Neural Network and Deep Learning

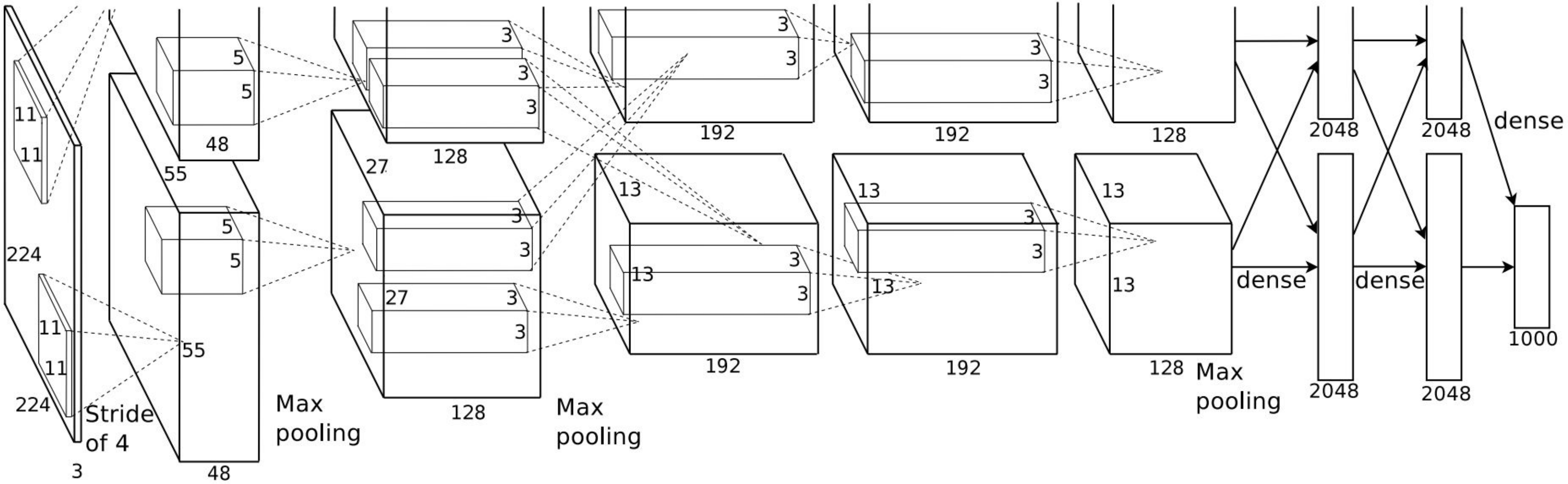
Generative Adversarial Network (GAN)

YANIKA KONGSOROT

AlexNet

$$\text{Output}_{\text{width}} = \left\lfloor \frac{I_W + 2P - N_W}{S_x} + 1 \right\rfloor$$

$$\text{Output}_{\text{height}} = \left\lfloor \frac{I_H + 2P - N_H}{S_y} + 1 \right\rfloor$$



The output volume size of the AlexNet (some layers)

Input: 227x227x3 images

First layer (CONV1): 96 11x11 filters applied at stride 4

What is the output volume size of this layer?

Second layer (POOL1): 3x3 filters applied at stride 2

What is the output volume size of this layer?

$$\mathrm{Output}_{\mathrm{width}} = \left\lfloor \frac{I_W + 2P - N_W}{S_x} + 1 \right\rfloor$$

$$\mathrm{Output}_{\mathrm{height}} = \left\lfloor \frac{I_H + 2P - N_H}{S_y} + 1 \right\rfloor$$

Generative Adversarial Network

Generative

The model is designed to generate new data that resembles the training data.

