Generative Project Milestone 2

Team Members (Group 11)

Lochan Enugula
Rishabh Joshi
Rutuja Jadhav
Yaswanth Kumar Reddy Gujjula

Github Link: https://github.com/rutuja-jadhav29/NNDL-GenerativeProject

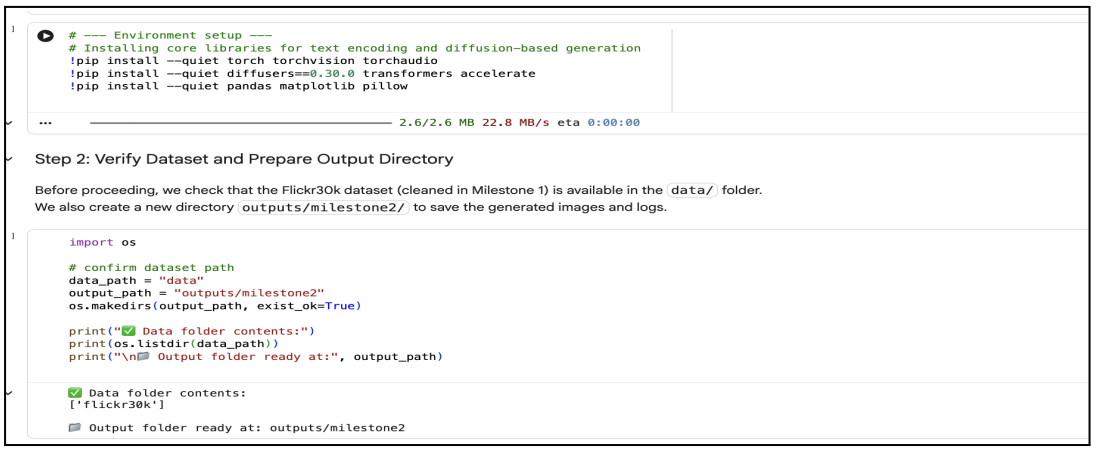
1. Objective:

The goal of this milestone was to connect the text-encoding and diffusion components to create a working text-to-image generation pipeline. We used the pretrained Stable Diffusion v1-5 model with its built-in CLIP text encoder to generate images from natural-language prompts. The main focus was to observe how changing parameters such as guidance scale and number of inference steps affected the quality and alignment of generated images.

2. Experimental Setup:

We selected five sample prompts from the cleaned Flickr30k dataset that included both human and object-based scenes. Each prompt was tested with three different guidance scales (5.0, 7.5, and 10.0). We also compared three inference step values (25, 50, and 75) to understand how the noise-removal process influences the final image. All generations were executed on a GPU runtime using the Hugging Face Diffusers library.

For implementation, we used the Stable Diffusion pipeline provided by the diffusers library in Python. The model was initialized using the pretrained weights of Stable Diffusion v1.5, which includes an internal CLIP text encoder and a UNet-based diffusion network. The experiment involved looping through multiple text prompts and systematically varying the guidance scale and inference step parameters. The generated outputs were saved for qualitative comparison, while a CSV training log recorded all configurations for Reproducibility.



```
# --- Environment setup ---
# Installing core libraries for text encoding and diffusion-based generation
!pip install --quiet torch torchvision torchaudio
!pip install --quiet diffusers==0.30.0 transformers accelerate
!pip install --quiet pandas matplotlib pillow
```

```
... ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.6/2.6 MB 22.8 MB/s eta 0:00:00
```

Step 2: Verify Dataset and Prepare Output Directory

Before proceeding, we check that the Flickr30k dataset (cleaned in Milestone 1) is available in the `data/` folder.
We also create a new directory `outputs/milestone2/` to save the generated images and logs.

```
import os

# confirm dataset path
data_path = "data"
output_path = "outputs/milestone2"
os.makedirs(output_path, exist_ok=True)

print("✅ Data folder contents:")
print(os.listdir(data_path))
print("\n📁 Output folder ready at:", output_path)
```

```
✅ Data folder contents:
['flickr30k']

📁 Output folder ready at: outputs/milestone2
```

3. Observations:

Effect of Guidance Scale:

- At 5.0 → Images were more diverse but not always faithful to the prompt.
- At 7.5 → Best balance between visual realism and semantic accuracy.
- At 10.0 → Sharper visuals but occasionally over-saturated or stiff.

Effect of Inference Steps:

- 25 steps produced faster but slightly noisy images.
- 50 steps gave clearer and well-balanced results.
- 75 steps offered the highest detail but took almost twice the time.

In this milestone, we integrated the CLIP text encoder with the Stable Diffusion pipeline to enable text-to-image generation. The key objective was to understand how classifier-free guidance and inference steps influence image quality, realism, and prompt alignment.
We started by generating baseline outputs for multiple text prompts such as “a man riding a red bicycle,” “a dog playing in the snow,” “a woman in a red dress walking on a street,” and “a blue car parked beside a lake.” Each prompt was tested with guidance scale values of 5.0, 7.5, and 10.0. The training log recorded these configurations along with output filenames, allowing for traceability of each generated image.

