CS7150 Deep Learning

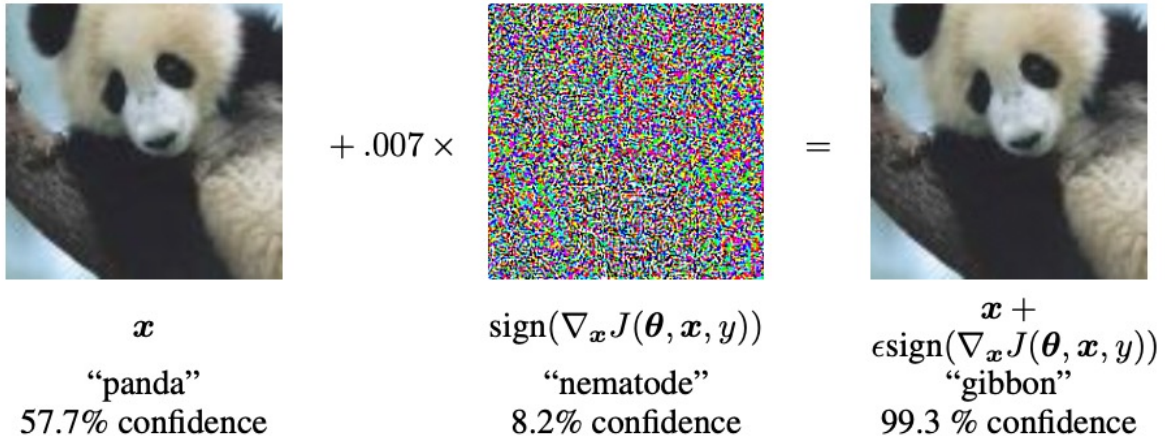
Jiaji Huang

https://jiaji-huang.github.io

03/23/2024

Recap of Last Lecture

- Compare Networks by Feature Similarity
- Attribute a network's decision to
  - Input features
  - Training samples
- Construct adversarial samples

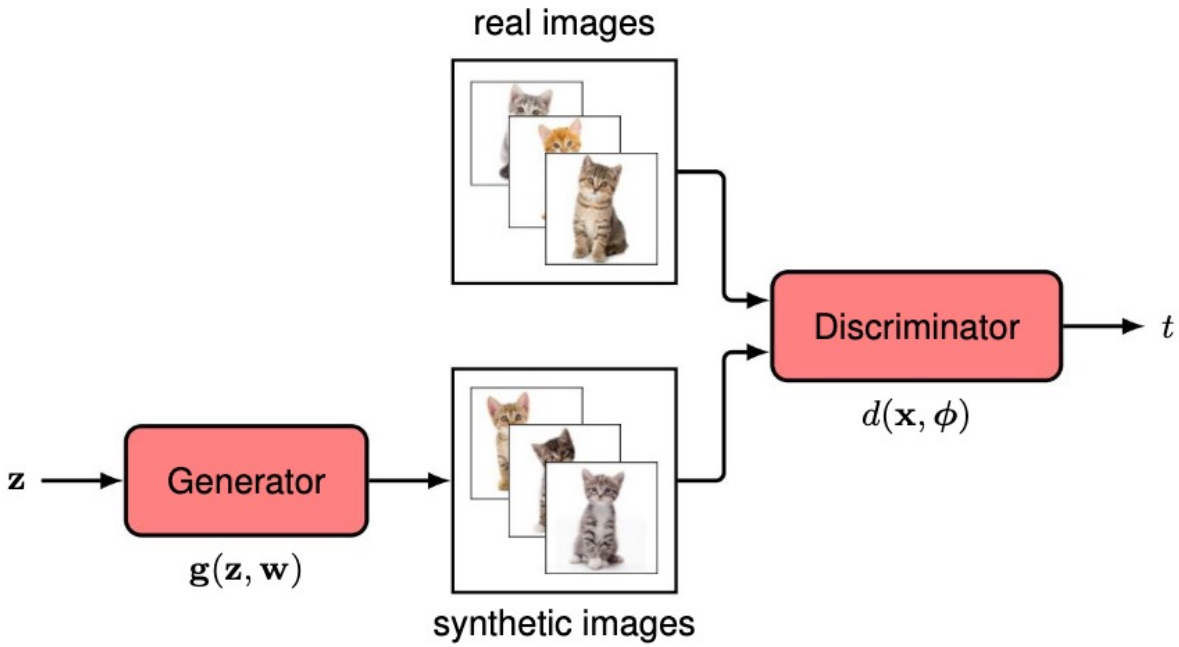


Agenda

- Generative Adversarial Network (GAN)
- Earth Mover Distance and Wasserstein GAN

Adversarial Training

- Generate datum “indistinguishable” from real ones
- Maximize a discriminator’s loss
- A competing game between
  generator and discriminator
- z: latent variable



Bishop book chapter 17

Formalize

- Discriminator: a binary classifier, $p(\boldsymbol{x}\ is\ real\ image) = d(\boldsymbol{x}, \phi)$

$$\max_{\phi} \frac{1}{N_{real}} \sum_{\boldsymbol{x}:real} \log d(\boldsymbol{x}, \phi) + \frac{1}{N_{gen}} \sum_{\boldsymbol{x}:gen} \log(1 - d(\boldsymbol{x}, \phi))$$

