K-NET AI Study Group

Hyungyu Kim
K-NET
03 Sep 2024

원래 주제는 “pix2pix”였는데…

- pix2pix 논문 중에 GAN이라고 하는 거대한 것이 있어서 미리 설명합니다.
- 오늘 얘기는 10%만 GAN의 오리지널 논문에 근거합니다.
- 디테일한 이야기보단 사전지식으로서 가볍게 설명합니다.

가짜를 보고 속은 경험이 있나요?

- 진짜같은 가짜 호나우지뉴가 있으니 메시도 속았네요.
- 아마 어머니가 봐도 속을 거에요…

가짜를 진짜처럼 잘 만들면…

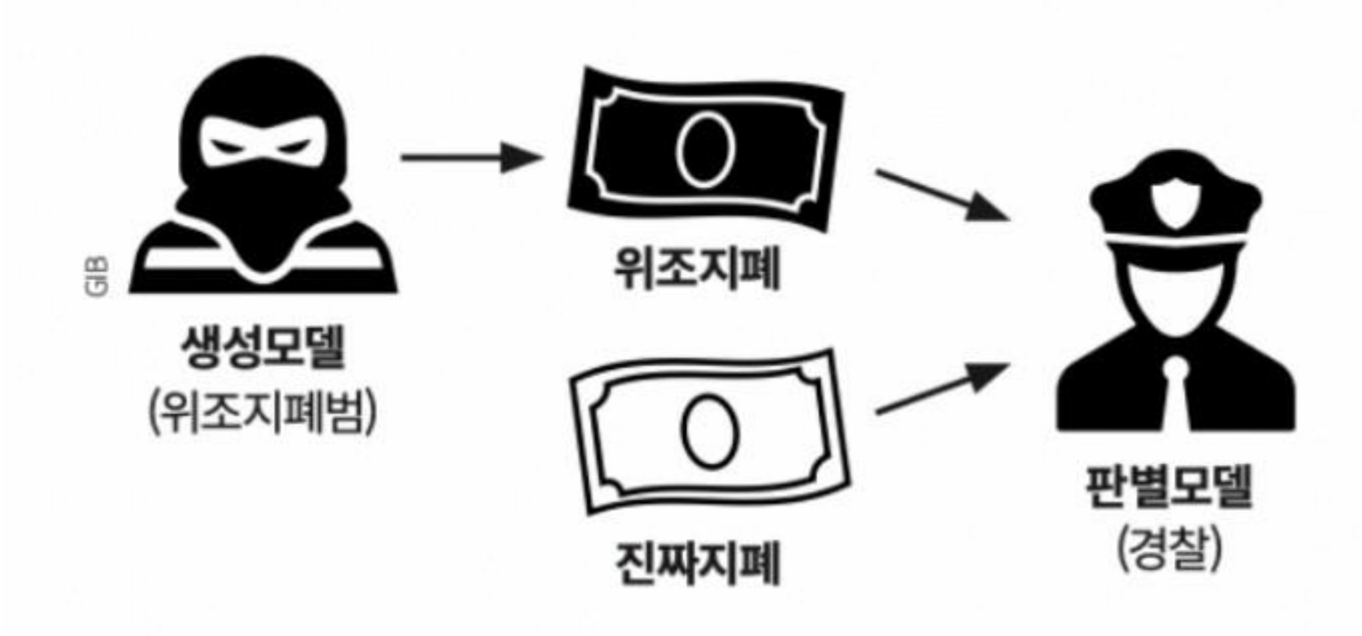
- GAN은 글자 그대로 생성 모델이에요.
- 가짜지만 진짜처럼 잘 만들어야 해요.
- 근데 "진짜같다"는 걸 누가 판단해주죠?

Yann LeCun described GANs as "the most interesting idea in the last 10 years in Machine Learning".

얀 르쿤이 말하길 GAN은 "지난 10년 동안 머신러닝에서의 가장 흥미로운 아이디어"라고 평했다.

