Discriminator Guidance in Score-based Diffusion Models

https://github.com/MehdiAbdellaoui/DiffusionGuidance

Mehdi Abdellaoui **Carl Persson** **Eduardo Santos Escriche**

Abstract

In this project, we attempt to reproduce some of the main results presented in the paper 'Refining Generative Process with Discriminator Guidance in Score-based Diffusion Models' [1], where the authors introduce a new method aiming to improve the sample generation process for pre-trained diffusion models, called Discriminator Guidance. Our initial experiments are able to produce very similar FID scores to those presented in the paper for the CIFAR-10 dataset and the EDM-G++ sampling method. In addition, we explore the statistical significance of those results, as well as the sensitivity of the S_{churn} hyperparameter. Lastly, we go beyond the scope of the paper by implementing and analyzing the performance of including the Low-Rank adaptation (LoRA) method for finetuning the discriminator, which we show can improve the FID score for unconditionally generated samples from 1.83 to 1.79.

1 Introduction

Many research papers, despite their contributions, lack sufficient transparency for others to replicate their results. This project aims to bridge this gap by attempting to reproduce key findings from 'Refining Generative Process with Discriminator Guidance in Score-based Diffusion Models' by Kim et al. (2022) [1]. In that paper, Kim et al. propose an innovative method to improve the sample generation process for pre-trained diffusion models by implementing the Discriminator Guidance method. This approach trains a discriminator network that discriminates between sampled images and training data, which is then used during sampling to adjust the score, thereby aligning the generated samples more closely with the distribution of the initial training data.

Our replication efforts aim not only to validate these findings but also to contribute to a broader understanding of discriminator-guided diffusion models in the field. This study will mainly focus on the experiments that employ the CIFAR-10 dataset. While larger datasets like ImageNet 256x256 offer more complexity, their high computational demands would constrain the scope and depth of our experiments given the project timeline. In addition to replicating the original findings, this study will venture beyond the initial paper's scope. This includes experiments related to best machine learning research practices, such as conducting statistical significance analysis and performing thorough hyperparameter searches, to ensure robustness and reliability in our findings.

Furthermore, this study will explore methods to enhance the results presented by Kim et al. (2022) [1]. The original paper utilized a discriminator architecture that involves using a pre-trained feature extractor whose weights are frozen during training. While effective in boosting the performance of diffusion models, this approach may not fully leverage the potential of discriminator guidance techniques. To address this, our research experiments with injecting LoRA (Low-Rank Adaptation [2]) layers into the pre-trained feature extractor. This modification allows for selective fine-tuning of the feature extractor without excessive computations and parameter updates. By integrating these

HT23 Deep Learning Advanced, DD2412, KTH.

LoRA layers, we aim to demonstrate a more efficient and potentially more effective approach to discriminator guidance in diffusion models.

2 Related Works

The original paper by Kim et al. (2022) [1] is an extension to diffusion models. In particular, the state-of-the-art EDM diffusion model [3] was one of the models chosen to showcase the capabilities of the discriminator guidance technique. The paper that presents the EDM model [3] makes multiple interesting changes to sampling and the training processes, especially in regards to stochastic sampling, where additional noise is added during the denoising steps.

Furthermore, the discriminator architecture proposed in [1] builds upon the ADM classifier presented in 'Diffusion Models Beat GANs on Image Synthesis' by Dhariwal et al. [4]. This work demonstrated the strong performance of diffusion models on image generation tasks by improving the architecture of diffusion models and exploring general improvements.

The experiments conducted in this paper will be focused on reproducing and analysing the statistical significance of the results. This mainly involves sample quality metrics and the sensitivity to hyperparameters, but also going beyond reproducibility by applying methods that may improve upon the work of Kim et al. (2022) [1]. Low-Rank adaptation (LoRA) [2] is a technique that allows for the finetuning of models of large models without excessive use of computational resources. This method may be well suited for finetuning the, otherwise frozen, feature extractor of the discriminator. Thus, we will explore this approach by injecting LoRA layers into the feature extractor of the discriminator.