$g(z, w)$

- Generator: minimize the reward

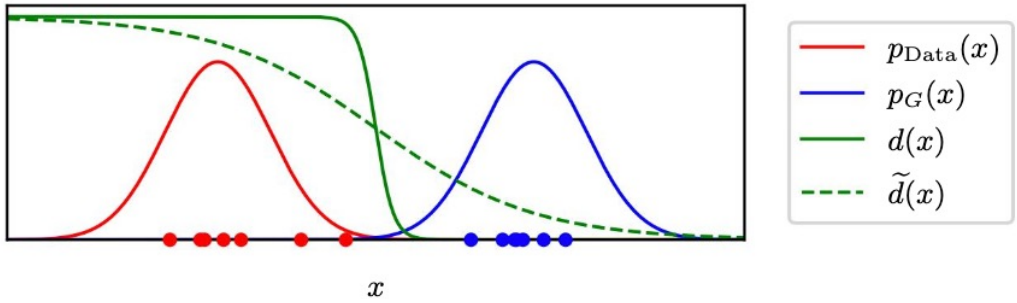
$$\min_{\boldsymbol{w}} \max_{\phi} \frac{1}{N_{real}} \sum_{\boldsymbol{x}:real} \log d(\boldsymbol{x}, \phi) + \frac{1}{N_{gen}} \sum_{\boldsymbol{z}} \log(1 - d(g(z, w), \phi))$$

- Grad ascent on ϕ. Grad descent on w:

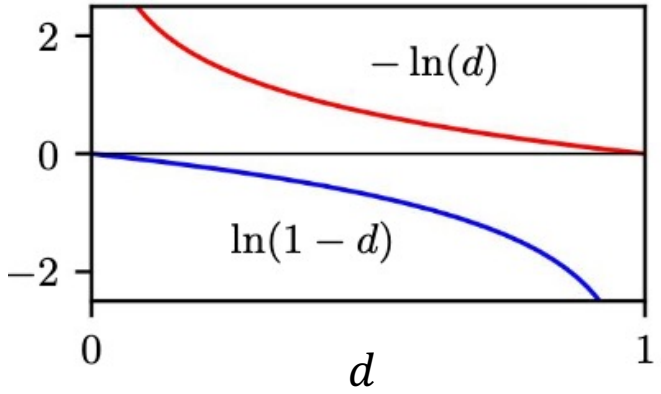
$$\Delta\phi = \eta \nabla_{\phi} L(w, \phi)$$
$$\Delta w = -\eta \nabla_{w} L(w, \phi)$$

Potential Issues

- Potential issue: “mode collapse”



- Too easy to discriminate, $\nabla_\phi E = 0$, stop learning
- Smooth the $d(\boldsymbol{x})$ to $\tilde{d}(\boldsymbol{x})$, enables gradients
- Consider instead $\log d(\boldsymbol{x}_{gen}, \boldsymbol{\phi})$



How it works

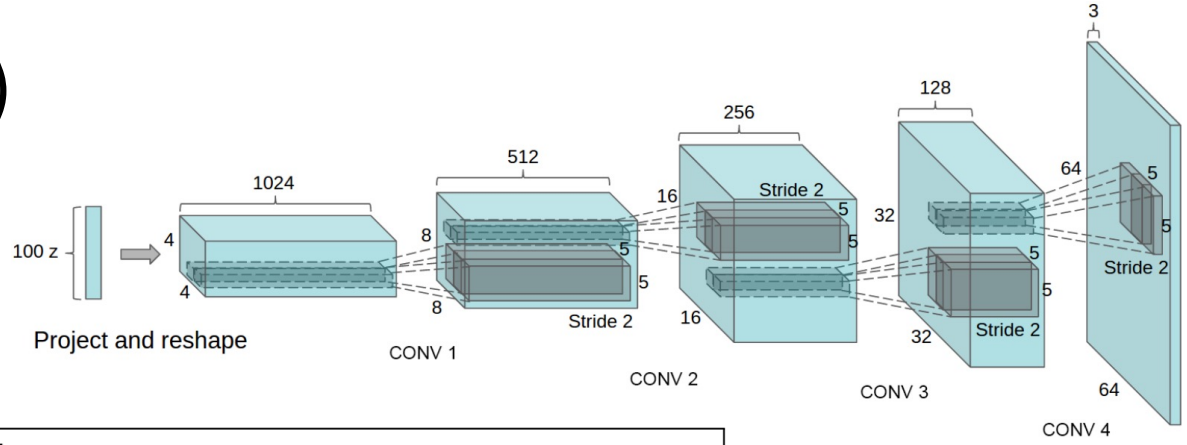
Generated Images and their nearest neighbors (in yellow box). Left: mnist; Middle: Toronto Face Dataset; Right: cifar-10

Generated images along a path of Interpolated z

Goodfellow et. al, 2014

Generator Architectures

- Deep Convolutional GAN (DCGAN)



Architecture guidelines for stable Deep Convolutional GANs

- Replace any pooling layers with strided convolutions (discriminator) and fractional-strided convolutions (generator).
- Use batchnorm in both the generator and the discriminator.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in generator for all layers except for the output, which uses Tanh.
- Use LeakyReLU activation in the discriminator for all layers.

