

★ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Decoding the Basic Math in GAN — Simplified Version

What is the mathematical foundation of generative adversarial networks?



Lakshmi Ajay · Follow

Published in Towards Data Science · 5 min read · Feb 24, 2022



200

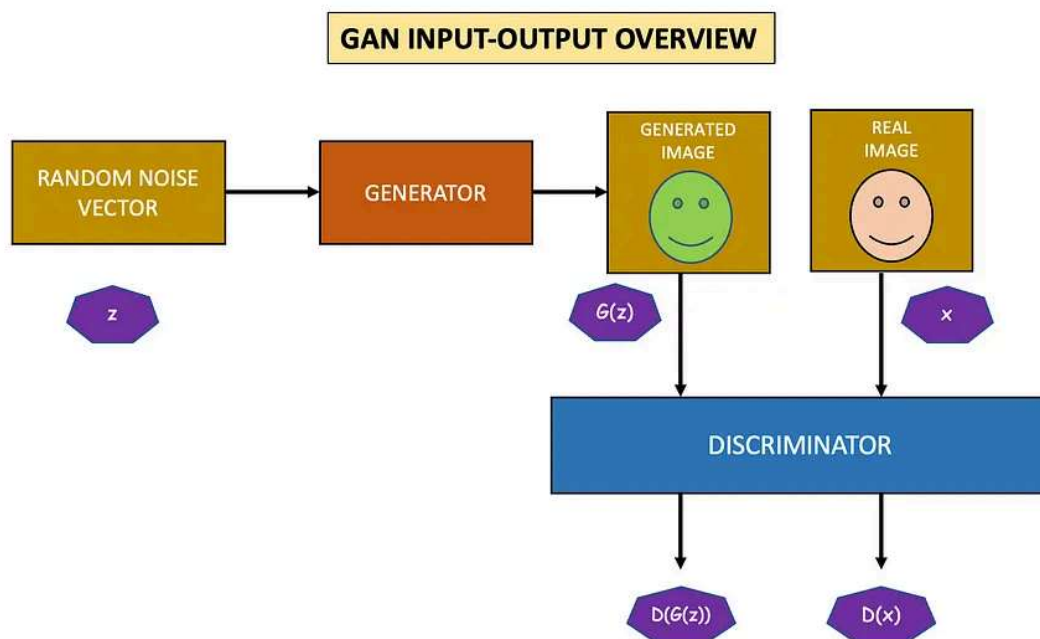


Generative Adversarial Networks or GANs have been a revolution in deep learning over the last decade. Since the first version of GAN that was released in 2014 by Ian Goodfellow et al., there have been numerous variants of GAN with some mind-blowing applications.

In this article we shall try to understand the basic mathematical foundation behind a GAN in simple terms.

To get a high-level intuition of how GANs work, it is recommended that you first browse through the following article: [An intuitive introduction to GANs](#)

GAN Overview



GAN input output flow (Image by Author)

The generator takes a random vector $[z]$ as input and generates an output image $[G(z)]$. The discriminator takes either the generated image $[G(z)]$ or a real image $[x]$ as input and generates an output $[D]$.

During the discriminator training, the generator's weights and biases are frozen. The discriminator is penalised when it predicts a real image as fake or a fake image as real. The weights and biases of the discriminator are updated through back propagation from the discriminator loss.

Similarly during the generator training, the discriminator's weights and biases are frozen. The goal of the generator is to deceive the discriminator. Hence the generator is penalised if the discriminator can identify that the image generated by the generator is a fake image. In this case the loss is back propagated through the generator network to update its weights and biases.

The two-player MiniMax Game

A GAN can be conceptually thought of as a minimax game played between the generator model and the discriminator model. Both models are trained simultaneously where one model tries to minimise the loss while the other tries to maximise the loss.

As per the original GAN paper, the loss function for GAN is as below

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

Image from [GAN — 2014 paper](#)

Let us try to derive this equation to understand it better.

Binary Cross-Entropy Loss

Both the generator and the discriminator use the binary cross-entropy loss to train the models. In general, binary cross entropy loss can be written as:

$$L(y, \hat{y}) = -(y \cdot \log \hat{y} + (1 - y) \cdot \log(1 - \hat{y}))$$

Discriminator Loss

The discriminator can have two possible inputs, real or fake.