3 Methods

In this section, we will present the main details of the Discriminator Guidance method introduced in [1], especially those concerning its implementation when combined with the EDM method.

3.1 Denoising Diffusion Probabilistic Models

Denoising Diffusion Probabilistic Models (DDPMs) can be described in terms of discrete time diffusion steps [5] as well as continuously through stochastic differential equations (SDEs) [6]. The discrete DDPMs works by adding iterative Gaussian noise to an input image x_0 through an approximate posterior distribution $q(x_{1:T}|x_0)$, generating T images that contain a varying amount of noise. The posterior distribution is fixed to a Markov chain such that $q(x_{1:T}|x_0) = \prod_1^T q(x_t|x_{t-1})$, where each $q(x_t|x_{t-1})$ has a Gaussian distribution [4]. The denoising model $p_\theta(x_{0:T})$ is defined as a Markov chain that is trained by minimizing the upper bound of the KL divergence, given by: [1]

$$D_{KL}(p_r(x_0)||p_\theta(x_0)) \leq D_{KL}(q(x_{0:T})||p_\theta(x_{0:T}))$$

where $p_r(x_0)$ is the original data distribution. The continuous interpretation of the DDPMs describes the diffusion process through an SDE: $dx_t = f(x_t, t)dt + g(t)dw_t$, where t is a continuous variable in [0, T], while $f(x_t, t)$ and $g(t)$ are the drift and volatility coefficients respectively[5], whereas w is the standard Wiener process (a.k.a., Brownian motion) [6]. This SDE has a unique reverse-time diffusion process given by:

$$dx_t = (f(x_t, t) - g^2(t)\nabla \log p_r^t(x_t))dt + g(t)dw_t \tag{1}$$

where it is desirable to train a model $s_\theta(x_t, t)$ that predicts the score (time dependent gradient field) of the perturbed data distribution $\nabla \log p_r^t(x_t)$ [6]. The probability density $p_r^t(x_t)$ is the perturbed data distribution that depends on the time t and follows the forward diffusion SDE outlined previously. Score based models, alternatively continuous DDPMs, are trained to minimize the loss function given by:

$$L_\theta = \frac{1}{2}\int_0^T \xi(t)E[||s_\theta(x_t, t) - \nabla \log p_{0t}(x_t|x_0))||_2^2]$$

where ξ is the temporal weight and p_{0t} is the transitional probability of going from x_0 to x_t and $s_\theta(x_t, t)$ is used to solve the reverse-time diffusion process [1].

3.2 EDM model

The EDM model is a score based model that works on the continuous methods mentioned above, but uses another loss function to train a model that indirectly predicts the score [3]. Furthermore, the model builds from the fact that all diffusion processes there is a corresponding deterministic process that share the same marginal probability densities $p_r^t(x)$. The corresponding deterministic process is the solution to an ordinary differential equation (ODE) [6]:

$$dx_t = (f(x_t, t) - \frac{1}{2} g^2(t) \nabla \log p_r^t(x_t))dt \tag{2}$$

The EDM model makes use of the deterministic ODE to sample images by using regular ODE solvers such as the second order Heun's method. However, the authors of the EDM model found that adding some temporary noise at each sampling step increased the quality of the sampled images compared to the deterministic sampling method, and it had the added benefit of adding diversity to the output images [3].