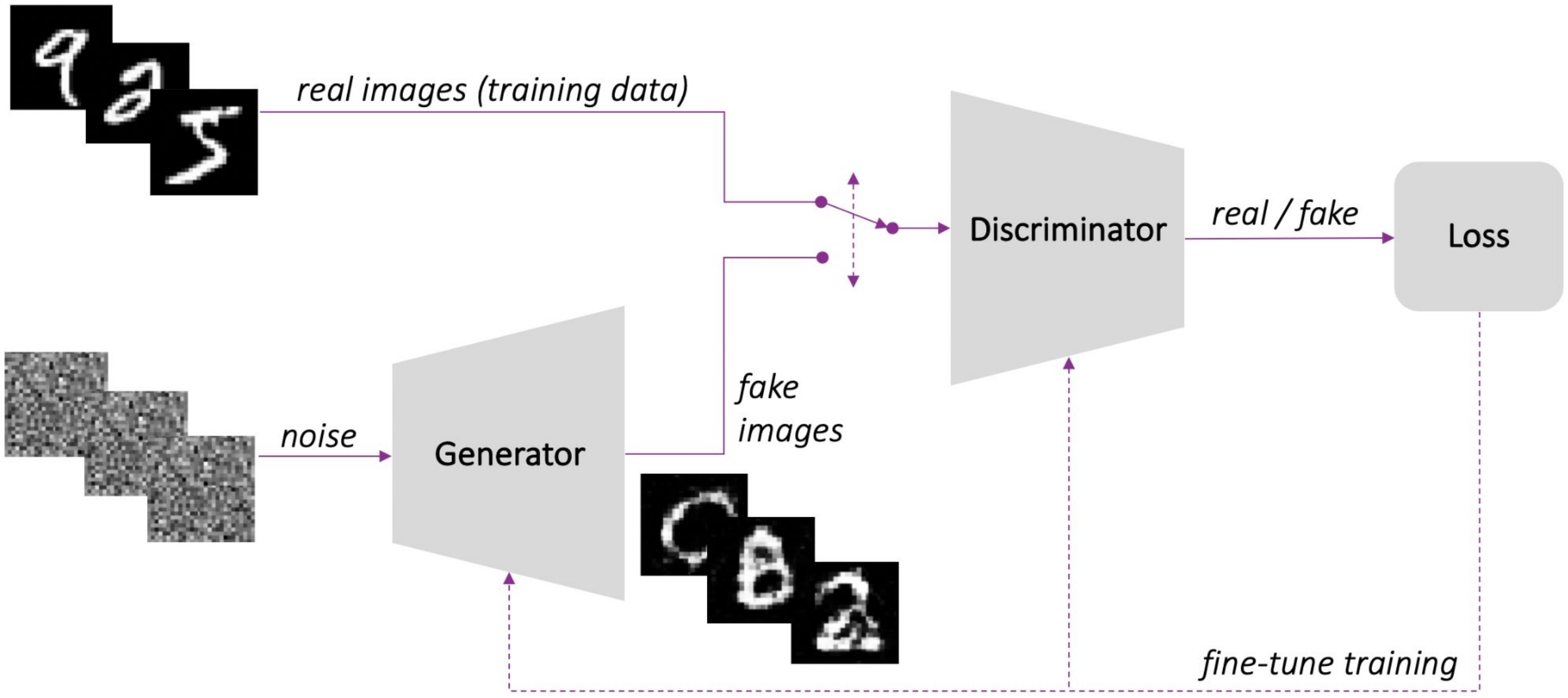
Adversarial

GANs involve a competition between two models (**generator vs discriminator**), which are trained in an adversarial manner.

Networks

Both the generator and discriminator are built using deep neural networks.

