Generating Images using a Denoising Diffusion Probabilistic Model

Anonymous author

Abstract

This project uses a Denoising Diffusion Probabilistic Model to accomplish image generation in 32x32 over the CIFAR-10 dataset. The model makes use of techniques such as Classifier Free Guidance,Time Step Encoding and Self Attention.

1 Methodology

This project uses a Denoising Diffusion Probabilistic Model (DDPM). DDPMs are a type of Markov Chain with a forward process and a reverse process. The forward process gradually adds gaussian noise to an image according to a noise schedule $\beta_1, ...\beta_T$ where T is the number of timesteps in the diffusion process. This forward process is defined by a posterior $q(\mathrm{x}_{1:T}|\mathrm{x}_0)$. The following equations come from [1]

$$q(\mathrm{x}_{1:T}|\mathrm{x}_0) = \prod_{t=1}^{T} q(\mathrm{x}_T|\mathrm{x}_{t-1}),\ q(\mathrm{x}_t|\mathrm{x}_{t-1}) = \mathcal{N}(\mathrm{x}_t; \sqrt{1-\beta}\mathrm{x}_{t-1}, \beta_t\mathrm{I}) \tag{1}$$

Usefully we can go one step further with our posterior q by making a function that returns the noised image at time step t. To do this we introduce new variables $\alpha_t = 1 - \beta_t$ and $\hat{\alpha} = \prod_{s=1}^{t} \alpha_t$. After we introduce these variables we can derive this function.

$$q(\mathrm{x}_t|\mathrm{x}_0) = \mathcal{N}(\mathrm{x}_t;\ \sqrt{\hat{\alpha}_t}\mathrm{x}_0,\ (1-\hat{\alpha}_t)\mathrm{I}) \tag{2}$$

The ultimate goal of the diffusion model is to generate an image by removing noise from a completely noised image. It does this by learning the reverse process. The iterative removal of Gaussian noise with model p_θ can be defined as follows [1].

$$p_\theta(\mathrm{x}_{t-1}|\mathrm{x}_t) = \mathcal{N}(\mathrm{x}_{t-1}; \mu_\theta(\mathrm{x}_t, t), \Sigma_\theta(\mathrm{x}_t, t)) \tag{3}$$

The above equation Consists of two neural networks: μ_θ (a neural network to predict the mean of the normal distribution) and Σ_θ (to predict the variance of the normal distribution). In my particular implementation, I chose to use a pre-computed fixed linear noise schedule [1]. This means that we do not need to predict the variance as it is given and therefore makes Σ_θ redundant.

1.1 Learning The Model

For this model to learn we need to define a loss function for μ_θ in the original DDPM paper [1] this process involves using Bayes theorem and log rules on our forward and reverse process equations. The simplified version of this involves the quantity ϵ which refers to the actual noise and ϵ_θ which refers to the model's predicted noise. It then follows we obtain this quantity. $Loss = ||\epsilon - \epsilon_\theta(x_t, t)||^2$

1.2 Classifier Free Guidance (CFG)

As the cifar-10 set contains labeled images I chose to incorporate CFG into the model, this means the model learns the denoising process with the assistance of knowing what the image is of. This leads to an increase in image quality as seen in [2]. In my particular implementation I used a guidance strength of 3 which is what was used in an example in [2].

1.3 The U-Net

To facilitate the learning of the reverse process I used the U-Net architecture first introduces in [5], which consists of an encoder and decoder the encoder is a series of convolutional layers that reduce the spatial size of the input image while increasing the number of feature maps. The decoder is symmetrical to the encoder only it upsamples the feature maps while decreasing the depth. Both Encoder and Decoder use 'Up' and 'Down' blocks which are comprised of convolutional layers. The down blocks take an input tensor x as well as a time step t, which is delivered to the block as a sinusoidal position embedding as seen in [6].To achieve CFG the label is concatenated to the time step. The Up blocks take a skip connection from the adjacent down block as input too, this skip connection is concatenated with the Up-sampled input of the up block. In between each up and down block is a self-attention block. I adopted this method from [3] where the authors showed the introduction of self-attention in the U-Net for diffusion increases sample quality.

