w.r.t.* Four Specialized Datasets

Table 1: Top-1 accuracy (%) on four specialized datasets.

	COVIDx	Came17	KVASIR-f	MVAD-l
Init	49.34	50.47	33.43	33.33
RF	50.01	54.82	34.66	48.17
GCap	50.86	55.77	32.66	27.33
B	50.42	54.41	32.57	43.21
LE	50.02	55.44	35.51	27.93
DPImg	49.14	61.06	33.35	37.03
PE	59.63	63.66	48.88	57.41
PE-EM	57.60	63.34	43.01	50.06
`PCEvolve`-GM	56.91	62.63	43.55	55.56
`PCEvolve`	**64.04**	**69.10**	**50.95**	**59.26**

We first assess `PCEvolve` across four datasets from specialized domains like healthcare and industry in Tab. 1, where "Init" refers to the downstream models with initial heads. "Came17" and "MVAD" abbreviate Camelyon17 and MVTecAD for space efficiency, respectively. For category III, we additionally consider two variants of PE and `PCEvolve` as baselines: PE-EM, which applies the EM $\mathcal{M}_u$ to PE's original similarity votes, and `PCEvolve`-GM, which applies the GM $\mathcal{M}_\sigma$ to `PCEvolve`'s u values.

As shown in Tab. 1, the methods from category III outperform the others, with `PCEvolve` achieving the best performance among them. Specifically, `PCEvolve` surpasses baselines by up to 5.44% in accuracy on Came17. This phenomenon stems from the privacy-preserving evolution loop, which iteratively improves image quality while maintaining privacy within a given cost (*i.e.*, the privacy cost ϵ_*). RF and GCap indirectly utilize private images via private-image-based post-filtering and captions. Without privacy-preserving techniques, they leak more privacy than DP-protected methods (Sander et al., 2024), despite accessing private data only once without evolution. In contrast, B and LE use only text prompts, ensuring full privacy protection. However, their performance is slightly below RF and GCap. DPImg adds DP noise to private images, rendering them unrecognizable for required privacy, leading to poor or negative performance (Croft et al., 2021), especially on hard tasks, *e.g.*, COVIDx and KVASIR-f. All methods perform better in less specialized domains (roughly, for two-class datasets, COVIDx (chest X-ray) is harder than Came17 (tumor tissue), and for three-class datasets, KVASIR-f (medical) is harder than MVAD-l (industrial)). Additionally, APIs (SD, LLaMA, LLaVA, *etc*.) do not always enhance

performance, as they can introduce noise (Barman et al., 2024).

Within category III, PE-EM and `PCEvolve`-GM underperform PE and `PCEvolve`. PE-EM lacks similarity score calibration, weakening the EM, while `PCEvolve`-GM applies the GM to u scores, requiring high sensitivity ($\Delta_f = K \times N$) and reducing utility. By design, the similarity votes in PE align with the GM, while the calibrated u scores in `PCEvolve` match the EM. Compared to PE, `PCEvolve` is better suited for specialized domains with few-shot private data, as we incorporate inter-class contrastive relationships and maximize the EM's effectiveness under the same privacy cost.
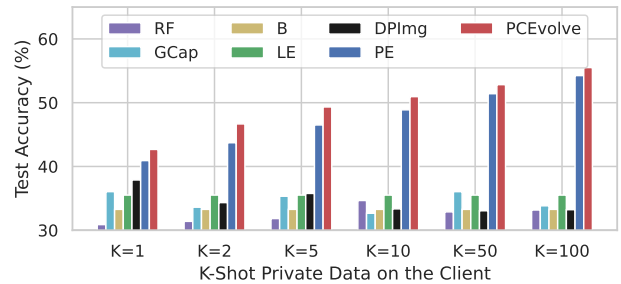
CAS *w.r.t.* K-Shot Private Data

Figure 3: Top-1 accuracy of ResNet-18 on KVASIR-f with varying shots of private data per class.

In the previous experiments, we used the default setting of $K = 10$ for K-shot private data on the client. We now explore different values of K to examine how varying the amount of private data affects downstream models. As shown in Fig. 3, methods that leverage private data to filter or evaluate generated synthetic data, such as RF, PE, and `PCEvolve`, tend to perform better given more private data. The performance improvement is particularly noticeable for PE and `PCEvolve`, since both methods rely on an evolution loop that iteratively accesses private data multiple times. This repeated access allows them to benefit more from richer private data, leading to a greater enhancement in the quality of the synthetic data generated and, consequently, the downstream model. In scenarios with extremely small amounts of private data, such as $K = 1$, the class center aggregation subroutine in our `PCEvolve` becomes invalid, as the aggregated private class centers are identical to the private data. Despite this, `PCEvolve` still outperforms other methods with our two key components: contrastive filter (g) and similarity calibrator (h). The performance of B and LE remains unaffected by K since they do not require private data. DPImg performs worse as the amount of private data increases since DP requires adding more noise to ensure privacy.

