Exploring Classifier-Free Guidance

Introduction to Machine Learning Final Project

Dhruman Gupta   Rushil Gupta

May 6, 2025

Table of Contents

Motivation

- In the past few years, diffusion models have gained popularity as a powerful generative modeling technique.
- These models can be trained on samples from a distribution and can generate new samples from that distribution.
  - So, given $D \sim P(x)$, we can sample $x_0 \sim P(x)$
- However, a natural question, and a key challenge, is can we sample from the conditional distribution $p(x_0 \mid y)$?

Applications of Conditional Generation

If we are able to sample from the conditional distribution, we can do a lot of things:

- Text to Image / Video Generation
- Text to Audio Generation
- Image Inpainting / Restoration

Table of Contents

Background: Diffusion Models

- Diffusion models are a class of generative models that iteratively add noise to data and then learn to reverse this process.
- Idea: add noise to an image, then learn to reverse the noise process to generate images.

Forward Process

- The forward process of adding noise is defined as:

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I\right), \quad t = 1, \ldots, T.$$

  Where x_0 is the original image, x_T is pure noise, and β_t is a variance schedule.

- In the reverse process, we want to learn $p(x_{t-1} \mid x_t)$.
  - If we are able to, then we can sample $x_T \sim \mathcal{N}(0, 1)$ and then iteratively sample from $p(x_{t-1} \mid x_t)$ to get x_0.

Reverse Process

- Learn to reverse noising via a neural network:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\big(x_{t-1};\, \mu_\theta(x_t, t),\, \sigma_t^2 I\big).$$

- Using score prediction, parameterize mean directly:

$$\mu_\theta(x_t, t) = x_t + \sigma_t^2 s_\theta(x_t, t),$$

where $s_\theta(x_t, t)$ is the neural network that predicts the score function $\nabla_{x_t} \log p(x_t)$.

Training Objective

- Score matching loss:

$$\mathcal{L}(\theta) = \mathbb{E}_{x_0,t}\left[\left\|s_{\text{true}}(x_t, x_0, t) - s_\theta(x_t, t)\right\|^2\right],$$

where $x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1 - \bar{\alpha}_t}\, \epsilon$.

- This means the network s_θ directly learns to predict the score of the noisy distribution.

Unconditional Sampling Algorithm

$x_T \sim \mathcal{N}(0, I)$
for $t = T, \ldots, 1$ **do**
    $s \leftarrow s_\theta(x_t, t)$ {Neural network predicts score directly}
    $\mu \leftarrow x_t + \sigma_t^2 s$
    $x_{t-1} \sim \mathcal{N}(x_{t-1}; \mu, \sigma_t^2 I)$
end for
return x_0

Table of Contents

Classifier Guidance: Concept

- Now, we have a model that can generate samples from $p(x_0)$.
- We want to be able to sample from $p(x_0 \mid y)$.
- How do we do that?

Classifier Guidance: Concept

- Let's say we want to sample from $p(x_0 \mid y)$.
- Update the reverse process to calculate the score of the condition y:

$$\nabla_{x_t} \log p(x_t) \to \nabla_{x_t} \log p(x_t \mid y)$$

$$\nabla_{x_t} \log p(x_t \mid y) = \nabla_{x_t} \log p(x_t) + \nabla_{x_t} \log p(y \mid x_t).$$

- The first term is exactly what diffusion models do. The second term is the score of the condition.

Classifier Guidance: Math

- How do we get the $\nabla_{x_t} \log p(y \mid x_t)$ term?
- We can train a classifier $p_\phi(y \mid x_t)$ to predict the condition y given the noisy image x_t.
- This was state-of-the-art in 2021, and significantly improved the quality of generated samples.

Table of Contents

Classifier-Free Guidance: Idea

- Having an auxiliary classifier is not always feasible, and it can lead to instability and mode collapse.
- We want to be able to have a more harmonious model that can be steered by a condition.
- How can we have an entirely generative model do this?

Classifier-Free Guidance: Concept

- Recall, our goal is to estimate $\nabla_{x_t} \log p(x_t \mid y)$.
- Currently, our model learns $\nabla_{x_t} \log p(x_t)$.
- This is the same as conditioning on nothing: $\nabla_{x_t} \log p(x_t \mid \varnothing)$.
- So our unconditional model is actually a special case of a conditional model.

Classifier-Free Guidance: Concept

- Let $s_\theta(x_t, t, c) \approx \nabla_{x_t} \log p(x_t \mid c)$.
- Train the diffusion model to predict both:

$$s_\theta(x_t, t, c) \quad \text{and} \quad s_\theta(x_t, t, \varnothing)$$

  by randomly dropping the condition c during training.
- At inference, combine:

$$s_{\text{CFG}} = (1 + w)\, s_\theta(x_t, t, c) - w\, s_\theta(x_t, t, \varnothing),$$

  where w is the guidance weight.
- **Advantage:** no extra classifier, single unified model.

CFG Sampling Algorithm

$x_T \sim \mathcal{N}(0, I)$
for $t = T, \ldots, 1$ **do**
  $s_c \leftarrow s_\theta(x_t, t, c)$
  $s_u \leftarrow s_\theta(x_t, t, \varnothing)$
  $s \leftarrow (1 + w)\, s_c - w\, s_u$
  $\mu \leftarrow x_t + \sigma_t^2 s$
  $x_{t-1} \sim \mathcal{N}(x_{t-1}; \mu, \sigma_t^2 I)$
end for
return x_0

Table of Contents

Implementation Overview

- We used the pre-trained VAE from stable diffusion v1.4 to encode and decode images.
- We used the pre-trained CLIP text encoder to encode the text prompts.
- The diffusion model is a UNet with 4 down/up blocks and 960 channels.

