Stable Diffusion - Thumbs Up Gesture

Nithyashree Manohar

September 25, 2023

1 Introduction

This document serves as a comprehensive overview of the assessment focused on fine-tuning the advanced Stable Diffusion XL (SDXL) model, a cutting-edge addition to diffusion models renowned for generating superior quality images. The assessment revolves around the application of the DreamBooth LoRA technique, experiments with learning rates, and deployment considerations. Challenges, insights, and methodologies employed during this assessment are detailed below. Google Colab's T4 GPU was used to fine-tune the model and to run the refiner NVIDIA A100 GPU on Azure was used.

2 Data Preparation

The Stable Diffusion XL (SDXL) model exhibits significantly enhanced capabilities compared to its predecessors, enabling fine-tuning with a minimal dataset. Typically, a collection of approximately 5-6 images is sufficient for fine-tuning SDXL on a single individual. However, more intricate subjects or varied images may necessitate a more substantial dataset. Each image was selected to represent the different backgrounds and angles, ensuring comprehensive and robust fine-tuning of the model.

For fine-tuning the model in this project, the following five images were utilized:

3 Approach

- The Stable Diffusion XL model from the huggingface diffusers library was employed for the training, leveraging the DreamBooth LoRA technique for fine-tuning. The details and documentation

of the model can be found at Diffusers Library on GitHub. (Note: Currently only DreamBooth fine-tuning via LoRA is supported for SDXL).

- All training was initially conducted on Google Colab; however, limitations were encountered as the refiner could not be run due to "CUDA out of memory" errors on Colab's provided T4 GPU. To overcome these constraints, an NVIDIA A100 GPU, boasting 80 GB of RAM, was utilized, with PyTorch compiled with CUDA enabled. The GPU was hosted on an Azure NC24ads instance. Due to high operational costs, this environment was reserved exclusively for inference tasks.
- In the pursuit of optimal results, extensive experimentation was conducted with learning rates of 1×10^{-4}, 1×10^{-5}, and 5×10^{-4}, executing 500 steps with checkpoints every 100 steps to discern the most favorable outcomes.
- Due to hardware limitations, attempts to fine-tune the model with prior preservation on Google Colab's T4 GPU failed. However, given access to suitable hardware this can be reattempted.
- Used the following prompts after fine-tuning to generate results
  - A picture of Ronaldo showing the thumbs up gesture on top of a hill
  - A picture of Ronaldo showing thumbs up gesture while holding an orange juice box in his right hand
  - A picture of Ronaldo showing the thumbs up gesture while swimming
  - A picture of Ronaldo showing the thumbs up gesture in a press conference
  - A picture of Ronaldo showing the thumbs up gesture on the moon

4 Results without refiner

The images displayed below are the best ones from multiple checkpoints, please refer to the GitHub repository for the complete set of results

4.1 Results for lr 1×10^{-4}

4.2 Results for lr 5×10^{-4}

4.3 Results for lr 1×10^{-5}

5 Results with refiner for lr 1×10^{-4}

The images displayed below are the best ones from multiple checkpoints for learning rate 1×10^{-4}, please refer to the GitHub repository for the complete set of results.

6 Analysis

A series of experiments were conducted to analyze the impact of different learning rates—$1e-4$, $1e-5$, and $5e-4$—on the quality of the generated images, focusing particularly on the facial features and the background of the images.

6.1 Learning Rate Analysis

- $1e-4$**:** This learning rate has demonstrated optimal results, producing the most accurate resemblance in facial features and maintaining integrity in the background. It is, therefore, deemed the most balanced and effective learning rate in the experiments.
- $1e-5$**:** While the backgrounds are reasonably well-preserved at this rate, the fidelity of facial features, particularly in resembling Ronaldo, is compromised, making it less optimal compared to a learning rate of $1e-4$.
- $5e-4$**:** The results at this learning rate, although generally satisfactory, exhibit signs of overfitting. The background, in particular, tends to lose its coherence in some instances, compromising the overall quality of the generated images.

6.2 Refiner Analysis

The incorporation of the refiner has shown to enhance the results in several instances, providing improved coherence and detail in the generated images. However, its effectiveness is not universally observed across all cases, indicating a selective advantage based on specific conditions or attributes of the input data.

In conclusion, a learning rate of $1e-4$ is recommended for achieving the most balanced and high-quality results in terms of both facial resemblance and background integrity. Additionally, while the refiner can offer enhancements in some cases, its application should be considered judiciously, given its inconsistent impact on the output quality.

7 Deployment Strategy

Deploying the custom SDXL model at scale necessitates a comprehensive and structured approach to ensure reliability, adaptability, and optimal performance. The following strategic components form the backbone of the deployment plan:

- **Training and Evaluation:** Optimal training data and rigorous evaluation are crucial to enhance the model's adaptability and accuracy in varied operational environments.
- **Deployment Environment:** Selection of a suitable deployment environment is paramount, with considerations extended to leading cloud solutions such as AWS, Google Cloud, or Azure to leverage their advanced and robust services.
- **Containerization:** The model and its dependencies will be containerized, utilizing technologies like Docker, to facilitate seamless deployment and management across different environments and to mitigate deployment-related discrepancies.
- **Monitoring and Maintenance:** Continuous monitoring is essential to ensure the model's alignment with expected behaviors and to identify and address any deviations or issues promptly.
- **Continuous Integration and Deployment (CI/CD):** A CI/CD pipeline will be established to automate the processes of building, testing, and deploying new versions of the model, ensuring smooth and efficient updates and enhancements.
- **Version Control:** Effective version control of the deployed models and associated code is crucial for tracking alterations, enabling rollbacks to stable versions when necessary, and facilitating collaboration.
- **Regular Updates and Retraining:** The model will undergo periodic updates and retraining with new or improved data to maintain its relevance and accuracy.

These components collectively contribute to the robustness and efficiency of the custom SDXL model in diverse application scenarios, ensuring its sustained excellence and adaptability.