Diffusion Model Report

Aajay Devaraj
2021001

Dhvanil Sheth
2021040

Divyansh Mishra
2021042

Priyash Shah
2021553

December 2023

1 Introduction

This report provides a comprehensive overview of diffusion models, a novel class of generative models that have shown promising results in synthesizing high-quality images and other complex data modalities. These models are based on principles from nonequilibrium thermodynamics and variational inference, representing a significant departure from traditional generative approaches like GANs and VAEs.

2 Methodology

Diffusion models operate by simulating a forward process that gradually adds noise to data, transforming the data distribution into a Gaussian distribution. The reverse process, parameterized by a neural network, learns to denoise this data, effectively reversing the diffusion process to generate samples from the original data distribution.

3 Model Architecture

The architecture of diffusion models typically involves deep neural networks, such as U-Nets, which are trained to parameterize the reverse process of the diffusion. These networks learn to predict the noise added at each step of the forward process and use this information to denoise and generate data samples in the reverse process.

4 Mathematical Formulation

The mathematical formulation of diffusion models involves two key processes: the forward diffusion process and the reverse denoising process.

4.1 Forward Process

The forward process is a Markov chain that adds Gaussian noise to the data over several steps. It can be described as:

$$x_t = \sqrt{1-\beta_t} x_{t-1} + \sqrt{\beta_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I) \tag{1}$$

where β_t are variance schedules and ϵ is Gaussian noise.

4.2 Reverse Process

The reverse process involves a neural network predicting the noise added in the forward process and denoising the data:

$$x_{t-1} = \frac{1}{\sqrt{1-\beta_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t) \right) \tag{2}$$

where $\epsilon_\theta(x_t, t)$ is the noise predicted by the neural network and $\alpha_t = 1 - \beta_t$.

4.3 Training Objective

The model is trained to minimize a variational lower bound on the data log-likelihood, encouraging accurate noise estimation at each diffusion step.

5 Training and Implementation

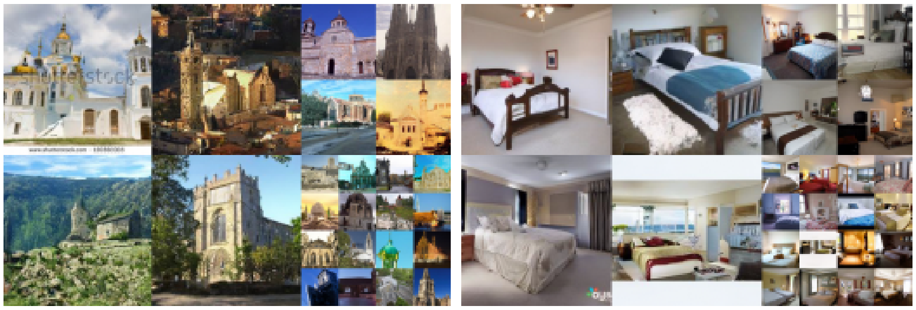
Training involves optimizing the variational bound using stochastic gradient descent. The forward process variances can be learned or fixed, and the reverse process is trained to accurately predict and remove the noise added during the forward process.

6 Applications and Results

Diffusion models have been successfully applied in high-quality image synthesis, achieving impressive results on datasets like CIFAR-10 and LSUN, and demonstrating potential in various other data modalities.

7 Limitations and Future Directions

While diffusion models excel in sample quality, they face challenges in terms of computational efficiency and log-likelihood performance compared to other generative models. Future research may focus on improving these aspects and exploring new applications.



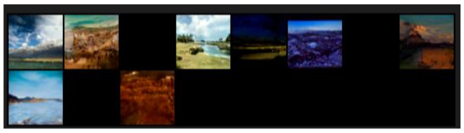
8 Conclusion

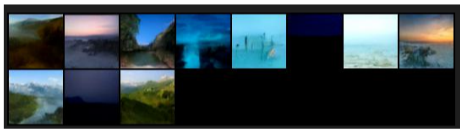
Diffusion models represent a significant advancement in generative modeling, offering a unique approach to data generation through the process of reversing diffusion. Their ability to synthesize high-quality data holds great promise for future developments in the field.

