PRIVACY-AWARE POST-TRAINING QUANTIZATION FOR MITIGATING MEMBERSHIP INFERENCE IN DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models are vulnerable to membership inference attacks that can reveal whether a sample was used during training. Existing defenses typically rely on privacy-aware training, which adds computational cost and can degrade generation quality. Meanwhile, model quantization provides an efficient way to compress AI models and can affect their membership privacy, yet existing quantization methods are not specified to optimize diffusion models for membership leakage. In this work, we propose PAQ-Diffusion, a privacy-aware post-training quantization framework that incorporates membership privacy into quantization optimization without requiring privacy-aware retraining. PAQ-Diffusion first performs layer-wise privacy-aware reconstruction by aligning the quantization residual distributions of member and non-member samples while preserving reconstruction fidelity. It then introduces timestep-specific activation quantization scales and optimizes them with a denoising-based privacy objective, allowing privacy optimization to be performed independently across timesteps. We evaluate the proposed PAQ-Diffusion on different diffusion models against representative membership inference attacks. Experimental results demonstrate that PAQ-Diffusion reduces membership leakage (e.g., reducing SecMI-NN attack AUC by up to 20.62%) while maintaining superior generation quality, providing an effective privacy-utility trade-off for quantized diffusion models.

1 INTRODUCTION

Diffusion models have emerged as one of the most powerful paradigms for image generation. By learning to progressively reverse a noise-injection process, diffusion models can generate diverse and high-fidelity images (Ho et al., 2020; Dhariwal & Nichol, 2021). Subsequent advances, including classifier-free guidance, accelerated sampling, and latent-space modeling, have further improved their controllability, sampling efficiency, and scalability (Ho & Salimans, 2021; Song et al., 2021; Rombach et al., 2022; Zhang & Chen, 2023). These developments have enabled diffusion models to serve as the foundation of modern image-generation systems.

Despite their impressive generation capabilities, diffusion models are often computationally expensive, even during inference. Their denoising networks typically contain many parameters and must be repeatedly sampled over multiple timesteps, resulting in substantial memory consumption and inference latency (Shang et al., 2023; He et al., 2023). These computational demands hinder deployment on resource-constrained and latency-sensitive platforms, e.g., smartphones and IoT devices. Model quantization has therefore received increasing attention as an effective approach to reduce storage and inference costs. Existing studies have investigated both quantization-aware training and post-training quantization for diffusion models (Shang et al., 2023; He et al., 2023; Huang et al., 2024; Zhao et al., 2024).

Meanwhile, the growing adoption of diffusion models has introduced substantial security and privacy concerns. These models may generate harmful or inappropriate content and can inadvertently memorize information contained in their training data (Carlini et al., 2023; Truong et al., 2025). In particular, recent studies have demonstrated that diffusion models are vulnerable to membership inference attacks, which aim to determine whether a given sample was included in the model's

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

training set (Duan et al., 2023; Matsumoto et al., 2023; Hu & Pang, 2023). More advanced attacks (Kong et al., 2024; Zhai et al., 2024; Zhao et al., 2025; Li et al., 2025) exploit reconstruction errors, likelihood-related signals, frequency characteristics, or carefully selected initialization strategies to improve membership inference attack performance. Such attacks pose serious privacy risks, especially when diffusion models are trained on sensitive personal, medical, or proprietary data.

Existing defenses against membership inference primarily rely on differential privacy during model training or fine-tuning (Liu et al., 2024; Tsai et al., 2025; Luo et al., 2025). Differential privacy provides a principled way to limit the influence of individual training samples. However, applying it to large diffusion models typically requires additional optimization and privacy-aware fine-tuning. The injected noise and constrained updates may also degrade the model's generation quality, particularly under strong privacy guarantees. Consequently, it remains challenging to mitigate membership leakage while simultaneously avoiding expensive training and preserving generation utility.

In this work, we shed light on an underexplored question of quantization in diffusion models: *Can quantization achieve both model compression and membership privacy mitigation at the same time?* Previous research (Zhang et al., 2025) reveals an important phenomenon: quantization can suppress membership inference signals in discriminative models, and stronger privacy requires lower-bit quantization (i.e., utility loss). However, directly extending this observation to diffusion models is nontrivial for several reasons: (1) Aggressively reducing the precision of the entire model can substantially degrade image generation quality. (2) More importantly, the privacy effect of quantization is not uniform throughout the denoising process. Different diffusion timesteps exhibit different sensitivities to membership inference, suggesting that applying a naive quantization policy across all timesteps is suboptimal for the privacy-utility trade-off.

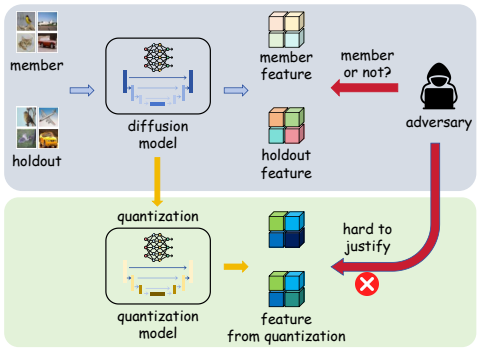


Figure 1: Motivation of PAQ-Diffusion.

This observation motivates a new perspective on diffusion-model quantization: exploring privacy-aware quantization that can maintain privacy and utility, as shown in Fig.1. Although recent work has begun to evaluate the relationship between post-training quantization and membership inference Zhang et al. (2025), privacy leakage has not yet been a primary objective in designing quantization policies for diffusion models. As a result, existing methods do not suffice to solve the optimization problem of quantization for privacy and utility simultaneously.

To address this limitation, we propose a privacy-aware post-training quantization framework for diffusion models (**PAQ-Diffusion**). PAQ-Diffusion is the first method to kill two birds with one stone, improving model efficiency while reducing membership leakage. The method uses a balanced calibration set of matched member and non-member samples and optimizes quantization parameters under joint reconstruction and privacy objectives. During layer-wise quantization, it aligns the member and non-member distributions of local quantization residuals while preserving the generation ability. It then performs timestep-wise post-reconstruction with an independent activation-scale set for each sampling timestep, aligning the corresponding denoising-error distributions independently. The entire procedure operates on a pretrained model and requires no privacy-aware retraining.