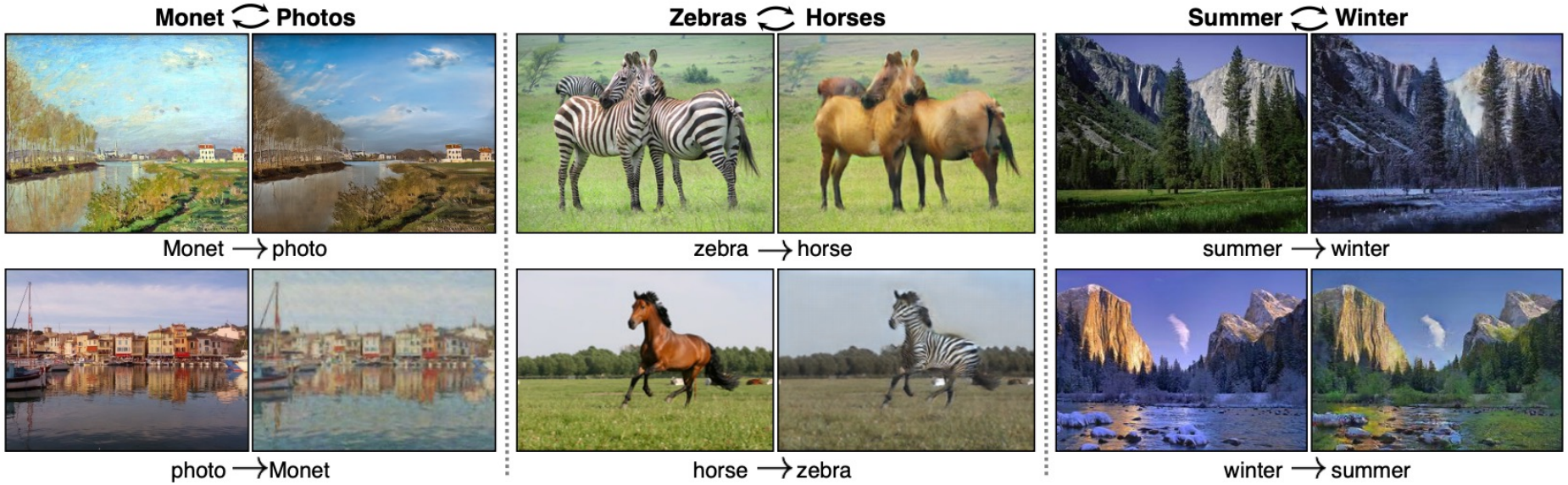
$g(z)$

Radford et. al, 2016

DCGAN

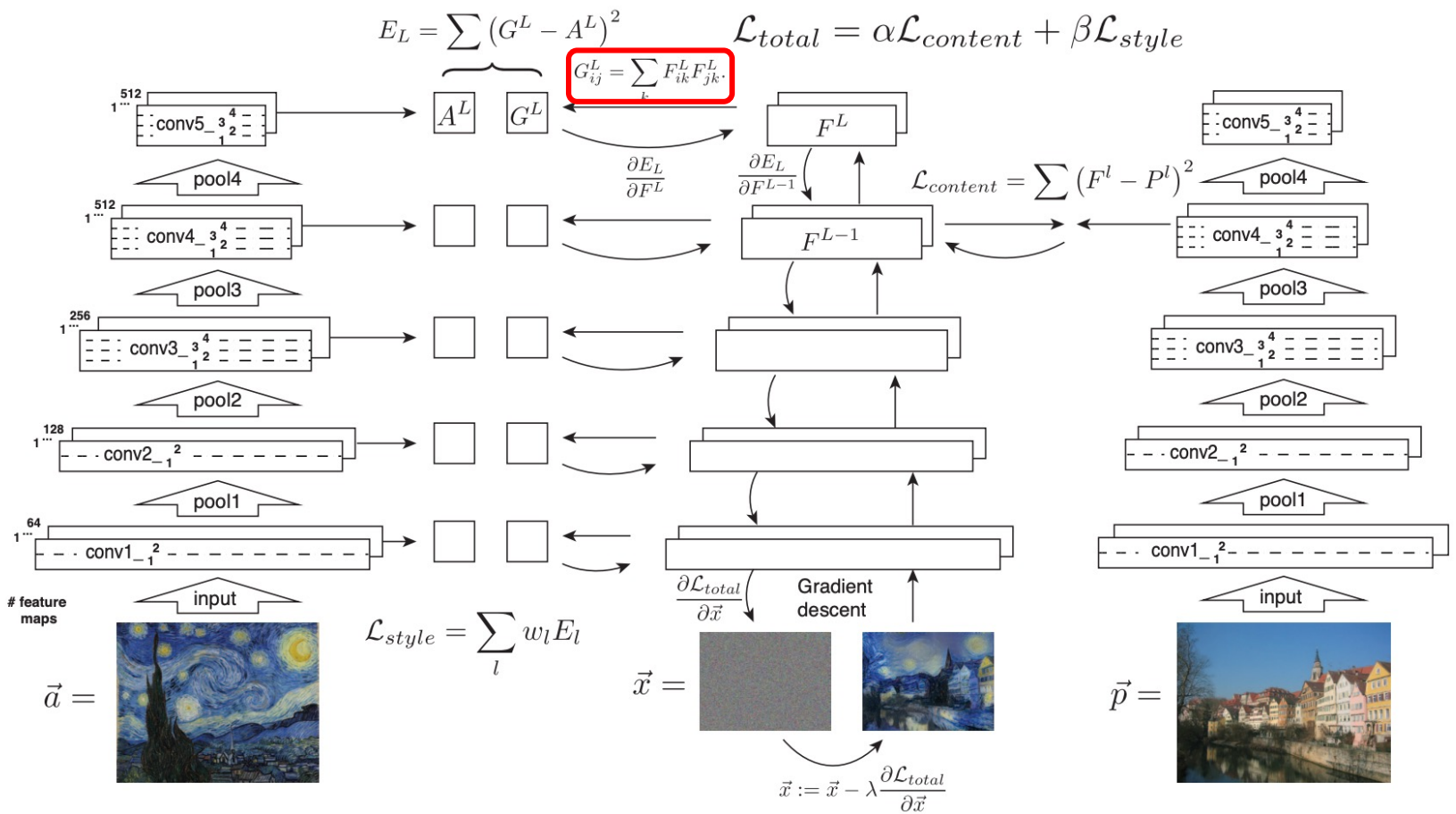
Cycle GAN

- Learn mapping between two collections



[Zhu et. al, 2020]

Revisit Image Style Transfer (Lecture 3)

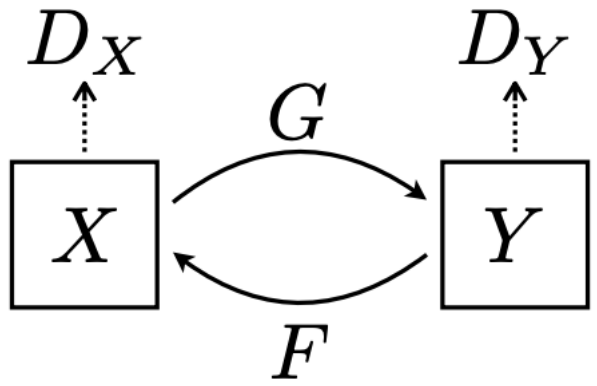


Formalize: Cycle GAN

