Iowa State University

Department of Computer Science

A Comparative Study of Image Generation Models

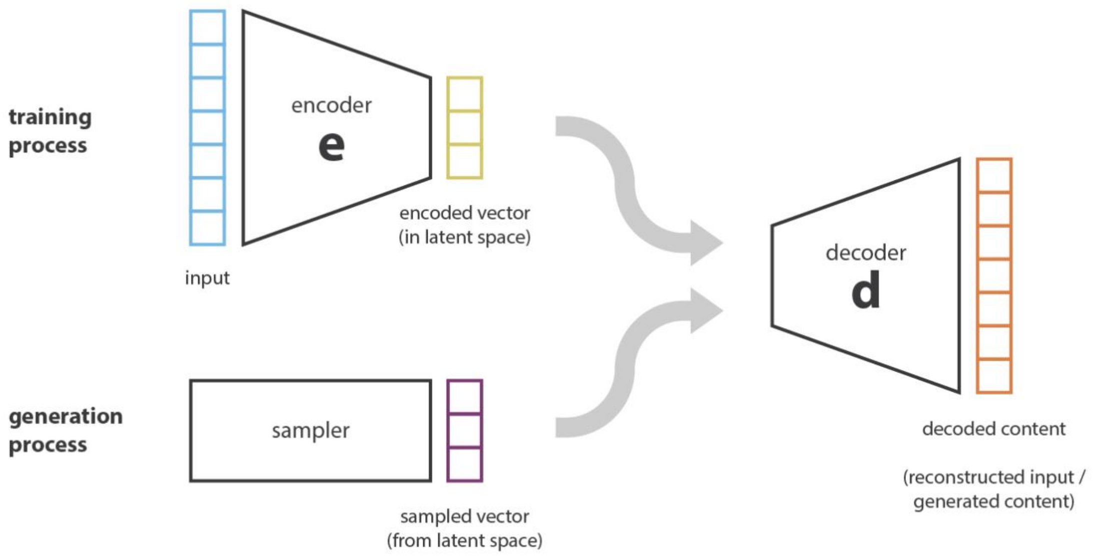
Benjamin Yen Kit Lee, Haniyeh, Fekrmandi, Srijita Chandra

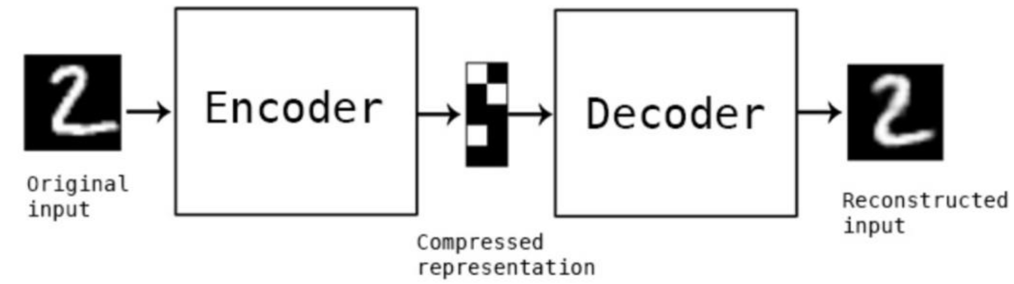
Introduction

What is image generation?

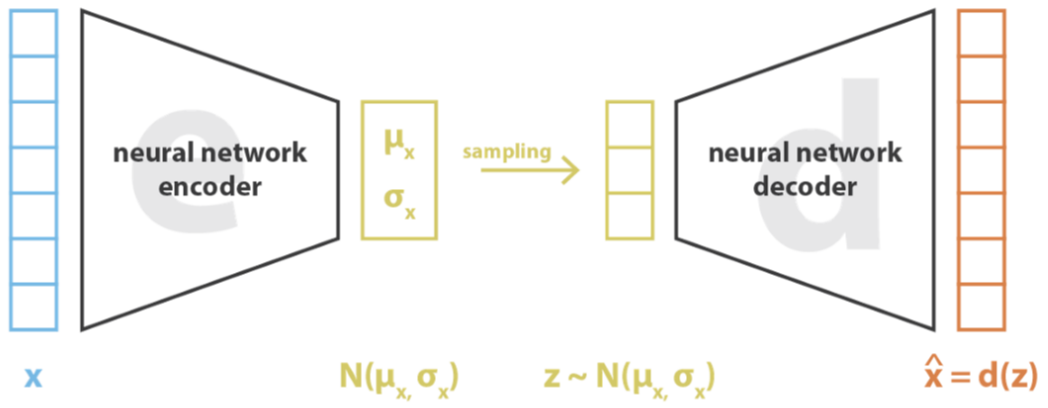
- Improving image resolution
- Restoring missing data
- Denoising
- Artificial face generation
- Image editing