The contributions of this work are summarized as follows.

- We propose PAQ-Diffusion, a privacy-aware post-training quantization framework that reduces membership leakage in pretrained diffusion models without privacy-aware retraining.
- We develop a two-stage privacy-aware quantization algorithm targeting structural and temporal dimensions. Layer-wise reconstruction aligns member and non-member quantization residuals, while timestep-wise post-reconstruction independently adapts activation scales to suppress timestep-specific error discrepancies.
- We evaluate PAQ-Diffusion on mainstream diffusion models against several membership inference attacks, where PAQ-Diffusion maintained promising generation quality and reduced privacy leakage.

2 Related Work

2.1 Quantization for Diffusion Models

Existing quantization methods for diffusion models fall broadly into post-training quantization (PTQ) and quantization-aware training (QAT). Among PTQ methods, PTQ4DM constructs timestep-aware calibration data to address the substantial activation-distribution variation across the denoising process (Shang et al., 2023). Q-Diffusion introduces a reconstruction-based quantization framework that accounts for both the temporal characteristics of diffusion sampling and the error accumulation from repeated denoising steps (Li et al., 2023a). PQD further improves post-training quantization by mitigating quantization errors and preserving generation performance under low-precision inference (Ye et al., 2025). For QAT approaches, Q-DM employs quantization-aware optimization to adapt diffusion models to low-bit weights and activations while maintaining denoising capability (Li et al., 2023b), whereas EfficientDM reduces QAT training overhead through efficient quantization-aware fine-tuning (He et al., 2024). Overall, existing methods primarily focus on preserving generation quality and improving storage or inference efficiency, while quantization-policy design rarely considers the privacy risks of quantized diffusion models.

2.2 Membership Inference Attacks on Diffusion Models

Existing membership inference attacks on diffusion models exploit the differences in denoising behavior between training members and non-members. SecMI (Duan et al., 2023) uses timestep-dependent denoising losses as membership signals, based on the observation that training samples generally produce smaller prediction errors. Wu et al. (Wu et al., 2023) investigate quantile regression attacks under different model-access settings. PIA constructs a proximal initialization around the target sample and measures the consistency between the reference and predicted diffusion trajectories, enabling efficient membership inference with fewer model queries (Kong et al., 2024). However, existing attacks primarily target full-precision diffusion models, while membership inference against quantized diffusion models and the influence of numerical precision on privacy leakage remain insufficiently studied.

This work will unite the two research branches into a privacy-aware quantization field.

3 Method

We propose **PAQ-Diffusion**, a privacy-aware post-training quantization framework for diffusion models, as illustrated in Figure 2. The method contains two complementary stages. First, we perform *layer-wise privacy-aware quantization* by aligning the quantization residual distributions of member and non-member samples during weight and activation reconstruction (Section 3.2). Second, we perform *timestep-wise privacy-aware post-reconstruction*, where each diffusion timestep is assigned an independent activation quantization scale so that privacy-oriented adaptation at one timestep does not interfere with the remaining denoising trajectory (Section 3.3).

3.1 Problem Formulation

Diffusion model. Diffusion models are a class of generative models that learn to synthesize data by inverting a gradual noising process. Given a clean sample $\mathbf{x}_0 \sim q(\mathbf{x}_0)$, the forward process progressively perturbs $\mathbf{x}_0$ through a sequence of Gaussian transitions:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}\Big(\sqrt{1-\beta_t}\,\mathbf{x}_{t-1},\, \beta_t \mathbf{I}\Big)\,, \tag{1}$$

where $\{\beta_t\}_{t=1}^{T}$ is a pre-defined variance schedule controlling the noise magnitude at each timestep. The generative process learns to invert this corruption through the reverse transition

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\big(\mu_\theta(x_t, t), \sigma_t^2 I\big)\,, \tag{2}$$

where t is represented by a timestep embedding. An ϵ-prediction network f_θ estimates the injected noise and parameterizes the mean as

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} f_\theta(x_t, t)\right). \tag{3}$$

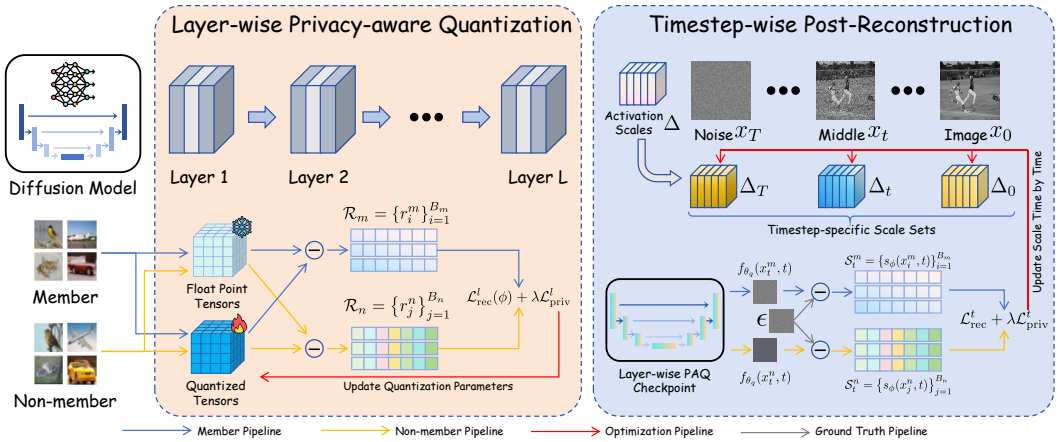


Figure 2: Overview of PAQ-Diffusion. Layer-wise privacy-aware quantization aligns the quantization residual distributions of member and non-member samples during weight and activation reconstruction. Timestep-wise post-reconstruction then independently adapts the activation scales at different denoising timesteps to reduce membership leakage while mitigating cross-timestep interference.