- Two collections of images: X and Y
- Learn mapping $G: X \mapsto Y$ to confuse the discriminator D_Y

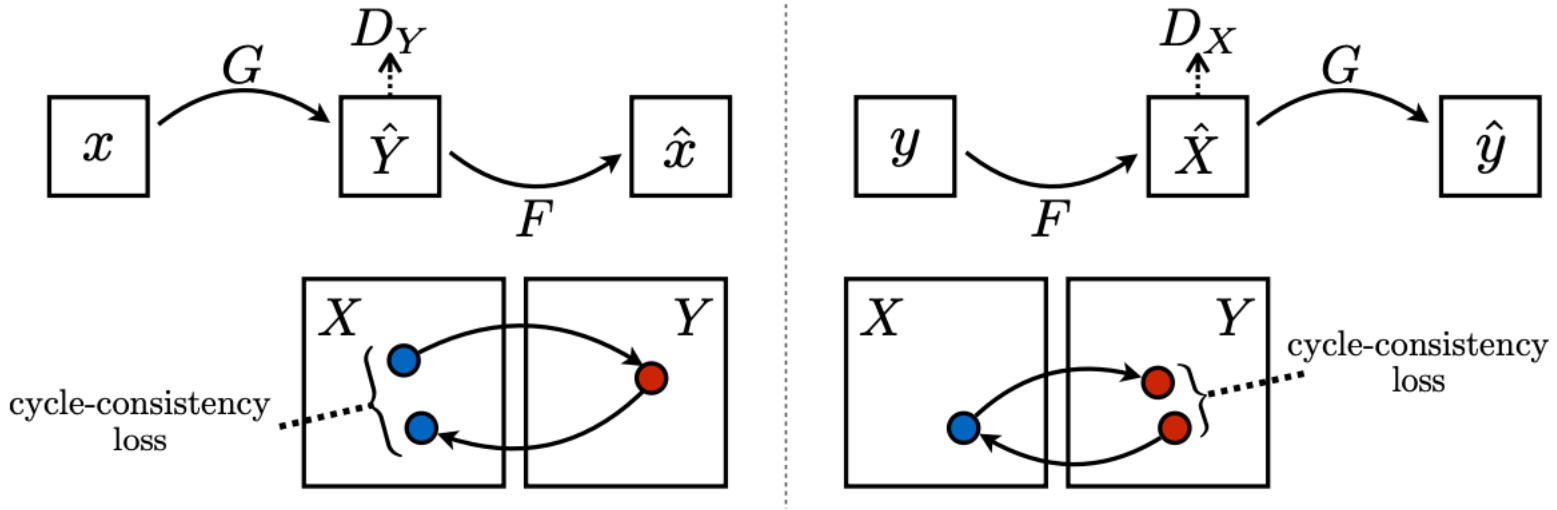
$$\min_{G} \max_{D_Y} \mathcal{L}\,(G, D_Y, X, Y)$$
$$= \mathbb{E}_{y \sim p(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p(x)}[\log(1 - D_Y(G(x)))]$$

