Evaluating the Efficacy of Fine-Tuning Methods for Concept Insertion in Pre-trained Diffusion Models

Gabriel Guralnick **Haoze Deng** **Miguel Weerasinghe**

Abstract

Latent diffusion models (Rombach et al., 2022) mark a new era in image generative models as they are able to capture insights from textual prompt and render faithful, high-fidelity images. Large pre-trained diffusion models such as Stable Diffusion have become essential tools for artists and researchers. However, these existing models lack the ability to synthesize specific objects due to the vastness of their training corpora. In this work, we investigate current techniques in fine-tuning pre-trained diffusion models, specifically for the task of inserting new concepts. We experiment with the three most popular methods: textual inversion, dreambooth, and low-rank adaptation (LoRA). Using CLIP cosine similarity and Fréchet Inception Distance, we quantify the efficacy of each fine-tuning approach with regards to how well the new concept is adapted to different environments[1].

1 Introduction

Large pre-trained latent diffusion models give users access to unlimited creativity, but what if you really want to incorporate your own stuffed animal in the generated images? Even with prior exposure, it is unlikely that the pre-trained models would produce your desired item with high fidelity and accuracy no matter how descriptive the prompt is. However, fine-tuning methods make such specific image generations possible. Fine-tuning image generation models involve utilizing a small dataset (3-5 images) containing images of the new concept to retrain the model slightly. The goal is to create a model that is able to faithfully place these new objects in new contexts.

2 Background and Related Works

All of our work here is based on latent diffusion models. Such a model generates a text embedding from an input prompt, which is then used to denoise a latent code iteratively in low dimensions using an UNet. The result is then fed forward through a variational autoencoder that decodes to the image of the desired resolution (Rombach et al., 2022). In a large pre-trained latent diffusion model, we are primarily interested in fine tuning the latent denoising UNet and the VAE decoder (see Figure 1).

3 Method

Unless specified otherwise, the diffusion and fine-tuning models used were imported from the HuggingFace Diffusers library (von Platen et al., 2022).

The evaluation models were imported from the HuggingFace transformers library (Wolf et al., 2020).

Throughout our experiment, we kept our hyperparameters consistent across fine tuning methods. Our generated images are of size 512×512.

[1] Our code is available at `https://github.com/gnguralnick/stable-diffusion-fine-tuning`

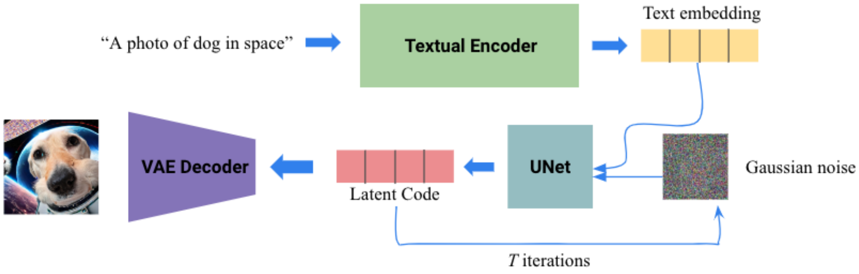

Figure 1: High level algorithm box for latent diffusion models at inference time.

3.1 Fine-Tuning Methods

Textual Inversion. Textual Inversion (Gal et al., 2022) diverges from typical fine-tuning methods in that it does not actually add or update the weights of the underlying image generator, or even train new parameters in general. Instead, it "inverts" the text-to-image process by using gradient descent to learn the best possible caption to generate the target output. The method adds a new pseudo-word S_* to the image generator's vocabulary and performs gradient descent to optimize the encoding of S_* in the model's embedding space. The training loss is calculated by prompting the model with a caption containing the pseudo-word (e.g., "a photo of S_*") and comparing the output to the provided target images. This method has proven effective both for object insertion and style transfer. The learned embedding also allows for total flexibility use of the fine-tuned model, as once training is complete the target concept can be used just like any other caption to prompt image generation. The primary limitation of textual inversion is the long optimization time. Several optimizations have been proposed (Voronov et al., 2023): the use of CLIP to initialize the starting embedding, the use of more specialized optimizers, and the use of a more advanced early stopping criteria to detect when optimization has stopped achieving noticeable improvements in output quality. However, our experiments do not make use of these optimizations.