The model is then trained with the denoising objective, which minimizes the denoising error between predicted and true noise:

$$\mathcal{L}_{\mathrm{DM}}(\theta) = \mathbb{E}_{x_0,t,\epsilon}\left[\|\epsilon - f_\theta(x_t, t)\|_2^2\right]. \tag{4}$$

Post-training quantization. Let f_{θ_q} denote the quantized diffusion model. For a tensor u in the model, a b-bit quantizer with scale δ, and integer range $[c_{min}, c_{max}]$ is defined by

$$q_b(u;\delta) = \mathrm{clip}\left(\left\lfloor \frac{u}{\delta} \right\rceil, c_{min}, c_{max}\right), \tag{5}$$

where $\lfloor\cdot\rceil$ denotes rounding to the nearest integer. The scale δ is calibrated from the weight or activation distribution. For weight quantization, fixed nearest rounding can introduce unnecessary reconstruction error. AdaRound (Nagel et al., 2020) instead assigns each weight a learnable rounding variable α and defines a soft-rounded quantized weight as follows:

$$q_b^{\mathrm{soft}}(w;\delta,\alpha) = \mathrm{clip}\left(\left\lfloor \frac{w}{\delta} \right\rfloor + h(\alpha), c_{min}, c_{max}\right), \qquad h(\alpha) = \mathrm{clip}\left(\sigma(\alpha), 0, 1\right), \tag{6}$$

where w means the weight layer, $h(\alpha)$ is the soft rounding function and σ is the sigmoid function. During reconstruction, α is optimized while the weight scale is fixed, and a rounding regularizer encourages $h(\alpha)$ toward either zero or one. After optimization, the soft decision is replaced by $\mathbb{I}[h(\alpha) \geq 1/2]$ to obtain integer-valued quantized weights.

Mitigating membership inference. Reconstruction fidelity alone does not ensure that a quantized model behaves similarly on training members and held-out samples. We assume access to a small privacy calibration set containing known members $\mathcal{D}_m$ and matched non-members $\mathcal{D}_n$. Let $z \in \{m, n\}$ denote membership status, and s be a membership signal, for example, the per-sample denoising error. If this signal follows different distributions for members and non-members, an adversary can use the discrepancy to infer membership. We therefore seek quantization parameters for which

$$P(s \mid z = m) \approx P(s \mid z = n), \tag{7}$$

while the quantized predictions remain close to those of the full-precision model. The resulting problem is to reduce membership distinguishability under fixed weight and activation bit widths without materially degrading denoising fidelity, thereby achieving a privacy-utility trade-off.

3.2 Layer-wise Privacy-Aware Quantization

We construct a balanced member-based calibration set with equal numbers of member and non-member samples. For diffusion models, each minibatch contains equal numbers of member and

non-member samples from the same class. Given a batch size B, we use $B_m = B_n = \frac{B}{2}$. This balanced calibration strategy enables direct comparison of quantization behavior between the two groups.

3.2.1 QUANTIZATION RESIDUAL ALIGNMENT

Consider a quantized model layer with output y_{q} and a floating-point model layer with output y_{fp}, we define the local quantization residual as

$$r = y_{\mathrm{q}} - y_{\mathrm{fp}}. \tag{8}$$

For member and non-member samples, the corresponding residual sets are

$$\mathcal{R}_m = \{r_i^m\}_{i=1}^{B_m}, \qquad \mathcal{R}_n = \left\{r_j^n\right\}_{j=1}^{B_n}. \tag{9}$$

The central idea of layer-wise PAQ is to reduce membership-dependent quantization behavior by aligning the residual distributions of member and non-member samples. To quantify the discrepancy between these distributions, we use squared maximum mean discrepancy (MMD) with a kernel $k(,)$:

$$\begin{aligned}\mathrm{MMD}^2(\mathcal{R}_m, \mathcal{R}_n) = \frac{1}{B_m^2}\sum_{i=1}^{B_m}\sum_{i'=1}^{B_m} k(r_i^m, r_{i'}^m) + \frac{1}{B_n^2}\sum_{j=1}^{B_n}\sum_{j'=1}^{B_n} k(r_j^n, r_{j'}^n) \\ - \frac{2}{B_m B_n}\sum_{i=1}^{B_m}\sum_{j=1}^{B_n} k(r_i^m, r_j^n).\end{aligned} \tag{10}$$

Specifically, the RBF kernel k is constructed from the per-dimension squared distance D. Let τ_ℓ be the adaptive bandwidth of each kernel, K the number of RBF kernels in the mixture, and ℓ the kernel index. The multi-kernel RBF similarity is defined as follows.

$$k(v_i, v_j) = \frac{1}{K}\sum_{\ell=1}^{K}\exp\left(-\frac{D(v_i, v_j)}{\tau_\ell}\right), \qquad D(v_i, v_j) = \frac{\|v_i - v_j\|_2^2}{d}. \tag{11}$$

Substituting Eq. (11) into Eq. (10) compares within-group similarity against cross-group similarity at multiple distance scales. We therefore define the local privacy loss for each layer l as

$$\mathcal{L}_{\mathrm{priv}}^l = \mathrm{MMD}^2\left(\mathcal{R}_m, \mathcal{R}_n\right), \tag{12}$$

which penalizes membership-dependent differences between the two quantization residual distributions while retaining sensitivity to discrepancies across multiple feature channels.

3.2.2 PRIVACY-AWARE LAYER RECONSTRUCTION

For weight and activation reconstruction stages, let ϕ denote the quantization variables optimized at the current stage, where $\phi = \alpha$ for the weight layers and $\phi = \delta$ for the activation layers. We first define a common reconstruction term from the member and non-member residuals:

$$\mathcal{L}_{\mathrm{rec}}^l(\phi) = \frac{1}{2}\left[\mathbb{E}_{x\sim\mathcal{D}_m}\|r^m\|_2^2 + \mathbb{E}_{x\sim\mathcal{D}_n}\|r^n\|_2^2\right]. \tag{13}$$