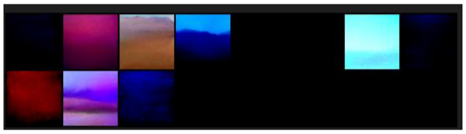
9 Model Comparison

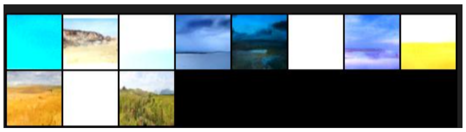
Let's get a quick comparison among images generated by these two models.

9.1 Images generated by Diffusion Model









9.2 Images generated by GAN Model

9.2.1 Observations

- Diffusion model was able to generate high-quality images in comparison to GAN models.
- Training diffusion model was computationally expensive compared to GAN models.
- Slight changes in training data caused the GAN model to generate a completely different image every time, making it sensitive to training data.
- We stuck to parameters provided by the diffusion model paper and got stable output.
- On changing the hyperparameters, the training became unstable for the Diffusion model, which wasn't the case for GAN models.

10 Theoretical Learnings

While exploring the theoretical aspects of Diffusion Models, we went through a few Research Papers, with the primary 4 (in chronological format) being:

1. Deep Unsupervised Learning using Nonequilibrium Thermodynamics
2. Denoising Diffusion Probabilistic Models
3. Improved Denoising Diffusion Probabilistic Models
4. Diffusion Models Beat GANs on Image Synthesis

The essential idea of diffusion models is to systematically and slowly destroy the data distribution through an iterative forward diffusion process. We then learn a reverse diffusion process that restores structure in data, yielding a highly flexible and tractable generative model of the data.

The DDPM paper from 2020 laid out three things the neural network could predict:

1. Predicting the mean of the noise at each time step
2. Predicting the original image directly
3. Predicting the noise in the image directly

Predicting the original image won't work well, so we can ignore this choice, and the first and the third option are the same just parametrized differently, and all authors decided to go with the third one.

Initially, we were only learning the mean as in the first option, and the variance was fixed. However, in the following papers by OpenAI authors, we decided that learning the variance leads to improvement in the log likelihoods.

The amount of noise added to the image is regulated by a schedule. A common schedule was the linear schedule, which was too rapid, hence some information was getting destroyed, and so modern models use a cosine schedule to make this step more gradual.

The first papers were using a UNet architecture and then the second paper by the openAI authors decided to make a few major changes to this architecture, which heavily improved the overall outcome. The updates that they made were that first of all they increased the Depth of the network and decreased the width then, they included more attention blocks than the original proposal and also increased the number of attention heads. They also took the residual blocks from big GAN and used this for the upsampling and downsampling blocks. Next they proposed what they call adaptive group normalization which is just a fancy name for the idea of incorporating the time step slightly differently and additionally also the class of the label.

11 Additional Learning

11.0.1 Server Setup

Due to several obstructions, we weren't able to run our models on a server that was already configured. Thus we had to request a separate server, which we configured from scratch. We were able to create environments necessary for running our models. Although we encountered endless problems while configuring the server, we learned a lot about how everything worked.

11.0.2 Paper Reading

We implemented all of our models by thoroughly studying academic papers and examining codebases from other developers. Reflecting on our journey, we acknowledge that we began as novices, but through the course of this project, we have elevated our skills beyond that initial stage. This experience has been instrumental in our growth, and we now find ourselves at a level of proficiency that surpasses our initial status as beginners.

References

1. https://arxiv.org/abs/1511.06434
2. https://pytorch.org/tutorials/beginner/dcgan$_{f}aces_{t}utorial.html$
3. https://github.com/dome272/Diffusion-Models-pytorch
4. https://arxiv.org/abs/2006.11239
5. https://arxiv.org/abs/2209.00796
6. https://www.youtube.com/watch?v=HoKDTa5jHvgt=580s
7. https://www.youtube.com/watch?v=fbLgFrlTnGU