1. For a real input, $y = 1$. Substituting this in the binary cross entropy loss function gives the below loss value:

$$L(1, \hat{y}) = -(1 \cdot \log(D(x)) + (1 - 1) \cdot \log(1 - D(x))) = -\log(D(x))$$

Where, $\hat{y} = D(x) \rightarrow$ output of the discriminator for a real input image

2. Similarly for a fake input, $y = 0$, which gives the below loss value:

$$L(0, \hat{y}) = -(0 \cdot \log(D(G(z))) + (1 - 0) \cdot \log(1 - D(G(z)))) = -\log(1 - D(G(z)))$$

Where, $\hat{y} = D(G(z))$

z is the input noise vector provided to the generator

$G(z)$ is the output of the generator

$D(G(z))$ is the output of the discriminator with generated data as the input

Combining both the above losses for the discriminator, one for a real input and the other for a fake input gives the below discriminator loss

$$L(\text{Discriminator}) = -[\log(D(x)) + \log(1 - D(G(z)))]$$

The goal of the discriminator is to minimise this loss,

$$L(\text{Discriminator}) = \min[-[\log(D(x)) + \log(1 - D(G(z)))]]$$

Removing the negative sign will change min to max, hence the final discriminator loss for a single datapoint can be written as:

$$L(\text{Discriminator}) = \max[\log(D(x)) + \log(1 - D(G(z)))]$$

Generator Loss

For the generator, loss is calculated from the discriminator loss. During the training of the generator, the discriminator is frozen. Hence only one input is possible to the discriminator, which is the fake input. This nullifies the first term in the discriminator loss equation to 0.

The generator is trying to fool the discriminator into classifying the fake data as real data. This implies that the generator tries to minimise the second term in the discriminator loss equation. The generator loss function for single generated datapoint can be written as:

Open in app ↗



Search

Write



GAN — Loss Equation

Combining both the losses, the discriminator loss and the generator loss, gives us an equation as below for a single datapoint.

$$L(GAN) = GminDmax [\log(D(x)) + \log(1 - D(G(z)))]$$

This is the minimax game played between the generator and the discriminator. The generator tries to minimise the loss whereas the discriminator tries to maximise the loss.

Generalising this equation to the expected values (E) for multiple datapoints gives the final equation as below,

$$L(GAN) = GminDmax [E_{x(real)} \log(D(x)) + E_{z(fake)} \log(1 - D(G(x)))]$$

Now compare this with the loss function given in the GAN paper to understand it better.

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

Image from [GAN — 2014 paper](#)

Limitations of GAN

Here are some of the commonly faced issues in GAN

1. Vanishing gradient issue due to which the generator training might fail
2. Model collapse where the generator might repeatedly create the same output
3. Failure to converge due to which the discriminator feedback can get less meaningful to the generator thus impacting its quality

Conclusion

There has been lot of improvements in GANs since its inception and researchers have identified ways to overcome these limitations to a large extent. Choosing the right loss function is one of the hot research topics and many alternatives are being proposed and evaluated.

Hope you enjoyed this article. Have fun in diving deeper into the various flavours of GAN.

Deep Learning

Gans

Generative Adversarial

Machine Learning

Computer Vision



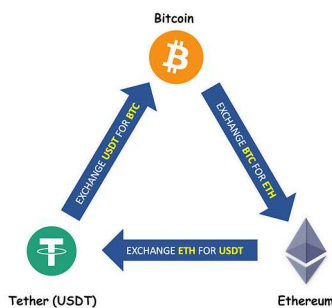
Written by Lakshmi Ajay

Follow

1.1K Followers · Writer for Towards Data Science

Data Science Enthusiast | Passionate Programmer | <https://www.linkedin.com/in/lakshmi-e-2321943/>

More from Lakshmi Ajay and Towards Data Science



Lakshmi Ajay in Geek Culture

Automated Triangular Arbitrage of Cryptos in 4 steps



Cristian Leo in Towards Data Science

The Math Behind Neural Networks

Dive into Neural Networks, the backbone of modern AI, understand its mathematics,...

Build your own arbitrage trading algorithm in Python

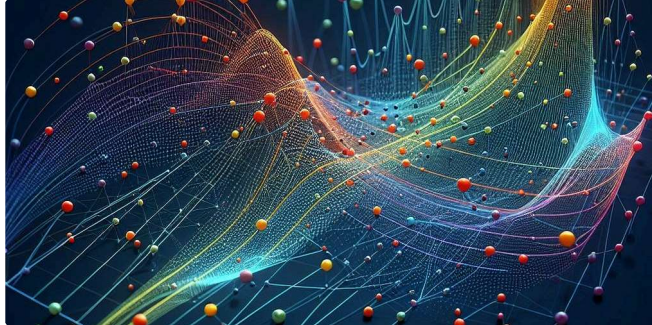
7 min read · Mar 3, 2022



843



21



Tim Sumner in Towards Data Science

A New Coefficient of Correlation

What if you were told there exists a new way to measure the relationship between two...