AutoEncoders





Variational AutoEncoders

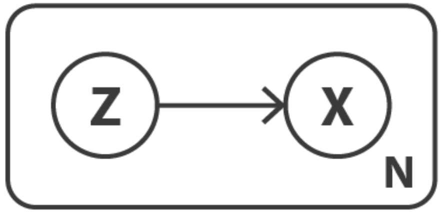


$$\text{loss} = \| x - \hat{x} \|^2 + \text{KL}[\, N(\mu_x, \sigma_x), N(0, I)\,] = \| x - d(z) \|^2 + \text{KL}[\, N(\mu_x, \sigma_x), N(0, I)\,]$$

In variational autoencoders, the loss function is composed of a reconstruction term and a regularization term.

VAEs from a probabilistic point of view

- Prior distribution p(z)
- Likelihood distribution p(x|z)
- Redefine our notions of encoder and decoder:

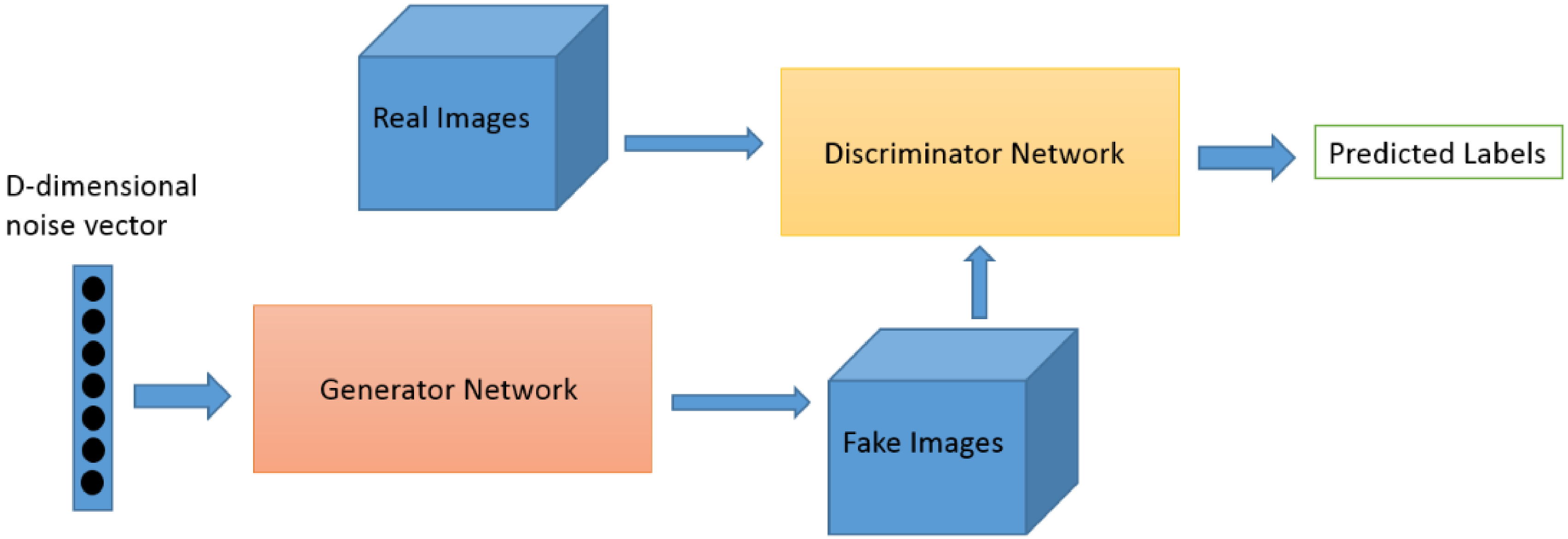


$$p(z) \equiv \mathcal{N}(0, I)$$
$$p(x|z) \equiv \mathcal{N}(f(z), cI) \qquad f \in F \qquad c > 0$$

$$f^* = \underset{f \in F}{\arg\max} \mathbb{E}_{z \sim q_x^*}(\log p(x|z))$$
$$= \underset{f \in F}{\arg\max} \mathbb{E}_{z \sim q_x^*}\left(-\frac{||x - f(z)||^2}{2c}\right)$$

Generative Adversarial Network

- The concept of GAN is to have two competing neural networks – the generator and the discriminator.
- The weights of both these networks are updated by backpropagation.