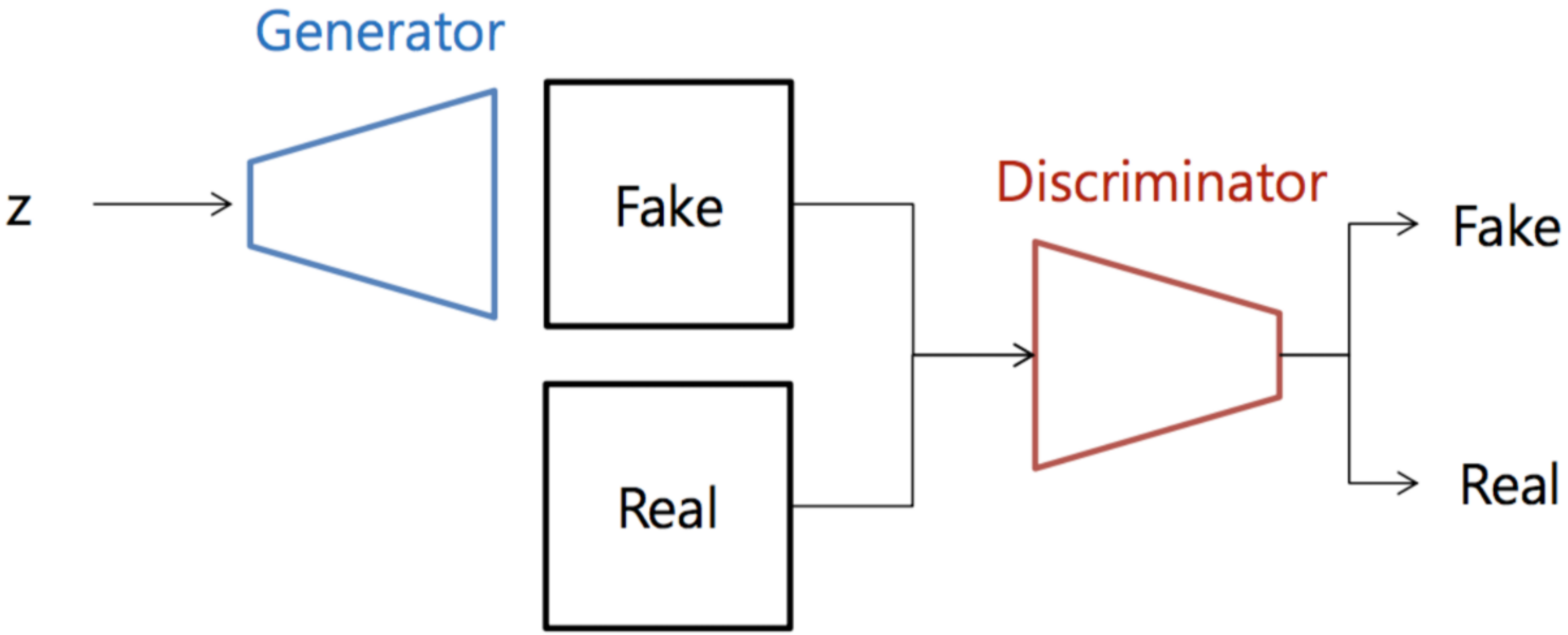
아이디어: 위조지폐범과 경찰

- 2014년에 GAN 논문을 발표한 이안 굿펠로우는 위조지폐범과 경찰에 비유했어요.
- 경찰(판별 모델)의 미션: 위조지폐와 실제지폐를 잘 구분해내는 것
- 위조지폐범(생성 모델)의 미션: 위조지폐를 진짜같이 만들어서 경찰을 속이는 것



신경망으로 그저 바꿔치면…

- GAN의 학습에서는 신경망 1개가 아니라 2개를 도입해요.
- Discriminator(판별 모델)의 미션: 가짜 데이터와 진짜 데이터를 잘 구분해내기
- Generator(생성 모델)의 미션: 진짜같은 가짜 데이터를 생성해서 판별자를 속이기



GAN의 손실 함수

- 참고 1: D는 데이터를 입력받아 진짜일 확률(0과 1 사이의 실수)을 출력합니다.
- 참고 2: G는 노이즈 입력 z를 받아서 데이터(이미지 등등)를 만들어냅니다.
- 아래 식이 무시무시한가요? D와 G의 미션이 뭐였는지 잊지 마시기 바랍니다.