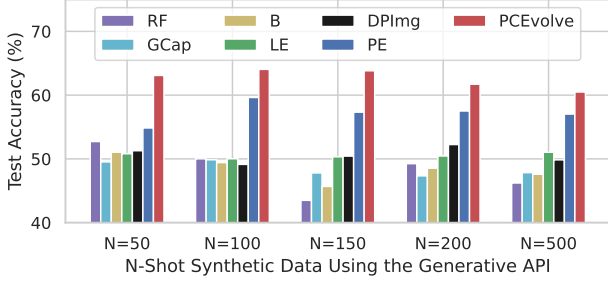**CAS *w.r.t.* N-Shot Synthetic Data**

Figure 4: Top-1 accuracy of ResNet-18 on COVIDx with varying synthetic data shots per class per iteration.

We default to generating $N = 100$ synthetic images per class on the client. This choice helps mitigate the risk of generative APIs introducing noise due to insufficient contextual information (Ronanki et al., 2024; Wang et al., 2023), which, in our scenario, comes from K-shot ($K = 10$) private data. The noise becomes even more pronounced when generating a large volume of synthetic data, as shown in Fig. 4 ($N = 150/200/500$), where accumulated noise and potential inaccuracies degrade the overall quality and utility of the dataset. Moreover, a larger N increases API resource consumption, which may become unaffordable for resource-constrained clients. On the other hand, if the synthetic data amount is too small (*e.g.*, $N = 50$), there is insufficient knowledge for downstream models to effectively learn from the generative APIs. However, our PCEvolve maintains its performance even at $N = 50$, with only a 0.95% drop in accuracy, whereas PE suffers a 4.78% decline. This demonstrates that PCEvolve is more adaptable to resource-constrained clients, highlighting its practical value.

CAS *w.r.t*. Various APIs

Table 2: Top-1 accuracy (%) on COVIDx and KVASIR-f using SD+IPA and OJ (online) APIs.

	COVIDx		KVASIR-f	
APIs	SD+IPA	OJ (online)	SD+IPA	OJ (online)
RF	45.03	47.91	27.22	36.55
GCap	53.70	47.42	28.77	37.11
B	46.61	50.22	26.27	36.61
LE	49.79	53.17	31.22	37.38
DPImg	50.58	49.61	36.89	35.05
PE	56.92	54.47	48.83	48.17
PCEvolve	**60.46**	**65.88**	**52.77**	**54.58**

To assess PCEvolve's applicability with other image generation APIs, we additionally use SD+IPA and OJ (online) on COVIDx and KVASIR-f. Generative models behind different APIs are pre-trained on diverse large-scale pub-

lic datasets (Schuhmann et al., 2022; Byeon et al., 2022), resulting in varying knowledge. Despite these differences, PCEvolve adapts to different generative APIs while maintaining its superiority, as shown in Tab. 2. Since SD, SD+IPA, and OJ (online) share a similar generative backbone, with OJ (online) fine-tuned on SD, we infer that SD+IPA and OJ (online) have been exposed to more public data than SD, leading to broader knowledge. From Tab. 1 and Tab. 2, we observe an interesting trend: PCEvolve benefits from APIs trained on larger public datasets, whereas PE experiences a slight performance decline under the same conditions.

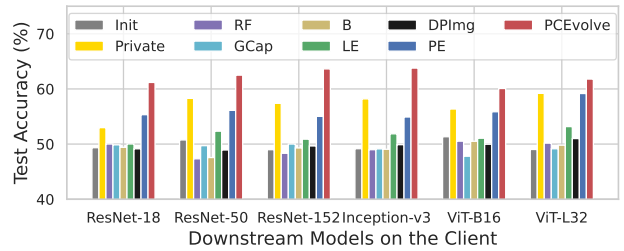**CAS *w.r.t*. Various Downstream Models**

Figure 5: Top-1 accuracy of various downstream models on COVIDx. "Private" represents an additional private baseline, which directly trains downstream models on few-shot private data.

To evaluate the versatility of the synthetic dataset for downstream models, we consider six models that are widely used in specialized image domains (Sarwinda et al., 2021; Manzari et al., 2023), as shown in Fig. 5. Following our setup in Sec. 5.1, we use pre-trained backbones and train newly initialized classification heads on the synthetic dataset. Here, we use the CLIP image encoder (Radford et al., 2021) as the encoder E_f following (Lin et al., 2024).

In Fig. 5, the synthetic image dataset generated by our PCEvolve is compatible with various downstream models of different architectures (*e.g.*, CNNs (LeCun et al., 2015) and Transformers (Vaswani et al., 2017)). Models trained on our PCEvolve's synthetic dataset achieve the best performance among all counterparts. Notably, PCEvolve surpasses the "Private" baseline by up to 8.20%, which directly trains downstream models on few-shot private data for the same number of training steps as PCEvolve. This demonstrates that our synthetic dataset incorporates valuable information from the generative API beyond what private data alone provides. In contrast, PE underperforms compared to "Private" when using larger models other than ResNet-18. While both PCEvolve and PE improve downstream model performance over their initial states, most one-time generation methods degrade, except for LE. This highlights the substantial domain gap between synthetic and private

datasets when an evolution loop is not used to incorporate sufficient private information.