3.3 Discriminator Guidance

The idea behind discriminator guidance is that the model trained to predict the score $s_{\theta_\infty}(x_t, t)$ is unlikely to converge to the global optimum θ_*, and instead converges to some local optimum θ_∞. However, it is possible to prove that the reverse-time diffusion process (Eq. 1) coincides with the diffusion process with an adjusted score [1]:

$$dx_t = (f(x_t, t) - g^2(t)(s_{\theta_\infty} + c_{\theta_\infty})(x_t, t))dt + g(t)dw_t, \qquad c_{\theta_\infty} = \nabla \log \frac{p_r^t(x_t)}{p_{\theta_\infty}^t(x_t)} \tag{3}$$

where p_{θ_∞} is the solution to the reversed-time diffusion process where $s_{\theta_\infty}(x_t, t)$ is the data score, while p_r^t and p_{θ_∞} are the marginal densities of the forward-time diffusion process starting from p_r and p_{θ_∞}. This only holds if $s_{\theta_\infty}(x, T) = \nabla \log \pi(x)$ where π is the prior distribution and $\log p_{\theta_\infty}$ is equal to its evidence lower bound L_{θ_∞} [1].

The correction term cannot be computed directly since it requires access to $p_r^t(x_t)$, which is the solution to the reverse-time diffusion process (Eq. 1). Instead the correction term is approximated by a discriminator neural network $d_{\theta_\infty}(x_t, t)$, that is trained to discriminate between generated samples using $s_{\theta_\infty}(x_t, t)$ and the original dataset. The discriminator is trained at all noise levels [0, T] using binary cross entropy with weights according to the temporal weight $\xi(t)$. The correction term is then approximated by: [1]

$$c_{\theta_\infty} \approx \nabla \log \frac{d_{\theta_\infty}(x_t, t)}{1 - d_{\theta_\infty}(x_t, t)}$$

The discriminator is implemented using the encoder part of a U-Net [7] and consists of two parts. A shallow encoder U-Net that is attached to a pre-trained ADM classifier [4] which acts as a feature extractor. The ADM classifier only consists of the encoder part of a U-Net and is frozen during training, while the shallow encoder U-Net is the only part of the discriminator network that is trained. The discriminator is conditioned on both the image class (if available) and the noise level t through residual blocks in the U-Net architecture.

The trained discriminator is applied to the pre-trained EDM diffusion model during sampling using Eq. 3. To assess the performance increase by utilizing the discriminator guidance technique, both NLL and FID is calculated for the pre-trained EDM diffusion model with and without discriminator guidance.

3.4 LoRA: Low Rank Adaptation

Low-rank adaption (LoRA) [2] is a method that allows fine-tuning of large pre-trained models without the need to update all the weights of the original model. This is done for some weight matrix $W_0 \in R^{d \times k}$ by constructing two smaller matrices $B \in R^{d \times r}$ $A \in R^{r \times k}$ where the rank $r << min(d, k)$. It is then possible to only fine tune the matrices A and B, while freezing the weights W_0 and applying these smaller matrices to the original matrix by $h = W_0 x + BAx$, essentially creating a new fine-tuned matrix. If the rank r is very small it is possible to save a lot of computing resources since only a few parameters need to be updated at each gradient pass [2].

4 Data

In this section, we will present the main characteristics of the dataset considered in this project, CIFAR-10 [8].

The CIFAR-10 dataset consists of 60,000 32x32 colour images organized in 10 mutually exclusive classes, with 6,000 images per class. There are 50,000 training images and 10,000 test images. This dataset is very well-known and extensively used in the litterature surrounding Computer Vision tasks, and as such, it is a very natural choice for the evaluation of methods related to image generation.

The best performing model on this dataset for image generation, according to [9], is the Consistency Trajectory Model (CTM) [10] with an FID score of 1.63. Moreover, we can highlight that the Discriminator Guidance approach that we are studying ranks second, with an FID score of 1.64 for conditional generation, and sixth with an FID of 1.77 for unconditional generation. An interesting aspect to point out is that among the top 10 best performing models, 8 of them are Diffusion related models.

5 Experiments and findings

In this section, we will present the main experiments and results that form the basis for our analysis of the reproducibility of the considered paper [1]. In the upcoming section we make use of the Fréchet Inception Distance (FID) score as proposed in [11], and the Negative Log Likelihood (NLL) with a Gaussian assumption.