- Likewise, Learn mapping $F: Y \mapsto X$ via $\min_{F} \max_{D_X} \mathcal{L}\,(F, D_X, X, Y)$
- Question: Key difference against style transfer?
  - Cycle GAN learns mapping between two styles
  - Style transfer only works for a specific image



Regularize

$$\mathcal{L}_{cycle} = \mathbb{E}_{x\sim p(x)}\left\|F\left(G(x)\right) - x\right\|_1 + \mathbb{E}_{y\sim p(y)}\left\|G\left(F(y)\right) - y\right\|_1$$



$$\min_{G,F}\left\{\max_{D_Y}\mathcal{L}(G, D_Y, X, Y) + \max_{D_X}\mathcal{L}(F, D_X, X, Y) + \mathcal{L}_{cycle}\right\}$$

GAN for Unsupervised Machine Translation

- Recap enc-dec models for machine translation

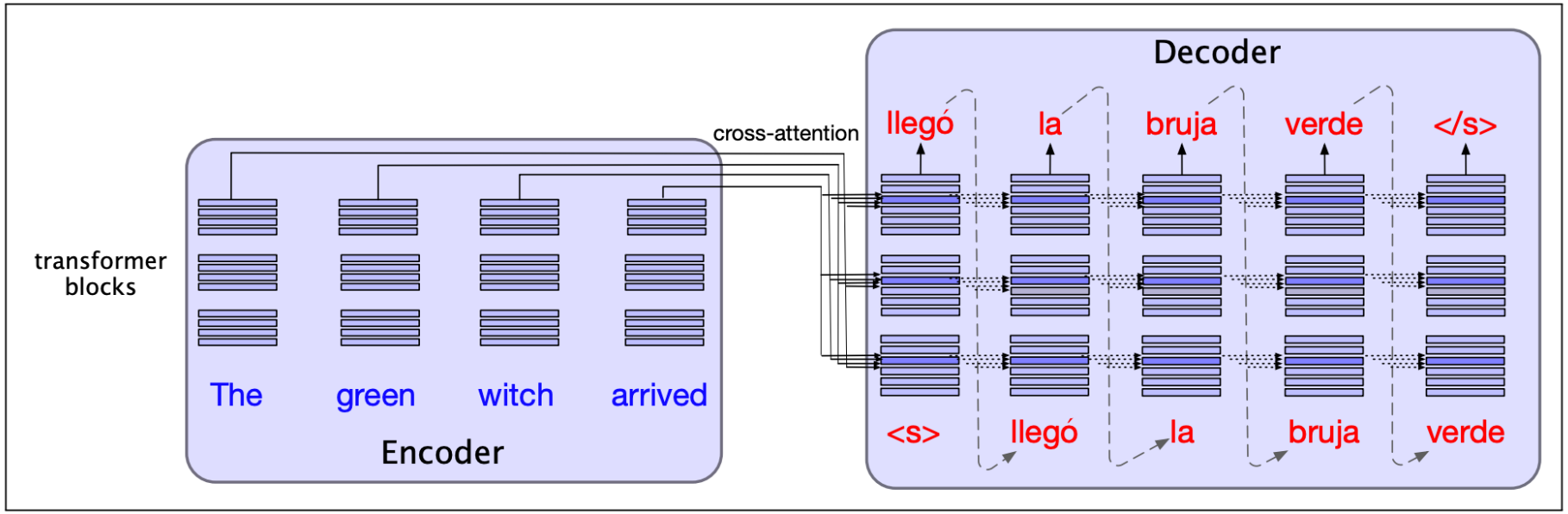
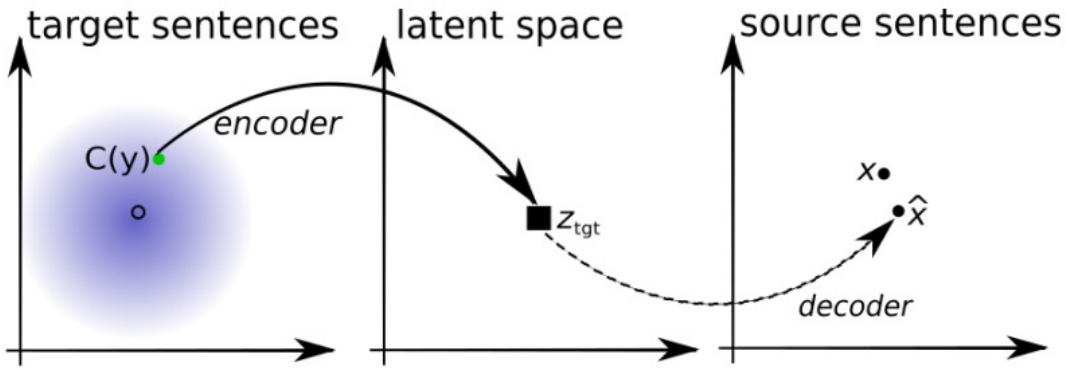


Figure 10.5 The encoder-decoder transformer architecture for machine translation. The encoder uses the transformer blocks we saw in Chapter 9, while the decoder uses a more powerful block with an extra **cross-attention** layer that can attend to all the encoder words. We'll see this in more detail in the next section.