The trainable variables differ between weight and activation reconstruction. For weight reconstruction, $\phi = \alpha$ are the AdaRound soft-rounding variables as in Eq. (6), and the corresponding regularizer is

$$\mathcal{L}_{\mathrm{round}} = \eta(1 - |2h(\alpha) - 1|^\gamma), \tag{14}$$

where η is the rounding weight and γ is the temperature parameter to control the AdaRound process. For activation reconstruction, the layers vary with the input and diffusion timestep, making persistent element-wise rounding variables impractical. We therefore set $\phi = \delta$ and optimize the shared scales to adapt the quantization range to the activation distribution. As a result, the stage-specific reconstruction term is

$$\mathcal{L}_{\mathrm{rec}}^l(\phi) = \begin{cases}\mathcal{L}_{\mathrm{rec}}^l(\alpha) + \mathcal{L}_{\mathrm{round}}(\alpha), & \phi = \alpha \quad \text{(layer } l \text{ is weight)}, \\ \mathcal{L}_{\mathrm{rec}}^l(\delta), & \phi = \delta \quad \text{(layer } l \text{ is activation)}.\end{cases} \tag{15}$$

Both stages use the same local residual-alignment objective derived above. Consequently, weight and activation reconstruction share the unified layer-wise objective

$$\mathcal{L}^l(\phi) = \mathcal{L}^l_{\text{rec}}(\phi) + \lambda \mathcal{L}^l_{\text{priv}}, \tag{16}$$

where λ is the privacy weight to control the privacy-utility trade-off. During quantization, we first optimize Eq. (16) with respect to α, convert soft rounding to hard rounding, and freeze the quantized weights. We then optimize the same objective with respect to δ. This produces the base privacy-aware quantized model used by the subsequent timestep-specific post-reconstruction stage.

3.3 Timestep-wise Post-Reconstruction

A diffusion model is evaluated repeatedly over a sequence of denoising timesteps. Using a single activation scale δ_l for layer l forces all timesteps to share the same scale: $a^q_l = q_b(a_l; \delta_l)$, where a_l and a^q_l denote the full-precision activation and its quantized representation at layer l, respectively. However, activation distributions and privacy leakage vary across timesteps. A privacy-oriented update obtained from a narrow timestep region can therefore shift a shared activation scale and degrade predictions at unrelated timesteps.

3.3.1 Timestep-specific Scale Set

To avoid such cross-timestep interference, we introduce a timestep-specific activation scale set. Starting from the layer-wise PAQ checkpoint produced by Section 3.2, we create an independent copy of the activation scales for each target timestep:

$$\Delta_t = \{\delta_{1,t}, \delta_{2,t}, \ldots, \delta_{L,t}\}. \tag{17}$$

Each scale set Δ_t is initialized from the same base PAQ activation scales and optimized independently. All other floating-point parameters, quantized weights, and weight quantization parameters remain frozen. Thus, for layer l and timestep t, the quantizer becomes

$$a^q_{l,t} = q_b(a_{l,t}; \delta_{l,t}), \tag{18}$$

where $\delta_{l,t}$ is independently maintained for each selected diffusion timestep.

3.3.2 Privacy-Aware Timestep Reconstruction

For a sample x_0 at timestep t, the quantized denoiser predicts $\hat{\epsilon}_q = f_{\theta_q}(x_t, t)$. We then define a per-sample denoising error as the privacy score

$$s_\phi(x, t) = \frac{1}{d} \left\| f_{\theta_q}(x_t, t) - \epsilon \right\|_2^2, \tag{19}$$

where d is the number of elements in the noise prediction and ϵ is the ground truth noise. For the member and non-member subsets of the minibatch, we obtain a privacy score $\mathcal{S}^m_t = \{s_\phi(x^m_i, t)\}_{i=1}^{B_m}$ and $\mathcal{S}^m_t = \{s_\phi(x^m_i, t)\}_{i=1}^{B_m}$ for member and non-member subsets, respectively. Based on the privacy scores, we compute the timestep-wise privacy discrepancy as follows.

$$\mathcal{L}^t_{\text{priv}} = \text{MMD}^2\left(\mathcal{S}^m_t, \mathcal{S}^n_t\right). \tag{20}$$

To preserve the denoising behavior of the floating-point model, we simultaneously minimize the output reconstruction loss

$$\mathcal{L}^t_{\text{rec}} = \frac{1}{2}\left[\mathbb{E}_{x\sim\mathcal{D}_m} \left\| f_{\theta_q}(x_t, t) - f_\theta(x_t, t) \right\|_2^2 + \mathbb{E}_{x\sim\mathcal{D}_n} \left\| f_{\theta_q}(x_t, t) - f_\theta(x_t, t) \right\|_2^2\right]. \tag{21}$$

Finally, the timestep-specific activation scales Δ_t are optimized using the following loss function

$$\mathcal{L}^t_{\text{post}} = \mathcal{L}^t_{\text{rec}} + \lambda \mathcal{L}^t_{\text{priv}}, \tag{22}$$

where λ is the privacy weight to control the privacy-utility trade-off.

In summary, PAQ-Diffusion mitigates membership leakage at two complementary levels. Layer-wise privacy-aware quantization suppresses membership-dependent quantization artifacts by aligning the residual distributions of member and non-member samples, while timestep-wise post-reconstruction adapts activation scales to the temporal variation in denoising behavior and privacy leakage. Together, these two stages produce a privacy-aware quantized diffusion model without retraining the full-precision model, balancing membership privacy with generation fidelity.

Table 1: Membership inference on DDPM-CIFAR10. Each cell reports AUC / TPR@1%FPR. Lower is better. The best results are shown in **bold**, and the second-best results are underlined.

Model	Loss	PIA	PIAN	SecMI-stat	SecMI-NN
FP	.8490 / 7.47%	.9177 / 24.96%	.8858 / 36.43%	.8864 / 8.00%	.9568 / 42.05%
Q-Diffusion	.8479 / 7.27%	.9164 / 23.10%	.9094 / 26.01%	.8308 / 2.69%	.8140 / 6.93%
PTQ4DM	.8485 / 7.20%	.9177 / 24.62%	.9093 / 31.96%	.8146 / 2.28%	.8285 / 6.41%
TFMQ-DM	.8463 / 7.13%	.9149 / 23.43%	.8861 / 33.84%	.8298 / 4.02%	.9099 / 20.57%
PAQ-Diffusion	**.8153 / 6.80%**	**.8830 / 15.17%**	**.8740 / 15.79%**	**.6417 / 1.15%**	**.6078 / 1.40%**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate PAQ-Diffusion along two axes: generation utility and resistance to membership inference attacks (MIAs). Appendix A.1 provides additional implementation and reproducibility details.

