



GAN

머신 러닝 계열에서 이런 문제를 풀 수 있는 솔루션 중 하나인 Auto-Encoder의 뒷부분 + Discriminator → GAN

Generator Model

A statistical model of the joint probability distribution to create an image that does not exist but is likely to exist

이미지 데이터의 분포를 근사하는 도메인을 만드는 것이 생성 모델의 목표

GAN (Generative Adversarial Network)

= Generator + Discriminator

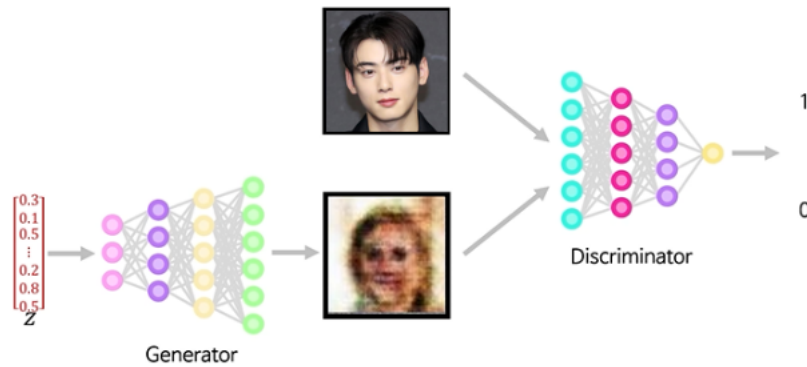
[Final Goal] Maximize the objective function

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

Ex) Noise가 z 개 인 데이터 → Generator가 이를 가지고 위조 지폐를 만들 → Discriminator가 위조 지폐와 실제 지폐를 구별함 → 해당 지폐들을 구분하는 것이 한 번의 epoch

이때 Generator가 잠재적인 코드와 해당 데이터를 가지고 위조 지폐를 만드는 것 → Unsupervised Learning

이때 Discriminator가 input 데이터를 분류 → Supervised Learning

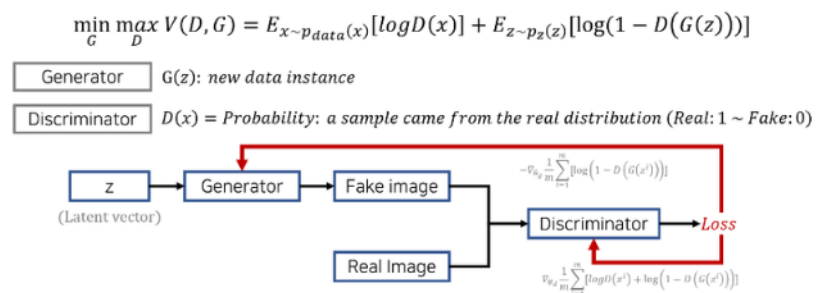


- 이때 Generator → Sigmoid 함수를 거쳐 → Discriminator로

GAN의 Discriminator Loss Function은 Entropy loss로, chain rule을 이용해서 역전파 기울기 계산

? chain rule이란?

Chain rule은 합성 함수의 미분을 계산하는 방법으로 가중치 W에 대해 미분할 때, 직접 계산할 수 없으므로, Chain Rule로 사용해서 각 층을 거쳐서 미분을 backpropagate → 출력에서 입력 방향으로 하나씩 곱하면서 전파하는



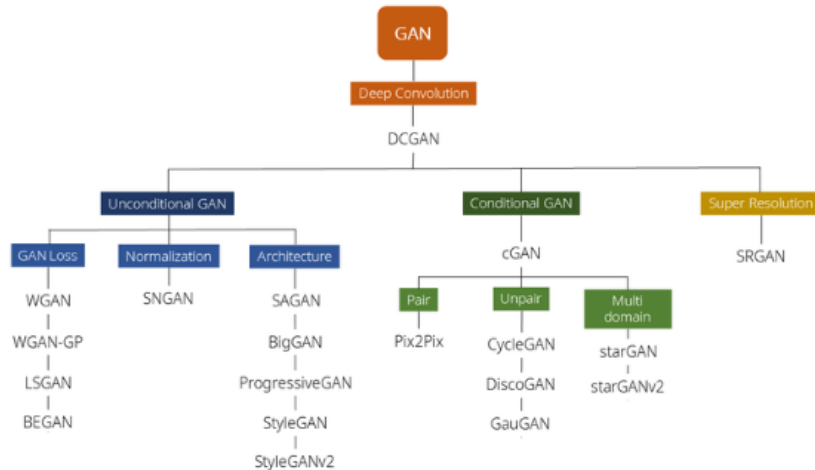
Discriminator가 가중치 업데이트 한 뒤, 그 가중치 사용해서 Generator를 학습

[한계]

GAN의 성능을 객관적 수치로 표현할 수 있는 방법이 없음 → 손실 값 자체가 Generator/Discriminator의 경쟁으로 인해 손실값이 직접적인 성능 지표가 될 수 없음

학습이 불안정한 구조 → 서로 경쟁하는 구조이기 때문에 → Minmax Game: 둘 중의 최적의 균형을 찾기 어려움

[GAN 종류]



DCGAN (Deep Convolutional Generative Adversarial Networks)

기존의 GAN의 MLP 대신 CNN을 사용

Generator :

Transposed Convolution을 사용해서 이미지 생성

Discriminator:

CNN 처럼 합성곱과 pooling으로 진짜 가짜 이미지 구분 → 마지막에 sigmoid/BCE Loss 로 0~1 사이의 확률값을 출력

이때 안정적인 학습 기법 by

- Batch Norm 적용
- FCN 제거
- Discriminator에서 Leaky ReLU activation func 사용