Dreambooth. Dreambooth (Ruiz et al., 2023) offers flexible concept insertion such that it utilizes class prior distribution to adapt new object into existing object classes. Utilizing a small training set of a novel object (3 to 5 images), dreambooth trains the pre-trained model with text prompts that contain an unique identifier for the class. For example, if the training image is a specific mug, the training prompt should be "A photo of a [V] mug" where [V] is the identifier. In this work, we use the identifer "sks". Additional training on UNet allows the model to associate the new concept with the identifier. In addition to training using the forward diffusion pass and reconstruction loss, dreambooth also employs class specific prior preservation loss which encourages generation of novel contexts.

Low-Rank Adaptation of Large Language Models (LoRA). Low-Rank Adaptation (LoRA) is a technique that adapts pre-trained Stable Diffusion models through low-rank modifications to model parameters, and accelerates the training of large models while consuming less memory. Thus offering a cost-effective and memory-efficient fine-tuning method (Hu et al., 2021). It adds pairs of rank-decomposition weight matrices to existing weights, and only trains those newly added weights. The LoRA matrices are generally added to the attention layers of the original model. Then you can control the extent to which the model is adapted toward new training images via a scale parameter. By retaining the core functionality of Stable Diffusion models, LoRA allows them to be effectively customized for domain-specific applications, while reducing the risk of catastrophic forgetting during fine-tuning as previous pretrained weights are kept frozen. This is relevant to the ability to make images via stable diffusion methods with fewer GPU resources. .

3.2 Evaluation Metrics

Contrastive Language-Image Pre-Training (CLIP) Similarity. CLIP (Radford et al., 2021) is an efficient and powerful image classification model. It consists of an image encoder and text encoder trained jointly to predict a set of (image, caption) pairs. At test time, then, it can predict a similarity score between a given image and a given caption, and can therefore provide the most probable caption for the image. The robustness of its embeddings for image features has prompted its use as an evaluation metric for comparing generated images to target images (see, e.g., (Gal et al., 2022)). Rather than calculating the similarity between an image and a caption, two sets of images can be projected into the CLIP embedding space to calculate their cosine similarity. We use this metric to evaluate our fine-tuning reconstruction accuracy. For each fine-tuning method, we pass both the target images and the generated images (for each target object) into CLIP and calculate the average pair-wise CLIP-space similarity between the embeddings.

Fréchet Inception Distance (FID). FID is an evaluation metric first proposed for Generative Adversarial Networks (GANs) (Heusel et al., 2018) and has become a popular metric for all generative models. The key idea of FID is to pass the generated images x^* and ground truth images x_{gt} through a pre-trained Inception V3 model on ImageNet (Szegedy et al., 2015). The coding layer of Inception V3 is then fitted with a Gaussian distribution for each image set. Mathematically, given x^* and x_{gt}, we obtain Gaussian distributions $X^* \sim N(\mu_1, \Sigma_1)$ and $X_{gt} \sim N(\mu_2, \Sigma_2)$, where μ_1 and μ_2 are the means and Σ_1 and Σ_2 are the covariances. Then the FID is calculated as

$$d_f^2 = ||\mu_1 - \mu_2||^2 + Tr(\Sigma_1 + \Sigma_2 - 2\sqrt{\Sigma_1 \Sigma_2})$$

Note that the dimension of the Inception V3 coding layer is 2048, however we have significantly less samples when calculating FIDs. This produces in complex number as the covariance matrices are not full rank. To address this issue we take the real component of the FID output in our final results.

3.3 Target Images

The target images used were a combination of images taken by the authors using mobile phones and a subset of the Dreambooth concept insertion dataset (Ruiz et al., 2023).

4 Experiments

Our experimental design and evaluation setups mimic those used by Gal et al. (2022) in their original evaluation of textual inversion. However, we now extend that evaluation to include the other fine-tuning methods.

First, for each target image class, we perform Textual Inversion, Dreambooth, and LoRA to generate fine-tuned models that contain the desired concept. Then, we generate 5 images from each fine-tuned model for each class using a basic prompt ("a photo of <*>"), and calculate CLIP similarity and Fréchet Inception Distance to the target images. We refer to the scores achieved under this evaluation setup as the "Image Similarity" scores, as they represent the similarity of the generated images to the target images.

In addition, we wanted to evaluate each fine-tuning methods' ability to incorporate the learned concepts in more complex prompts. We used the trained models to generate 5 images for each of the more complex prompts (e.g., "a photo of <*> in a forest") and calculated the similarity to the generated output for corresponding prompts that do not contain the learned concept (e.g., "a photo of in a forest"). Greater similarity here demonstrates better incorporation of the learned concept in a more complex prompt and demonstrates increased editability of the fine-tuned model. We refer to the scores achieved under this setup as the "Prompt Similarity" scores, as they represent the ability of the fine-tuned models to incorporate text prompts in their generations.