10 min read · Mar 31, 2024



2.7K



35



🌟 · 28 min read · Mar 29, 2024



2.9K



20



Lakshmi Ajay in Towards Data Science

Creating a Fantasy Cricket Team— Application of Linear Programming

A step-by-step guide on Linear Programming using Python (PuLP package)

11 min read · Jan 9, 2021



73



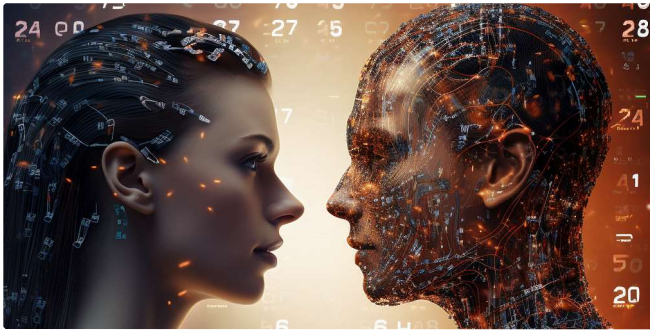
2




See all from Lakshmi Ajay

See all from Towards Data Science

Recommended from Medium



 Marco Del Pra

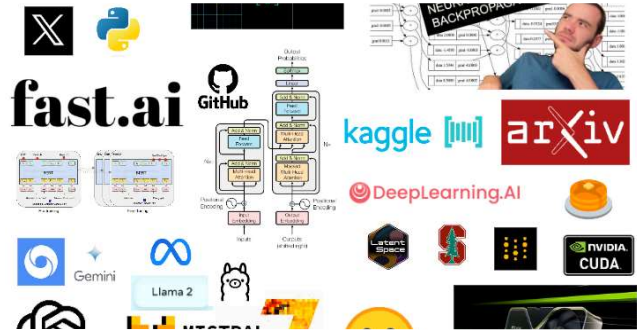
Generative Adversarial Networks

Generative Adversarial Networks (GANs) have garnered significant attention in the field...

14 min read · Oct 30, 2023



22



 Benedict Neo in bitgrit Data Science Publication

Roadmap to Learn AI in 2024

A free curriculum for hackers and programmers to learn AI

11 min read · Mar 11, 2024



10.2K



110



Lists



Predictive Modeling w/ Python

20 stories · 1113 saves



Practical Guides to Machine Learning

10 stories · 1334 saves



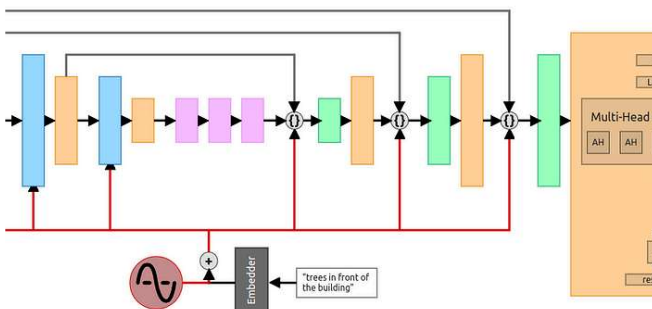
Natural Language Processing


1386 stories · 880 saves



data science and AI

40 stories · 130 saves



 Kemal Erdem (burnpiro)



 Mirko Pet... in Mirko Peters—Data & Analytics ...

Step by Step visual introduction to Diffusion Models.

How the diffusion models works under the hood? Visual guide to diffusion process and...

15 min read · Nov 10, 2023



259



1



Jonathan Kernes in Towards Data Science

Diffusion Models

What are they, how do they work, and why now?

23 min read · Dec 13, 2022



367



5



Understanding Generative Adversarial Networks (GAN) in...

Generative Adversarial Networks, or GANs, have revolutionized the field of machine...



10 min read · Mar 2, 2024



23



Detect



Condense



DhanushKumar

Convolution and ReLU

Convolution is a mathematical operation that combines two functions to produce a third. I...

9 min read · Nov 23, 2023



54



See more recommendations