FPAN: Mitigating Replication in Diffusion Models through the Fine-Grained Probabilistic Addition of Noise to Token Embeddings

Jingqi Xu* Chenghao Li* Yuke Zhang Peter A. Beerel
University of Southern California
{jingqixu, cli78217, yukezhan, pabeerel}@usc.edu

Abstract

Diffusion models have demonstrated remarkable potential in generating high-quality images. However, their tendency to replicate training data raises serious privacy concerns, particularly when the training datasets contain sensitive or private information. Existing mitigation strategies primarily focus on reducing image duplication, modifying the cross-attention mechanism, and altering the denoising backbone architecture of diffusion models. Moreover, recent work has shown that adding a consistent small amount of noise to text embeddings can reduce replication to some degree. In this work, we begin by analyzing the impact of adding varying amounts of noise. Based on our analysis, we propose a fine-grained noise injection technique that probabilistically adds a larger amount of noise to token embeddings. We refer to our method as ***Fine-grained Probabilistic Addition of Noise (FPAN)****. Through our extensive experiments, we show that our proposed FPAN can reduce replication by an average of 28.78% compared to the baseline diffusion model without significantly impacting image quality, and outperforms the prior consistent-magnitude-noise-addition approach by 26.51%. Moreover, when combined with other existing mitigation methods, our FPAN approach can further reduce replication by up to 16.82% with similar, if not improved, image quality.*

1. Introduction

Diffusion models [6, 11, 22] have become a dominant paradigm in generative modeling due to their strong capabilities in producing high-quality and diverse images. Compared to traditional approaches like Variational Autoencoders (VAEs) [12] and Generative Adversarial Networks (GANs) [8], diffusion models offer superior fidelity, diversity, and controllability. In particular, text-to-image diffusion models such as DALL·E [20], Stable Diffusion [22], and Imagen [24] excel at generating images that are both semantically aligned with input captions and photorealistically detailed. These models iteratively denoise Gaussian noise based on textual prompts, producing aligned outputs after a fixed number of steps. Despite their success, recent studies [2, 27, 28] have shown that diffusion models are susceptible to memorizing training data, often generating outputs that closely resemble specific training images. This replication raises concerns over copyright infringement and privacy leakage, especially when models are fine-tuned on custom or sensitive small datasets [27].

Prior mitigation strategies [4, 9, 14, 15, 21, 25, 28, 31, 32] that address the replication issue in diffusion models fall into three categories: optimization of the input image or text embeddings during training, modification of the cross-attention module, and architectural changes to the denoising backbone model. In the category of optimization of the input image or text embeddings, Somepalli et al. [28] introduce Random Token Replacement and Addition (RT), which randomly replaces tokens or inserts additional tokens into captions at random positions. Li et al. [14] introduce the Dual Fusion method (DF), which leverages large language models (LLMs) to generalize captions and further mitigates replication by weighted fusing fine-tuning data with data from another source. Modifications of the cross-attention module try to prevent diffusion models from overemphasizing tokens that are likely to lead to replication via masking [21]. Within the category of architectural modifications, Li et al. [15] improve the U-Net [23] architecture by dynamically modifying the skip connections at specific timesteps to limit the impact of the replication-causing direct connections between the upsampling and downsampling blocks.

Prior work shows that perturbing text embedding with a small amount of noise provides a straightforward mitigation strategy [28], as specific captions have been shown to contribute to replication in diffusion models [14, 28]. To enable finer-grained control over noise addition, and to explore the potential of increasing noise intensity for stronger replication mitigation, we studied the impact of a much wider range of noise on token embeddings, and found that

*Equal contribution.

as noise intensity increases, the quality of the generated images first decreases, then improves, and eventually degrades again. Using a modified CLIPScore [17] to measure the degree of model overfitting, we show that this trend can be attributed to the model overfitting, well-fitting, and underfitting the training data, respectively.

Building upon this insight, we propose a **Fine-grained Probabilistic Addition of Noise (FPAN)** to balance the trade-off between generation quality and replication. Our fine-tuning strategy operates at the fine-grained token embedding level and probabilistically injects relatively high-intensity noise into the tokens. Our empirical results demonstrate that our proposed method, in comparison to the baseline model, can significantly reduce replication score while preserving high generated image quality. When integrated with prior replication mitigation techniques [14, 15, 21, 28], our method consistently enhances their performance, highlighting its potential for achieving synergistic improvements in diffusion model training.

We summarize our contributions as follows.1) We observe that as we increase the intensity of injected noise to token embedding the image quality tends to first becoming worse, then better, and then worse again and we show that this trend can be attributed to the model being in the states of overfitting, well-fitting, and underfitting, respectively. 2) We propose our FPAN strategy, which probabilistically injects high-intensity noise into fine-grained token embeddings during training. 3) We present experimental results demonstrating that our strategy provides competitive trade-offs between generation quality and replication. 4) We further show that our method can be effectively integrated with other mitigation techniques to achieve significant synergistic effects in further reducing replication.

2. Background and Related works

2.1. Diffusion Models

Denoising Diffusion Probabilistic Models (DDPMs) [11] employ a forward process that gradually adds Gaussian noise to an image and a reverse process that reconstructs the original image by progressively removing noise. Due to high computational overhead in DDPMs, Latent Diffusion Models (LDMs), such as Stable Diffusion (SD) [22], have been explored. LDMs apply a conditional diffusion process to a compressed latent space transformed by a Variational Autoencoder (VAE) [12].

Fine-tuning a pretrained SD model leverages a dataset consisting of N image-caption pairs, expressed as $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$, where $x^{(i)}$ denotes the i^{th} image, and $y^{(i)}$ represents its corresponding caption. During the forward corruption process, each clean image x is progressively corrupted through the incremental addition of Gaussian noise over T discrete timesteps, resulting in pure Gaussian noise. The noisy representation of an image x at timestep t is denoted by x_t, and the noise introduced at this timestep is represented as ϵ_t. In the backward denoising process, the model M is fine-tuned to estimate the noise added at each timestep t, conditioning on the caption y. The predicted noise is subsequently removed from the noisy input, facilitating a step-wise reconstruction of the original clean image. Formally, the optimization objective during training is expressed as follows:

$$\mathcal{J}(\theta) = \mathbb{E}_{t\in[1,T],\epsilon_t\sim\mathcal{N}(0,I)}\left[\|\epsilon_t - M(x_t, t, e)\|_2^2\right], \quad (1)$$

where e is the text embedding obtained by applying CLIP [19] text encoder to original caption y.