Models and datasets. We use a pixel-space DDPM trained on CIFAR-10 and two text-conditioned latent diffusion models trained on Pokémon and LAION subsets, denoted SD-Pokémon and SD-Laion, respectively. The DDPM experiments use a diffusion horizon of $T = 1000$. Unless stated otherwise, all quantized models use W8A8 weights and activations. We compare PAQ-Diffusion with the full-precision (FP) checkpoint and three representative PTQ methods: Q-Diffusion (Li et al., 2023a), PTQ4DM (Shang et al., 2023), and TFMQ-DM (Huang et al., 2024). Each method starts from the same FP checkpoint for a given dataset.

Quantization and calibration. The reported DDPM model uses $N = 1000$ calibration images and excludes calibration images from the attack evaluation pool. We select the privacy weight $\lambda = 2$ using the privacy-utility analysis in Section 4.5. We use 20,000 optimization iterations for weight reconstruction and 5,000 iterations for activation reconstruction. Calibration covers 100 DDIM sampling positions. The compared PTQ baselines follow their official reconstruction procedures and recommended configurations.

Attacks and Metrics. We evaluate five attacks spanning complementary membership signals. The Loss attack (Hu & Pang, 2023) uses the negative noise-prediction error as its score. PIA and PIAN (Kong et al., 2024) use errors induced by proximal initialization, with and without normalization, respectively. SecMI-stat (Duan et al., 2023) applies a scalar statistic to the reverse-process residual, whereas SecMI-NN trains a neural classifier on the same type of residual feature.

For privacy, we report area under the ROC curve (AUC) and true-positive rate at 1% false-positive rates. We report AUC as $\max(\mathrm{AUC}, 1 - \mathrm{AUC})$ to eliminate the effect of attack-score orientation. Therefore, lower values indicate less membership leakage. For generation utility, we report FID and sFID. DDPM quality metrics use 50,000 generated images for calculation, DDIM sampling with 100 steps and $\eta = 0$, and the full CIFAR-10 training split as the reference distribution.

4.2 PRIVACY PRESERVATION ON DDPM

Table 1 reports AUC and TPR@1%FPR for all five attacks. PAQ-Diffusion consistently yields the lowest AUC. Relative to the strongest utility-oriented PTQ baseline for each attack, PAQ-Diffusion reduces AUC by 3.10 points for the loss attack, 3.19 points for PIA, 1.21 points for PIAN, 17.29 points for SecMI-stat, and 20.62 points for SecMI-NN. The improvement is especially pronounced for SecMI, whose reverse-process features are more closely related to the member/non-member residual gap in our objective target. At the operationally relevant 1% FPR, PAQ-Diffusion also reduces SecMI-NN TPR to 1.40%. The gains across loss-, initialization-, and residual-based attacks show that PAQ-Diffusion does not merely overfit a single attack score.

Table 2: Membership inference on latent diffusion models. Each cell reports AUC / TPR@1%FPR. Lower is better. The best results are shown in **bold**, and the second-best results are underlined.

	Model	Loss	PIA	SecMI-stat	SecMI-NN
SD-Pokémon	FP	.9057 / 16.34%	.9841 / 44.37%	.8646 / 12.76%	.8973 / 11.23%
	Q-Diffusion	.8866 / 12.50%	.9783 / 46.55%	.5328 / .24%	.6295 / 2.72%
	PTQ4DM	.8867 / 12.73%	.9794 / 46.09%	.5571 / .48%	.5799 / 2.65%
	TFMQ-DM	.8856 / 12.50%	.9786 / 35.48%	**.5166 / .00%**	.5863 / 2.01%
	PAQ-Diffusion	**.8556 / 7.82%**	**.9476 / 20.58%**	.5425 / .63%	**.5685 / 1.95%**
SD-Laion	FP	.7519 / 28.08%	.7341 / 23.08%	.6617 / 19.92%	.7874 / 30.27%
	Q-Diffusion	.7512 / 30.52%	.7376 / 22.60%	.7188 / 1.04%	.7129 / 21.97%
	PTQ4DM	.7512 / 28.52%	.7371 / 23.64%	**.5095** / 2.04%	.6542 / 12.13%
	TFMQ-DM	.7513 / 28.72%	.7374 / 23.76%	.5264 / .08%	.6199 / 10.19%
	PAQ-Diffusion	**.7492 / 26.52%**	**.6877 / 18.75%**	.5240 / **.05%**	**.5972 / 6.15%**

4.3 Privacy Preservation on Stable Diffusion

Table 2 evaluates transfer to text-conditioned latent diffusion. On SD-Pokémon, PAQ-Diffusion improves loss, PIA, and SecMI-NN AUC over all baselines, including reducing SecMI-NN AUC from 0.8973 for FP and from 0.5799 for the best conventional PTQ baseline to 0.5685. On SD-Laion, it gives the best loss, PIA, and SecMI-NN results, with SecMI-NN AUC decreasing from 0.7874 for FP to 0.5972. We omit the PIAN attack from Table 2 because it already performs poorly on the full-precision (FP) model, making further evaluation on quantized models less informative. These results demonstrate that the proposed method works for both pixel-space DDPMs and latent diffusion models.

4.4 Generation Quality

Figure 3 and Table 3 evaluate the qualitative and quantitative generation quality, respectively. On DDPM, PAQ-Diffusion achieves the lowest FID of 7.80, improving over the FP model and all utility-oriented PTQ baselines, while its sFID of 5.92 is within 0.01 of the best result. On SD-Pokémon, its FID and sFID remain within the range of the conventional PTQ baselines, although all quantized models exhibit some degradation relative to FP. On SD-Laion, PAQ-Diffusion obtains the lowest FID of 24.71, with a moderate increase in sFID. The qualitative examples in Figure 3 further show that PAQ-Diffusion preserves the principal semantic content and visual appearance of both member and non-member samples across all three datasets. Together, these results indicate that the privacy improvements are achieved without indiscriminately influence the model outputs.