5.1 Reproducibility of NLL and FID scores

Our initial experiment attempts to reproduce the NLL and FID scores presented in the paper for CIFAR-10 using the EDM-G++ model (EDM with discriminator guidance). Note that the FID-50k score references the FID score computed by comparing 50 thousand generated images and 50 thousand images from the CIFAR-10 dataset, which was used during training. Likewise, the FID-10k score references the FID score computed by comparing 10 thousand generated images and 50 thousand images from the CIFAR-10 dataset.

The discriminator guidance was implemented as described in Section 3.3 and the discriminator was trained using the hyperparameters as given in the original paper. Table 1 and Figure 1 show the results specified in the original paper for the EDM and EDM-G++ models in combination with the results obtained through our implementation. There are slight discrepancies between our reported results and the results reported in the original paper [1], especially for the unconditional EDM-G++ model, where the FID-50k score is higher by quite a large margin. The FID plot seen in Figure 1 is similar to the plot given in the original paper, but the results aren't exactly comparable as we use FID-10k scores instead of FID-50k.

Model	**Unconditional**		**Conditional**
	NLL ↓	**FID-50k ↓**	**FID-50k ↓**
EDM (random seed)	2.60	2.03	1.82
EDM (manual seed)	2.60	1.97	1.79
EDM-G++ (random seed)	2.55	1.77	1.64
EDM (ours)	3.53	1.96	1.85
EDM-G++ (ours)	3.29	1.83	1.66

Table 1: Comparison between our reproduction results and the results of the original paper on the CIFAR-10 data-set. Our results were generated using a random seed for discriminator training and a set seed for image generation. The seed was chosen randomly and set to be constant in order to gain consistent results.

This could be due to slight differences in implementation caused by ambiguities in the paper and/or due to the natural variances in training the discriminator. The samples that we took from the original EDM model [3] and used for training, had a low FID-50k score, which could have impacted the

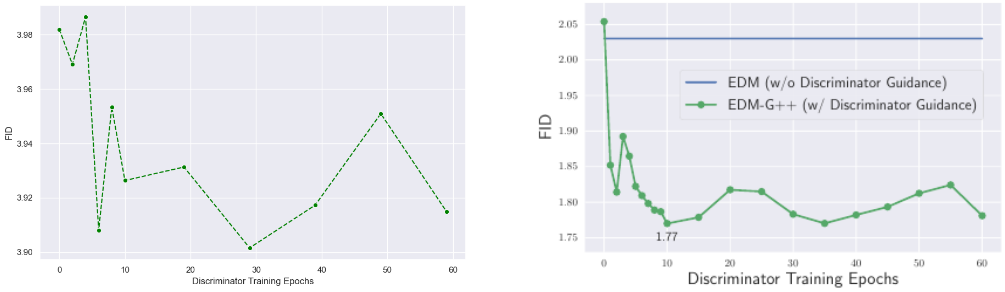

Figure 1: Left: Our results for FID-10k score as a function of the number of discriminator training epochs. (Note that FID-10k has a higher variance that FID-50K). Right: Discriminator guidance refines FID on CIFAR-10 (plot from [1]).

performance of our trained model. Nevertheless, we see a considerable decrease in FID-50k score for both the conditional and unconditional EDM-G++ models, where the results of the conditional EDM-G++ model were very close to the original reported results.

It is interesting to note that the authors of the original paper [1] are likely to have trained many more models compared to our re-production, and are likely to have fine-tuned their hyperparameters to their batch of sampled images from the original EDM model. This could partly explain their lower FID scores, as they are likely to report the best scores achieved throughout their experiments. Our attempts to reproduce the NLL scores weren't completely successful, since they deviated significantly from those reported in the original paper. However, we can still observe a similar trend, which shows that the proposed Discriminator Guidance model is able to provide a lower NLL score compared to the original EDM method. This discrepancy can likely be attributed to a difference in implementation of the NLL calculation, see Section 6 for more details.