2.2. Replication Score

To evaluate the degree of replication in generated images, we leverage the Replication Score [14, 15, 27, 28], denoted as R. R is defined as the 95th-percentile statistic of the image-level similarity score between the generated images and their nearest matches in the training set.

In other words, a top-1 similarity for every generated image x_{gen} is computed as:

$$Sim_{\text{Top1}}(x_{\text{gen}}) = \max_{x_d\in\mathcal{D}} \textit{sim}\big(\phi(x_{\text{gen}}),\, \phi(x_d)\big), \quad (2)$$

where ϕ extracts image embeddings using SSCD [18] [1], and *sim* is typically dot product. Then $Sim_{\text{Top1}}(x_{\text{gen}})$ among all x_{gen} are collected into a set to compute the replication score R as:

$$R(\mathcal{G}) = \mathcal{Q}_{0.95}\{Sim_{\text{Top1}}(x_{\text{gen}})|x_{\text{gen}} \in \mathcal{G}\}, \quad (3)$$

where $\mathcal{G}$ is the generated image set and $\mathcal{Q}_{0.95}$ means the 95th-percentile value in the set.

The reason why R only focuses on top 5% of generated images by similarity is to ensure the evaluation focuses on replicated samples rather than the entire dataset, which otherwise may be misleading because most generations may be nowhere near direct copies, but a small fraction of generated images can still be very close to a training image. By zooming in on the right-hand tail of similarity score distribution, R captures the worst-case copying behavior. Specifically, a higher R indicates a higher level of replication.

2.3. Mitigation strategies

Prior researches [14, 28] suggest that replication is primarily driven by image duplication and highly specific captions. To address this issue, several mitigation strategies have been proposed, which can be broadly categorized into

[1]Here we use pretrained sscd_disc_large model. It can be found and downloaded from https://github.com/facebookresearch/sscd-copy-detection/tree/main

three classes: model input-based strategies, cross-attention-based strategies, and architectural modifications. Within the category of model input-based strategies, Li et al. [14] proposed a generality score and leveraged a large language model (LLM) [1, 29] to increase caption abstraction. They also introduced a dual fusion technique that merges training images with external image-caption pairs to address duplication. Anti-Memorization Guidance (AMG) [4] employs despecification, deduplication, and dissimilarity guidance to mitigate replication. Multiple Captions(MC) [28] uses BLIP to generate 20 captions per image and randomly samples one during each fine-tuning iteration. Random Caption Replacement (RC) [28] uses random words to replace the caption of an image. Caption Word Repetition(CWR) [28] randomly choose a word from the given caption and insert it into a random location in the caption.

In addition, various cross-attention-based strategies have been proposed. Ren et al. [21] adjusted attention scores to reduce reliance on "trigger tokens" during inference, while chen et al. [3] introduced the Bright Ending (BE) mask to lower dependence on final prompt tokens. Zhang et al. [35] reduce the influence of specific tokens through attention resteering, effectively suppressing the model's reliance on memorized concepts. Hintersdorf et al. [10] proposed identifying and removing neurons in cross-attention layers responsible for replication.

Furthermore, architectural modifications have also been explored. Li et al. [15] proposed RAU-Net, incorporating an Information Transfer Block into U-Net's skip connections to prevent direct transmission of high-resolution information.

Our FPAN offers a mitigation approach without sacrificing generation quality by probabilistically adding appropriate high-intensity noise to fine-grained token embeddings. Because of its orthogonal operational mechanism that ensures non-interference with existing techniques, FPAN can be synergistically combined with prior mitigation strategies to further reduce their replication scores.

3. Fine-Grained Probabilistic Addition of Noise (FPAN)

3.1. Adding noise to token embeddings

Research has shown that specific captions, corresponding to specific text embeddigns, can cause diffusion models to replicate training images [14, 28]. Since noise can reduce caption specificity [13], one straightforward approach to mitigate this issue is to inject noise into the text embeddings. In FPAN, we propose to inject noise at the token embedding level to enable finer-grained control over the semantic perturbation.

Let $\psi = \{\tau_i\}_{i=1}^{L}$ denote a text embedding consisting of L token embeddings, where each $\tau_i \in \mathbb{R}^{1\times d}$ represents

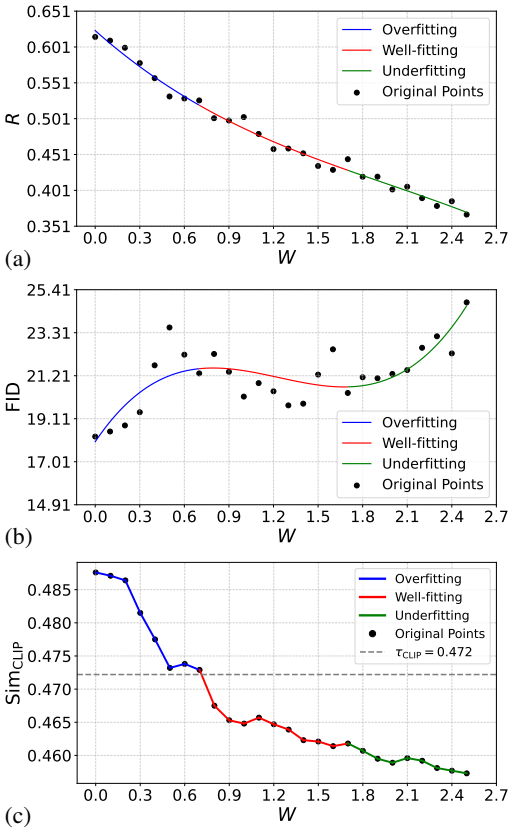


Figure 1. Results under different noise intensities. (a) R, (b) FID, (c) Sim_{CLIP}. Three stages are shown as, overfitting stage (blue) when $W \leq 0.7$; well-fitting stage (red) when $0.7 < W \leq 1.7$; underfitting stage (green)when $W > 1.7$.

the i-th token embedding in a d-dimensional space. The following noise injection process are applied during each training iteration,

$$\tau_i' = \tau_i + \boldsymbol{\xi}_i,\ \boldsymbol{\xi}_i \sim W \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{4}$$

where $\boldsymbol{\xi}_i$ denotes a Gaussian noise, and $W \in \mathbb{R}_+$ controlling the noise intensity. Let $\psi' = \{\tau_i'\}_{i=1}^{L}$ denote the noisy text embedding, where $\tau_i' \in \mathbb{R}^{1\times d}$ represents the noisy embedding of the i-th token.

