



Keras ile Derin Evriřimsel GAN

Enes avuş

Derin Öğrenme ve Evriřimli Sinir Ağları - Proje

İçerik

- Çalışma Hakkında
- Generative Adversarial Networks (GANs) Nedir?
- Veri Seti İncelemesi
- Çalışmadaki Ön İşlemler
- GAN – Generator
- GAN – Discriminator
- GAN Mimarisini Anlamak
- Model Katmanları İncelemesi – Generator
- Model Katmanları İncelemesi – Discriminator
- Çıktılar – Sonuçların Değerlendirilmesi
- Proje Çıktıları
- Kaynaklar

Çalışma Hakkında

- Bu sunumda anlatılacak olan çalışma Generative Adversarial Networks(GANs) – Çekişmeli Üretici Ağları hakkındadır.
- Ele alınan çalışmada hedeflenen, Fashion MNIST veri seti yani kıyafetler, pantolonlar, ayakkabılar gibi giyim ürünlerine ait görseller içeren veri seti kullanılarak, GAN ağları yardımı ile bu veri setinde olmayan ama benzeri ve gerçekçi görsellerin çıktı olarak elde edilmesidir.
- GANlar, modellerin kendi kendine bir şeyler üretmesi ile ilgili çalışmalarda kullanıldığından dolayı UnSupervised(Gözetimsiz Öğrenme) alt sınıfında yer almaktadır.
- Sonraki slaytlarda, GANlar hakkında tanımlamalar yapılacak, veri seti incelenecek, kullanılan derin öğrenme ağı incelenecek, çıktılar üzerinden yorumlamalar yapılacaktır.

Generative Adversarial Networks (GANs)

Nedir?

- GAN, derin öğrenme çalışmalarında çok kez adının duyulduğu ve 'Derin Öğrenme' kitabının yazarlarından biri olan Ian Goodfellow tarafından 2014 yılında geliştirilmiştir.
- GAN'lar derin öğrenme metodlarını kullanarak, gerçek hayatta hiç varolmamış fakat gördüğümüzde gerçek mi sahte mi olduğunu ayırt edemeyeceğimiz kadar gerçekçi görseller üretebilen ağlardır.
- Aynı zamanda GANlar, veri artırma amaçlı kullanılabilmekte, bu sayede farklı derin öğrenme uygulamalarına dolaylı yoldan dahil olabilmektedir. Örneğin yüz tanıma algoritması için GANlar tarafından oluşturulmuş görseller kullanılarak geliştirilmiş projeler bulunmaktadır.

Generative Adversarial Networks (GANs)

Nedir?

- ➔ GAN'lardaki çalışma prensibi kısaca şöyle açıklanabilir; bir derin öğrenme ağı rastgele piksel değerleri ile görseller üretmektedir, bizim belirleyeceğimizi veri setindeki içerikler ile (çiçek, kitap, rakam vb) bu görselleri karşılaştıran başka bir derin öğrenme ağı da üretilen görsellerin gerçek mi sahte mi olduğunu değerlendirmekte ve ona göre geri bildirim yapmaktadır.
- ➔ Bu geri bildirimler doğrultusunda, üretici ağıımız kendi içerisinde değişiklikler yapmakta ve her epoch'da daha gerçekçi görseller üretebilecek hale gelmektedir. En sonunda ise karşılaştırmaya yapan ağıımız, görsellerin gerçek mi sahte mi olduğunu ayıramayacak duruma gelebilmektedir.
- ➔ Yandaki görselde GAN tarafından oluşturulmuş sahte bir insan yüzü görülmektedir.