5.3. Properties of **PCEvolve**

| (a) Initial | (b) PE | (c) `PCEvolve` | (d) Private |

Figure 6: Generated leather surface images *w.r.t.* MVAD-l for industry anomaly detection. The three rows show normal images, cut defects, and droplet defects. "Initial" denotes the initial synthetic images in PE and `PCEvolve`. "Private" denotes the real images from MVAD-l.

Synthetic Images

As illustrated in Fig. 6, the initial synthetic images differ significantly in meaning, content, color, and detail from few-shot private images. While PE employs an evolution loop to refine the initial images iteratively, its DP-protected similarity voting becomes nearly random with a few-shot private dataset. This limitation results in synthetic images that retain much information from the initial data while remaining distant from the private data. For instance, the synthetic images in Fig. 6(b) fail to accurately depict the cut defect, instead emphasizing the boundary of a leather surface, as in Fig. 6(a). Although the normal and droplet defect images align with the meaning of private images, they still differ in style and detail, leading to a significant domain gap. Such synthetic images fail to enhance downstream tasks and may even negatively impact them (Hataya et al., 2023). In contrast, our `PCEvolve` effectively tackles the few-shot challenge by leveraging inter-class contrastive relationships via the contrastive filter (g), resulting in class-discriminative leather images. As illustrated in Fig. 6(c), our synthetic images demonstrate greater diversity, such as varied lighting angles, across classes with additional knowledge extracted from APIs, while preserving a high degree of similarity to private images. This high similarity is achieved through our similarity calibrator (h), which enhances the likelihood of selecting the most similar candidates as prototypical syn-

thetic images.

Ablation Study

Table 3: Top-1 accuracy (%) on four datasets using `PCEvolve` ($u = h \circ g$) variants with different u. h' denotes the initial h function in Eq. (2).

	$u = h \circ g$	$u = g$	$u = h' \circ g$	$u = h$
COVIDx	64.04	56.59	56.11	55.58
Came17	69.10	66.51	65.29	59.39
KVASIR-f	50.95	44.61	50.67	47.78
MVAD-l	59.26	55.74	55.56	53.71

The design of the utility function u is pivotal to `PCEvolve`'s effectiveness. In Tab. 3, we replace u with alternative variants to demonstrate the significance of each component in `PCEvolve`. Notably, the contrastive filter (g) plays an important role, achieving strong performance even when used alone. When combined with the similarity calibrator (h), `PCEvolve` achieves up to a 7.45% accuracy improvement on COVIDx. However, directly using the uncalibrated similarity scores from $h' \circ g$ leads to a performance drop of up to 7.93% accuracy on COVIDx compared to `PCEvolve`. Furthermore, removing g leads to an even more significant performance drop, with accuracy decreasing by 9.71% on Came17 at most.

Hyperparameter Study

Table 4: Top-1 accuracy (%) on four datasets using `PCEvolve` with varying τ values.

	$\tau = 1$	$\tau = 5$	$\tau = 10$	$\tau = 20$	$\tau = 100$
COVIDx	59.34	60.70	64.04	60.61	57.90
Came17	65.85	66.84	69.10	68.42	68.05
KVASIR-f	46.89	47.29	50.95	57.08	54.66
MVAD-l	40.70	53.71	59.26	55.75	55.55

`PCEvolve` has only one hyperparameter: the similarity calibrating factor τ in the similarity calibrator (h) (Eq. (3)). As shown in Tab. 4, selecting an appropriate τ enhances `PCEvolve`'s performance, with the optimal value being $\tau = 10$ for COVIDx, Came17, and MVAD-l, while KVASIR-f achieves its best performance at $\tau = 20$. When τ is too small (*e.g.*, $\tau = 1$), the u scores (*i.e.*, outputs of h) become similar among class-discriminative synthetic data, leading to nearly uniform selection probabilities after applying the EM $\mathcal{M}_u$ (Definition 3.3). Conversely, when τ is too large (*e.g.*, $\tau = 100$), most u scores become zero, as u computes $e^{-\ell \cdot \tau}$, where ℓ is the normalized ℓ_2 distance in Eq. (3). In this case, only the best candidate (with $\ell = 0$) retains a u score of 1, while all others—including non-discriminative candidates—approach (near) 0, inadvertently increasing the

likelihood of selecting poor candidates according to Definition 3.3. To balance this trade-off, we choose an appropriate $\tau = 10$ for `PCEvolve`.

6. Conclusion and Limitation

Our proposed `PCEvolve` effectively addresses the few-shot private data challenge in DP generation with APIs, particularly for specialized domains like healthcare and industry, as shown by `PCEvolve`'s superiority on four specialized datasets with various scenarios. By leveraging extra inter-class contrastive relationships in private data and proposing an adapted EM, `PCEvolve` outperforms existing API-assisted methods, enabling high-quality DP synthetic images while leaving more practical few-shot scenarios for future exploration.

Acknowledgments

Impact Statement

This work highlights practical privacy-preserving generative API applications with a Differential Privacy (DP) guarantee, demonstrating the effectiveness of API-assisted training-free DP synthetic data generation in specialized domains. Apart from this contribution, we do not identify any significant societal implications that require specific attention.