3.2. Analyzing impact of token noise intensity

An important question that arises is how to determine the appropriate intensity W of the added noise. Intuitively, adding noise with excessively high intensity may effectively distort the captions that are specific but can also degrade the sematic information in the caption and result in poor quality of generated images. Conversely, extremely low intensity noise may fail to effectively reduce the caption specificity and mitigate replication.

This section explores the impact of noise intensity by conducting experiments and comparing both replication and

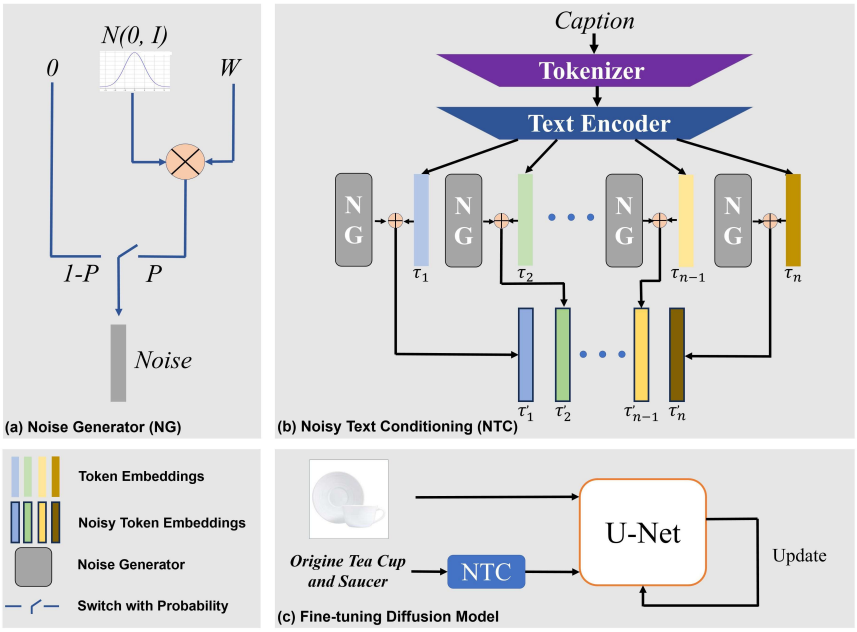


Figure 2. Overview of our proposed framework. (a) Probabilistic selection of noise. (b) Generation of noisy token embeddings. (c) Diffusion model fine-tuning process including noisy token embedding.

image quality across different values of W. Our experiments are based on Stable Diffusion 2.1 [22]. We fine-tune this diffusion model on a random subset of 10,000 samples from the LAION-2B dataset [26], incorporating noise addition to the text embeddings as described in Equation 4. We vary W from 0 to 2.5 in steps of 0.1. To evaluate performance, we use the replication score R [14, 15, 27, 28] to quantify the replication, and the Fréchet Inception Distance (FID) [16] to assess the fidelity and diversity of the generated images. Additional details on the experimental setup are provided in Section 4.1.

Our exprimental results are presented in Figure 1, where (a) illustrates how R varies with the intensity of the injected noise by applying third-order polynomial fitting to the experimental data points. As W increases, the replication score R monotonically decreases. This is consistent with prior work [28] which argues that replication in text-to-image diffusion models is largely due to the overlearning of overly detailed semantic information in text embeddings and their associations with training images. As W increases, the level of semantic information contained in token embeddings is gradually degraded, making it more difficult to learn the relationship between text conditioning and their associated images, leading to a gradual reduction in replication. As a result, R exhibits a decreasing trend as W increases.

The behavior of FID is less obvious as it shows a nonlinear N-shaped increase–decrease–increase pattern as W increases. As shown in Figure 1 (b), we hypothesize that as W increases, the model falls into one of three-stage, overfitting, well-fitting and finally underfitting. As W increases in the overfitting stage, we see FID increases reflecting a reduction in overfitting, as adding noise to training data acts as a form of regularization that mitigates overfitting [7]. In the well-fitting stage, as W increases, FID begins to decline, a point that potentially be attributed to fully overcoming the overfitting stage and the ability to start to generalize. As W increases further, FID begins to rise again, which may be attributed to the model entering a stage of underfitting the training data.

To justify this hypothesis, we leverage a metric proposed in [17], which we denote as Sim_{CLIP}. This metric measures the average CLIP [19] embedding similarity between the generated image set and training image set. Specifically, for each generated image, we compute the average cosine similarity between its embedding and the embeddings of all images in the training dataset. The final Sim_{CLIP} is then obtained by averaging these values across all generated images [17]. Sim_{CLIP} serves as an indicator of overfitting, as higher values suggest that the generated images closely resemble those in the training set. In particular, we define a threshold τ_{CLIP} to distinguish between well-generalized and overfitted models that is based on large pretrained models that are generally trained on diverse and extensive datasets and thus unlikely to overfit our small fine-tuning dataset [34]. In particular, we generate a reference image set with the pretrained model and our fine-tuning prompts and set the threshold τ_{CLIP} to the Sim_{CLIP} measured between the ref-

erence and fine-tuning image sets. Fine-tuned models with Sim_{CLIP} above τ_{CLIP} are flagged as potentially overfitting.

In Figure 1 (c), we present the Sim_{CLIP} score for diffusion models fine-tuned with different noise intensities and compare them to $\tau_{CLIP} = 0.472$ measured with Stable Diffusion 2.1 [22]. In the overfitting stage, we can see the Sim_{CLIP} score is always larger than τ_{CLIP} and gradually decreases as W increases, supporting our hypothesis. In the well-fitting stage, when W increases beyond 0.7, the Sim_{CLIP} score drops below τ_{CLIP} and we assert our fine-tuned model no longer is overfitting. In the underfitting stage, excessive noise severely disrupts the semantic content of the captions, leading the model to generate significantly degraded images.