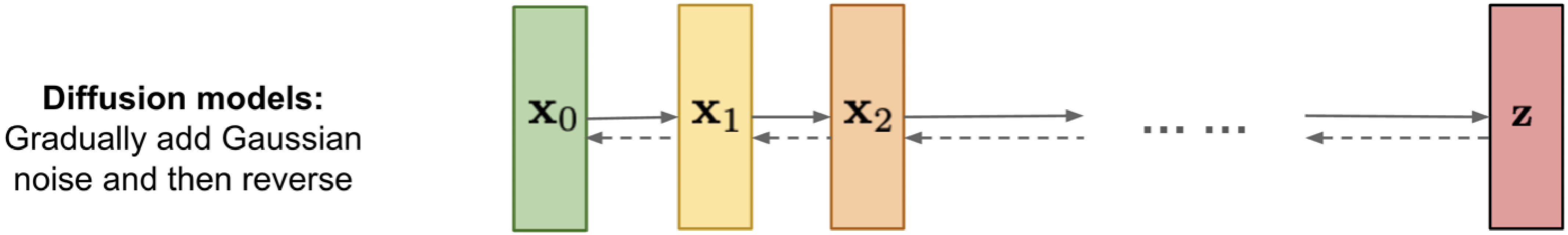
Generative Adversarial Network – contd.

- The networks are a collection of layers the weights of which are manipulated using the loss generated to get an accurate image generated or for accurate classification.
- The losses can be calculated using the following equation

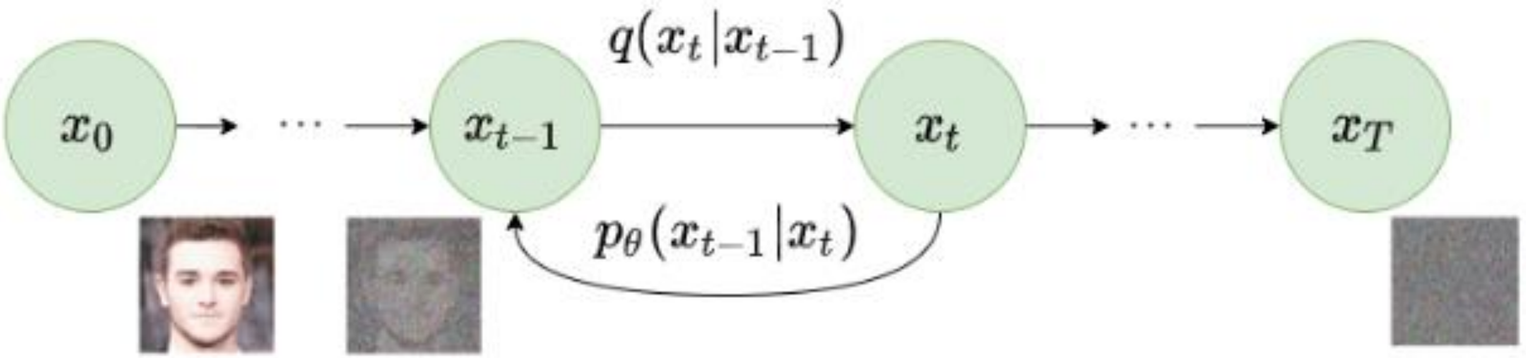
$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$$

Diffusion model

The essential idea of diffusion model is to systematically and slowly destroy structure in a 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



