Denoising Diffusion Probabilistic Model

Aman Patel

28/03/24

1 Introduction

Over the past few years, generative models have gained an immense ability to generate human-like natural text, and now it comes to the generation of high quality synthetic images through the concept of diffusion models. But wait, Why Diffusion? Infact, What is Diffusion? The first thing that comes to mind about diffusion is the movement of random particles from high concentration to low concentration. The models here work on a similar concept, differences here being - a) the random particles here are noise tensors generated from standard normal distribution and b) their movement is quantized and structured whether being added or subtracted on a given image tensor at a particular time.

Diffusion models are a class of likelihood-based models which has shown to produce high-quality images while offering desirable properties and easy scalability. A Diffusion is a probabilistic model, that means it is parametrized by Markov chains that rely on simple concept that the present state is dependent on past few states.

Diffusion starts with a forward diffusion process, a sequence of deterministic steps (does not requie any training of neural networks) that gradually adds noise to our original data (e.g image) in a controlled manner. The denoising here refers to reversing a gradual noising process i.e removing noise from a more noised image to a less noised image, conditioned on the original image. But then an obvious question comes up - what's the novelty of diffusion

process as it is just adding and removing the noise in subsequent processes? Also, where's the new generation of images happening? It comes after the training of reverse diffusion process - that's what denoising refers to, and we will discuss about it in detail in later sections.

2 Model

Diffusion models are latent variable models that is the representation of intermediate states of same dimensionality, and x_0 is gradually corrupted with noise to produce a sequence of noisy states x_1,...,x_T and similarly in reverse diffusion process(denoising process), the model is trained to predict sequence of latent variables x_{t-1}, x_{t-2},..., x_0 from highly corrupted state x_T and with **conditioning** x_0. These predicted latent variables represent the intermediate denoised states that the model goes through to recover the original data x_0 from the noisy input x_T. The main challenges here are to choose the variance scheduling β_t, model architecture and Gaussian distribution of the reverse process.

2.1 Forward Diffusion Process

It is a deterministic sequence of steps where goal is to gradually corrupt the original data x_0 with increasing levels of gaussian noise. But why gradually? Why not just add the whole noise in one timestep and make it a fully corrupted image? The key reason is that gradual change ensures smooth transition over sequence of steps which is crucial for denoising (reverse diffusion) part as model should go through enough iterations to learn the underlying data distribution to differentiate between original high quality image and noisy version x_t, and accordingly remove the noise to get a better view and repeat the process until we get our clear image x_0.

The simple equation that may govern adding more noise to x_{t-1} to get the next state x_t could be like-

$$x_t = \alpha x_{t-1} + \beta N(0, 1)$$

The above equation depicts that noise of constant variance β^2 is added in each step but it would take a large number of steps (experimentally, 1000) to get the image full of noise and after halfway, most of images are full of noise only, so it is a less efficient technique. Hence, in-practice we increase

the noise with time by linear or cosine scheduling(preferred) that increases our variance and we get to our final noised image x_t in fewer timesteps(10) and replacing α with $\sqrt{1-\beta_t}$ and β with $\sqrt{\beta_t}$, we get the corrupted image as standard normal distribution with mean 0 and variance 1, and that's where we start from in the denoising process.

$$x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}N(0,1)$$

Reiterating the above equation in terms of x_{t-1} and so on to get x_t in terms of our original image x_0, and replacing $\sqrt{1-\beta_t}$ with α_t, we get the most general equation on which forward diffusion processes rely.

$$x_t = \sqrt{\bar{\alpha}_t}x_{t-1} + \sqrt{1-\bar{\alpha}_t}N(0,1)$$

where, $\bar{\alpha}_t = \prod_{i=1}^{t}\alpha_t$ and N is the noise sampled from gaussian distribution; a more formal notation of markov chain that gradually adds a gaussian noise according to variance schedule β_1,β_2,...,β_t can be shown as,