3.3. Probabilistic addition of noise

In Section 3.2, we show that the maximum noise intensity associated with the well-fitting stage will maximally mitigate replication without excessively degrading generation quality. However, compared to the case with no noise, the FID score can still be somewhat degraded. To compensate for the FID score increase and improve the replication-FID trade-off, we propose a fine-tuning process with Fine-grained Probabilistic Addition of Noise (FPAN) strategy. In particular, our method **probabilistically** adds this maximal-intensity noise to each token embedding during each fine-tuning iteration. More precisely, instead of consistently adding high-intensity noise to all token embeddings, our probabilistic mechanism defines a probability factor P, which controls the probability of whether we inject high-intensity noise into a specific token embedding or not. The intensity W of the injected noise is set to the maximum noise associated with the well-fitting stage because of its associated low R and relative local minimum FID, as shown in Figures 1 (a) and (b). An overview of the proposed method is given in Figure 2.

More formally, let $\psi = \{\tau_i\}_{i=1}^{L}$ denote a text embedding consisting of L token embeddings, where each $\tau_i \in \mathbb{R}^{1\times d}$ represents the i-th token embedding in a d-dimensional space. During each fine-tuning iteration, we independently sample a noise term $\boldsymbol{\xi}_i \in \mathbb{R}^{1\times d}$ for each token embedding from the following distribution:

$$\boldsymbol{\xi}_i \sim z_i \cdot \mathcal{N}(\mathbf{0}, W^2\mathbf{I}), z_i \sim \text{Bernoulli}(P), \tag{5}$$

where z_i is a Bernoulli random variable that equals to 1 with probability P. Therefore, $\boldsymbol{\xi}_i$ is sampled from $\mathcal{N}(\mathbf{0}, W^2\mathbf{I})$ with probability P, and is set to zero with probability $1-P$. The noise $\boldsymbol{\xi}_i$ is then injected into the corresponding token embedding as

$$\tau_i' = \tau_i + \boldsymbol{\xi}_i, \tag{6}$$

where $\tau_i' \in \mathbb{R}^{1\times d}$ denotes the i-th noisy token embedding. All τ_i' are aggregated into a noisy text embedding $\psi' = \{\tau_i'\}_{i=1}^{L}$, which is used as the conditioning input for fine-tuning the diffusion model during the current fine-tuning iteration. More detailed pseudocode for this approach is presented in Algorithm 1.

Algorithm 1 Fine-tuning with FPAN

Input: Pre-trained model M; Fine-tuning dataset $\mathcal{D}$; Total timesteps T; Number of iterations N; Image Encoder $\mathcal{E}(\cdot)$; Text Encoder CLIP$(\cdot)$; Noise weights W; Probability P.
Output: Fine-tuned model M.
M.train()
iter $\leftarrow 0$
while iter $< N$ **do**
  for each batch $\{(x, y)\} \in \mathcal{D}$ **do**
    Image Encoding $I \leftarrow \mathcal{E}(x)$
    Text Encoding $\psi \leftarrow$ CLIP(y)
    for each token embedding $\tau_i \in \psi$ **do**
      Generate noise distribution:
      $\Delta \leftarrow \mathcal{N}(0, W^2\mathbf{I})$
      Sample Bernoulli variable:
      $z_i \sim$ Bernoulli(P)
      Sample noise: $\boldsymbol{\xi}_i \leftarrow z_i \cdot \Delta$
      Add noise: $\tau_i' \leftarrow \tau_i + \boldsymbol{\xi}_i$
    end for
    Noisy text embedding: $\psi' \leftarrow \{\tau_i'\}$
    Sample $t \sim$ Uniform$(0, T)$
    Update the model: $M \leftarrow$ update(M, I, ψ', t)
    iter $\leftarrow$ iter $+ 1$
  end for
end while
return M

4. Experimental Mitigation Results

In this section, we perform a systematic tuning of the hyperparameters for our proposed method and provide analysis. Furthermore, to evaluate the effectiveness of our approach, we present empirical evidence demonstrating its ability to mitigate replication, both as a standalone strategy and in combination with existing state-of-the-art methods.

4.1. Experimental Setup

Model Selection and Dataset We build upon Stable Diffusion 2.1 [22], an advanced text-to-image diffusion model pretrained on the complete LAION dataset [26]. We fine-tune the model using a random subset of 10,000 samples from the LAION-2B dataset [26]. Each sample includes an image paired with a descriptive caption, thereby capturing a wide variety of visual and textual content. The unmodified fine-tuned model serves as our baseline. Our study specifically targets adjustments to the noise addition strategy on text token embeddings during fine-tuning, leaving

$W = 1.5$											
P	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
R↓	0.615	0.559	0.538	0.512	0.491	0.477	0.474	0.461	0.457	0.451	0.434
FID↓	18.24	17.37	16.43	17.45	17.77	18.15	18.91	19.87	18.22	18.83	21.26
$W = 1.6$											
P	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
R↓	0.615	0.559	0.525	0.488	0.475	0.469	0.457	0.450	0.452	0.440	0.429
FID↓	18.24	17.26	16.35	17.07	17.21	17.44	17.83	20.44	19.31	22.27	22.49
$W = 1.7$											
P	0	0.1	0.2	**0.3$_{(A_1)}$**	**0.4$_{(A_2)}$**	**0.5$_{(A_3)}$**	**0.6$_{(A_4)}$**	0.7	0.8	0.9	1.0
R↓	0.615	0.557	0.515	**0.491**	**0.452**	**0.452**	**0.438**	0.450	0.446	0.430	0.444
FID↓	18.24	16.29	16.66	**15.89**	**17.81**	**17.17**	**17.96**	18.17	18.63	20.71	20.36

Table 1. Replication score and FID under different values of P for three configurations: $W = 1.5$, $W = 1.6$, and $W = 1.7$. For $W = 1.7$, the values of $P = 0.3$, 0.4, 0.5, and 0.6 correspond to points A_1, A_2, A_3, and A_4 in Figure 3, respectively.

the model architecture intact.

Fine-tuning process During fine-tuning, all components except the U-Net remain frozen. We adhered to the fine-tuning configuration detailed in [27, 28], running the process for 100,000 steps with a learning rate of 5×10^{-6} and incorporating a warm-up phase over the first 5,000 steps. The diffusion process is executed with $T = 1000$ timesteps. For evaluation, we generate 10,000 images during the inference process using 50 inference steps, with prompts identical to those used in the fine-tuning set. More implementation details are provided in Appendix 4.1.