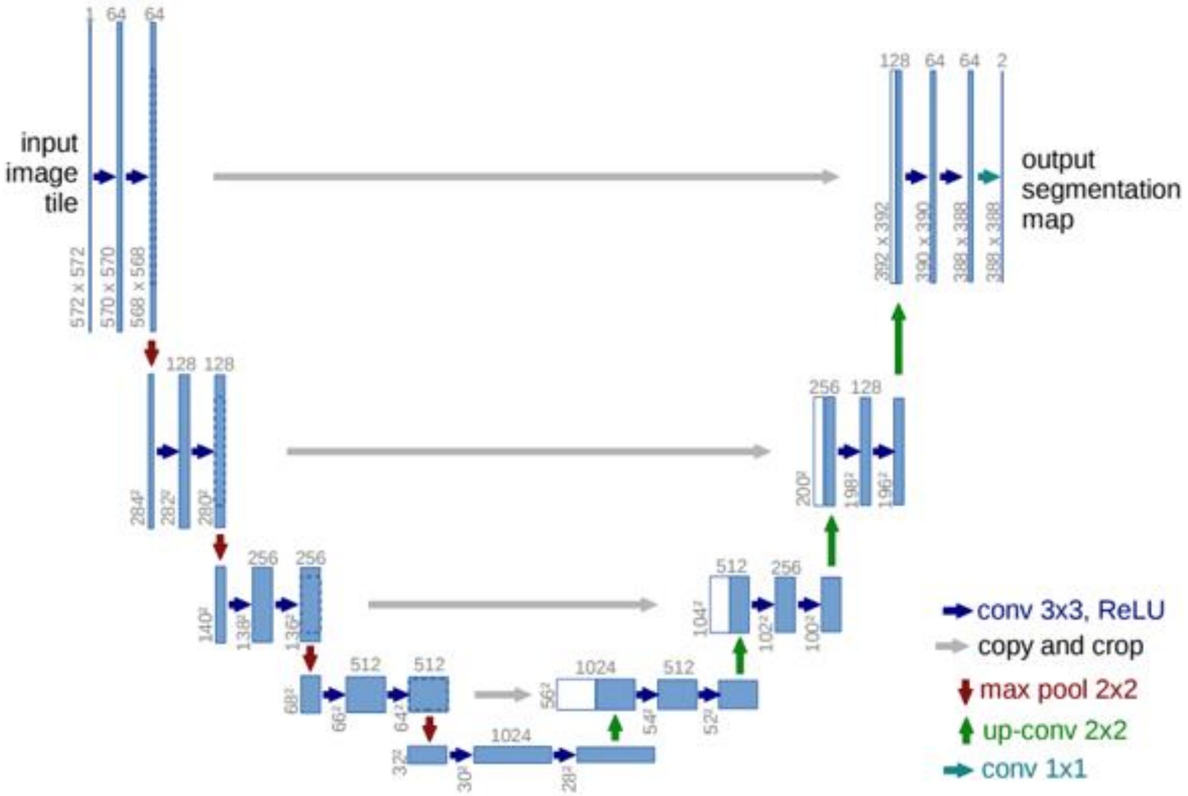
Forward and reverse diffusion process



Forward diffusion process: $$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

Reverse diffusion process: $$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$

U-Net neural network



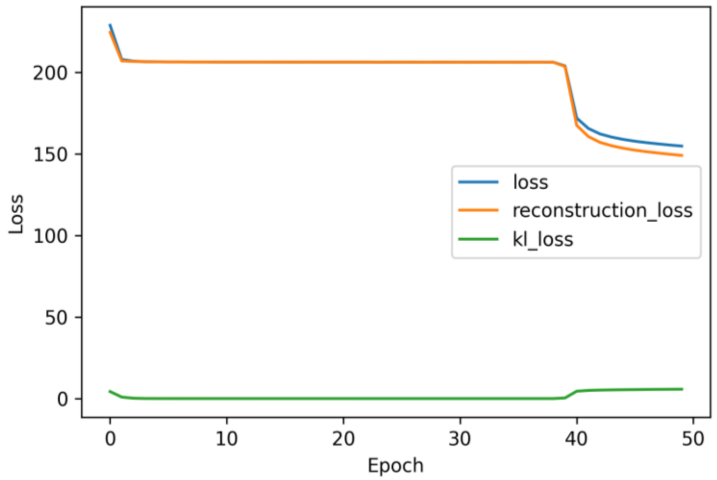
Loss : $$L_{\text{simple}}(\theta) := \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2\right]$$

Experiments

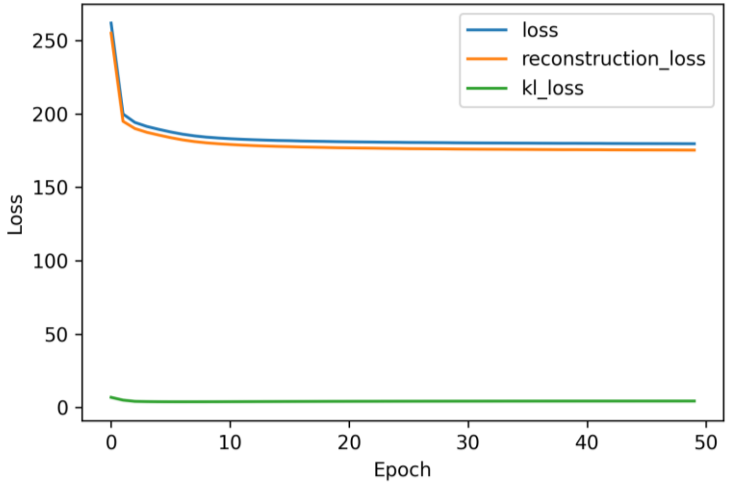
- MNIST dataset – consists of 60000 handwritten images of which 50000 training images, 10000 test images.
- The three models were trained on the MNIST dataset using 4 different optimizers to get the best results and compare the models with the best results.
- We limited the number of epochs to 50.
- Adam optimizer performed the best in all the models.
- Tensorflow and Keras python libraries were used.