References

Abou Baker, N., Rohrschneider, D., and Handmann, U. Parameter-efficient fine-tuning of large pretrained models for instance segmentation tasks. *Machine Learning and Knowledge Extraction*, 6(4):2783–2807, 2024.

Barman, D., Guo, Z., and Conlan, O. The dark side of language models: Exploring the potential of llms in multimedia disinformation generation and dissemination. *Machine Learning with Applications*, pp. 100545, 2024.

Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Boland, M. R., Karczewski, K. J., and Tatonetti, N. P. Ten

simple rules to enable multi-site collaborations through data sharing, 2017.

Brooks, T., Holynski, A., and Efros, A. A. Instructpix2pix: Learning to follow image editing instructions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Byeon, M., Park, B., Kim, H., Lee, S., Baek, W., and Kim, S. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022.

Carlini, N., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, 2022.

Chen, J., Tang, J., and Li, W. Industrial edge intelligence: Federated-meta learning framework for few-shot fault diagnosis. *IEEE Transactions on Network Science and Engineering*, 10(6):3561–3573, 2023.

Chen, J.-W., Yu, C.-M., Kao, C.-C., Pang, T.-W., and Lu, C.-S. Dpgen: Differentially private generative energy-guided network for natural image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, 2020a.

Chen, X., Tang, S., Zhu, R., Yan, S., Jin, L., Wang, Z., Su, L., Zhang, Z., Wang, X., and Tang, H. The janus interface: How fine-tuning in large language models amplifies the privacy risks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 2024.

Chen, Y., Qin, X., Wang, J., Yu, C., and Gao, W. Fedhealth: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, 35(4):83–93, 2020b.

Croft, W. L., Sack, J.-R., and Shi, W. Obfuscation of images via differential privacy: From facial images to general images. *Peer-to-Peer Networking and Applications*, 14: 1705–1733, 2021.

De Cristofaro, E. Synthetic data: Methods, use cases, and risks. *IEEE Security & Privacy*, 2024.

Dong, J., Durfee, D., and Rogers, R. Optimal differential privacy composition for exponential mechanisms. In *International Conference on Machine Learning (ICML)*, 2020.

Duan, H., Dziedzic, A., Papernot, N., and Boenisch, F. Flocks of stochastic parrots: Differentially private prompt learning for large language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Dwork, C. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, 2008.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. *Theory of Cryptography*, pp. 265–284, 2006.

Dwork, C., Roth, A., et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

Frolov, S., Hinz, T., Raue, F., Hees, J., and Dengel, A. Adversarial text-to-image synthesis: A review. *Neural Networks*, 144:187–209, 2021.

Fu, J., Hong, Y., Ling, X., Wang, L., Ran, X., Sun, Z., Wang, W. H., Chen, Z., and Cao, Y. Differentially private federated learning: A systematic review. *arXiv preprint arXiv:2405.08299*, 2024a.

Fu, J., Ye, Q., Hu, H., Chen, Z., Wang, L., Wang, K., and Ran, X. Dpsur: Accelerating differentially private stochastic gradient descent using selective update and release. *Proceedings of the VLDB Endowment*, 2024b.

Gao, J., Pi, R., Yong, L., Xu, H., Ye, J., Wu, Z., ZHANG, W., Liang, X., Li, Z., and Kong, L. Self-guided noise-free data generation for efficient zero-shot learning. In *International Conference on Learning Representations (ICLR)*, 2023.

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

Guan, H. and Liu, M. Domain adaptation for medical image analysis: a survey. *IEEE Transactions on Biomedical Engineering*, 69(3):1173–1185, 2021.

Hao, Y., Chi, Z., Dong, L., and Wei, F. Optimizing prompts for text-to-image generation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Harmon, S. A., Sanford, T. H., Xu, S., Turkbey, E. B., Roth, H., Xu, Z., Yang, D., Myronenko, A., Anderson, V., Amalou, A., et al. Artificial intelligence for the detection of covid-19 pneumonia on chest ct using multinational datasets. *Nature communications*, 11(1):4080, 2020.

Hataya, R., Bao, H., and Arai, H. Will large-scale generative models corrupt future datasets? In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

He, R., Sun, S., Yu, X., Xue, C., Zhang, W., Torr, P., Bai, S., and Qi, X. Is synthetic data from generative models ready for image recognition? In *International Conference on Learning Representations (ICLR)*, 2023.

Hou, C., Shrivastava, A., Zhan, H., Conway, R., Le, T., Sagar, A., Fanti, G., and Lazar, D. Pre-text: Training language models on private federated data in the age of llms. In *International Conference on Machine Learning (ICML)*, 2024.

Hu, R., Guo, Y., Ratazzi, E. P., and Gong, Y. Differentially private federated learning for resource-constrained internet of things. *arXiv preprint arXiv:2003.12705*, 2020.

Hu, Y., Wu, F., Li, Q., Long, Y., Garrido, G. M., Ge, C., Ding, B., Forsyth, D., Li, B., and Song, D. Sok: Privacy-preserving data synthesis. In *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2024.

Ji, W. and Chung, A. C. Diffusion-based domain adaptation for medical image segmentation using stochastic step alignment. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024.