Illustration from https://web.stanford.edu/~jurafsky/slp3/13.pdf

GAN for Unsupervised Machine Translation

- Setup: monolingual corpora, no translation pair
- Encoder and decoder are language agnostic
- Back-translation:
  - Sample source sentence x, translate to target y using current model
  - Train as if supervised case, on $(x, C(y))$, where $C(\cdot)$ adds noise



$$\mathcal{L}(\theta_{enc}, \theta_{dec}, C(y), x)$$

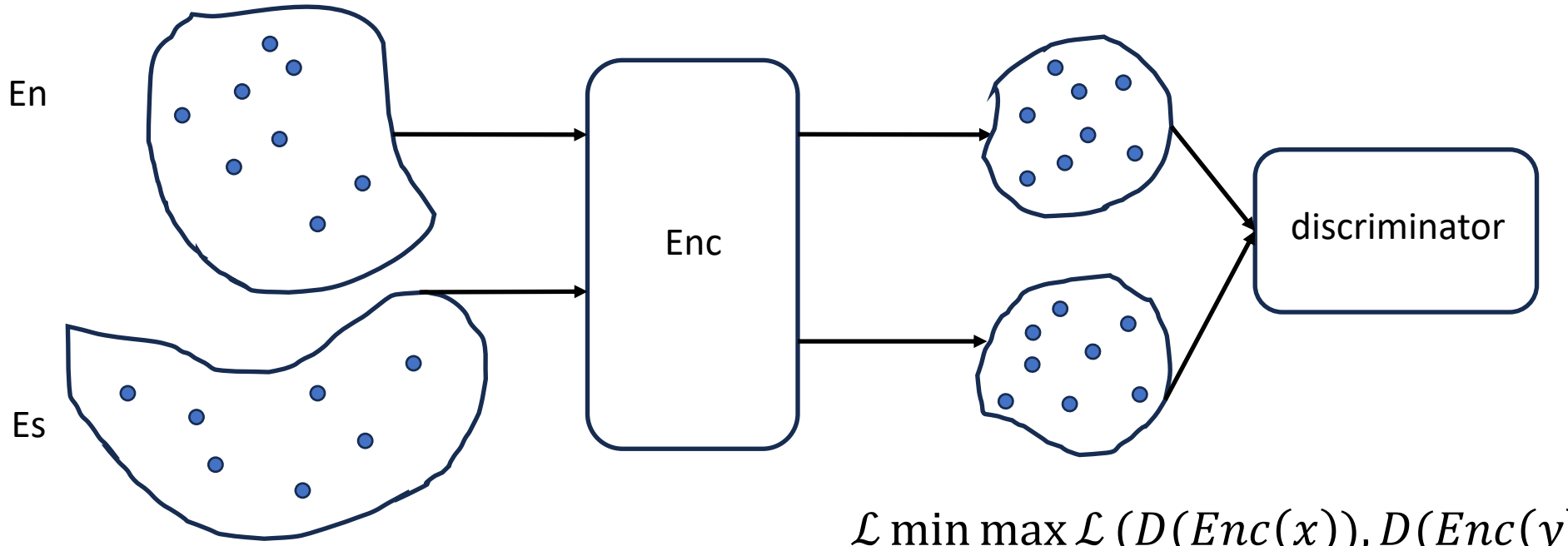
And likewise

$$\mathcal{L}(\theta_{enc}, \theta_{dec}, C(x), y)$$

Lample et. al, 2018

GAN for Unsupervised Machine Translation

- Languages should share some traits
- Enc(x)'s and Enc(y)'s should be indistinguishable



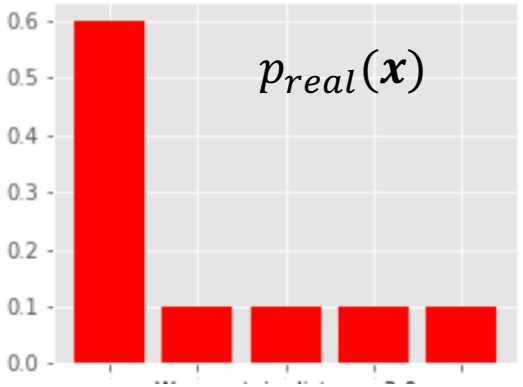
$$\mathcal{L} \min_{\theta_{enc}} \max_{D} \mathcal{L}\left(D(Enc(x)), D(Enc(y))\right)$$