In other words, D and G play the following two-player minimax game with value function $V(G, D)$:

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

GAN의 손실 함수 (cont’d)

- min, max 밑에 G, D라고 적혀있는 걸 보니, 실수 공간에서 최대화를 하는 게 아니라 함수 공간에서 최적화를 하네요.
- 즉, 손실함수를 최적화하는 파라미터(실수)를 찾는 게 아니라 그냥 신경망(함수) G, D을 찾음
- 근데 왜 G 입장에선 최소화를 하고 D 입장에선 최대화를 할까요?

$$\min_{G} \max_{D} E_{x \sim p_{data}}[\log D(x)] + E_{z \sim p_z}[\log(1 - D(G(z))]$$

GAN의 손실 함수 (cont'd)

- 아래 식의 x는 진짜 데이터입니다. z는 노이즈입니다.
- D(x)는 판별 모델이 진짜 데이터를 진짜라고 예측한 확률입니다.
- 로그는 underflow를 막기 좋은 수단입니다.
- 기대값은 전체 데이터의 정보를 말할 수 있는 좋은 수단입니다.

$$E_{x \sim p_{data}}[\log D(x)] + E_{z \sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

GAN의 손실 함수 (cont'd)

- G(z)는 가짜 데이터입니다.
- D(G(z))는 판별 모델이 가짜 데이터를 진짜라고 예측한 확률입니다.
- 1−D(G(z))는 판별 모델이 가짜 데이터를 가짜라고 예측한 확률입니다.
- 로그, 기대값을 똑같이 씌워줍니다.

$$E_{x \sim p_{data}}[\log D(x)] + E_{z \sim p_z}\left[\log\left(1 - D(G(z))\right)\right]$$

GAN의 손실 함수 (cont'd)

- 그러니까 아래 식의 의미는:

"진짜 데이터는 진짜라고 예측할 평균적 (로그)확률
+ 가짜 데이터는 가짜라고 예측할 평균적 (로그)확률"

- 이런 의미를 가진 식을 D와 G 입장에선 어떻게 최적화해야 할까요?
- D와 G의 역할을 까먹은 건 아니죠?

$$E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

GAN의 손실 함수 (cont'd)

- D의 입장에선 진짜 데이터는 진짜라고, 가짜 데이터를 가짜 데이터라고 잘 골라내야 할 임무가 있습니다.
- 그러니까 아래 식을 최대화해야 하는 것이죠:
“진짜 데이터는 진짜라고 예측할 평균적 (로그)확률
+ 가짜 데이터는 가짜라고 예측할 평균적 (로그)확률”

$$\max_{D} E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

GAN의 손실 함수 (cont'd)

- G는 D와 경쟁하는 (적대적) 입장에 있습니다.
- 그래서 Discriminator와 반대되는 일을 합니다.
- $E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p_z}[\log(1 - D(G(z)))]$ 이 커지는 것은 그 의미 상 Generator에게 손해이므로 G 입장에선 줄이려 노력해야 합니다.

  (더 정확히는, D에 대해 최대화된 값을 최소화합니다.)

$$\arg\min_{G} \max_{D} E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p_z}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

참고: GAN의 학습

- D를 먼저 학습하고 G를 학습하는 것에 유의

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**
  for k steps **do**
  • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  • Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  • Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)\right].$$

  end for
  • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  • Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

참고: GAN의 학습

- GAN의 학습은 Generator의 출력 분포를 데이터의 분포에 가깝게 만드는 역할 수행

Theorem 1. *The global minimum of the virtual training criterion $C(G)$ is achieved if and only if* $\boxed{p_g = p_{data}.}$ *At that point, $C(G)$ achieves the value* $-\log 4$.