Kather, J. N., Ghaffari Laleh, N., Foersch, S., and Truhn, D. Medical domain knowledge in domain-agnostic generative ai. *NPJ digital medicine*, 5(1):90, 2022.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

Kulis, B. et al. Metric learning: A survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013.

LeCun, Y., Bengio, Y., and Hinton, G. Deep Learning. *Nature*, 521(7553):436–444, 2015.

Lee, T., Yasunaga, M., Meng, C., Mai, Y., Park, J. S., Gupta, A., Zhang, Y., Narayanan, D., Teufel, H., Bellagente, M., et al. Holistic evaluation of text-to-image models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Li, K., Gong, C., Li, Z., Zhao, Y., Hou, X., and Wang, T. {PrivImage}: Differentially private synthetic image generation using diffusion models with {Semantic-Aware} pretraining. In *33rd USENIX Security Symposium (USENIX Security 24)*, 2024.

Lin, Z., Gopi, S., Kulkarni, J., Nori, H., and Yekhanin, S. Differentially private synthetic data via foundation model apis 1: Images. In *International Conference on Learning Representations (ICLR)*, 2024.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Luo, X., Chang, X., and Ban, X. Regression and classification using extreme learning machine based on l1-norm and l2-norm. *Neurocomputing*, 174:179–186, 2016.

Manzari, O. N., Ahmadabadi, H., Kashiani, H., Shokouhi, S. B., and Ayatollahi, A. Medvit: a robust vision transformer for generalized medical image classification. *Computers in Biology and Medicine*, 157:106791, 2023.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2007.

Meskó, B. Prompt engineering as an important emerging skill for medical professionals: tutorial. *Journal of medical Internet research*, 25:e50638, 2023.

Moor, M., Banerjee, O., Abad, Z. S. H., Krumholz, H. M., Leskovec, J., Topol, E. J., and Rajpurkar, P. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265, 2023.

OpenAI. Fine-tuning models - openai platform, 2024. URL https://platform.openai.com/docs/guides/fine-tuning#vision. Accessed: 2025-01-22.

Pan, Z., Zhou, X., and Tian, H. Arbitrary style guidance for enhanced diffusion-based text-to-image generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.

Pogorelov, K., Randel, K. R., Griwodz, C., Eskeland, S. L., de Lange, T., Johansen, D., Spampinato, C., Dang-Nguyen, D.-T., Lux, M., Schmidt, P. T., et al. Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, 2017.

PromptHero. Openjourney, 2023. URL https://openjourney.art/.

Qi, X., Zeng, Y., Xie, T., Chen, P.-Y., Jia, R., Mittal, P., and Henderson, P. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations (ICLR)*, 2024.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

Ravuri, S. and Vinyals, O. Classification accuracy score for conditional generative models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Ronanki, K., Cabrero-Daniel, B., and Berger, C. Prompt smells: An omen for undesirable generative ai outputs. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI*, 2024.

Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., and Gehler, P. Towards total recall in industrial anomaly detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Samuel, D., Ben-Ari, R., Raviv, S., Darshan, N., and Chechik, G. Generating images of rare concepts using pre-trained diffusion models. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2024.

Sander, T., Yu, Y., Sanjabi, M., Durmus, A. O., Ma, Y., Chaudhuri, K., and Guo, C. Differentially private representation learning via image captioning. In *International Conference on Machine Learning (ICML)*, 2024.

Sarwinda, D., Paradisa, R. H., Bustamam, A., and Anggia, P. Deep learning in image classification using residual network (resnet) variants for detection of colorectal cancer. *Procedia Computer Science*, 179:423–431, 2021.

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Seo, M., Cho, S., Lee, M., Misra, D., Choi, H., Kim, S. J., and Choi, J. Just say the name: Online continual learning with category names only via data generation. *arXiv preprint arXiv:2403.10853*, 2024.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, 2017.

Song, Y., Wang, T., Cai, P., Mondal, S. K., and Sahoo, J. P. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Computing Surveys*, 55(13s):1–40, 2023.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is All You Need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Wang, L., Lin, Z. Q., and Wong, A. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific reports*, 10(1):19549, 2020.

Wang, Y., Shen, S., and Lim, B. Y. Reprompt: Automatic prompt editing to refine ai-generative art towards precise expressions. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, 2023.

Wornow, M., Xu, Y., Thapa, R., Patel, B., Steinberg, E., Fleming, S., Pfeffer, M. A., Fries, J., and Shah, N. H. The shaky foundations of large language models and foundation models for electronic health records. *npj Digital Medicine*, 6(1):135, 2023.

Xie, C., Lin, Z., Backurs, A., Gopi, S., Yu, D., Inan, H. A., Nori, H., Jiang, H., Zhang, H., Lee, Y. T., et al. Differentially private synthetic data via foundation model apis 2: Text. In *International Conference on Machine Learning (ICML)*, 2024.

Xu, J., Luo, X., Pan, X., Li, Y., Pei, W., and Xu, Z. Alleviating the sample selection bias in few-shot learning by removing projection to the centroid. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., and Yang, M.-H. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

Yang, S., Liu, L., and Xu, M. Free lunch for few-shot learning: Distribution calibration. In *International Conference on Learning Representations (ICLR)*, 2021.