Evaluation Metrics Our evaluation framework utilizes three key metrics. (1) Replication score (R) [14, 15, 27, 28], which quantifies the degree of replication; (2) Frechet Inception Distance (FID) [16], which assesses the fidelity and diversity of the generated images; and (3) the R-FID curve [15], which illustrates the trade-off between replication and generation quality created by varying a shared hyperparameter across different methods. A second-order polynomial function is then fitted to the (R, FID) pairs to form a continuous trade-off curve. Curves that lie closer to the origin reflect more favorable trade-offs.

4.2. Hyperparameters Tuning

Based on the observations from Figure 1, we consider three noise weight intensities: $W = 1.5$, $W = 1.6$, and $W = 1.7$, and vary the probability P uniformly over the interval $[0, 1.0]$ with a step size of 0.1. To assess the effectiveness of different high-intensity noise levels, we leverage R-FID curves obtained by varying the probability parameter P for all three values of W. Interestingly, we observed an interesting shift: as P decreases from 1 to approximately 0.2, the FID score decreases, but then rises sharply below this point. A detailed analysis of this behavior is provided in Appendix A.2. For a clearer visual comparison of the R-FID curves, Figure 3 presents curves fitted using results with $0.2 < P \leq 1$.

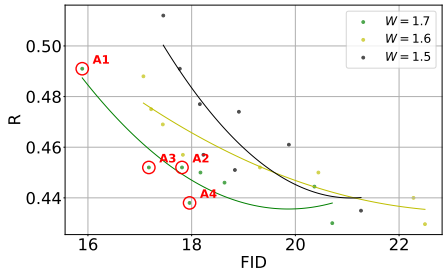


Figure 3. R-FID curves corresponding to different values of W, including four favorable points corresponding to $W = 1.7$. $A_1(15.89, 0.491)$ when $P = 0.3$, $A_2(17.81, 0.452)$ when $P = 0.4$, $A_3(17.17, 0.452)$ when $P = 0.5$, and $A_4(17.96, 0.438)$ when $P = 0.6$.

The results show that the R-FID curve corresponding to $W = 1.7$ lies closest to the origin, indicating this value leads to the most favorable trade-off between image quality and replication mitigation. In Table 1, we present the detailed results for all values of P under different settings of W. It can be seen that when $0.3 \leq P \leq 0.6$ under $W = 1.7$, replication is effectively mitigated while the corresponding FID scores remain lower than that of the baseline model. Specifically, the values $P = 0.3$, $P = 0.4$, $P = 0.5$, and $P = 0.6$ correspond to points A_1, A_2, A_3, and A_4 in Figure 3, respectively.

4.3. Standalone Performance

We employ our mitigation strategy with $W = 1.7$ under probability settings $P \in \{0.3, 0.4, 0.5, 0.6\}$, and independently evaluate its effectiveness against a range of exist-

	Baseline	GN [28]	MC [28]	RC [28]	CWR [28]	LD [15]	DF [14]	TMAA [21]	FPAN(Ours) (P = 0.3 / 0.4 / 0.5 / 0.6)
$R\downarrow$	0.615	0.596	0.420	0.565	0.614	0.378	0.412	0.309	0.491 / 0.452 / 0.452 / 0.438
FID $\downarrow$	18.24	19.50	16.83	15.98	16.73	19.17	17.47	38.18	15.89 / 17.81 / 17.17 / 17.96

Table 2. Comparison of FPAN with prior works.

ing approaches. These include Gaussian Noise (GN) [28], Multiple Captions (MC) [28], Random Caption Replacement (RC) [28], Caption Word Repetition (CWR) [28], Loyal Diffusion (LD) [15], Dual Fusion (DF) [14] and an inference-time method involving token masking and attention score adjustment (TMAA) [21]. The corresponding results are summarized in Table 2.

Compared to the baseline, our method reduces the replication score R by up to 28.78%, while improving the FID score by up to 2.35. Compared to the GN, RC, and CWR methods, our approach achieves maximum reductions in R by 26.51%, 22.47%, and 28.66%, respectively, without incurring significant degradation in the FID score. Moreover, compared to MC, LD, DF, and TMAA, our method yields improvements in FID by up to 0.94, 3.28, 1.58, and 22.29, respectively, while maintaining good R scores.

The above findings suggest that our method outperforms most other methods, when deployed as a standalone strategy, offers similar if not improved trade-off between memorization mitigation and generative quality over existing approaches.

4.4. Synergy with Prior Art

To further demonstrate the benefits of our method, we investigate its synergistic effects when integrated with prior approaches that target replication mitigation from different perspectives. Specifically, we combine our strategy with four representative methods: Multiple Captions (MC) [28], Dual Fusion (DF) [14], LoyalDiffusion (LD) [15], and Token Masking and Attention score Adjustment (TMAA) [21]. The achieved best experimental results with corresponding P in our method are shown in Table 3. When combined with prior methods, our approach yields up to a 16.82% reduction in R. In fact, our approach, when combined with LD yields the lowest $R = 0.357$ across all prior methods tested that have a reasonable FID of less than 20. The only lower achieved R is for TMAA with FPAN, where we get an $R = 0.257$ but at the cost of a much higher $FID = 36.93$.

We may also note that the results show that in most cases, the addition of FPAN not only yields improvements in R but also improvements in FID. The exception is MC with FPAN which shows a small increase in FID compared to MC alone. This may be attributed to the fact that the MC's multiple captions may already have sufficient randomness in the semantic information and adding more randomness using FPAN may be leading to some semantic degradation. However, the FID of MC with FPAN is still similar to baseline's FID, and thus it maintains high generation quality.

Method	w/FPAN	R↓	FID↓
Baseline	✗	0.615	18.24
	✓ (P=0.6)	0.438	17.96
MC [28]	✗	0.420	16.83
	✓ (P=0.4)	0.378	18.45
LD [15]	✗	0.378	19.17
	✓ (P=0.6)	0.357	19.78
TMAA [21]	✗	0.309	38.18
	✓ (P=0.6)	0.257	36.93
DF [14]	✗	0.412	17.47
	✓ (P=0.6)	0.371	17.58

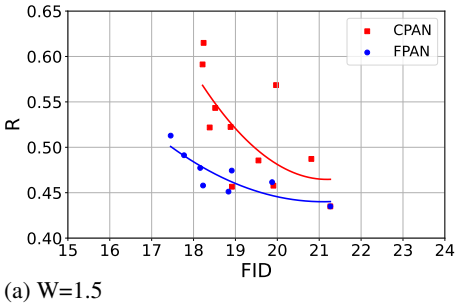
Table 3. The impact of combining FPAN with prior works. ✗ represents that the method is perform standalone and ✓ means FPAN is also used with the method.