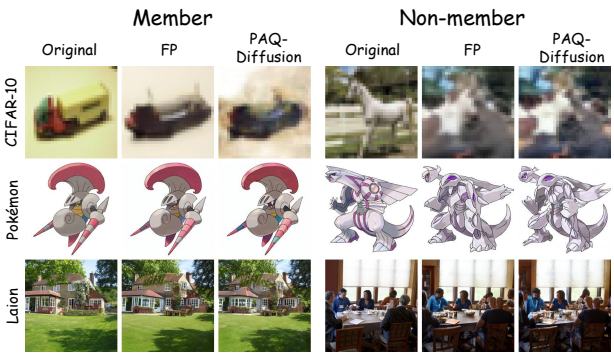


Figure 3: Qualitative comparison on member and non-member samples from CIFAR-10, Pokémon, and LAION. Each group shows the reference image and the corresponding outputs of the FP model and PAQ-Diffusion.

4.5 Ablation Experiments

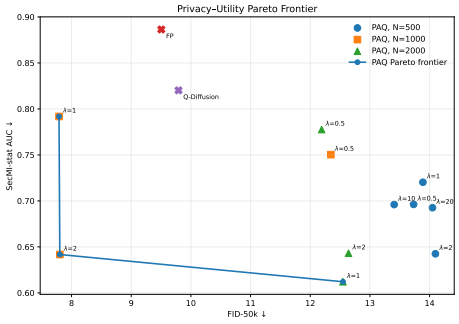
Table 4 evaluates the two stages of PAQ-Diffusion. Relative to Q-Diffusion, layer-wise privacy-aware quantization reduces the SecMI-NN AUC from 0.8140 to 0.7203 and TPR@1%FPR from 6.93% to 2.32%, while decreasing FID from 8.85 to 8.21. Adding timestep-specific post-reconstruction (i.e., complete PAQ-Diffusion) further lowers the AUC and TPR to 0.6078 and 1.40%, respectively, while improving FID to 7.80. These results show that residual alignment provides the

Table 3: Generation quality of DDPM and Stable Diffusion models evaluated using FID and sFID. The best results are shown in **bold**, and the second-best results are underlined.

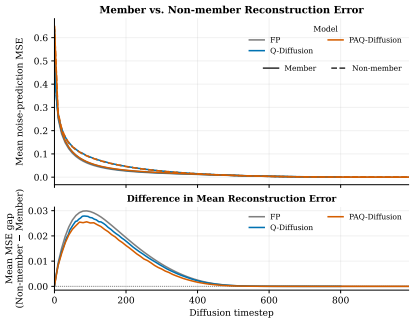
Model	DDPM		SD-Pokémon		SD-Laion	
	FID ↓	sFID ↓	FID ↓	sFID ↓	FID ↓	sFID ↓
FP	9.19	5.93	**49.73**	**191.12**	24.95	17.65
Q-Diffusion	8.85	5.95	53.84	194.79	24.99	18.82
PTQ4DM	9.71	5.95	55.97	196.21	27.63	**18.26**
TFMQ-DM	10.00	**5.91**	52.72	194.35	24.76	18.75
PAQ-Diffusion	**7.80**	5.92	55.25	195.27	**24.71**	18.89

Table 4: Ablation of PAQ-Diffusion on DDPM, with SecMI-NN attack. The best results are shown in **bold**, and the second-best results are underlined.

Model	FID ↓	AUC / TPR@1%FPR↓
FP	9.19	.9568 / 42.05%
Q-Diffusion	8.85	.8140 / 6.93%
Layer-wise	8.21	.7203 / 2.32%
PAQ-Diffusion	**7.80**	**.6078 / 1.40%**



(a) Privacy–utility Pareto frontier for different calibration sizes N and privacy weights λ. Lower FID-50k and SecMI-stat AUC are better.



(b) Member and non-member noise-prediction errors across diffusion timesteps. The lower panel shows the non-member–member error gap.

Figure 4: Privacy–utility trade-off and timestep-wise membership behavior on DDPM-CIFAR10.

primary privacy improvement, and timestep-specific adaptation further strengthens privacy while recovering generation quality.

Calibration and privacy-weight sensitivity. Figure 4a studies the calibration size N and privacy weight λ on DDPM. With $N = 1000$, increasing λ from 0.5 to 1 worsens SecMI-stat AUC from 0.6962 to 0.7918, while $\lambda = 2$ recovers an AUC of 0.6417. This behavior is expected because the privacy loss changes the optimization landscape rather than imposing a direct monotonic constraint on the final attack score. The two most informative operating points expose the desired trade-off. The $N = 2000, \lambda = 1$ configuration gives the strongest SecMI privacy (AUC 0.6121/0.5850) but has FID 12.55. The $N = 1000, \lambda = 2$ configuration achieves nearly the same privacy (0.6417/0.6078) while improving FID to 7.80, which is also better than the FP checkpoint. We therefore use $N = 1000, \lambda = 2$ as the main configuration. These results also show why choosing λ solely by magnitude is unreliable; model selection must jointly validate privacy and generation quality.

Timestep-wise leakage. Figure 4b shows that membership leakage depends strongly on the timestep. For all three models, members have lower mean noise-prediction error than non-members in the early and intermediate trajectory. The gap peaks near $t = 100$ and becomes negligible after roughly $t = 500$. PAQ-Diffusion consistently reduces this gap relative to FP and Q-Diffusion while retaining a similar overall error trajectory. This supports focusing privacy adaptation on specific timesteps rather than perturbing regions where the distributions already overlap.

5 Conclusion

In this work, we presented PAQ-Diffusion, a privacy-aware post-training quantization framework that mitigates membership leakage in pretrained diffusion models. It combines layer-wise quantization with timestep-wise post-reconstruction to balance membership privacy and denoising fidelity. Experiments on different diffusion models demonstrate reduced leakage across multiple attacks while maintaining competitive generation quality.