5.2 Statistical Significance Analysis

In order to test the statistical significance of our results we employed the one sample location test and trained multiple unconditional discriminator models. The unconditional discriminator model was chosen for statistical significance testing as it required less time for training. The model was trained six times, each time using random initialization of weights and shuffling order of the training data. Each model was used to sample 50k images and the FID score was calculated for each model, the results of which can be seen in Figure 2.

Model (Unconditional)	N Samples	Mean	Standard Deviation	Standard Error
EDM-G++	6	1.8433	0.01082	0.004417

[1] All reported statistical values represent samples from their true respective distributions.

Table 2: The sampled FID score statistics generated through the training of multiple models using different initialization of weights and shuffling of training data.

In order to check the statistical significance of our results it would have been beneficial to sample FID scores multiple times from the original EDM model and compare the distributions, but due to limitations in time and resources a simpler approach was chosen. The null hypothesis is that the mean FID score of the unconditional EDM-G++ model is the same as the FID score of the samples generated from the original EDM model. The test is one tailed since we are only interested in a relative decrease in FID score for the unconditional EDM-G++ model as compared to the original EDM model, this yields a p value of (p values obtained using a t-table with 5 DoF):

$$t = \frac{\bar{X} - \mu_{EDM}}{SE} = \frac{1.8433 - 1.96}{0.004417} \approx -26.42 \qquad \Longrightarrow \qquad p < 0.0005$$

Thus it can be concluded that the result obtained using discriminator guidance for the unconditional case is statistically significant, but only under the assumption that the FID scores generated from the unconditional EDM-G++ is normally distributed and that the sampled FID score of the original EDM model is representative of the true mean of the original EDM model. However, note that the assumption that the FID scores are normally distributed is not that strong (but should still be checked for validity) and our FID scores for the original EDM model is close to the values presented in the original paper.

5.3 Hyperparameter sensitivity

At each time step during the sampling/de-noising procedure, a bit of noise is added to the image that is removed during the same time step. This is done to allow the model to correct errors made in previous de-noising steps [3]. The S_{churn} parameter governs how much extra noise is added at odd time steps when the log density ratio is low, where the log density ratio is defined as: $\log d_{\theta_\infty}(x_t, t)/(1 - d_{\theta_\infty}(x_t, t))$. As such, when the discriminator is confidently predicting that the image is not from the training dataset (i.e. when the log density ratio is low) we can expect that there is an error from previous de-noising steps. Increasing the noise for these samples can aid in correcting the error and S_{churn} governs how much extra noise is added to these samples. This technique is only applied to the conditional discriminator.

Most of the hyperparameters introduced in the original paper were thoroughly explored. However, the hyperparameter S_{churn} was set manually by the authors without further investigation, and thus we chose to study it further. In order to explore the effects of this variable, both a rough and fine hyperparameter search were carried out with regards to the FID-10k score. The rough search interval was chosen by checking large and small values of S_{churn} to ensure that the chosen range contained the optimal value. The best performing value of S_{churn} for the rough search was then used to decide the interval of the fine hyperparameter search. The results of this search are shown in the left plot of Figure 2.

The hyperparameter S_{churn} was found to converge to a FID-10k score of around 3.85 for values larger than 10 and not using the S_{schurn} parameter was found to yield a FID-10k score of around 3.76. As such the rough search was carried out in the interval between 0 and 10, the results of which can be seen in Figure 2. The optimal value for S_{churn} found during the rough search was 2, and consequently, the fine search was carried out for values between 1 and 3. The results of the fine search can be seen in the right plot of Figure 2 and had fairly high variance. However, we can conclude that the S_{churn} value of 2 yielded the best FID-10k score in the fine search as well. The FID-50k score was calculated for the conditional model with a value of $S_{churn} = 2$ which yielded a FID-50k score of 1.65. This is a modest decrease in FID score with respect to the result shown in Table 1.