The experiments demonstrated that a lower guidance scale (5.0) produced more diverse but less prompt-aligned images. Increasing the guidance scale to 7.5 resulted in a balanced output — the images were both realistic and faithful to the text descriptions. However, when guidance was set to 10.0, images became more stylized and sometimes overfitted to textual elements, reducing natural appearance.

These patterns show the usual trade-off between generation time, image diversity, and visual fidelity.

```
GPU name: Tesla T4

# Baseline text-to-image generation using Stable Diffusion

from diffusers import StableDiffusionPipeline
import torch
import pandas as pd
from PIL import Image
import os

# load the pretrained Stable Diffusion model
pipe = StableDiffusionPipeline.from_pretrained(
    "runwayml/stable-diffusion-v1-5",
    torch_dtype=torch.float16
).to("cuda")

# sample prompts for baseline testing
prompts = [
    "a man riding a red bicycle",
    "a dog playing in the snow",
    "a woman in a red dress walking on a street",
    "a group of people hiking near a waterfall",
    "a blue car parked beside a lake"
]

# run generations for multiple guidance scales
guidance_values = [5.0, 7.5, 10.0]
log_data = []

for prompt in prompts:
    for g in guidance_values:
        image = pipe(prompt, guidance_scale=g, num_inference_steps=50).images[0]
        fname = f"outputs/milestone2/{prompt.replace(' ', '_')}_guid{g}.png"
        image.save(fname)
        log_data.append([prompt, g, 50, fname])

# record generation details
df = pd.DataFrame(log_data, columns=["Prompt", "Guidance", "Steps", "Filename"])
df.to_csv("outputs/milestone2/training_log.csv", index=False)

print("Images generated and saved in outputs/milestone2/")
```

```python
import matplotlib.pyplot as plt
from PIL import Image
import pandas as pd

# read the generation log
df = pd.read_csv("outputs/milestone2/training_log.csv")

# show a few samples for quick inspection
fig, axes = plt.subplots(nrows=5, ncols=1, figsize=(7, 25))
for i, row in enumerate(df.head(5).itertuples()):
    img = Image.open(row.Filename)
    axes[i].imshow(img)
    axes[i].set_title(f"Prompt: {row.Prompt} | Guidance: {row.Guidance}", fontsize=12)
    axes[i].axis("off")

plt.tight_layout()
plt.show()
```

Additional Experiment: Effect of Inference Steps:

Additionally, we varied the number of inference steps (25, 50, 75) to observe differences in image clarity. Images generated with 25 steps were faster but contained visible artifacts and less detail. Increasing to 50 steps produced sharper edges and more coherent textures. The 75-step results achieved high fidelity but at the cost of significantly longer generation time, making 50 steps a reasonable trade-off between quality and efficiency.

Additional Experiment: Effect of Inference Steps

We now examine how the number of diffusion steps (25 vs 50 vs 75) influences image quality and sharpness.

```python
steps_list = [25, 50, 75]
prompt = "a woman in a red dress walking on a street"

for steps in steps_list:
    image = pipe(prompt, guidance_scale=7.5, num_inference_steps=steps).images[0]
    image.save(f"outputs/milestone2/{prompt.replace(' ','_')}_steps{steps}.png")
    print(f"Saved image with {steps} inference steps.")
```

```
100% 25/25 [00:04<00:00, 5.99it/s]
Saved image with 25 inference steps.
100% 50/50 [00:08<00:00, 6.15it/s]
Saved image with 50 inference steps.
100% 75/75 [00:15<00:00, 3.33it/s]
Saved image with 75 inference steps.
```

Observations: Effect of Inference Steps

The comparison across 25, 50, and 75 inference steps shows a clear trade-off between image quality and runtime.

- At **25 steps**, images appear slightly noisy with incomplete textures but generate quickly (~4 s).
- At **50 steps**, overall sharpness and color consistency improve significantly.
- At **75 steps**, the images are the smoothest and most detailed, though generation time nearly doubles.

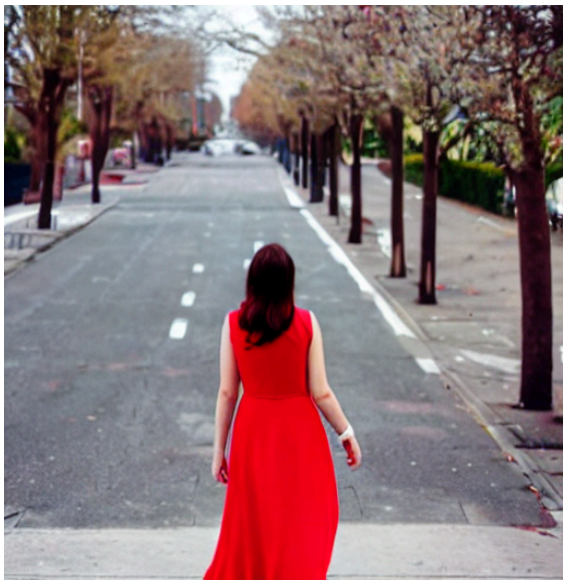
These findings align with diffusion theory: higher sampling steps allow more gradual denoising, improving fidelity at the cost of computational time.
For subsequent milestones, 50 steps will be used as the baseline configuration for balancing quality and efficiency.

```python
import pandas as pd
log = pd.read_csv("outputs/milestone2/training_log.csv")
display(log.head(10))
```