Architecture of GAN

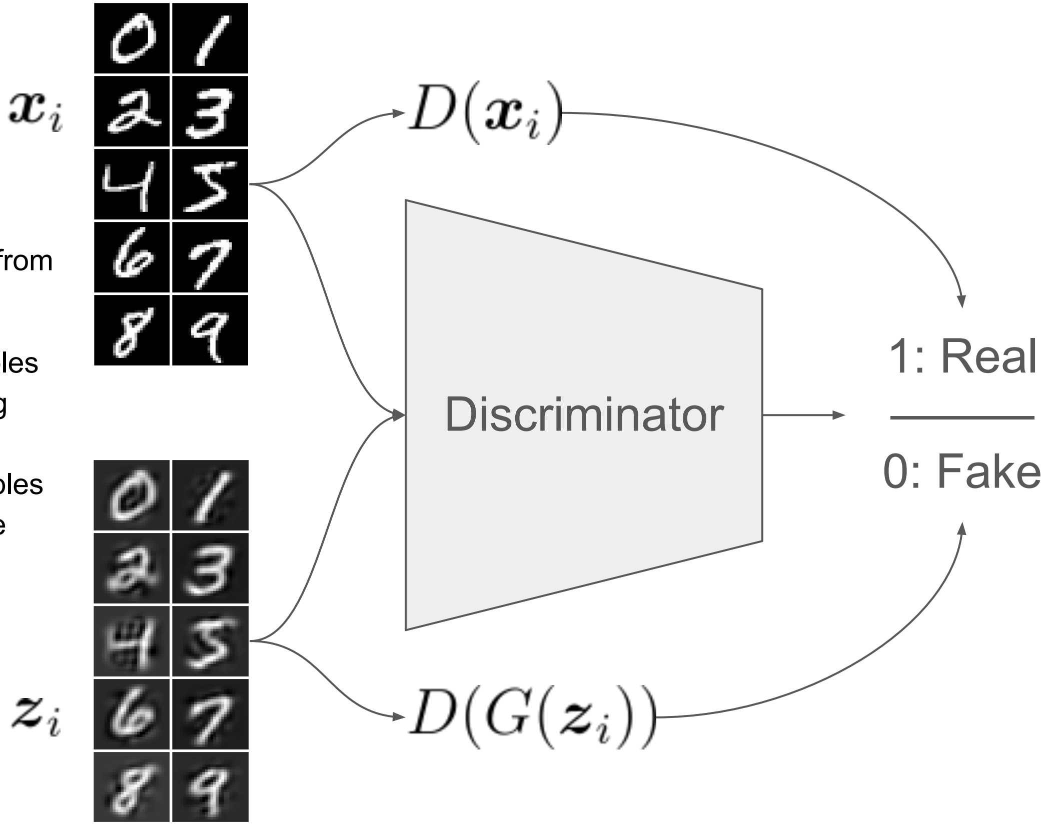


Discriminator

The Discriminator is the component responsible for distinguishing between real and generated (fake) data.

Input: It receives data from two sources:

1. Real data samples from the training dataset.
2. Fake data samples produced by the generator.



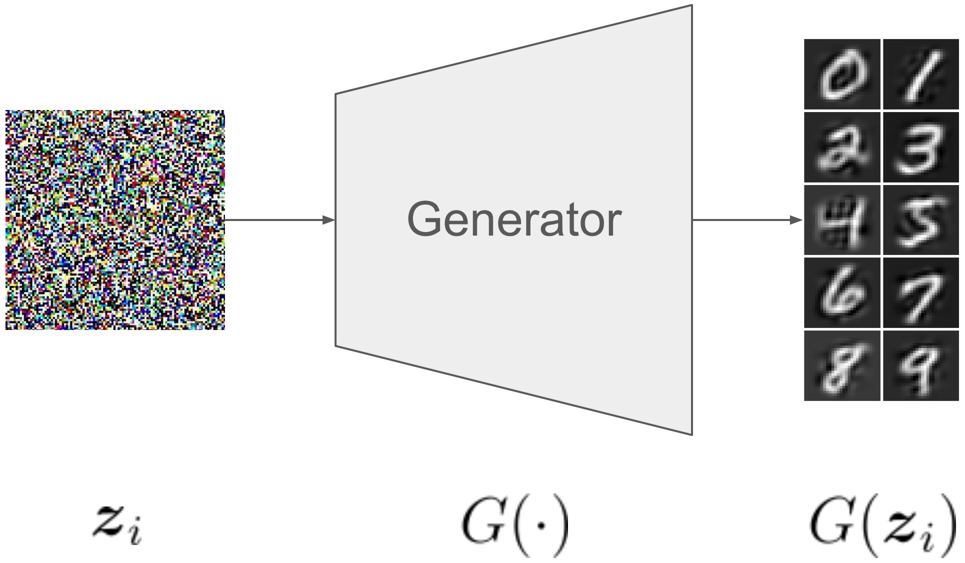
Output: It outputs a probability, typically between 0 and 1, where:

- A value close to 1 indicates that the data is real.
- A value close to 0 indicates that the data is fake.

