**ML Engineer Assignment @ Ayna – REPORT**

**Introduction: -**

The problem statement involved two major components:-

1. Generate images based on the given prompt

2. Focus on creating a photo-realistic image(high-resolution image)

**Research Background**

***Diffusion Models***

Diffusion models are a type of generative artificial intelligence that create high-quality images from textual descriptions. Imagine starting with a blank canvas that initially just contains random, noisy visual data. Through a sophisticated process known as denoising, the model gradually removes this noise, transforming the chaos into a detailed picture that matches the provided text prompt. This denoising process is inspired by thermodynamic processes observed in nature, where systems naturally evolve from disorder to order. Technically, diffusion models operate by reversing a diffusion process over multiple steps, using a deep learning framework. A trained neural network guides this transformation, conditioning the evolving image to increasingly resemble the desired output. The result is remarkably realistic images that not only closely align with complex textual descriptions but also allow for precise control over the generated content. This capability makes diffusion models powerful tools for applications in creative and design fields, where detailed and controlled image creation is crucial.

*Image Resolution Models*

Super-resolution techniques, such as Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR) and Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN), are advanced methods used to improve the resolution and quality of images. EDSR focuses on reconstructing high-resolution images from their low-resolution counterparts without using Batch Normalization, enhancing detail while maintaining efficiency. ESRGAN, an improvement on traditional GANs, uses a perceptual loss function to produce images that are not only high resolution but also texturally rich and more appealing to the human eye*.*

*Fine-tuning (LoRA’s)*

Locally Receptive Fields (LoRAs) represent another innovative approach, where additional trainable parameters are inserted into existing model architectures. This allows for fine-tuning and adaptation without extensive retraining of the entire network, enhancing model flexibility and responsiveness to specific tasks or data types.

*Mixture of Models*

Combining different models, or a mixture of models, leverages the strengths of various architectures to address complex challenges in image generation. For example, using one model to generate an initial image and another to refine details or add stylistic elements can achieve superior results, particularly in tasks requiring high fidelity and specific aesthetic qualities. This hybrid approach maximizes the benefits of each model, leading to enhanced performance and output quality.

**Experimental Setup**

The complete experiment was majorly tested on a local setup, with the following configurations:

1. Nvidia GTX-1080 (VRAM-8GB)
2. RAM - 32 GB
3. CPU- Intel 7820HK
4. OS - Ubuntu

The above local setup was used instead of Colab to understand how to build setups on Linux hardware, set libraries from the base using pip, and work under certain hardware constraints, especially those that are faced during deployments.

Certain codes were still tested on Colab as GPU requirements were not met.

**Model Selection and Experimentation:**

*Note: - All models and libraries used in the Experiment were open-source.*

For the current problem statement, I have just used a pre-existing trained model, to see if it can solve the problem statement as effectively as possible. Here are the results:

* For Image Generation(Diffusion Models) multiple models were tried. Some models that are heavy on resources that were tested are Stable Diffusion v1.5, Stable Diffusion XL Base 1.0, and F-I-XL-v1.0, etc. Additionally, other models like aMused, Mini-Dalle, and CompVis/stable-diffusion-v1-4 (Latent Diffusion Model) were also experimented with.
* For a basic comparative study 3 models, **CompVis/stable-diffusion-v1-4**(Model-1), **aMused**(Model-2), and **Stable Diffusion XL Base 1.0**(Model-3) were selected. Model-3 had the best results but used the most resources whereas Modle-1 gave good results using fewer resources. Model 2 had bad results but used the least resources.
* To achieve the desired resolution of 2048\*2048, I have used the EDSR model to improve the resolution of the image.

**Link to Colab Notebook: -** https://colab.research.google.com/drive/1kYtSRZbf7lG7a2MaUBghfrBEOCJmGi8z?usp=sharing

**Link to Git Repo: -**

https://github.com/Vedaang-Chopra/Computer\_Vision\_Projects/tree/main/Ayna\_Generative\_AI\_Assignment

**Results and Observations:**

The following are the results from the experiment: -

1. Increasing the resolution is dampening the quality of the generated image, and super-resolution currently needs more work
2. Generating images using the same prompt can give multiple outputs, so re-training on custom data to avoid large variations is necessary.

**Challenges and Limitations:**

The following challenges were observed while working on the problem:-

1. One of the biggest challenges was GPU. The majority of the time the GPU went out of Memory since the entire RAM was consumed.
2. Data Constraints: - To re-train lightweight models more data needed to be generated for better quality retraining.

**Optimal Pipeline and Recommendations(Future Work):**

The following tasks need to be implemented to improve the solution for the problem statement: -

1. Retraining AI model(stable diffusion -XL) on a custom dataset, to improve accuracy
2. Fixing hyperparameter and finetuning tuning of EDSR, to improve super-resolution

**Conclusion:**

The submitted code is a stepping-stone solution, towards the complex problem statement. Much more refinement can be added to improve the solution

**References:**

1. <https://huggingface.co/eugenesiow/edsr-base>
2. https://huggingface.co/CompVis/stable-diffusion-v1-4