Experiments – VAE graph

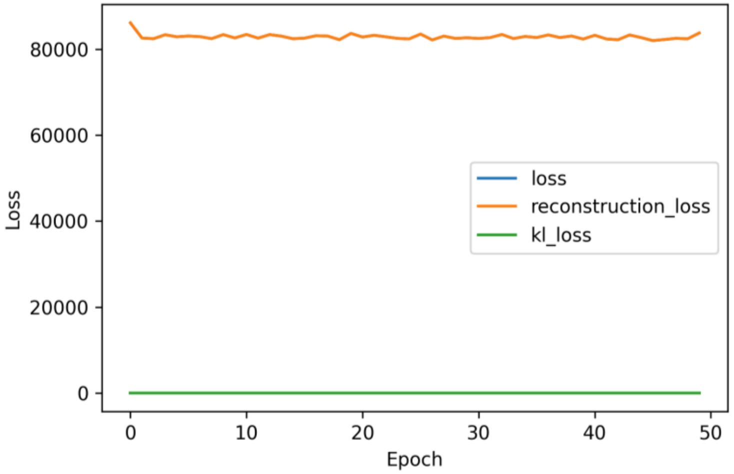
Adam optimizer



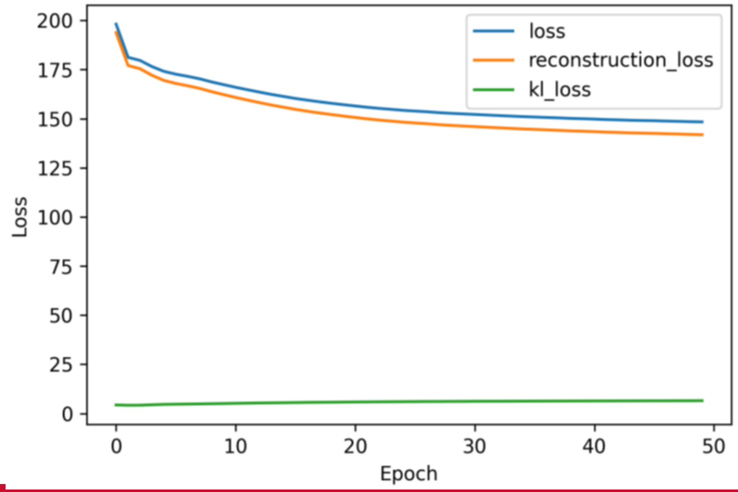
Adagrad optimizer



SGD optimizer

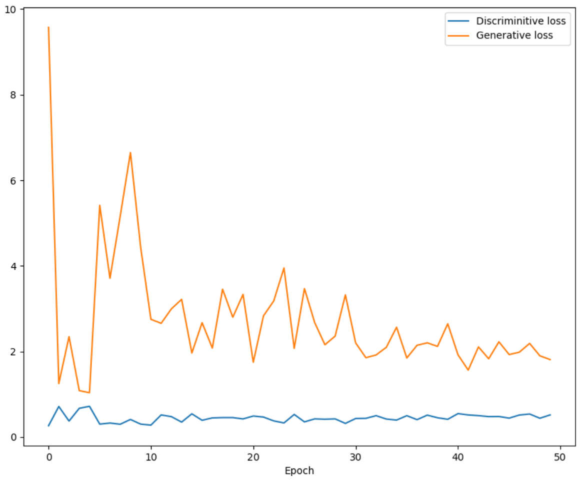


RMSprop optimizer

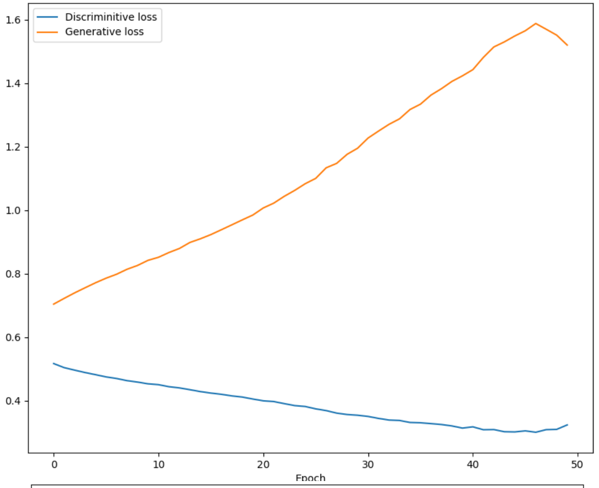


Experiments – GAN graph