This analysis showcased the importance of studying the sensitivity of the hyperparameters of a considered model, since, the specified value for the S_{churn} can have a significant effect on the final FID score. It is interesting to note that the value chosen by the authors ($S_{churn} = 4$) had almost no effect on the FID score of the generated samples, as observed in our generated plot in Figure 2.

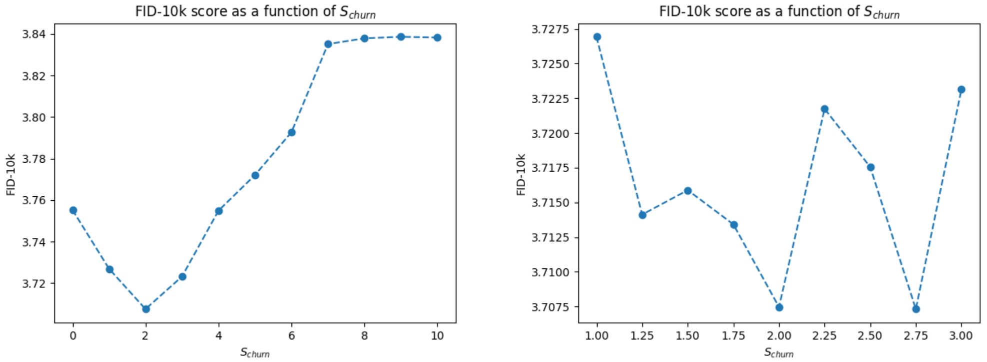

Figure 2: FID-10k plotted as a function of S_{churn}. FID-10k is calculated using the complete CIFAR-10 training set of 50k images and 10k generated images. Left: Rough search around the author's proposed value. Right: Fine search around the optimal value found in the left plot.

5.4 LoRA: Low rank adaptation

The low rank adaptation method (LoRA) was applied to the pre-trained ADM classifier model [4] which serves as a feature extractor for the discriminator model. The LoRA layers were injected into the ADM model at the first convolutional layer of every residual block. During training the regular layers were frozen, while the LoRA layers were fine-tuned. This meant that the ADM feature extractor was trained in tandem with the shallow encoder U-Net, potentially resulting in a more powerful discriminator model. The discriminator models with LoRA layers were trained using a lower rank of 16 and a higher rank of 128. The results of which can be seen in Table 3 and some sample images can be seen in Appendix A.

Model		**Unconditional**	**Conditional**
	LoRA rank	**FID-50k** ↓	**FID-50k** ↓
EDM	N/A	1.96	1.85
EDM-G++	N/A	1.83	1.66
LoRA EDM-G++	16	1.83	1.65
LoRA EDM-G++	128	1.79	1.68

Table 3: Our FID-50k results gained from sampling the EDM model, our implementation of the discriminator guidance model EDM-G++ as well as the discriminator guidance model trained with injected LoRA layers LoRA EDM-G++.

It is difficult to draw conclusions from the results in Table 3, since we see both decreases in FID-50k score as well as increases depending on the model. In the case where a low LoRA rank of 16 is used, we see no improvement in FID-50k score for the unconditional model and only a slight decrease for the conditional model. However, the results for the models trained with a LoRA rank of 128 show a sizeable decrease in FID-50k score for the unconditional model and a slight increase for the conditional model.

The higher amount of flexibility gained by injecting the LoRA layers into the feature extractor does seem to generally decrease the FID-50k score, and it is likely that the increase in FID score seen in the conditional model was a result of variance during training. However, it is important to note that the introduction of LoRA layers increased training times substantially and might have allowed the model to overfit to the training data and thereby increasing the FID score for the conditional models. To fully conclude that injecting the LoRA layers helped in decreasing the FID scores, a statistical analysis would need to be conducted.