AI USE STATEMENT

ChatGPT was used as an assistive tool to draft and reorganize portions of the manuscript, polish language, and edit LaTeX. It was not used to generate or alter experimental measurements. The authors reviewed and verified all AI-assisted content and made all scientific decisions.

ETHICS STATEMENT

This work aims to reduce the privacy risks of diffusion models by mitigating membership inference. Our experiments use existing research datasets and pretrained checkpoints and do not involve newly collected human-subject data or user studies. Membership inference techniques are inherently dual-use; we evaluate existing attacks only to measure privacy leakage and defense effectiveness. PAQ-Diffusion does not provide a formal differential-privacy guarantee, and deployments should therefore assess residual leakage under their own threat models. Our study also inherits the data-provenance, licensing, and bias limitations of the underlying datasets and models.

REPRODUCIBILITY STATEMENT

Section 3 specifies the quantization procedure, privacy objectives, and optimized variables. Section 4.1 reports the models, datasets, baselines, bit widths, calibration settings, optimization budgets, sampling protocol, attacks, and evaluation metrics. Appendix A.1 provides additional implementation and reproducibility details. The main results and ablations further report the privacy–utility operating points and the effects of the two optimization stages. Together with the cited baseline implementations, these details are intended to support reproduction of the reported experiments.

REFERENCES

Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX security symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Jinhao Duan, Fei Kong, Shiqi Wang, Xiaoshuang Shi, and Kaidi Xu. Are diffusion models vulnerable to membership inference attacks? In *International Conference on Machine Learning*, pp. 8717–8730. PMLR, 2023.

Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Ptqd: Accurate post-training quantization for diffusion models. *Advances in Neural Information Processing Systems*, 36:13237–13249, 2023.

Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Efficientdm: Efficient quantization-aware fine-tuning of low-bit diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 15731–15750, 2024.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

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

Hailong Hu and Jun Pang. Loss and likelihood based membership inference of diffusion models. In *International Conference on Information Security*, pp. 121–141. Springer, 2023.

Yushi Huang, Ruihao Gong, Jing Liu, Tianlong Chen, and Xianglong Liu. Tfmq-dm: Temporal feature maintenance quantization for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7362–7371, 2024.

Fei Kong, Jinhao Duan, Heng Tao Shen, Xiaoshuang Shi, Xiaofeng Zhu, Kaidi Xu, et al. An efficient membership inference attack for the diffusion model by proximal initialization. In *International Conference on Learning Representations*, volume 2024, pp. 57876–57896, 2024.

Jingwei Li, Jing Dong, Tianxing He, and Jingzhao Zhang. Towards black-box membership inference attack for diffusion models. In *International Conference on Machine Learning*, pp. 34271–34287. PMLR, 2025.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17535–17545, 2023a.

Yanjing Li, Sheng Xu, Xianbin Cao, Xiao Sun, and Baochang Zhang. Q-dm: An efficient low-bit quantized diffusion model. *Advances in neural information processing systems*, 36:76680–76691, 2023b.

Michael F Liu, Saiyue Lyu, Margarita Vinaroz, and Mijung Park. Differentially private latent diffusion models. *Transactions on Machine Learning Research*, 2024.

Zihao Luo, Xilie Xu, Feng Liu, Yun Sing Koh, Di Wang, and Jingfeng Zhang. Privacy-preserving low-rank adaptation against membership inference attacks for latent diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5883–5891, 2025.

Tomoya Matsumoto, Takayuki Miura, and Naoto Yanai. Membership inference attacks against diffusion models. In *2023 IEEE Security and Privacy Workshops (SPW)*, pp. 77–83. IEEE, 2023.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International conference on machine learning*, pp. 7197–7206. PMLR, 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1972–1981, 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Vu Tuan Truong, Luan Ba Dang, and Long Bao Le. Attacks and defenses for generative diffusion models: A comprehensive survey. *ACM Computing Surveys*, 57(8):1–44, 2025.

Yu-Lin Tsai, Yizhe Li, Chia-Mu Yu, Xuebin Ren, Po-Yu Chen, Zekai Chen, and Francois Buet-Golfouse. Differentially private fine-tuning of diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4561–4571, 2025.

Steven Wu, Shuai Tang, Sergul Aydore, Michael Kearns, and Aaron Roth. Membership inference attack on diffusion models via quantile regression. In *NeurIPS 2023 Workshop on Regulatable ML*, 2023.

Jiaojiao Ye, Zhen Wang, and Linnan Jiang. Pqd: post-training quantization for efficient diffusion models. In *Proceedings of the winter conference on applications of computer vision*, pp. 150–156, 2025.

Shengfang Zhai, Huanran Chen, Yinpeng Dong, Jiajun Li, Qingni Shen, Yansong Gao, Hang Su, and Yang Liu. Membership inference on text-to-image diffusion models via conditional likelihood discrepancy. *Advances in Neural Information Processing Systems*, 37:74122–74146, 2024.

Chenxiang Zhang, Tongxi Qu, Zhong Li, Tian Zhang, Jun Pang, and Sjouke Mauw. Bits for privacy: Evaluating post-training quantization via membership inference. In *2025 IEEE 24th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 853–860. IEEE, 2025.

Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. In *The Eleventh International Conference on Learning Representations*, 2023.

Tianchen Zhao, Tongcheng Fang, Enshu Liu, Wan Rui, Widyadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation, 2024.

Xinkai Zhao, Yuta Tokuoka, Junichiro Iwasawa, and Keita Oda. Frequency-calibrated membership inference attacks on medical image diffusion models. In *MICCAI Workshop on Deep Generative Models*, pp. 298–307. Springer, 2025.

A Appendix

A.1 Reproducibility Details

This appendix documents the model configurations, dataset construction, and quantization settings used in our experiments.

A.1.1 Compute and software environment.

We conducted all experiments on a Linux server with 8 NVIDIA L40S GPUs (48 GB each) and two AMD EPYC 9355 32-core CPUs. Each quantization, post-reconstruction, sampling, or attack-evaluation run uses a single GPU. The experiments were validated under Ubuntu 24.04 with Python 3.10.19, PyTorch 2.10.0 (CUDA 12.8), torchvision 0.25.0, diffusers 0.37.0, transformers 5.3.0, and scikit-learn 1.7.2.