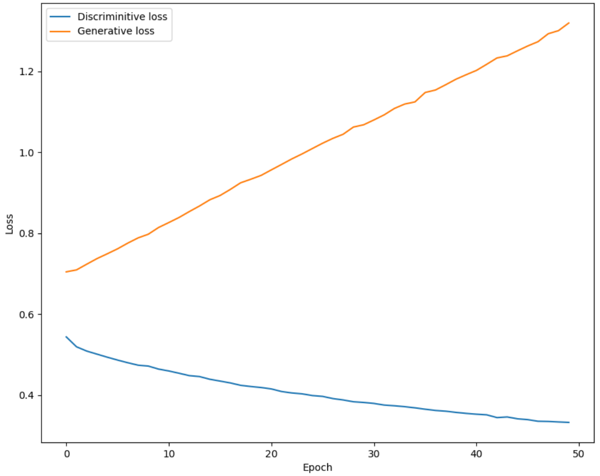
Adam optimizer



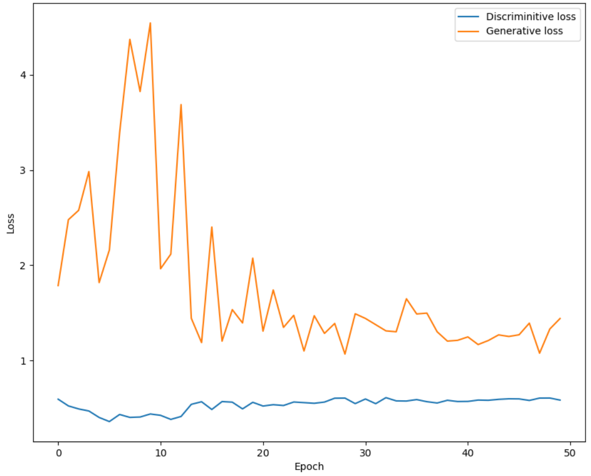
Adagrad optimizer



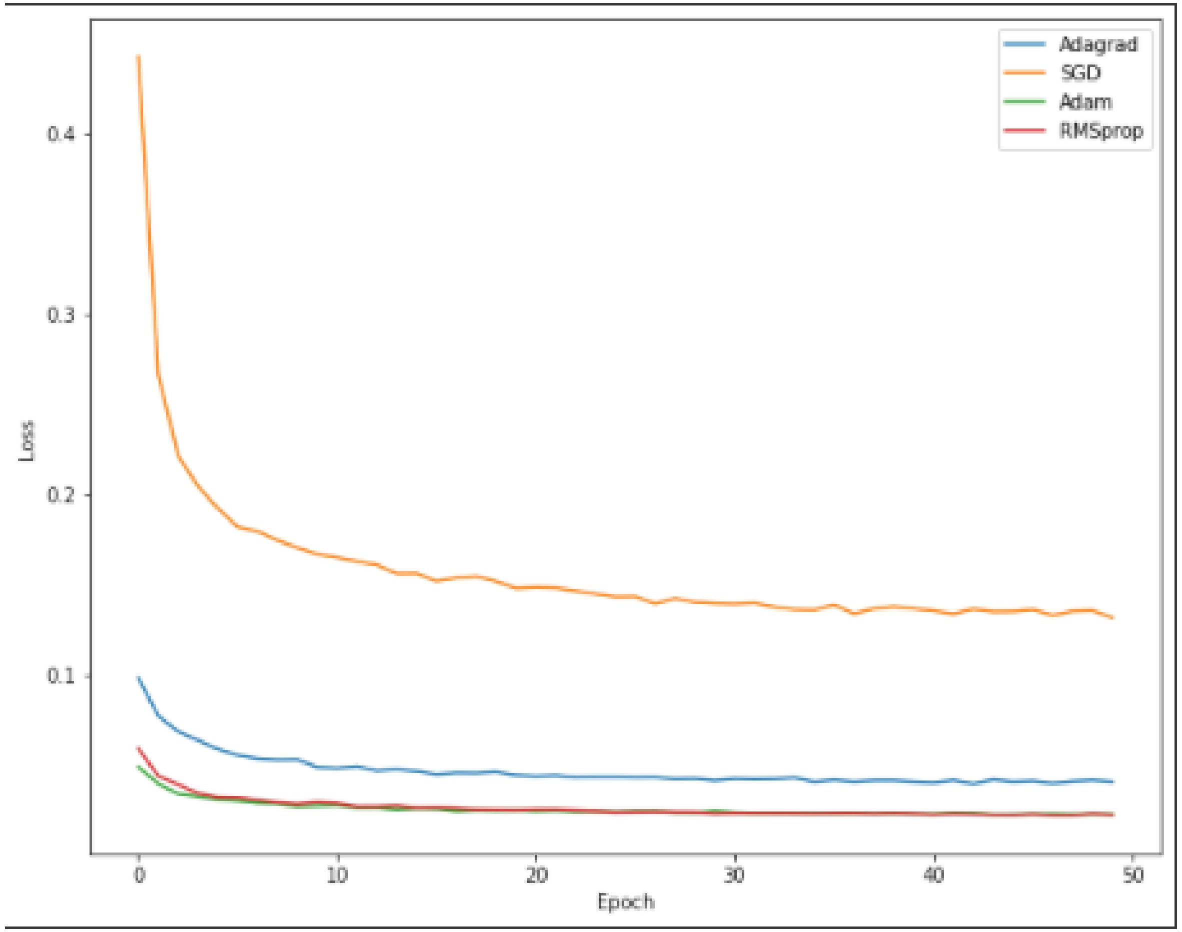
SGD optimizer



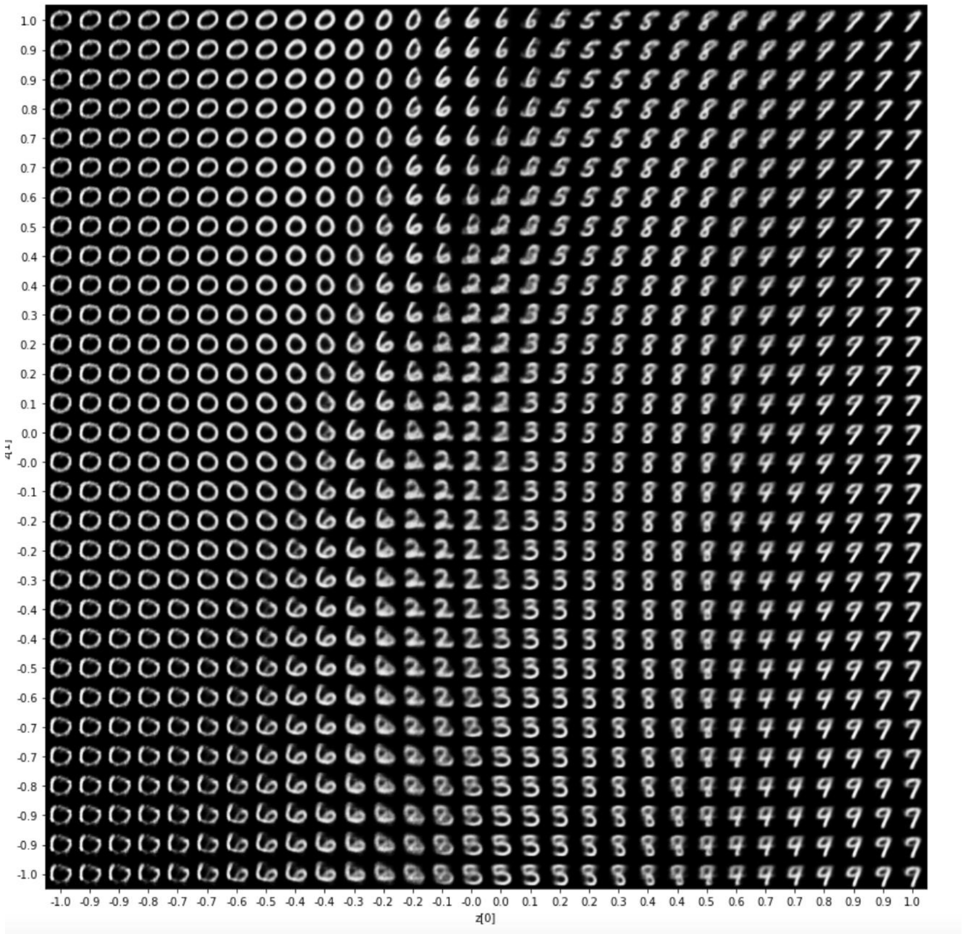
RMSprop optimizer



Experiments – Diffusion Model graph