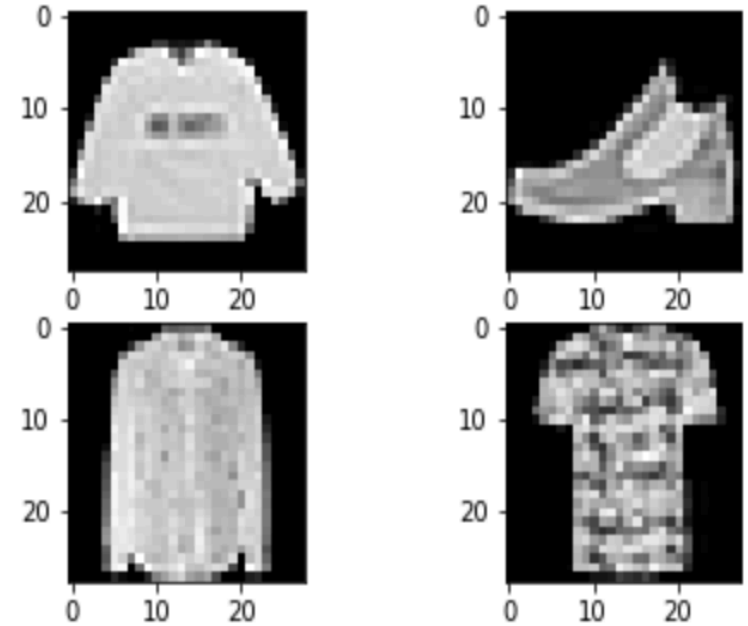
Veri Seti İncelemesi

- Veri setinde rakamlardan oluşan MNIST datasetine benzer yapıda görseller bulunmaktadır.
- Bu görseller siyah-beyaz-gri olacak şekilde basitleştirilmiş giyim-kıyafet figürlerinden oluşmaktadır.
- GAN ile kullanımının uygun olmasının sebebi sınıfların rakamlardaki gibi kendine özgün olmalarıdır.
- Örneğin bir pantolonun yapısı bellidir ve üretici ağ yapısı geri bildirimler sayesinde gerçekçi çıktılar oluşturabilecektir.



Veri Seti İncelemesi

- ➔ Görseller 28 x 28 piksel boyutlarındadır.
- ➔ Ceket, çanta, ayakkabı elbise, t-shirt benzeri eşyalardan oluşun 10 sınıf içermektedir.
- ➔ 60.000 eğitim – 10.000 test verisi olarak ikiye ayrılmıştır.