5. Ablation Studies

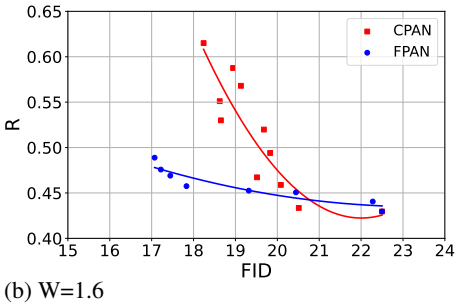
5.1. Fine-Grained vs. Coarse-Grained

To justify the effectiveness of targeting fine-grained token embeddings, we compare FPAN against a coarser grained probabilistic addition of noise we refer to as CPAN. In particular, while adopting the same sampling scheme for the noise term as described in Equation 5, CPAN applies the noise term to the entire text embedding rather than to individual token embeddings. More specifically, in FPAN, each token has an independent probability of being perturbed by noise or left unchanged. However, in CPAN, if a text embedding is selected to be perturbed, all token embeddings within this text embedding are injected with noise. Otherwise, no token embedding within the text embedding will have injected noise.

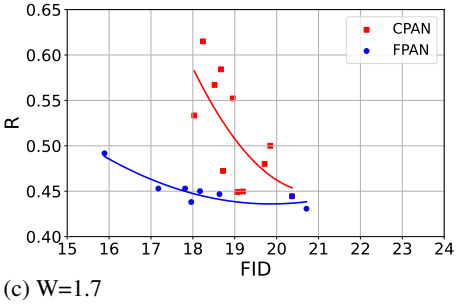
To compare the two approaches, we adopt the same hyperparameter settings as described in Section 4.2. For each



(a) W=1.5



(b) W=1.6



(c) W=1.7

Figure 4. Comparison of R-FID curves between FPAN and CPAN for various values of W.

possible value of W, we observe R-FID curves by varying probability P for both FPAN and CPAN. The results are presented in Figure 4. Across all values of W, FPAN consistently produces R-FID curves that lie closer to the origin, indicating FPAN is more effective in mitigating R while preserving FID.

We attribute the superior performance of our proposed FPAN approach to its more fine-grained and flexible noise injection strategy. Unlike CPAN, which often perturbs all token embeddings that can severely degrades semantic content, FPAN injects noise into individual token embeddings. This fine-grained process increases the chance of targeting specific tokens most responsible for replication, while leaving others intact. In principle, precisely controlling noise injection for specific tokens could offer even greater benefits, but doing so would require substantial computational overhead due to the need for token-level importance estimation or gradient-based analysis.

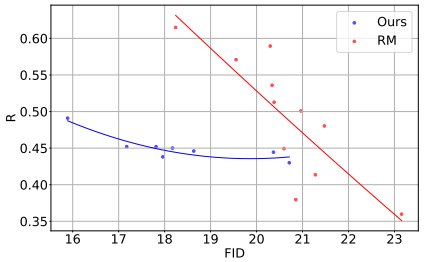


Figure 5. Comparison of R-FID curves between FPAN and RM.

5.2. Random Masking vs. FPAN

We refer to the approach in which each token embedding in a text embedding is masked with a probability Q during fine-tuning as the Random Masking (RM) strategy. We compare it with our FPAN approach because both of them aim to despecify text conditioning for diffusion models.

We obtain R-FID curves by varying the probability P and Q for both approaches, as shown in Figure 5. Since RM does not exhibit the sudden shift phenomenon, its curve is fitted using results from $0 \leq Q \leq 1$. For a clearer visual comparison, the curve corresponding to FPAN is fitted using the same P settings as described in Section 4.2, under the optimal hyperparameter setting of $W = 1.7$. The R-FID curve of FPAN lies closer to the origin and, in particular, shows FPAN outperforms RM in lower FID situations. In addition, more detailed analysis is given in Appendix A.3

6. Conclusions and Future Work

In this work, we propose Fine-grained Probabilistic Addition of Noise (FPAN), a fine-tuning strategy designed to mitigate replication in diffusion models while maintaining generation quality. Our method probabilistically adds high-intensity noise to fine-grained token embeddings during each fine-tuning iteration. The choice of appropriate high-intensity noise is determined by our finding on how different amount of noise affect replication and generation quality. Through extensive experiments, FPAN demonstrates a significant reduction in replication compared to baseline models and prior works, without compromising the FID score. Moreover, FPAN can be combined with recent mitigation methods to produce synergistic effects, further enhancing their performance.

While FPAN demonstrates strong performance, there are limitations to its current design. Specifically, both the noise intensity W and the probability parameter P are fixed throughout the training process. However, the optimal values of W and P may vary across different stages of training. Future research could explore dynamically adjusting W and P as training progresses, allowing more precise control over noise perturbation and further improving the trade-off between generation quality and replication mitigation.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3

[2] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023. 1

[3] Chen Chen, Daochang Liu, Mubarak Shah, and Chang Xu. Exploring local memorization in diffusion models via bright ending attention. *arXiv preprint arXiv:2410.21665*, 2024. 3

[4] Chen Chen, Daochang Liu, and Chang Xu. Towards memorization-free diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8425–8434, 2024. 1, 3

[5] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020. 1

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1

[7] Oussama Dhifallah and Yue Lu. On the inherent regularization effects of noise injection during training. In *International Conference on Machine Learning*, pages 2665–2675. PMLR, 2021. 4

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 1

[9] Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *arXiv preprint arXiv:2310.02664*, 2023. 1

[10] Dominik Hintersdorf, Lukas Struppek, Kristian Kersting, Adam Dziedzic, and Franziska Boenisch. Finding NeMo: Localizing neurons responsible for memorization in diffusion models. *Advances in Neural Information Processing Systems*, 37:88236–88278, 2024. 3