1.4 The U-Net Architecture

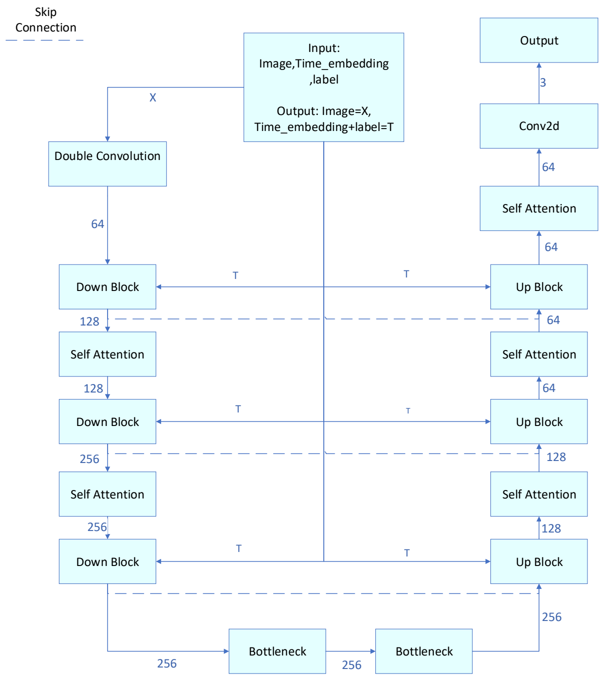


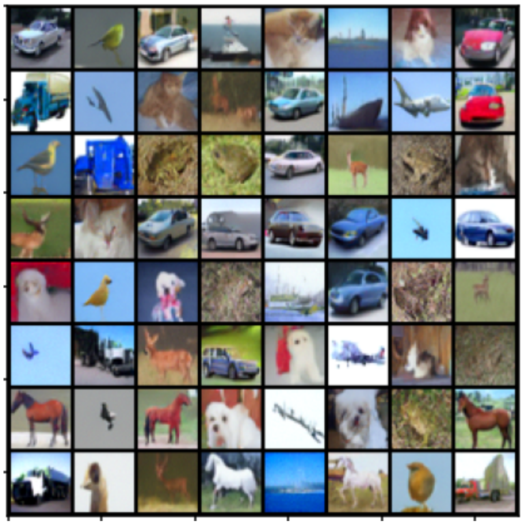
Figure 1: High-level U-Net architecture

2 Training

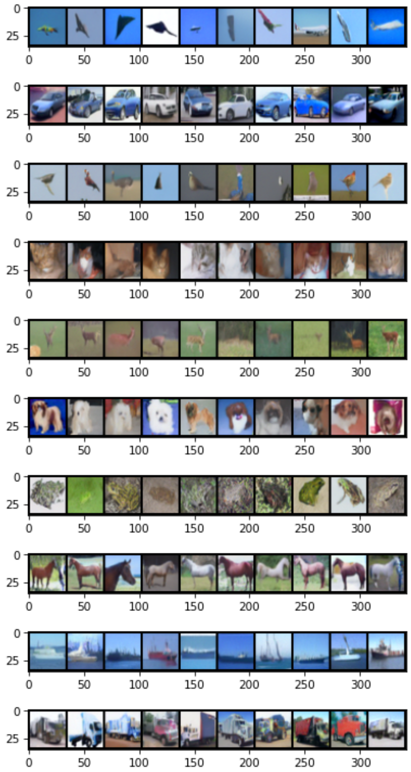
The model detailed was trained on the CIFAR-10 dataset at a 32x32 resolution for a total of 600 epochs. The time steps T for the diffusion process was set to 1000 the model was trained with Classifier free guidance 90% of the time as seen in [2]. The AdamW optimizer was implemented with betas from 0.9 to 0.999 and a learning rate of 0.001.

3 Results

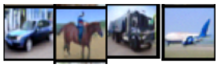
The resulting sample shows good image fidelity with very realistic renderings across many classes of CIFAR10, the only drawback would be some coloring issues e.g. the red horse.



The next results show images generated by interpolating between points in the latent space:



And here are some cherry-picked samples that show the best outputs the model has generated:



4 Limitations

As the results show the generated images from my model are realistic, but there are some inconsistencies in sampling like the wrong colouring being used in an image e.g. the red horse this issue is however not too common. Another problem with my model can be seen in the interpolation between points in the latent space. It seems nearby images have little similarity, therefore, couldn't show a smooth interpolation between two different points. In the future, I would like to adapt this model on a more high-resolution dataset such as STL-10 I was unable to do this for this project as the google colab GPU did not have enough memory to facilitate training at this resolution. I would also like to experiment using a different noise scheduling algorithm such as 'cosine noise scheduling' as this has shown to improve model performance as seen in [4].

Bonuses

This submission has a total bonus of -4 marks (a penalty), as it is trained only on CIFAR-10.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising Diffusion Models". In: (2020).

[2] Jonathan Ho and Tim Salimans. "Classifier-Free Diffusion Guidance". In: (2022).

[3] Susung Hong et al. “Improving Sample Quality of Diffusion Models Using Self-Attention Guidance”. In: (2022).

[4] Alex Nichol and Prafulla Dhariwal. “Improved Denoising Diffusion Probabilistic Models”. In: (2021).

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: (2015).

[6] Ashish Vaswani et al. “Attention Is All You Need”. In: (2017).