What Is a UNet?

- **Purpose:**
  - Designed for image-to-image tasks (e.g., denoising, segmentation).
  - Learns to reverse a corruption process by progressively refining features.
- **Key Idea:**
  - Combines high-resolution spatial information with deep, coarse features.
  - The idea is to capture both local and global context.
- **Origin:**
  - First introduced for biomedical image segmentation (Ronneberger et al., 2015).

UNet Architecture Overview

- **Encoder (Contracting Path):**
  - Stacks of Conv→ReLU→Conv→ReLU.
- **Bottleneck:**
  - deepest layer; captures the most abstract features.
  - No pooling—just convolutions.
- **Decoder (Expanding Path):**
  - Upsamples (transposed conv) to restore spatial size, halves channels each step.
- **Skip Connections:**
  - Concatenate encoder feature maps to decoder at each level.
  - Preserve fine-grained details lost during downsampling.

UNet Architecture Overview

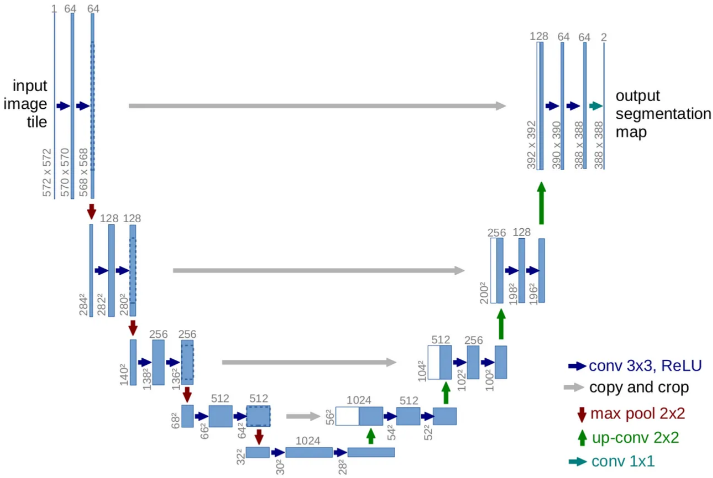

Figure: UNet Architecture

Data Pipeline & Caption Dropout

- Dataset: MS COCO 2014 captions
  - Contains 200K images having annotations for object detection, segmentation, and captioning
  - Comprises of 80 categories including common objects like cars, bicycles, and animals, as well as more specific categories such as umbrellas, handbags, and sports equipment
- Images resized to 128×128 and center-cropped

Training Details

- The diffusion model's process is set to 1000 timesteps.
- Training consisted of 450 epochs.

Inference & Sampling

- Sweep guidance weights $w \in \{1, 3, 5, 7\}$
- Fixed seed for consistent comparisons
- Generate 3×3 grids per prompt and weight

Table of Contents

Results

- We train the model for 450 epochs.
- Results are not the most impressive, since we are using a small model and dataset - due to compute constraints.

Results - Snowy Mountain

(a) $w = 1.0$ (b) $w = 3.0$ (c) $w = 5.0$ (d) $w = 7.0$

Figure: "a beautiful snowy mountain landscape" with different guidance weights

(a) $w = 1.0$ (b) $w = 3.0$ (c) $w = 5.0$ (d) $w = 7.0$

Figure: "a beach" with different guidance weights

Results - Tennis Court

(a) $w = 1.0$ (b) $w = 3.0$ (c) $w = 5.0$ (d) $w = 7.0$

Figure: "a tennis court" with different guidance weights

Table of Contents

Discussion

- Increasing guidance weight enhances prompt adherence but reduces sample diversity, leading to mode collapse at high w.
- Due to our small dataset and limited model parameters, high guidance weights can introduce artifacts.
- Failure cases on novel objects highlight dataset limitations: COCO lacks sufficient examples of uncommon items.
- The chosen UNet size balances capacity and compute but may underfit complex scenes.

Table of Contents

Future Work

- Expand the training corpus with larger captioned datasets (e.g., OpenImages, LAION) to improve object diversity.
- Use parameter-efficient fine-tuning (LoRA) to enable larger backbone models under compute constraints.
- Explore alternative noise schedules and fewer timesteps to accelerate training and improve image sharpness.

Table of Contents

Conclusion

- We implemented classifier-free guidance in a custom diffusion model and systematically explored its qualitative impact.
- CFG offers a simple yet powerful control knob for balancing fidelity and diversity.
- Our findings underscore the importance of dataset scale and model capacity for reliable generative performance.

Table of Contents

Bibliography

Sohl-Dickstein, Jascha, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli.
Deep Unsupervised Learning using Nonequilibrium Thermodynamics.
In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel.
Denoising Diffusion Probabilistic Models.
In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Dhariwal, Prafulla and Alex Nichol.
Diffusion Models Beat GANs on Image Synthesis.
In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Ho, Jonathan and Tim Salimans.
Classifier-Free Diffusion Guidance.
In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.