6 Challenges

Regarding the implementation of the discriminator-guided sampler, we found the explanations in the paper and the corresponding appendix to be quite superficial since they mostly focused on general strategies, but didn't explain how the actual sampling was implemented for a particular method. In that sense, we had to look at the authors' code, from which we deduced that the lack of explanation of sampling was due to the fact that the authors had reused code from the implementation of the EDM model [3]. Therefore, we also used that code as the basis of our implementation, and then proceeded to generate the needed code to add the required modifications as they were specified in the Discriminator Guidance paper [1].

An additional challenge was that we noticed that the value of some hyperparameters, e.g. the period for adapting the S_{churn} parameter in the conditional setting, differed between what was specified in the paper ("every odd denoising step") and what was specified in the code (value of 5). In this case, we proceeded by following the explanations of the paper.

Furthermore, we found it difficult to choose viable experiments due to the long training and sampling times. This resulted in a few of the experiments using the FID-10k score instead of FID-50k, as it saved a considerable amount of time. This also applied to LoRA, where the training time increased drastically due to the additional compute needed to fine tune the feature extractor. Additionally, we also found a challenge when trying to compute the Negative Log-Likelihood Loss (NLL), since the methodology used in the paper was not specified. External resources had to be used in order to find a way to calculate the NLL.

7 Conclusion

Our re-implementation of the Discriminator Guidance method was able to provide very similar results to those presented in the paper despite the aforementioned challenges. Therefore, we conclude that our attempt at reproducing some of the main results of [1] was successful, since the difference in performance is more likely due to variances during training or slight differences in implementation. This initial work was quite useful in order to understand the possible challenges that can arise from reimplementing machine learning methods, as well as how to address them.

In addition, we explored the best machine learning research practices by studying the sensitivity to the values of the hyperparameter S_{churn} and the statistical significance of the previously shown results. In particular, we show that the results are statistically significant for the considered model, and that the S_{churn} value selected by the authors might not be the optimal one. Through these experiments, we learned about the importance of analyzing the sensitivity of the hyperparameters, since as we observed, the results can vary a lot depending on the selected value.

Lastly, we also go beyond the scope of the paper, and study the inclusion of the LoRA technique for finetuning layers in the pre-trained feature extractor that is part of the discriminator. Our results show that this modification produces a better performance for the unconditional generation when using a large enough rank.

Regarding possible future work, if more computational resources and time had been available, we would also have liked to explore the possibility of considering discriminator architectures different from the U-Net, e.g. ResNet18 or a transformer. Moreover, it would have also been interesting to consider other datasets and methods that worked with a latent diffusion space, e.g. LSGM-G++.

8 Ethical consideration, societal impact, alignment with UN SDG targets

Regarding the ethical considerations derived from the experiments presented in the studied paper and in our analysis, we believe that those would be related to the potential usages of Diffusion or generative models in general, since the proposed methods simply refine their performance. In that sense, the ethical considerations would mainly relate to potential negative societal effects of such models, such as disinformation, harmful biases, or emphasizing certain stereotypes [12].

That problem of copyright is also of great significance currently, which has led to the *US Copyright Office* to recently create a request for comments on copyright and generative AI [13]. Multiple tech companies have submitted their comments mainly arguing that the usage of this type of models falls under the category of fair use, but artists have presented multiple complaints about the potential harms of those models, including reputational damage and economic loss [14].

Lastly, taking into account the aforementioned potential negative effects derived from the usage of this type of generative model, we can also consider how they relate to the United Nations' Sustainable Development Goals (SDGs) targets [15]. In that sense, we observe that they would mostly relate to goal 10 "Reduced inequalities" since the potential disinformation and biases showcased by this models could potentially hinder the progress towards that goal.

