DENOISING DIFFUSION PROBABILISTIC MODELS

INTRODUCTION TO PROBABILISTIC GRAPHICAL MODELS AND DEEP GENERATIVE MODELS

Antonin Barbe *, Mathias Grau †

[1]M2 MVA, Ecole Normale Supérieur de Paris Saclay, Gif-su-Yvette, France

*antonin.barbe@polytechnique.edu, †mathias.grau@polytechnique.org

CONTEXT

Generative modeling has been a foundational area in machine learning research, facilitating the creation of realistic data distributions for various applications, including image synthesis and text generation. Traditional methodologies such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) have demonstrated significant advancements; however, they are accompanied by inherent limitations.

In 2020, the groundbreaking work titled "*Denoising Diffusion Probabilistic Models*" (DDPM) (Ho; Jain; Abbeel, 2020) introduced a novel generative modeling paradigm centered on iterative denoising, commonly referred to as diffusion models. These models deviate from conventional approaches by conceptualizing the data generation process as the inverse of a gradual noise-application mechanism, termed the forward process. By employing parameterized Gaussian transitions, this probabilistic framework effectively bridges the forward and reverse processes, offering a principled methodology for generating high-fidelity samples.

PROBABILISTIC DIFFUSION MODELS

Diffusion models are latent variable models of the form

$$p_\theta(x_0) := \int p_\theta(x_{0:T})\, dx_{1:T},$$

where $x_1, \dots, x_T$ are latent variables with the same dimensionality as the data $x_0 \sim q(x_0)$. The joint distribution $p_\theta(x_{0:T})$, known as the reverse process, is defined as a Markov chain with learned Gaussian transitions starting from a prior $p(x_T) = \mathcal{N}(x_T; 0, I)$:

DENOISING PROCESS (BACKWARD PROCESS

$$p_\theta(x_{0:T}) := p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t),$$
$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \quad (1)$$

What sets diffusion models apart from other latent variable models is the approximate posterior $q(x_{1:T}|x_0)$, referred to as the forward process or diffusion process. This process is fixed as a Markov chain that progressively adds Gaussian noise to the data x_0 following a variance schedule $\beta_1, \dots, \beta_T$:

DIFFUSION PROCESS (FORWARD PROCESS)

$$q(x_{1:T}|x_0) := \prod_{t=1}^{T} q(x_t|x_{t-1}),$$
$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I). \quad (2)$$

Using the notation $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$, we can write:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}\, x_0, (1-\bar{\alpha}_t) I). \quad (3)$$

The forward process posterior $q(x_{t-1}|x_t, x_0)$ is Gaussian (very useful for the future optimization)

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I) \quad (4)$$

with mean and variance :

$$\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t, \quad \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

EVIDENCE LOWER BOUND (ELBO)

The evidence lower bound (ELBO) is a variational approach to optimizing the model distribution $q(x)$. Starting from the log evidence, we have the ELBO :

$$\log p_\theta(x_0) \geq \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right] = \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log p(x_T) + \sum_{t \geq 1} \log \frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})} \right]$$

DECOMPOSING THE ELBO

The ELBO can be decomposed into three terms:

$$\text{ELBO} = \underbrace{D_{\mathsf{KL}}(q(x_T|x_0) \parallel p(x_T))}_{\mathcal{L}_T} + \sum_{t>1} \underbrace{D_{\mathsf{KL}}(q(x_{t-1}|x_t, x_0) \parallel p_\theta(x_{t-1}|x_t))}_{\mathcal{L}_{t-1}} - \underbrace{\log p_\theta(x_0|x_1)}_{\mathcal{L}_0}$$

$$\text{ELBO} = \mathcal{L}_T + \sum_{t=1}^{T-1} \mathcal{L}_t - \mathcal{L}_0, \quad (5)$$

- $\mathcal{L}_T$, which compares the prior $p(x_T)$ and the forward process posterior $q(x_T|x_0)$,
- $\mathcal{L}_{t-1}$, which compares $q(x_{t-1}|x_t, x_0)$ (forward posterior) with the reverse process model $p_\theta(x_{t-1}|x_t)$
- $\mathcal{L}_0$, a reconstruction term for x_0 given x_1.