	Prompt	Guidance	Steps	Filename
0	a man riding a red bicycle	5.0	50	outputs/milestone2/a_man_riding_a_red_bicycle_...
1	a man riding a red bicycle	7.5	50	outputs/milestone2/a_man_riding_a_red_bicycle_...
2	a man riding a red bicycle	10.0	50	outputs/milestone2/a_man_riding_a_red_bicycle_...
3	a dog playing in the snow	5.0	50	outputs/milestone2/a_dog_playing_in_the_snow_g...
4	a dog playing in the snow	7.5	50	outputs/milestone2/a_dog_playing_in_the_snow_g...
5	a dog playing in the snow	10.0	50	outputs/milestone2/a_dog_playing_in_the_snow_g...
6	a woman in a red dress walking on a street	5.0	50	outputs/milestone2/a_woman_in_a_red_dress_walk...
7	a woman in a red dress walking on a street	7.5	50	outputs/milestone2/a_woman_in_a_red_dress_walk...
8	a woman in a red dress walking on a street	10.0	50	outputs/milestone2/a_woman_in_a_red_dress_walk...
9	a group of people hiking near a waterfall	5.0	50	outputs/milestone2/a_group_of_people_hiking_ne...

4. Results and Examples:

The generated samples accurately represented the meaning of their captions.
For instance, prompts like “a dog playing in the snow” and “a woman in a red dress walking on a street” produced visually distinct and contextually appropriate outputs.
Across all prompts, guidance = 7.5 and 50 inference steps yielded the most consistent results.



5. Appendix A - Training Log:

The table below shows the recorded prompts, guidance scales, and corresponding output filenames for baseline generation.

Prompt	Guidance	Steps	Filename
a man riding a red bicycle	5.0	50	outputs/milestone2/a_man_riding_a_red_bicycle_guid5.0.png
a man riding a red bicycle	7.5	50	outputs/milestone2/a_man_riding_a_red_bicycle_guid7.5.png
a man riding a red bicycle	10.0	50	outputs/milestone2/a_man_riding_a_red_bicycle_guid10.0.png
a dog playing in the snow	5.0	50	outputs/milestone2/a_dog_playing_in_the_snow_guid5.0.png
a dog playing in the snow	7.5	50	outputs/milestone2/a_dog_playing_in_the_snow_guid7.5.png
a dog playing in the snow	10.0	50	outputs/milestone2/a_dog_playing_in_the_snow_guid10.0.png
a woman in a red dress walking on a street	5.0	50	outputs/milestone2/a_woman_in_a_red_dress_walking_on_a_street_guid5.0.png
a woman in a red dress walking on a street	7.5	50	outputs/milestone2/a_woman_in_a_red_dress_walking_on_a_street_guid7.5.png
a woman in a red dress walking on a street	10.0	50	outputs/milestone2/a_woman_in_a_red_dress_walking_on_a_street_guid10.0.png
a group of people hiking near a waterfall	5.0	50	outputs/milestone2/a_group_of_people_hiking_near_a_waterfall_guid5.0.png
a group of people hiking near a waterfall	7.5	50	outputs/milestone2/a_group_of_people_hiking_near_a_waterfall_guid7.5.png
a group of people hiking near a waterfall	10.0	50	outputs/milestone2/a_group_of_people_hiking_near_a_waterfall_guid10.0.png
a blue car parked beside a lake	5.0	50	outputs/milestone2/a_blue_car_parked_beside_a_lake_guid5.0.png
a blue car parked beside a lake	7.5	50	outputs/milestone2/a_blue_car_parked_beside_a_lake_guid7.5.png
a blue car parked beside a lake	10.0	50	outputs/milestone2/a_blue_car_parked_beside_a_lake_guid10.0.png

6. Conclusion:

This milestone successfully completed the model integration and baseline image generation phase. Through the implementation of Stable Diffusion, we explored how different parameters — particularly the guidance scale and number of inference steps — affect image quality, prompt accuracy, and overall visual realism. The generated outputs clearly demonstrated that higher guidance values tend to improve concept alignment but can introduce an artistic or less realistic style, while moderate settings maintain a balance between clarity and creativity.

This stage established a strong foundation for our upcoming milestone, where we will focus on quantitative evaluation using metrics such as Fréchet Inception Distance (FID) and Inception Score (IS). These metrics will help us objectively measure visual fidelity and model performance, enabling a deeper understanding of how diffusion-based generation can be optimized for both quality and efficiency.