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

2.2 Reverse Diffusion Process

The reverse diffusion process is typically modeled using a neural network, such as U-Net architecture or transformer based models, where the core training of diffusion model happen. It is responsible for learning to remove the noise from the corrupted data generated at each step, wherein underneath it learns the conditional probability distribution which represents the probability of previous less noisy state x_{t-1} , given the current noisy state and *original data* x_0. Below given is the main idea around which training revolves.

$$x_{t-1} = q(x_{t-1}|x_t, x_0)$$

Since we have the scheduling of β_t from forward diffusion process, it may seem - if we just keep on removing the same amount of noise as added, we will eventually get the clear original image. But that is not the case and here lies the challenge! Since everytime we sample a noise tensor(image), it is completely random and not the same as what was added - so doing that would just end up getting worse images, and model will not learning anything. What the model should actually learn is to examine the difference

of data distribution (i.e. values of pixels) between x_{t-1} and x_0 and try to predict the total noise from the initial image to that state , not only from x_t to x_{t-1} and accordingly remove some noise(β_t of that state is also provided). That is because the learning happens conditioned on the original image x_0, otherwise the model won't know what is the final image required and how much noise to remove.

2.3 Sampling and Generation of New Images

The key insight of new generation lies in learning the denoising process where the model implicitly learns the underlying data distribution of the clean images. During the generative step, sampling is performed starting from pure noise (e.g.Gaussian noise tensor) and iteratively applying the learnt reverse diffusion process *without* conditioning on any specific x_0. Since the initial pure noise tensor was completely random and not coherent to any sample of x_0, we provide any random sample of x_0 and model attempts to predict the noise and output ,given x_0, but since the noise tensor did not resemble the given x_0, it will generate a new image of a similar fashion (similar semantics but different spatial arrangement) with the same high quality as the model learnt by denoising process.

3 Why Diffusion?

Diffusion models have surpassed the performance of other generative models such as GANs, VAEs due to their ability to generate high quality samples across various domains. Diffusion beat GANs as they don't suffer through the issues of stable training which include problems of mode collapse and non-convergence. These issues in GANs refer to the situation where the Generator in GAN has limited subset of training examples leading to less diverse images and since it's a minimax game convergence is a issue which sometimes lead to poor quality of samples and diffusion handles this by progressive denoising process. Another major reason is explicit likelihood modelling that is diffusion models explicitly learns the underlying data distribution of high quality images in denoising process whereas GANs just generate a simple gaussian noise and transforms it to complex distribution through series of weights and predicts whether it is real or fake and backpropogates which is very less efficient. The 'posterior collapse' issue of VAE is that the encoder narrows down the input to a much lower dimensnion and becomes

solely dependent on decoder to reconstruct back from latent variables, where diffusion simply addresses this by skip connections between encoder and decoder networks in the U-Net architecture.

4 Future Scope of Diffusion models

Currently diffusion models have shown promising results in high-quality image generation, however their application can be extended to other modalities such as audio and video. For example, in case of audio generation of a particular voice, one could give a text prompt about what to speak and 20-30 seconds of audio of person's voice and the model could learn the frequency and tone of voice and could generate a whole text similar to his voice. The interpretability of diffusion models is one of the biggest scope of improvements if we can interpret the model's decisions by adding some specific information related to output. There's an interesting research ongoing in this domain called Control-Net which allows users to edit with desired styles, fonts and colors from provided SVG (Scalable Vector Graphics) output of diffusion models.

5 References

[1] Jonathan Ho, Ajay Jain, Pieter Abbeel. Denoising Diffusion Probabilistic models. arXiv:2006.11239v2 [cs.LG] 16 Dec 2020.

[2] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, Surya Ganguli. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. arXiv:1503.03585v8 [cs.LG] 18 Nov 2015.

[3] Prafulla Dhariwal, Alex Nichol. Diffusion models beats GANs on Image Synthesis. arXiv:2105.05233v4 [cs.LG] 1 Jun 2021.

[4] Kemal Erdem. Step by step Visual Introduction to Diffusion Models. medium.com.

[5] lilianweng.github.io/. What are diffusion models?