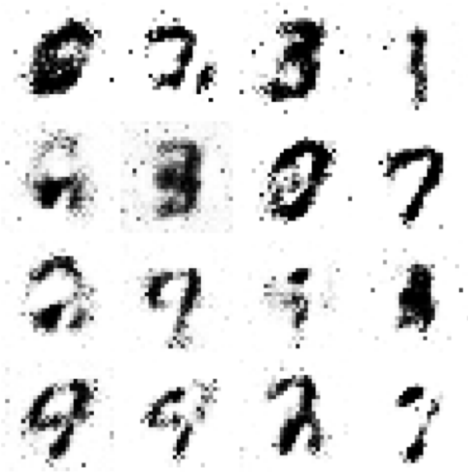


Results – VAE Model

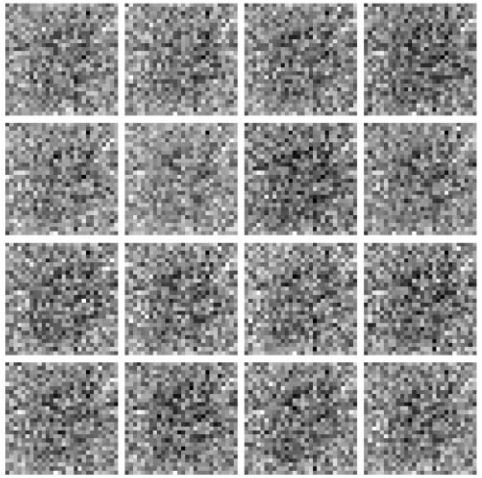


Results – GAN Model

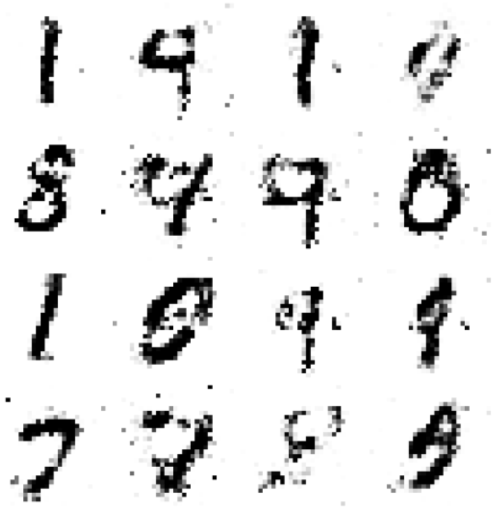
- Images generated in the 50th epoch during GAN training



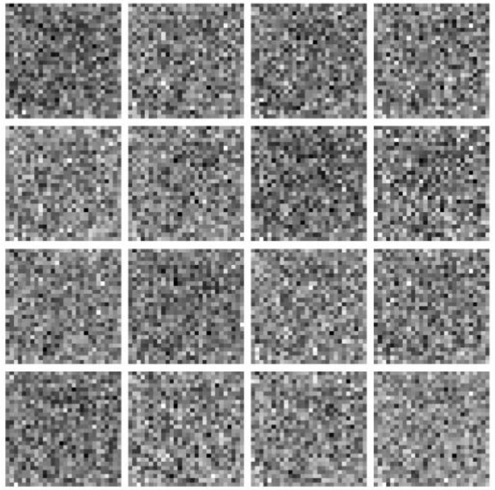
Adam optimizer



Adagrad optimizer