9 Self Assessment

Overall, our experiments show that we have successfully re-implemented some of the main novelties introduced by the studied paper [1]. We have also carried out several additional experiments, mostly related to analyzing the consideration of the best practices in machine learning in the project paper.

Additionally, we went beyond the scope of the paper's method and included the LoRA method for finetuning the considered pre-trained ADM classifier during the training process of the discriminator. The results in this case show an improvement of the FID-50k values obtained without LoRA mainly for unconditional generation, but also for some conditional generations.

Considering this, we believe that the aforementioned implementations and results, together with the consideration of the challenges mentioned in Section 6, make this project deserving of the grade we aimed for in our project proposal, i.e. an A.

We also want to self nominate for a bonus point as we believe that the difficulty of the implementation and the faced challenges was significantly higher compared to other papers.

References

[1] Dongjun Kim, Yeongmin Kim, Wanmo Kang, and Il-Chul Moon. Refining generative process with discriminator guidance in score-based diffusion models. *arXiv preprint arXiv:2211.17091*, 2022.

[2] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[3] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

[4] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021.

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[6] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[8] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

[9] Papers With Code. CIFAR-10 Benchmark (Image Generation). `https://paperswithcode.com/sota/image-generation-on-cifar-10`.

[10] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279*, 2023.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[12] Pamela Mishkin, Lama Ahmad, Miles Brundage, Gretchen Krueger, and Girish Sastry. Dall·e 2 preview - risks and limitations. 2022.

[13] U.S. Copyright Office. Artificial Intelligence Study. `https://www.copyright.gov/policy/artificial-intelligence/`.

[14] Harry H Jiang, Lauren Brown, Jessica Cheng, Mehtab Khan, Abhishek Gupta, Deja Workman, Alex Hanna, Johnathan Flowers, and Timnit Gebru. Ai art and its impact on artists. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pages 363–374, 2023.

[15] United Nations Development Programme. Sustainable Development Goals. `https://www.undp.org/sustainable-development-goals`.

A Generated images

Figure 3 includes some of the images generated by our improved unconditional EDM-G++ model including LoRA (left), compared to samples generated by the unconditional pre-trained EDM model [3].

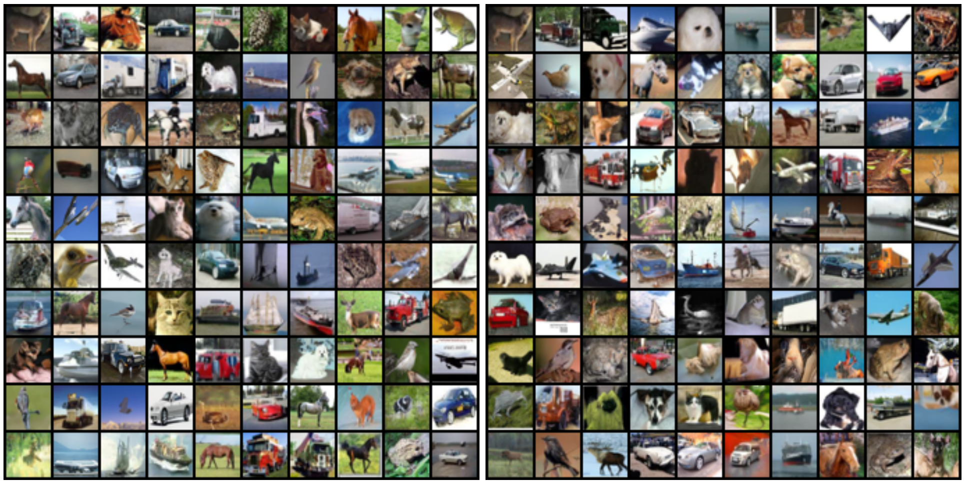

Figure 3: Left: 100 samples generated from the trained unconditional EDM-G++ model using LoRA (FID: 1.79). Right: 100 samples generated from the unconditional pre-trained EDM model (FID: 1.96).