Ye, H., Zhang, J., Liu, S., Han, X., and Yang, W. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

Ye, J., Gao, J., Wu, Z., Feng, J., Yu, T., and Kong, L. Progen: Progressive zero-shot dataset generation via in-context feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2022.

Zhang, J., Chen, Y., and Li, H. Privacy leakage of adversarial training models in federated learning systems. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Zhang, J., Hua, Y., Wang, H., Song, T., Xue, Z., Ma, R., and Guan, H. FedALA: Adaptive Local Aggregation for Personalized Federated Learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2023a.

Zhang, J., Liu, Y., Hua, Y., and Cao, J. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2024a.

Zhang, J., Liu, Y., Hua, Y., Wang, H., Song, T., Xue, Z., Ma, R., and Cao, J. Pfllib: A beginner-friendly and comprehensive personalized federated learning library and benchmark. *Journal of Machine Learning Research*, 26(50):1–10, 2025.

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *IEEE International Conference on Computer Vision (ICCV)*, 2023b.

Zhang, Y., Liang, X., Du, R., and Tian, J. Dp-discriminator: A differential privacy evaluation tool based on gan. In *Proceedings of the 21st ACM International Conference on Computing Frontiers*, 2024b.

Ziller, A., Usynin, D., Braren, R., Makowski, M., Rueckert, D., and Kaissis, G. Medical imaging deep learning with differential privacy. *Scientific Reports*, 11(1):13524, 2021.

A. Experimental Details

We have included the necessary experimental details in the main body, and show more details here.

A.1. Image Generation APIs

We consider three image generation APIs: Stable Diffusion (SD) (Rombach et al., 2022), SD with the IP-Adapter (SD+IPA) (Ye et al., 2023), and the online OpenJourney (OJ) API (PromptHero, 2023) (OJ (online)). Following PE (Lin et al., 2024), we manually implement SD on a server, providing an SD API with both text-to-image (t2i) and image-to-image (i2i) features. Specifically, we use the pre-trained open-source SD v1.5 model from HuggingFace[4] and wrap it to expose only the API interface and serve as an API server, keeping the model details hidden from clients. SD+IPA is implemented similarly, by integrating a pre-trained open-source IP-Adapter[5] to the SD API. For the OJ (online) API, we use the getimg.ai platform to access the online OJ API[6], which requires payments before usage. We retain the default settings (such as `num_inference_steps` =50, `guidance_scale` =7.5, *etc*.) for all APIs to ensure better generalization. Similar to PE, we initialize the `strength` at 0.8 and anneal it to 0.6 with a 0.02 decrease per iteration for the i2i API. Additionally, we set `scale` to 0.5 for the IP-Adapter. Further details are available in our code.

A.2. Few-Shot Datasets

We evaluate `PCEvolve` on four datasets across two specialized domains under K-shot settings. In healthcare, we use (1) COVIDx[7] (Wang et al., 2020) (chest X-ray images for COVID-19, two classes), (2) Camelyon17[8] (Koh et al., 2021) (tumor tissue patches from breast cancer metastases, two classes), and (3) KVASIR-f[9] (endoscopic images for gastrointestinal abnormal findings detection subset from KVASIR (Pogorelov et al., 2017), three classes). In industry, we use MVTecAD-l[10] (leather surface anomaly detection subset from MVTecAD (Bergmann et al., 2019), three classes). Specifically, KVASIR-f is a subset of KVASIR containing all pathological finding images, while MVTecAD-l is a subset of MVTecAD focused on leather surface images. By default, we set $K = 10$ for MVTecAD-l, containing only 19 leather surface images per class. We use 10 images for synthetic image generation and reserve the remaining 9 images for evaluating the test accuracy of the fine-tuned downstream models. This value of K is typical for few-shot image tasks (He et al., 2023). We create few-shot subsets from these datasets to represent realistic scenarios. In Tab. 5, we list each dataset's details. We use a uniform simple text prompt $\mathcal{T} :=$ "A DOMAIN image with LABEL", where "DOMAIN" and "LABEL" are placeholders for respective domain and label names, for all datasets and tasks. More details are available in our code.

Table 5: The details of four few-shot datasets from two specialized domains.

Dataset	Image Size	Test Set Size	Domain	Labels
COVIDx	256x256	8482	"chest radiography (X-ray)"	["", "COVID-19 pneumonia"]
Camelyon17	96x96	85054	"histological lymph node section"	["", "breast cancer with a tumor tissue"]
KVASIR-f	256x256	600	"pathological damage in mucosa of gastrointestinal tract"	["esophagitis", "polyps", "ulcerative-colitis"]
MVTecAD-l	256x256	27	"leather texture"	["", "cut defect", "droplet defect"]

A.3. Baselines

We compare `PCEvolve` with six baselines across three categories, all of which focus on generating image datasets using untrusted black-box API(s), without training:

(I) *Using t2i APIs for image generation*:
• B (He et al., 2023), which uses only a t2i API with a simple text prompt $\mathcal{T}$ that includes only the domain and class label