To provide reference baselines for the results, we also calculate the CLIP similarity and FID between each target image set and itself, and the CLIP similarity and FID between sets of images generated using only the evaluation prompts. The evaluation results achieved for each of these are suffixed with "-image" and "-prompt" outputs, respectively (so the point labeled "lora-prompt" represents the results of the LoRA model for a prompt that did not include the learned concepts).

5 Results

Figure 2 show how each of the models performed in terms of ability to reproduce the original target images ("image similarity") versus ability to incorporate the learned concepts in more complex prompts ("prompt similarity"). For CLIP, the theoretical maximum is the point (1, 1).

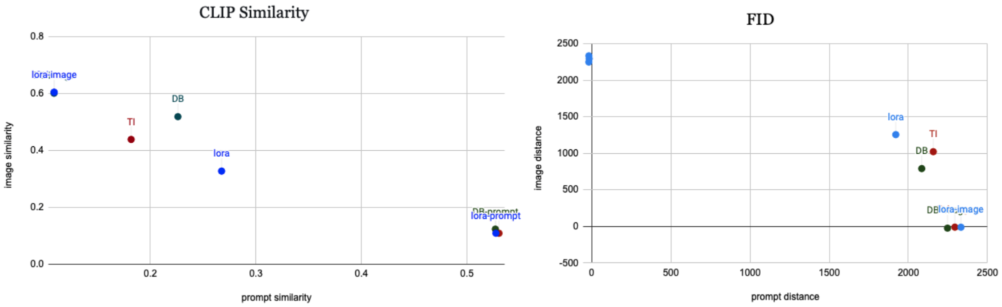

Figure 2: Plots for CLIP similarity score and FID. For CLIP similarity, the point closest to (1,1) is better. For FID, the points closer to baselines is better. Ideally without issue with complex number the FID should be at (0, 0) for minimum. The overlapping points on each graph along the axes are the baseline values, which are similar for all fine-tuning methods since the underlying Stable Diffusion model and set of target images was the same.

Overall, Dreambooth achieves the best results in terms of reproducing the target images, while LoRA performs better at adhering to the more complex prompt types. This holds for both evaluation metrics.

We were able to observe the trade-off between the image similarity and prompt similarity in the model training. The higher the image similarity, the more likely the model will produce images similar to the training data. The higher the prompt similarity, the better the model is able to create images based on a wide variety of prompts. The ideal case would be a (prompt-similarity, image-similarity) of (1,1) which would indicate that the model has high flexibility while also being able to create outputs that fulfill the purpose of the model fine-tuning. From our limited experiment, we observe that Dreambooth generally performs best when recreating the inserted concepts, but it tends to overfit the model as it doesn't perform well in complex prompts (i.e. prompts that place new concept into novel contexts). We also see that textual inversion places last in both regards compared to the other techniques.

Based on the results, we also see that the methods of training were not all that effective for our experiment as our prompt vs image coordinates were far off of (1,1) and the distances were not close to the baselines. We can attribute the lower values of prompt-similarity and image-similarity and prompt distance and image distance to the lack of dataset size and time constraints on training. A larger dataset than the image sets we used would have produced a better result in the models.

6 Conclusion

The tradeoff between image similarity and prompt similarity is known as the distortion-editability curve (Gal et al., 2022) and is a well-known phenomena in the field of image generation. Overall, we conclude that Dreambooth achieved the best fine-tuning results. Textual Inversion lost on all accounts, as it had both the worst prompt and image scores under both evaluation metrics. Meanwhile, although Dreambooth had worse performance on the novel prompts, the difference in its prompt similarity performance from that of LoRA is less than the difference in image similarity from LoRA to Dreambooth.

Ideally, in future experiments, we would like to explore the trade-off between image set sizes and model efficacy. It would be of interest to also have the resources to train a model for long periods of time uninterrupted as that could facilitate the creation of a more versatile model. We would also like to test more recent versions of stable diffusion as the technology behind stable diffusion is quite new, and improvements to the core technology can aid in the understand and research of the field. (von Platen et al., 2022). The last improvement that could be explored in future research is the optimal hyper-parameters that maximize the utility of each method of fine tuning (i.e. lowering the learning rate for dreambooth to prevent overfitting).

References

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

M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models, 2021.

A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2022.

N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015. URL `http://arxiv.org/abs/1512.00567`.

P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, and T. Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

A. Voronov, M. Khoroshikh, A. Babenko, and M. Ryabinin. Is this loss informative? speeding up textual inversion with deterministic objective evaluation, 2023.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/2020.emnlp-demos.6`.