Generator

The Generator is responsible for generating new data that mimics the real data.

Input: It takes in a random noise vector, usually sampled from a simple distribution like a Gaussian or uniform distribution.



Output: The generator transforms the random noise into data points that resemble the training data. For example, if you're working with images, the output will be an image with similar characteristics to the real dataset.

Loss function

$$\mathrm{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right]$$

$$\begin{aligned}
&\text{Discriminator}\\
\mathcal{L}_D^{\text{Real}} &= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[1\times\ \log(D(\boldsymbol{x})) + (1-1)\log(1-D(\boldsymbol{x}))\right]\\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(D(\boldsymbol{x}))\right]\\
\mathcal{L}_D^{\text{Fake}} &= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[0\times\ \log(D(G(\boldsymbol{z}))) + (1-0)\log(1-D(G(\boldsymbol{z})))\right]\\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(1-D(G(\boldsymbol{z})))\right]\\
\mathcal{L}_D &= \mathcal{L}_D^{\text{Real}} + \mathcal{L}_D^{\text{Fake}}\\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] - \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1-D(G(\boldsymbol{z})))]\\
&\text{Generator}\\
\mathcal{L}_G &= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[1\times\ \log(D(G(\boldsymbol{z}))) + (1-1)\log(1-D(G(\boldsymbol{z})))\right]\\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(D(G(\boldsymbol{z})))\right]
\end{aligned}$$

Loss function — Example

$$\mathcal{L}_D^{\text{Real}} = -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[1 \times \log(D(\boldsymbol{x})) + (1-1)\log(1-D(\boldsymbol{x}))\right]$$
$$= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(D(\boldsymbol{x}))\right]$$

$$D(\boldsymbol{x}) = [0.6, 0.7, 0.8]$$
$$= -\frac{\log(0.6) + \log(0.7) + \log(0.8)}{3}$$
$$= 0.157$$

Loss function — Example

$$
\begin{aligned}
\mathcal{L}_D^{\text{Fake}} &= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[0 \times \log(D(G(\boldsymbol{z}))) + (1-0)\log(1 - D(G(\boldsymbol{z})))\right] \\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(1 - D(G(\boldsymbol{z})))\right] \\
D(G(\boldsymbol{z})) &= [0.1, 0.2, 0.3] \\
&= -\frac{\log(1-0.1) + \log(1-0.2) + \log(1-0.3)}{3} \\
&= 0.099
\end{aligned}
$$

Loss function — Example

$$\begin{aligned}
\mathcal{L}_D &= \mathcal{L}_D^{\text{Real}} + \mathcal{L}_D^{\text{Fake}} \\
&= -\mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] - \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))] \\
&= 0.157 + 0.099 \\
&= 0.256
\end{aligned}$$

Loss function — Example

$$\begin{aligned}
\mathcal{L}_G &= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[1 \times \log(D(G(\boldsymbol{z}))) + (1-1)\log(1-D(G(\boldsymbol{z})))\right] \\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[\log(D(G(\boldsymbol{z})))\right] \\
D(G(\boldsymbol{z})) &= [0.1, 0.2, 0.3] \\
&= -\frac{\log(0.1) + \log(0.2) + \log(0.3)}{3} \\
&= 0.740
\end{aligned}$$

Variants of GAN

CycleGAN

Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision (pp. 2223-2232).

Pix2Pix

Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1125-1134).

StyleGAN

Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4401-4410).

SRGAN

Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., & Acosta, A. (2017, July). Photo-realistic single image super-resolution using a generative adversarial network [C]. In Proceedings of the IEEE conference on computer vision and pattern recognition (Vol. 2017, pp. 4681-4690).

