

Text to Image Generation using Stable Diffusion

Presented by Mintu Adak(2311MC22)



Project Overview


Objective:

- ❑ Generate high-quality images from text prompts
- ❑ Deploy a real-time demo to allow users to customize parameters for better results

Technology Used: `runwayml/stable-diffusion-v1-5`

Frameworks & Libraries:

- Diffusers: For image generation
- Torch: Deep learning framework for computations
- Gradio: User interface for web-based interaction
- PIL & Matplotlib: Image processing & visualization
- Python



MODEL & PIPELINE SETUP

Pretrained Model: [runwayml/stable-diffusion-v1-5](#)

Pipeline Initialization:

1. Utilizes the StableDiffusionPipeline from Hugging Face
2. Run on CPU(Since free Hugging Face space does not support GPU)

Scheduler for Image Generation:

1. Euler Discrete
2. LMS Discrete
3. PNDM
4. DPM Solver Multistep
5. DDIM

IMAGE GENERATION PROCESS

Inputs:

Prompt: Description of the desired image

Negative Prompt: What to exclude

Inference Steps: Number of denoising steps

Guidance Scale: Balances prompt adherence

Image Dimensions: Height & width selection

Batch Size: Number of images generated

Scheduler: Algorithm selection

Processing:

1. Selects the scheduler based on user input
2. Sets a random seed for reproducibility (if provided)
3. Generates an image output list

Gradio Web App and Deployment

Frontend Features:

Textboxes: User input for prompt & negative prompt

Sliders: Fine-tune image generation parameters

Dropdown: Select different scheduler algorithms

Gallery Output: Displays generated images

Web app deployment:

Hugging Face Spaces:

Web demo hosted on free tier (CPU-based) → Slower execution.

Link: https://huggingface.co/spaces/cyrus007/Text_to_image_generation_using_stable_diffusion

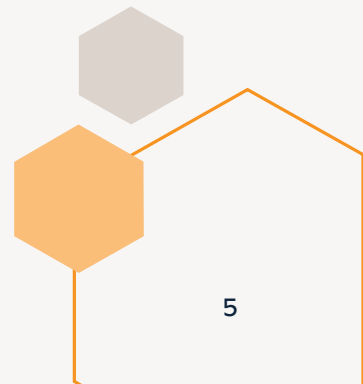
For Faster Execution:

Run the IPYNB file on Google Colab from GitHub.

Link: https://github.com/cyrus0001/cyrus007-Text_to_image_generation_using_stable_diffusion

Enable GPU: Runtime → Change runtime type → GPU

Faster processing compared to a CPU.



Thank you