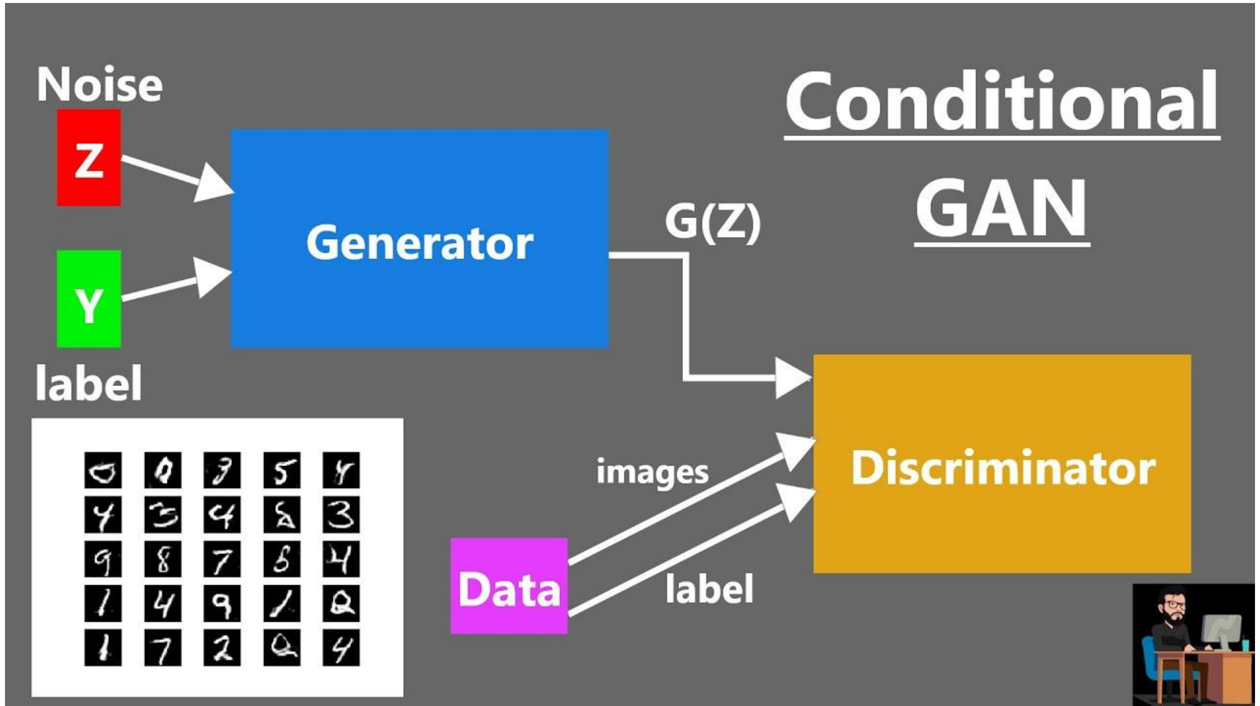
Proposition 2. *If G and D have enough capacity, and at each step of Algorithm 1, the discriminator is allowed to reach its optimum given G, and p_g is updated so as to improve the criterion*

$$\mathbb{E}_{\boldsymbol{x}\sim p_{data}}[\log D^*_G(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}\sim p_g}[\log(1 - D^*_G(\boldsymbol{x}))]$$

then $\boxed{p_g \text{ converges to } p_{data}}$

GAN에 추가 정보를 같이 섞으면?

- conditional GAN(cGAN)은 데이터(x) 이외에 추가 정보(y)를 넣을 수 있습니다.
- 추가 정보 예시: 이미지의 레이블(클래스)나 다른 modality의 데이터 등
- 기존의 GAN에선 할 수 없었던, 생성되는 데이터를 조절하는 일이 가능해집니다.



GAN에 추가 정보를 같이 섞으면?

- 아래 보면 알겠지만 더 설명할 건 이게 다예요. 거의 동일해요.
- 추가 정보 y는 조건부로 들어가요.

In the generator the prior input noise $p_{\boldsymbol{z}}(\boldsymbol{z})$, and $\boldsymbol{y}$ are combined in joint hidden representation, and the adversarial training framework allows for considerable flexibility in how this hidden representation is composed. [1]

In the discriminator $\boldsymbol{x}$ and $\boldsymbol{y}$ are presented as inputs and to a discriminative function (embodied again by a MLP in this case).

The objective function of a two-player minimax game would be as Eq 2

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x}|\boldsymbol{y})] + \mathbb{E}_{\boldsymbol{z} \sim p_z(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}|\boldsymbol{y})))]. \qquad (2)$$

GAN에 추가 정보를 같이 섞으면?