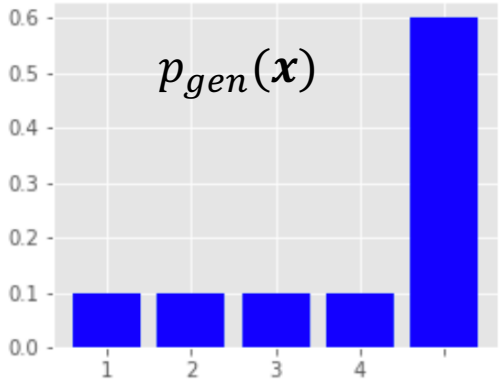
Agenda

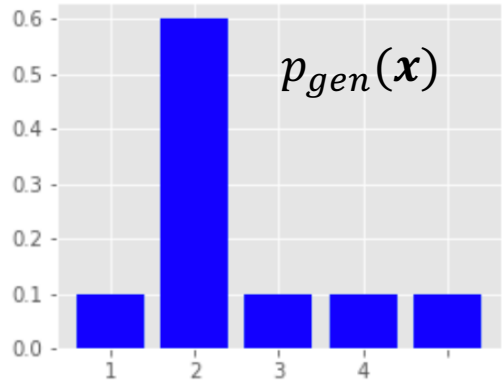
- Generative Adversarial Network (GAN)
- Wasserstein GAN

Improve Objective

- Don’t completely rely on discriminator
- Can we penalize the distance between $p_{real}(\boldsymbol{x})$ and $p_{gen}(\boldsymbol{x})$?
- e.g., KL divergence: $\sum_{\boldsymbol{x}\sim p_{real}} \log \frac{p_{real}(\boldsymbol{x})}{p_{gen}(\boldsymbol{x})}$





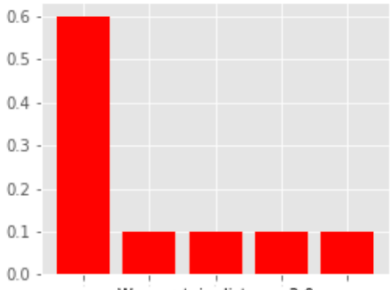
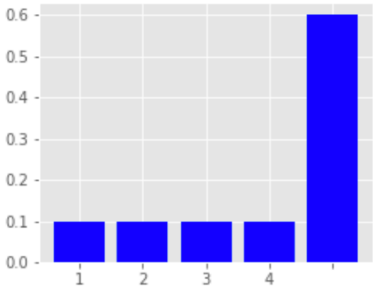
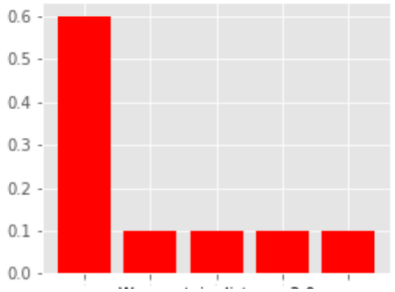
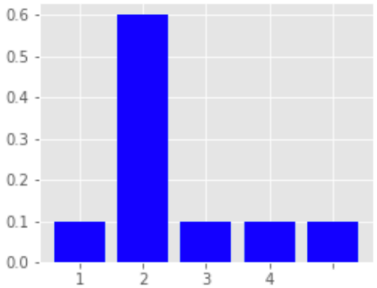


- Same KL-div 0.8959

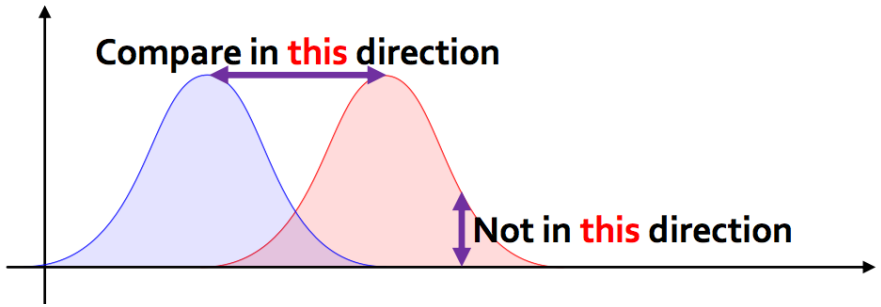
Example from [stackechange]