OPTIMIZATION OF THE ELBO

- $\mathcal{L}_T$: The authors use β_t as constant values. The approximate posterior $q(x_{1:T}|x_0)$ has no learnable parameters, so $\mathcal{L}_T$ is a constant during training.
- $\mathcal{L}_{t-1}$: Both terms are Gaussians, the optimisation is :

$$\mathcal{L}_{t-1} = \mathbb{E}_q \left[\frac{1}{2\sigma_t^2} \|\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2 \right] + C \quad (6)$$

  Optimization process can be straightforward : μ_θ predicts $\tilde{\mu}_t$. But, they reparametrize the optimization for the noise because : $x_t(x_0, \epsilon) = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$. So ϵ_θ is a function approximator designed to predict ϵ. We get :

$$\mathbb{E}_{x_0, \epsilon} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2 \right] \quad (7)$$

  And the sampling simplify of $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$ involves computing: $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z$
- $\mathcal{L}_0$: Data Scaling, Reverse Process Decoder. The images are integers in $\{0,1,\dots,255\}$ scaled linearly to $[-1,1]$. They trick to make it a discrete decoder without variance for the last timestep $t = 0$.

SIMPLIFIED LOWER BOUND TO OPTIMIZE

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{x_0, \epsilon, t} \left[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2 \right] \quad (8)$$

TRAINING

repeat
  $x_0 \sim q(x_0)$
  $t \sim \text{Uniform}(\{1, \dots, T\})$
  $\epsilon \sim \mathcal{N}(0, I)$
  Gradient : $\nabla_\theta \left(\epsilon - \epsilon_\theta \left(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t \right) \right)$
until converged

SAMPLING

$x_T \sim \mathcal{N}(0, I)$
for $t = T, \dots, 1$ **do**
  $z \sim \mathcal{N}(0, I)$ if $t > 1$, else $z = 0$
  $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z$
end for
return x_0

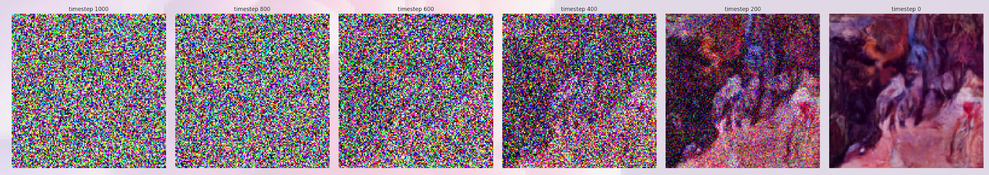
IMPLEMENTATIONS AND RESULTS

Training on 2287 impressionist paintings from (WikiArt, 2024) with Hugging Face Diffusers library. (Code available on our GitHub).

Figure 1 : Generation $T = 1000$, $\beta_1 = 10^{-4}$, $\beta_T = 0.02$ and U-Net

Figure 2 (right): Interpolations varying the number of diffusion steps prior to latent mixing

Figure 3 : All steps of denoising



IDEAS FOR IMPROVEMENT

- Introduce conditions on the generation : Conditional Score-based Diffusion Models, Tashiro et al. (2021)
- Improve training : Immiscible Diffusion with Noise Assignment, Li et al. (2024)
- Use VAE to diffuse on the latent space, improving image quality : High-Resolution Image Synthesis with Latent Diffusion Models, Rombach et al. (2022)
- Learned variance for denoising : Diffusion Models With Learned Adaptive Noise, Sahoo et al. (2024)

CONCLUSION

We managed to implement (Ho; Jain; Abbeel, 2020) diffusion model and train it on a new impressionist dataset. This article was one of the first in therm of DDPM, paving the way to many improvement regarding sampling efficiency, conditioning, Multi-Task Learning, and decreasing computational cost of diffusion models.

REFERENCES

HO, Jonathan; JAIN, Ajay; ABBEEL, Pieter. **Denoising Diffusion Probabilistic Models**. [*S. l.: s. n.*], 2020. arXiv: 2006.11239 [cs.LG]. Disponível em: https://arxiv.org/abs/2006.11239.

WIKIART. **WikiArt Dataset**. [*S. l.: s. n.*], 2024. Accessed: 2024-11-25. Disponível em: https://www.wikiart.org/.

ACKNOWLEDGEMENT

Professors Pierre Latouche and Pierre-Alexandre Mattei from UCA, Ecole Polytechnique and INRIA

AUTHORS

Antonin Barbe and Mathias Grau