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2

[12] Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019. 1, 2

[13] Seanie Lee, Minki Kang, Juho Lee, and Sung Ju Hwang. Learning to perturb word embeddings for out-of-distribution QA. *arXiv preprint arXiv:2105.02692*, 2021. 3, 1

[14] Chenghao Li, Dake Chen, Yuke Zhang, and Peter A Beerel. Mitigate replication and copying in diffusion models with generalized caption and dual fusion enhancement. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7230–7234. IEEE, 2024. 1, 2, 3, 4, 6, 7

[15] Chenghao Li, Yuke Zhang, Dake Chen, Jingqi Xu, and Peter A Beerel. LoyalDiffusion: A diffusion model guarding against data replication. *arXiv preprint arXiv:2412.01118*, 2024. 1, 2, 3, 4, 6, 7

[16] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? A large-scale study. *Advances in neural information processing systems*, 31, 2018. 4, 6

[17] Rubén Pascual, Adrián Maiza, Mikel Sesma-Sara, Daniel Paternain, and Mikel Galar. Enhancing DreamBooth with LoRA for generating unlimited characters with Stable Diffusion. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2024. 2, 4

[18] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14532–14542, 2022. 2

[19] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 2, 4

[20] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021. 1

[21] Jie Ren, Yaxin Li, Shenglai Zeng, Han Xu, Lingjuan Lyu, Yue Xing, and Jiliang Tang. Unveiling and mitigating memorization in text-to-image diffusion models through cross attention. In *European Conference on Computer Vision*, pages 340–356. Springer, 2024. 1, 2, 3, 7

[22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 4, 5

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 1

[24] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 1

[25] Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22522–22531, 2023. 1

[26] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo

Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. LAION-5B: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022. 4, 5

[27] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? Investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6048–6058, 2023. 1, 2, 4, 6

[28] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023. 1, 2, 3, 4, 6, 7

[29] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[30] Jean Ure. Lexical density and register differentiation. *Applications of linguistics*, 23(7):443–452, 1971. 1

[31] Ryan Webster, Julien Rabin, Loic Simon, and Frederic Jurie. On the de-duplication of LAION-2B. *arXiv preprint arXiv:2303.12733*, 2023. 1

[32] Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 1

[33] Haike Xu, Zongyu Lin, Jing Zhou, Yanan Zheng, and Zhilin Yang. A universal discriminator for zero-shot generalization. *arXiv preprint arXiv:2211.08099*, 2022. 1

[34] Weili Zeng, Yichao Yan, Qi Zhu, Zhuo Chen, Pengzhi Chu, Weiming Zhao, and Xiaokang Yang. Infusion: Preventing customized text-to-image diffusion from overfitting. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3568–3577, 2024. 4

[35] Gong Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Forget-Me-Not: Learning to forget in text-to-image diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1755–1764, 2024. 3

FPAN: Mitigating Replication in Diffusion Models through the Fine-Grained Probabilistic Addition of Noise to Token Embeddings

Supplementary Material

A. Appendix

A.1. Additional Experimental Setup Details

All experiments are conducted using NVIDIA A100 Tensor Core GPUs equipped with 40 *GB* of memory, supporting both the training and inference process. During training, we set the batch size to 16 and fix the image resolution to 256. Optimization is performed using the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, along with a weight decay factor of $1e^{-2}$. For the inference process, we generate samples using $S = 50$ steps, uniformly spacing across the full diffusion process.

A.2. Analysis of the Sudden Shift in FID

In this section, we provide a detailed analysis of the nonlinear trend observed in the FID score as the probability parameter P decreases. In Figure 6, we present third-order polynomially fitted curves based on the FID scores obtained across different W for $0 \leq P \leq 1$. We observe that the FID score steadily decreases as P decreases from 1 to approximately 0.2, but begins to rise abruptly below this point. We provide a possible explanation for this phenomenon.

According to [30], approximately 40% of the words in a sentence contribute significantly to its overall semantics, while the remaining 60% are function words that carry relatively little semantic content. When $P = 1$, all token embeddings that corresponding to high-information words are perturbed with strong noise, substantially disrupting the semantic integrity of the text embedding. Consequently, the quality of generated images is severely degraded, resulting in high FID scores. As P decreases, the proportion of high-information token embeddings subjected to strong noise gradually decreases. This leads to a progressive restoration of the global semantics encoded in the text embedding, thereby improving the generation quality and reducing the FID score. Furthermore, Clark et al. [5] shows that when less than approximately 20% of the words in a sentence are perturbed or replaced, a semantic classifier is still capable of identifying the key semantic content, indicating that the overall semantics are preserved. Building on this insight, when P decreases to approximately 0.2, the proportion of token embeddings affected by strong noise reduces to around 20%. This may explain the plateau observed in FID reduction: at this point, the original semantic information in the text embedding is almost fully recovered, and further decreases in P may no longer contribute to generation quality improvement. However, as P continues to decrease below 0.2, the number of function word embeddings subjected to strong noise continues to decrease. Although function words individually encode limited semantic content, perturbing them introduces stochasticity that acts similarly to data augmentation [13], thereby enhancing the model's generalization ability. We attribute the sudden increase in FID scores to the diminishing effectiveness of this regularization effect, which in turn may weaken the model's generalization capacity and lead to a decline in image generation quality.

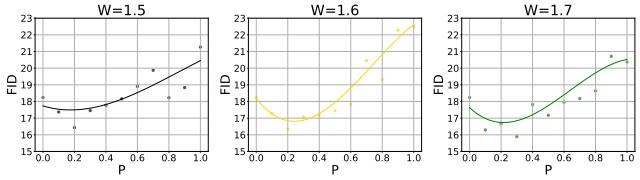


Figure 6. FID curve of FPAN as we vary P for three different values of $W = 1.5, 1.6, 1.7$.

A.3. Comparison between FPAN and RM

In Section 5.2, we empirically demonstrated that FPAN achieves a more favorable balance between generation quality and replication compared to RM. In this section, we provide a theoretical interpretation of the differences between the two approaches.

We compare the impact of FPAN and RM on the distribution of token embeddings. We assume that each token embedding, denoted as τ, is sampled from a distribution with mean μ_τ and variance σ_τ^2.