---

[4] https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5
[5] https://huggingface.co/h94/IP-Adapter
[6] https://dashboard.getimg.ai/models
[7] https://www.kaggle.com/datasets/andyczhao/covidx-cxr2
[8] https://wilds.stanford.edu/datasets/#camelyon17
[9] https://datasets.simula.no/kvasir/
[10] https://www.mvtec.com/company/research/datasets/mvtec-ad

name. • LE (Seo et al., 2024), which extends B with a LLaMA (Touvron et al., 2023) API to enhance $\mathcal{T}$. Specifically, we use an additional text prompt to enhance $\mathcal{T}$ with the LLaMA API: "refine this description of images to introduce rich context: ".
• RF (Samuel et al., 2024), which filters out bad t2i-generated data that closely resemble private data from different classes.
• GCap, which generates images using a t2i API with a LLaVA (Liu et al., 2023) API for extracting private image captions.

(II) *Using i2i APIs for image generation with DP*:
• DPImg, which directly adds DP (GM) noise to few-shot private images to generate DP replicas, which are then input to an i2i API. DPImg adapts RG (He et al., 2023) to ensure DP while avoiding modifications to the generative API. We compute the σ for GM based on Definition 3.2 given a total privacy cost ϵ_*.

(III) *Using t2i and i2i APIs for image generation with DP*:
• PE (Lin et al., 2024), like `PCEvolve`, generates DP synthetic image datasets using private data along with both t2i and i2i APIs within a privacy-preserving evolution loop. In few-shot scenarios, there are too few votes but too much noise in PE, making the thresholding operation on similarity votes meaningless. Therefore, we set $H = 0$ for PE. Additionally, as shown in the PE paper, PE performs similarly for $H \geq 0$ when $\epsilon_* > 2$.

A.4. Implementation Details

To maximize performance in few-shot scenarios, following (He et al., 2023), we train a new classification head on a pre-trained backbone model using the final synthetic dataset $\mathcal{D}_s$. The downstream model training during evaluation is done with a batch size of 16, a learning rate of 0.001, and 100 epochs. We report the Top-1 accuracy on the entire downstream test sets (see Tab. 5). Top-1 accuracy, also known as the classification accuracy score (CAS) (Ravuri & Vinyals, 2019), is a widely used metric for assessing the quality of synthetic datasets in downstream tasks (Frolov et al., 2021; Lee et al., 2024). By default, we use ResNet-18 (He et al., 2016; Zhang et al., 2025; 2023a; 2024a) as the pre-trained backbone model and encoder E_f due to its broad applicability across resource-constrained clients. For DP methods, the overall privacy cost ϵ_* is set to 10, 8, 8, and 10 for COVIDx, Camelyon17, KVASIR-f, and MVTecAD-l, respectively. For GM in DPImg and PE, we set δ to 10^{-5}, higher than EM's 0 in `PCEvolve`. By default, we set the total generation iteration $T = 20$ for `PCEvolve`. Most of our experiments are run on a machine with 64 Intel(R) Xeon(R) Platinum 8362 CPUs, 256GB of memory, eight NVIDIA 3090 GPUs, and Ubuntu 20.04.4 LTS. While most experiments are completed within 48 hours, those involving a large N for N-shot image generation may take up to a week.

B. CAS *w.r.t.* Overall Privacy Cost ϵ_*

Table 6: Top-1 accuracy (%) on four specialized datasets with varying overall privacy cost ϵ_*.

Dataset	DP Algorithm	$\epsilon_* = 4$	$\epsilon_* = 8$	$\epsilon_* = 10$	$\epsilon_* = 20$
COVIDx	PE	52.76	57.83	59.63	62.44
	`PCEvolve`	56.74	60.21	64.04	63.82
Camelyon17	PE	62.29	63.66	63.37	65.29
	`PCEvolve`	68.05	69.11	69.58	69.95
KVASIR-f	PE	43.72	48.88	51.01	51.83
	`PCEvolve`	50.44	50.95	51.67	52.11
MVTecAD-l	PE	50.21	55.85	57.41	58.02
	`PCEvolve`	51.84	57.41	59.26	60.67

To study the impact of the overall privacy cost ϵ_* on iterative generation algorithms, such as PE and our `PCEvolve`, we vary ϵ_* and present the results in Tab. 6. We observe that both PE and `PCEvolve` achieve lower accuracy with a smaller ϵ_* and perform better with larger values, consistent with the DP literature (Dwork et al., 2014). To balance the privacy-utility trade-off, selecting an appropriate ϵ_* is crucial for different tasks and environments (Lin et al., 2024).
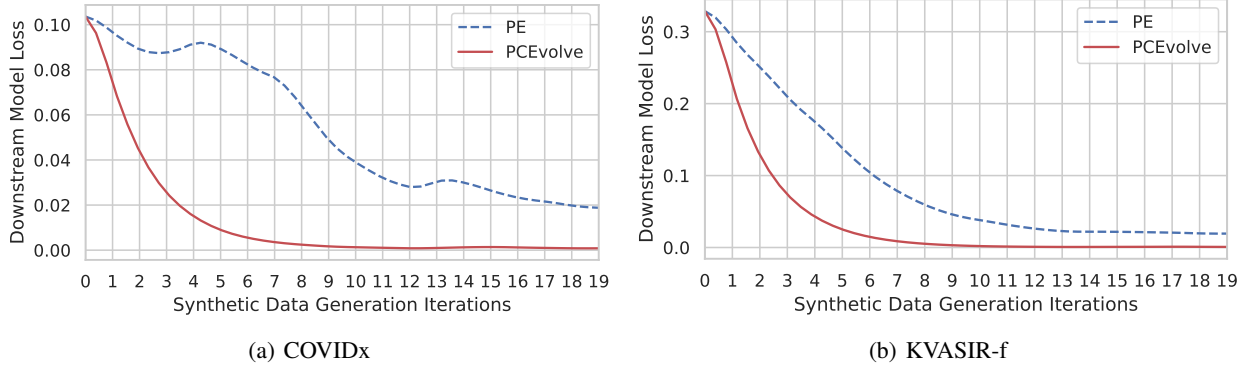
(a) COVIDx