Issue with KL-div

- But Intuition is

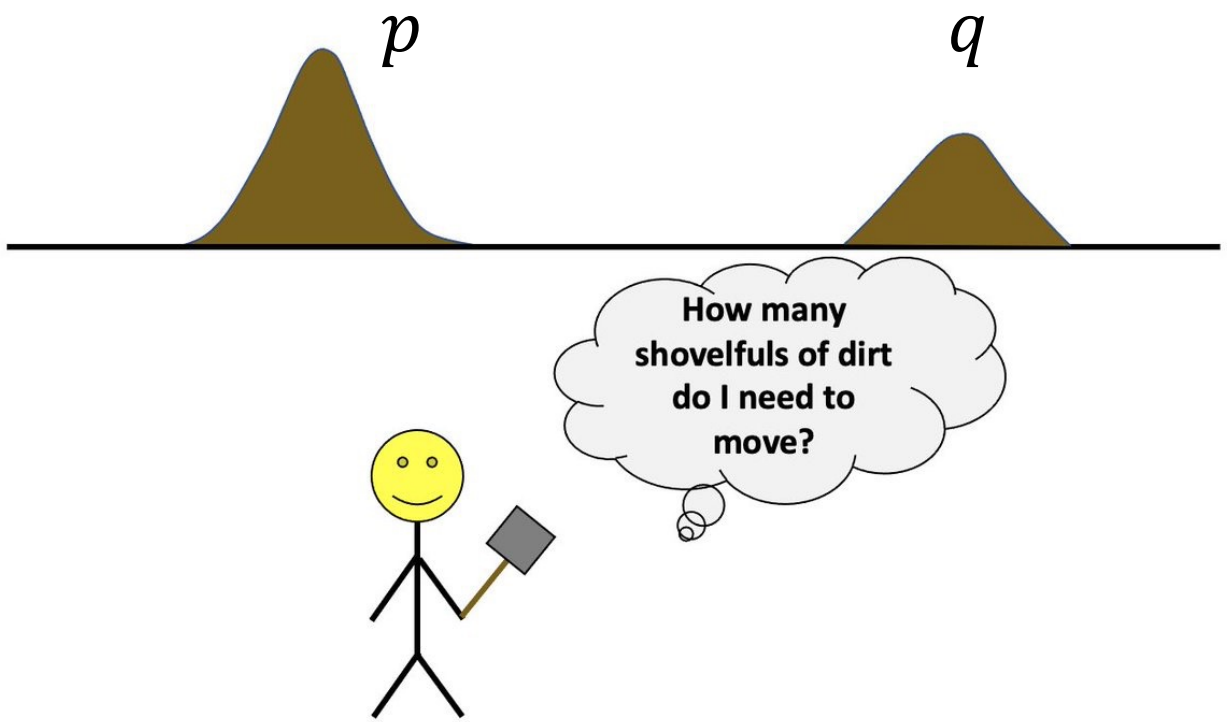
$D($   $) > D($   $)$

- We should consider the difference along the x axis also!



Arjovsky, et. al, 2017

Earth Mover Distance (EMD)



Earth mover's distance is an intuitive metric that denotes the minimum cost flow between two distributions (in other words, minimum work that needs to be done to turn one into the other)

Cartoon from [post on X]

- Discretized case

$$\min_{M} \sum_{i,j} \gamma_{i,j} D_{i,j}$$

s.t. $\sum_i \gamma_{i,j} = q_j$ and $\sum_j \gamma_{i,j} = p_i$

- γ is joint distribution with marginals p and q

Generalize: Wasserstein Distance (WD)

- Generalize the form of $d_{i,j}$, and continuous support

$$WD(p,q) = \min_{\gamma \in \Gamma(p,q)} \mathbb{E}_{(x,y)\sim\gamma}[d(x,y)^p]^{1/p}$$

- So for generative model, we seek to minimize

$$WD\left(p_{real}, p_{gen}\right)$$

- Seems quite hard, but there is a duality

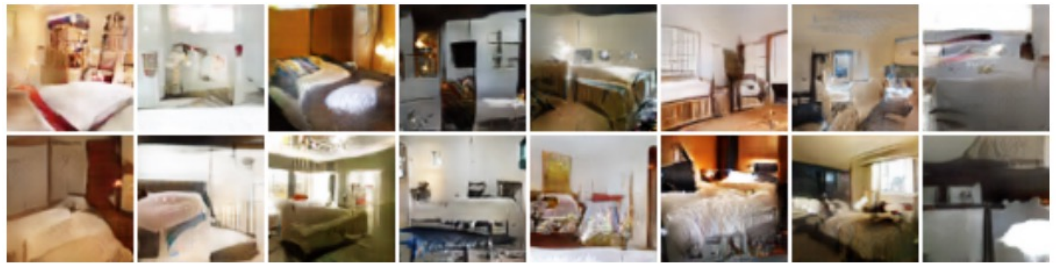
$$WD(p,q) = \max_{f} \mathbb{E}_{x\sim p}\left[f(x)\right] - \mathbb{E}_{x\sim q}[f(x)]$$

- Parameterize $f(\cdot)$ as a deep network with parameter ϕ (Wasserstein-GAN)

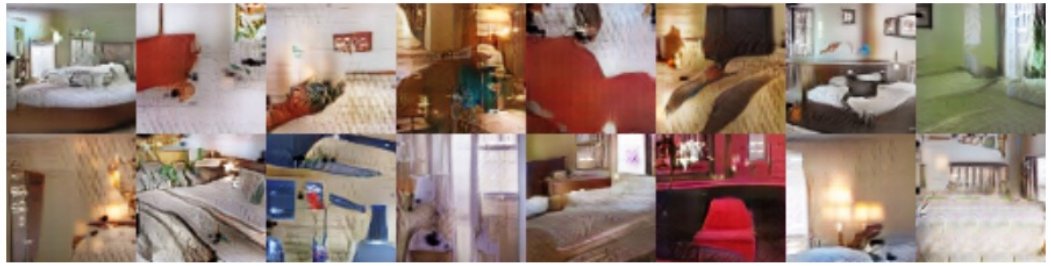
$$\min_{w} \max_{\phi} \frac{1}{N_{real}} \sum_{\boldsymbol{x}:real} f(\boldsymbol{x};\phi) - \frac{1}{N_{gen}} \sum_{z} f(g(z,w))$$

Compare to Traditional GAN

W + DC



DC



W + DC (tweaks removed)

DC (tweaks removed)