A.1.2 Models and Quantization Scope

We evaluate a pixel-space DDPM on CIFAR-10 and two latent diffusion models, SD-Pokémon and Stable Diffusion v1.5 (SD-Laion). The DDPM operates at 32×32 resolution with $1{,}000$ diffusion steps and a linear variance schedule from 2×10^{-4} to 2×10^{-2}. SD-Pokémon and SD-Laion operate at 512×512 resolution, use ϵ prediction, and employ a scaled-linear schedule from 8.5×10^{-4} to 1.2×10^{-2} over $1{,}000$ steps. For the latent diffusion models, we quantize only the UNet; the VAE, CLIP text encoder, tokenizer, and scheduler remain in floating point.

All primary PAQ-Diffusion models use simulated W8A8 quantization. Convolutional and linear modules, including the UNet attention projections, are quantized. Weights are quantized per output channel and activations per tensor. We use AdaRound for weight rounding, while activation scales are learnable. Biases, normalization layers, and unwrapped timestep embeddings remain in floating point. At inference time, we convert the learned AdaRound variables to hard rounding decisions.

A.1.3 Dataset Splits

All attack methods follow the configuration in their original codes. For the dataset, we adjust the splits to test our method.

DDPM-CIFAR10. We use the $50{,}000$-image CIFAR-10 training split. Membership follows the saved SecMI split artifact, which contains $25{,}000$ target-model training members and $25{,}000$ non-members excluded from target training. Both groups index the same torchvision training set.

SD-Pokémon. We use the 833-example `lambdalabs/pokemon-blip-captions` dataset. The persisted dataset contains 416 members and 417 non-members. Images are bilinearly resized, center-cropped to 512×512, converted to tensors, and normalized to $[-1, 1]$; captions are tokenized to 77 CLIP tokens.

SD-Laion. We follow the SecMI-LDM pairing of $2{,}500$ LAION image–caption records as members and $2{,}500$ COCO2017 validation images as non-members. LAION ordering and captions are specified by the captions. We sort COCO image identifiers before applying the held-out positions, and we use the first annotated caption for each selected image. Preprocessing matches SD-Pokémon.

Table 5: Dataset sizes and disjoint calibration and attack splits. Counts are reported as members/non-members.

Model	Source pools	PAQ calibration	SecMI-NN auxiliary	SecMI test	Quality samples
DDPM	25,000/25,000	500/500	5,000/5,000	15,000/15,000	50,000
SD-Pokémon	416/417	100/100	100/100	216/217	5,000
SD-Laion	2,500/2,500	100/100	1,000/1,000	2,400/2,400	5,000

Table 6: Layer-wise PAQ reconstruction settings. Iteration counts are per reconstruction unit.

Setting	DDPM	SD-Pokémon	SD-Laion
Calibration members/non-members	500/500	100/100	100/100
Batch size	32	4	4
Weight reconstruction units	32	282	282
Activation reconstruction units	38	282	282
Weight iterations per unit	2,000	1000	1000
Activation iterations per unit	5,000	500	500
Activation learning rate	4×10^{-4}	4×10^{-4}	4×10^{-4}
Privacy coefficient	2	2	2

We exclude calibration images before constructing the shared SecMI-NN auxiliary and test sets. The auxiliary and test sets are mutually disjoint and are reused across FP, Q-Diffusion, PAQ Diffusion, and all ablations. For both latent diffusion models, we sample calibration records without replacement, and the complementary indices form the attack set; therefore, calibration and attack records have zero overlap.

A.1.4 Privacy-Aware Quantization

Both reconstruction stages use the biased multi-kernel RBF MMD estimator. We flatten the member and non-member samples, then normalize pairwise squared distances by the residual dimension. The detached median of the positive off-diagonal distances in the joined minibatch defines the base bandwidth σ^2, which is clamped to at least 10^{-12}. Four kernels with multipliers $m \in \{0.5, 1, 2, 4\}$ are averaged. The within-group terms include diagonal entries. Layer-wise reconstruction applies MMD to the local residual $y_q - y_{\text{fp}}$, whereas post-reconstruction applies it to the per-sample noise-prediction MSE.

Every reconstruction minibatch contains equal numbers of members and non-members evaluated at a shared diffusion timestep. Weight reconstruction optimizes AdaRound variables with Adam at learning rate 10^{-3}, squared ℓ_2 reconstruction, MMD coefficient 1, and rounding regularization weight 0.01. We disable the rounding regularizer for the first 20% of iterations, after which we anneal its temperature from 20 to 2. Activation reconstruction optimizes the scale parameters with Adam, $\ell_{2.4}$ reconstruction, the same MMD coefficient, and a cosine learning-rate schedule.

DDPM calibration is class-balanced, with 50 members and 50 non-members from each CIFAR-10 class. It spans the DDIM-100 grid $\{0, 10, \ldots, 990\}$, yielding 50,000 logical image–timestep pairs. For the latent diffusion models, each iteration samples a timestep uniformly from the same grid and uses equal member and non-member minibatch halves.

Post-reconstruction starts from the completed layer-wise checkpoint. The objective combines balanced full-precision output reconstruction with MMD on the member/non-member noise-prediction errors. DDPM uses 5,000 iterations, batch size 32, learning rate 4×10^{-5}, and 149 optimized activation scales. SD-Pokémon and SD-Laion each use 5,00 iterations, batch size 4, learning rate 10^{-5}, and 282 optimized activation scales. All three models use Adam with a cosine schedule ending at zero.

For DDPM, the timestep-specific scale set is for 100 DDIM inference timesteps, with a timestep interval of 10. For latent diffusion models, we use all 50 DDIM inference timesteps, with a timestep interval of 20.

A.2 LIMITATIONS.

PAQ-Diffusion has several limitations. Its privacy gains depend on the attack, and some loss-based attacks, such as PIA, still achieve AUC above 0.8 in certain settings. Our experiments use a single random seed, focus on W8A8 quantization, and do not consider adaptive attackers or DP-based baselines. In addition, the method requires access to both member samples and matched non-member samples for calibration, which may limit its applicability when such data are unavailable.