(b) KVASIR-f

Figure 7: The loss curves of ResNet-18 (He et al., 2016), which is *retrained* at each iteration of synthetic data generation for algorithm performance evaluation. We use a CLIP image encoder (Radford et al., 2021) as the encoder.

C. Iterative Improvement in the Evolution Loop

In the main body, following (He et al., 2023), we train a new classification head for a pre-trained downstream model (*e.g.*, ResNet-18) on the final synthetic dataset $\mathcal{D}_s$. To demonstrate the iterative improvement in the evolution loop, we retrain a new classification head for the given downstream model at each generation iteration and track the loss value after training on the synthetic data. The loss curves are shown in Fig. 7. Although the downstream model starts with the same initial loss value at the 0th iteration for all methods, our `PCEvolve` rapidly reduces the loss in early iterations and consistently maintains a near-zero loss value in subsequent iterations, demonstrating steady iterative improvement. In contrast, PE also starts with the same initial loss but experiences fluctuations throughout the evolution process, ultimately reaching a higher loss. This is due to its GM-based similarity voting approach, which results in noisy synthetic images that hinder its performance.
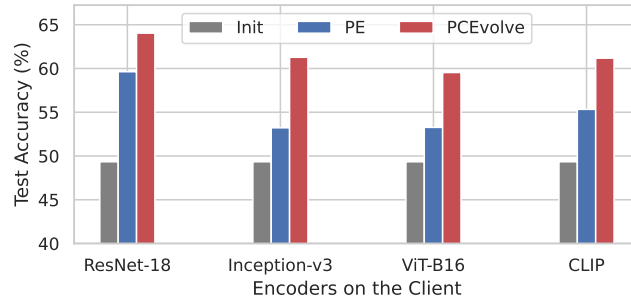
D. CAS *w.r.t.* Various Encoders E_f

Figure 8: Top-1 accuracy of ResNet-18 on COVIDx using four encoders. "CLIP" is short for CLIP image encoder.

Here, we analyze PE and `PCEvolve` to assess the impact of different encoders, which map images into feature vectors for efficient distance computation when evaluating synthetic data quality against private data. In Fig. 8, we observe that the downstream model (ResNet-18) achieves optimal performance when paired with the same encoder (ResNet-18). When using other encoders, our `PCEvolve` shows less performance degradation (4.50%) compared to PE (6.41%). A small encoder like ResNet-18 (0.01B parameters) is more practical than a large one like CLIP image encoder (0.3B parameters), as encoding is performed on resource-constrained clients and is required throughout the iterative synthetic data generation process. Both `PCEvolve` and PE improve the performance of the initial downstream models with different encoders.

E. CAS on the Mixture of Synthetic and Private Data

We follow PE (Lin et al., 2024) to evaluate the quality of the synthetic image data in the main body, considering scenarios where synthetic data is widely utilized in various downstream tasks outside the private client, and the original private dataset is typically not accessible. In specific cases where private data owners wish to augment their local private datasets with synthetic data, the private data can be accessed, and the synthetic and private data can be mixed for augmentation.

Table 7: Top-1 accuracy (%) on the mixture of synthetic and private data based on KVASIR-f. "Syn" is short for synthetic.

	Syn	Syn + Private
Init		33.43
Private		83.17
RF	34.66	81.61 (-1.56)
GCap	32.66	84.78 (+1.61)
B	32.57	81.01 (-2.16)
LE	35.51	84.61 (+1.44)
DPImg	33.35	62.94 (-20.23)
PE	48.88	88.67 (+5.50)
PCEvolve	**50.95**	**90.51 (+7.34)**

For this scenario, we follow He et al. (2023) and apply the mix training with the default downstream model (ResNet-18). As shown in Tab. 7, most baselines improve the performance when mixing synthetic and private data, with PCEvolve showing the highest improvement by enhancing the class-discriminability of synthetic data, which benefits classification tasks. Although some t2i baselines (*e.g.*, GCap and LE) perform poorly when evaluated on synthetic data alone, they show positive performance when applied to mixed datasets, as they bring additional valuable knowledge from APIs to the private data. In contrast, DPImg can negatively impact private data, as its synthetic data contains significant noise.