For FPAN, we denote the token embedding after adding noise $\boldsymbol{\xi}$ as τ', where $\boldsymbol{\xi}$ is independent of τ and is sampled from the distribution described in Equation 5. The resulting τ' follows a distribution with mean μ'_τ and variance σ'^2_τ, both of which can be derived as follows,

$$\mu'_\tau = \mathbb{E}[\tau + \boldsymbol{\xi}] = \mathbb{E}[\tau] + \mathbb{E}[\boldsymbol{\xi}] = \mu_\tau + 0 = \mu_\tau \tag{7}$$

$$\begin{aligned} \sigma'^2_\tau &= \text{Var}(\tau + \boldsymbol{\xi}) = \text{Var}(\tau) + \text{Var}(\boldsymbol{\xi}) \\ &= \sigma_\tau^2 + P \cdot W^2 \end{aligned} \tag{8}$$

From the derivation above, it is evident that our method preserves the original mean of the token embedding distribution. According to [33], the mean of token embeddings serves as a strong representation of the overall sentence semantics. Thus, our method effectively retains the semantic content of the caption, which is crucial for maintaining the

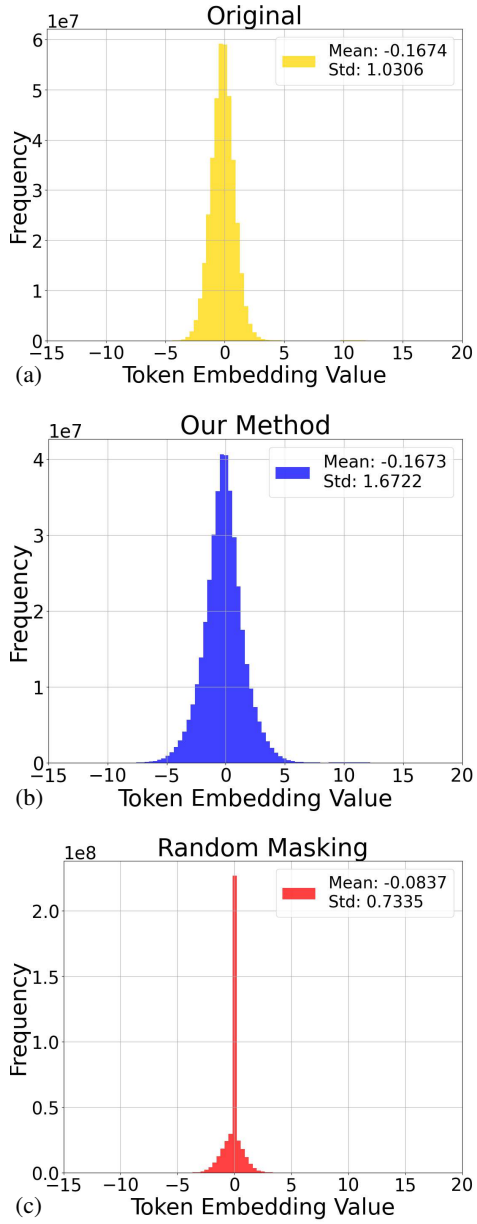


Figure 7. Comparison of token embedding distributions. (a) Original token embeddings with mean -0.1674 and standard deviation 1.0306; (b) Token embeddings of FPAN with mean -0.1673 close to that of the original embeddings but with a larger standard deviation of 1.6722; (c) Token embeddings of RM with smaller magnitude mean of -0.0837 and smaller standard deviation of 0.7335.

high quality of generated images. In addition, our method increases the variance of the token embedding distribution. This increased variance allows for greater diversity in token embeddings, thereby reducing the frequency with which the model encounters identical token embeddings during training, which in turn reduces the model's tendency to memorize specific token information.

For RM, we denote the token embedding processed by RM as τ''. The embedding τ'' is computed as follows,

$$\tau'' = m \cdot \tau, \quad m \sim \text{Bernoulli}(1-Q), \tag{9}$$

where m is a Bernoulli random variable that equals 0 with probability Q. We denote the mean and variance of the distribution that τ'' follows as μ''_τ and σ''^2_τ, respectively. The values of μ''_τ and σ''^2_τ can be derived as follows:

$$\mu''_\tau = \mathbb{E}[m \cdot \tau] = \mathbb{E}[m]\mathbb{E}[\tau] = (1-Q)\mu_\tau \tag{10}$$

$$\begin{aligned}\sigma''^2_\tau &= \mathbb{E}[(\tau'')^2] - (\mathbb{E}[\tau''])^2 \\ &= \mathbb{E}[m^2 \cdot \tau^2] - (1-Q)^2\mu_\tau^2 \\ &= (1-Q)(\sigma_\tau^2 + \mu_\tau^2) - (1-Q)^2\mu_\tau^2 \\ &= (1-Q)\sigma_\tau^2 + Q\mu_\tau^2(1-Q).\end{aligned} \tag{11}$$

From the derivation results, we observe that because $1-Q<1$, that the mean of the tokens after random masking μ''_τ is smaller in magnitude than the original μ_τ. Furthermore, given that $1-Q<1$ and μ_τ is close to zero, it follows that $(1-Q)\sigma_\tau^2 < \sigma_\tau^2$, and the term $Q\mu_\tau^2(1-Q)$ approaches zero. As a result, the total variance $\sigma''^2_\tau = (1-Q)\sigma_\tau^2 + Q\mu_\tau^2(1-Q)$ is less than $\sigma'^2_\tau = \sigma_\tau^2 + P \cdot W^2$. We conclude that RM alters the original semantic information and yields a smaller increase in variance than our method. These derivations suggest that RM is less effective than our approach in balancing the trade-off between generation quality and replication, possibly because it alters the mean of the token embedding distribution and fails to effectively increase the variance.

To experimentally show the differences of FPAN over RM, we analyze the impact of both methods on captions in the fine-tuning set. Specifically, we randomly select 5000 captions from the fine-tuning set, use $W=1.7$ and $P=0.6$ as the hyperparameters of FPAN and $Q=0.5$ for RM. Figure 7 presents the resulting means and standard deviations of the token embeddings after noise injection using each method. It can be observed that the mean of the token embeddings processed by our method remains nearly unchanged from the original token embeddings. In contrast, the magnitude of the mean of the embeddings under RM is approximately 50% of the original value. Moreover, the standard deviation of the token embeddings produced by our method increases by 62.25% relative to that of the original token embeddings, whereas the standard deviation resulting from RM decreases by 28.82%. These results empirically validate our theoretical analysis.