- 실제 결과들

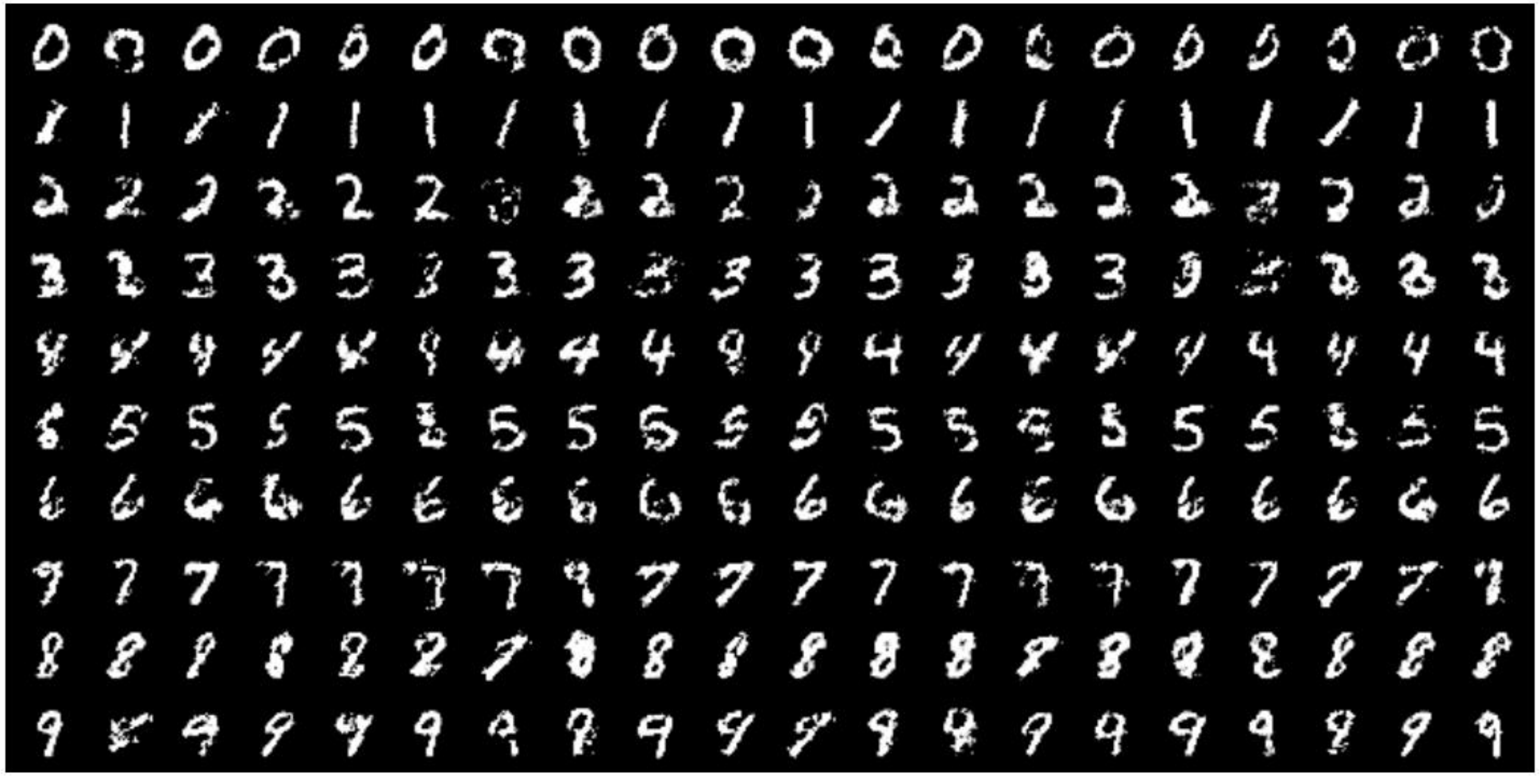


Figure 2: Generated MNIST digits, each row conditioned on one label

이제 다시 질문에 답할 차례입니다

- GAN은 글자 그대로 생성 모델이에요.
- 가짜지만 진짜처럼 잘 만들어야 해요.
- 근데 "진짜같다"는 걸 누가 판단해주죠?

In other words, D and G play the following two-player minimax game with value function $V(G, D)$:

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

다음 시간에 다룰 주제

- pix2pix