Variants of GAN

CycleGAN

Pix2Pix

StyleGAN

SRGAN

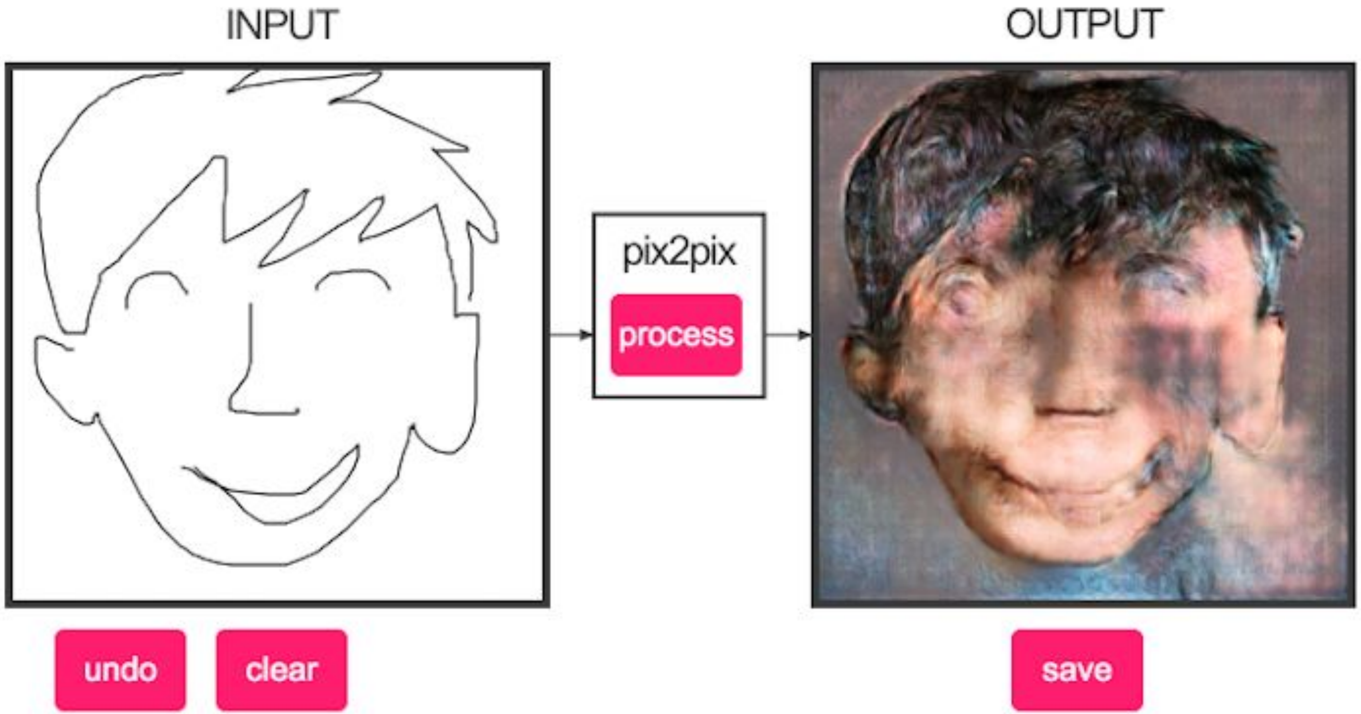
Variants of GAN

CycleGAN

Pix2Pix

StyleGAN

SRGAN



Variants of GAN

CycleGAN

Pix2Pix

StyleGAN

SRGAN

Variants of GAN

CycleGAN

Pix2Pix

StyleGAN

SRGAN

Hand On

Calculate the loss function for GAN using following given values

$$D(\boldsymbol{x}) = [0.56, 0.67, 0.64, 0.98, 0.89, 0.96, 0.48, 0.65, 0.56, 0.78]$$
$$D(G(\boldsymbol{z})) = [0.12, 0.24, 0.23, 0.42, 0.36, 0.32, 0.16, 0.18, 0.27, 0.56]$$