Generative Adversarial Networks/GAN

Motivation and Architecture

AUA, DS330 Deep Learning 2021, Hovhannes Tamoyan

Discriminative Modeling

In **Supervised** setup we model a system to predict a class label given an example of input variables. This is the Classification task, which is often referred to as **Discriminative Modeling**. This is because a model must discriminate examples of input variables across classes; it must choose or make a decision as to what class a given example belongs.

Discriminative Modeling - use the training data to find a discriminant function $f(x)$ *that maps each* x *directly onto a class label, thereby combining the inference and decision stages into a single learning problem.*

Generative Modeling

Unsupervised models that summarize the distribution of input variables may be able to be used to create or generate new examples in the input distribution. As such, these types of models are referred to as **Generative models**.

In fact, a really good generative model may be able to generate new examples that are not just plausible, but indistinguishable from real examples from the problem domain.

Approaches that explicitly or implicitly model the distribution of inputs as well as outputs are known as generative models, because by sampling from them it is possible to generate synthetic data points in the input space.

GAN

Generative Adversarial Network, or **GAN**, is a deep-learning-based generative model.

First described by Ian Goodfellow et. al. in 2014.

A standardized approach called Deep Convolutional Generative Adversarial Networks, or **DCGAN**, that led to more stable models was later formalized by Alec Radford, et al. in 2015.

Most **GAN**s today are at least loosely based on the **DCGAN** architecture.

Architecture

The **GAN** model architecture involves two sub-models: a **generator** model for generating new examples and a **discriminator** model for classifying whether generated examples are real, from the domain, or fake, generated by the generator model.

- **Generator**. Model that is used to generate new plausible examples from the problem domain.
- **Discriminator**. Model that is used to classify examples as real (from the domain) or fake (generated).

The Generator

The **generator** model takes a fixed-length random (from Gaussian distribution) vector as input and generates a sample in the domain. After training, points in this multidimensional vector space will correspond to points in the problem domain, forming a compressed representation of the data distribution.

This vector space is referred to as a **latent space**.

Latent variables, are those variables that are important for a domain but are not directly observable.

The Discriminator

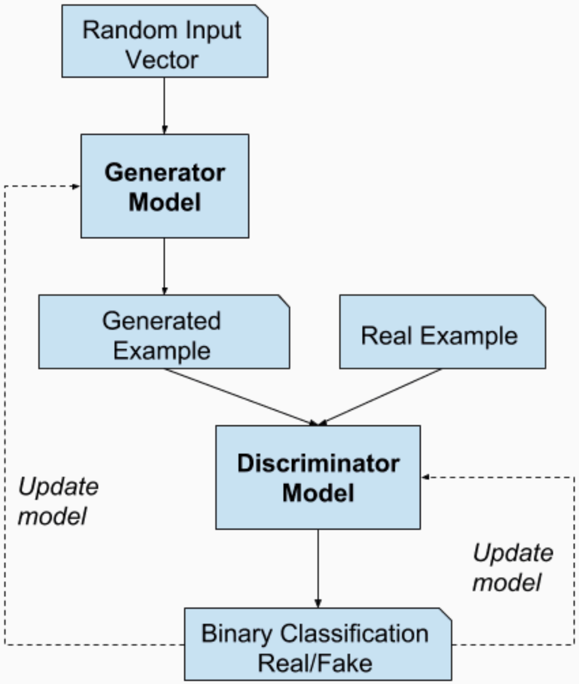
The **discriminator** model takes an example from the domain as input (real or generated) and predicts a binary class label of real or fake (generated).

The real example comes from the **training dataset**. The generated examples are output by the **generator model**.

After the training process, the discriminator model is discarded as we are interested in the generator.

GAN

At a limit, the **generator** generates perfect replicas from the input domain every time, and the **discriminator** cannot tell the difference and predicts “unsure” (e.g. 50% for real and fake) in every case. This is just an example of an idealized case; we do not need to get to this point to arrive at a useful generator model.



The Objective of a GAN

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

Where:

G - the Generator

D - the Discriminator

z - randomly generated noise (p_z)

x - real data (p_{data})

GANs and CNNs

GANs typically work with image data and use Convolutional Neural Networks, or CNNs, as the generator and discriminator models.

The first description of the technique was in the field of CV and used CNNs and image data, and because of the remarkable progress that has been seen in recent years using CNNs more generally to achieve state-of-the-art results on a suite of computer vision tasks such as object detection and face recognition.

Conditional GANs

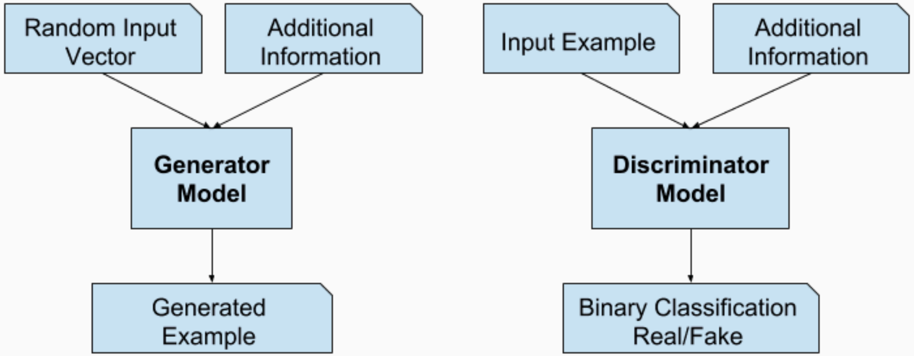
The generative model can be trained to generate new examples from the input domain, where the input, the random vector from the latent space, is provided with (conditioned by) some **additional input**.

The **additional input** could be a class value, such as male or female in the generation of photographs of people, or a digit, in the case of generating images of handwritten digits.

The discriminator is also conditioned, meaning that it is provided both with an input image that is either real or fake and the **additional input**.

Conditional GANs contd.

Taken one step further, the GAN models can be conditioned on an example from the domain, such as an image. This allows for applications of GANs such as text-to-image translation, or image-to-image translation, style transfer, photo colorization, transforming photos from summer to winter or day to night, and so on.



Thank You