RMSprop optimizer



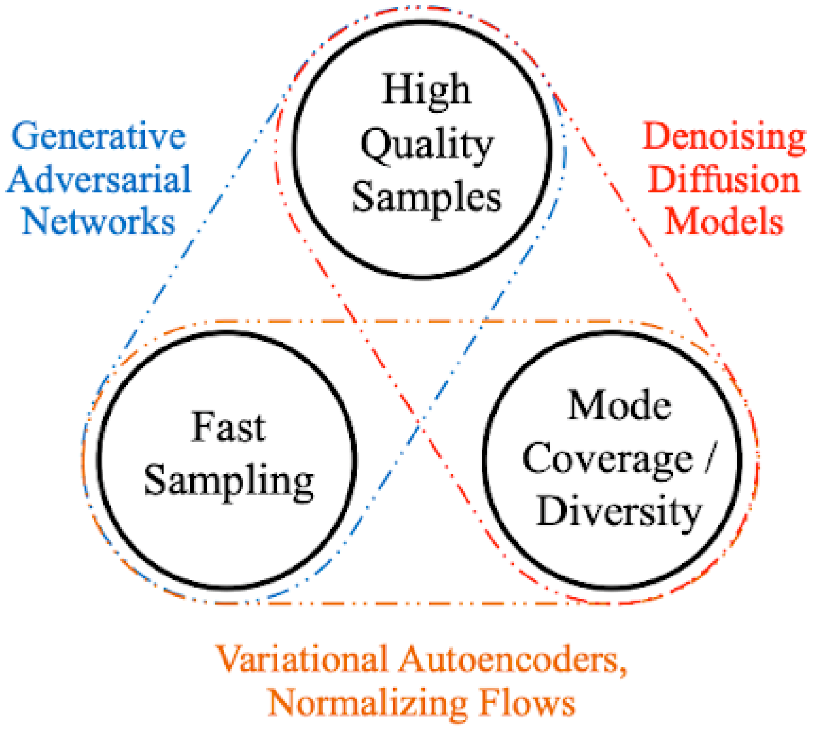
SGD optimizer

Results – Diffusion Model

Conclusion

- The models have a clear precedence in terms of performance.

  VAE < GAN < Diffusion model
- Each model has its own shortcoming which are eliminated by the other model.



Thank you!