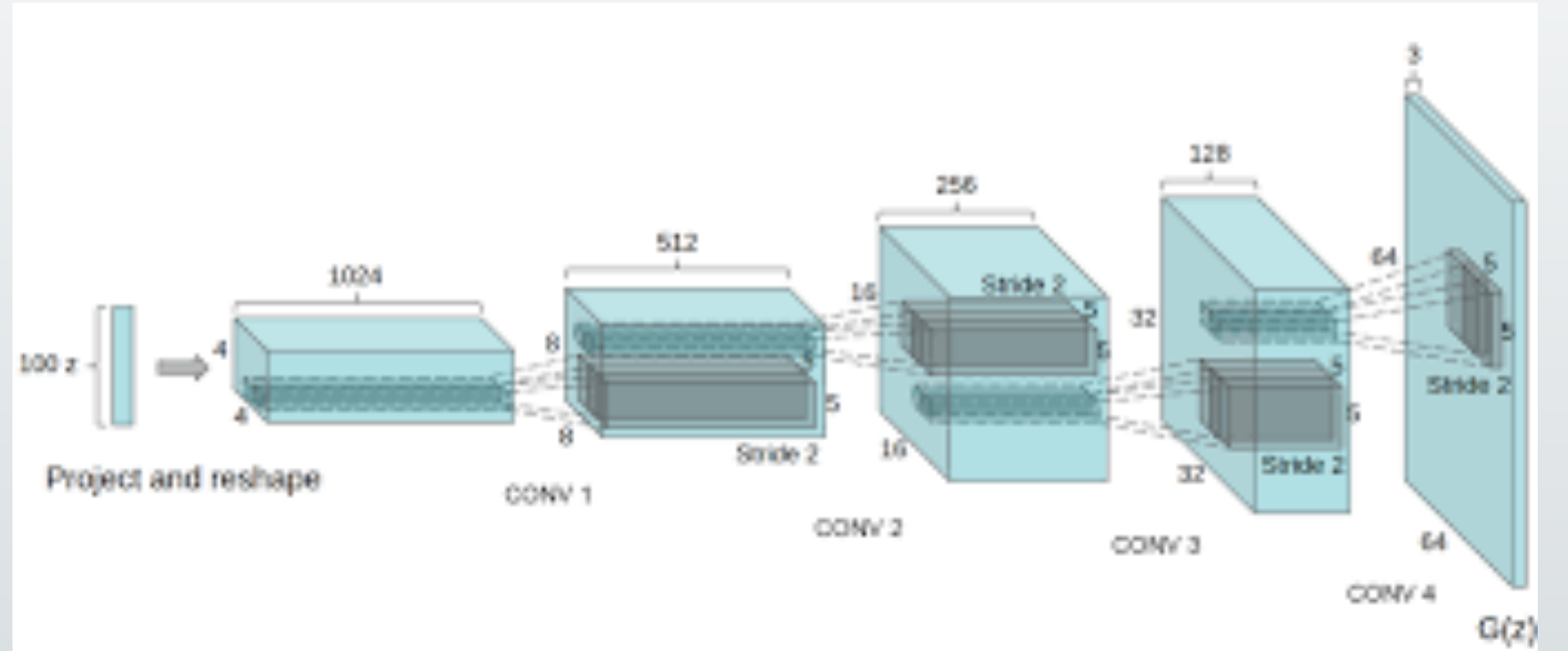


Çalışmadaki Ön İşlemler

- GANlardaki veri ön işlemleri ile dönüştürme adımları diğer görüntü işleme çalışmaları ile benzer şekildedir. Görseller hafızaya yüklenir, fashion mnist keras içerisinde mevcuttur.
- Görseller yine eğitim ve test verileri şeklinde ayrılmış olmalıdır.
- Daha sonra modellerin anlayacağı şekle getirilmek üzere piksellerin yeniden kodlanması(encoding) ve tip dönüşümleri gerçekleştirilir.
- Modelin mimarisini tanımlamadan önce birkaç hiperparametre tanımlamaları yapılmalıdır.

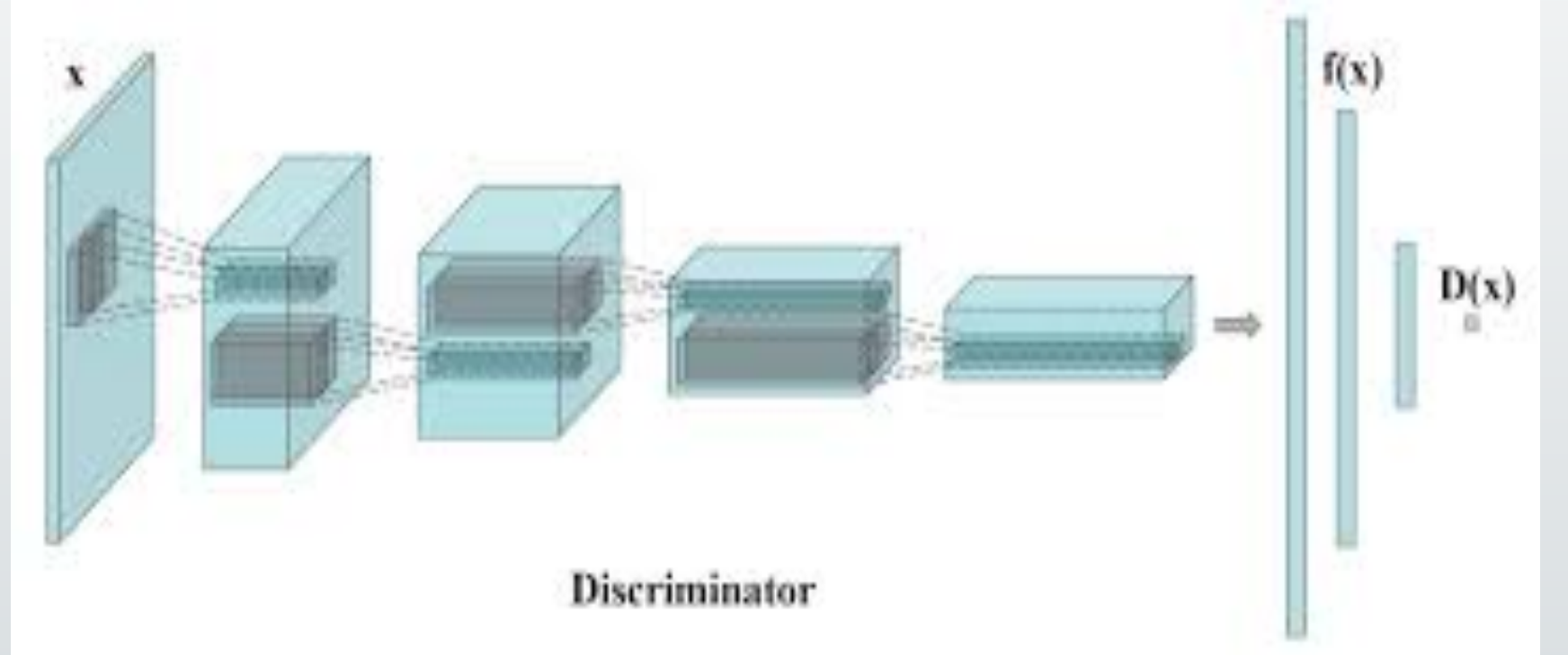
GAN - Generator

- Derin öğrenme ağında ise farklılıklar bulunmaktadır. Mimaride iki farklı ağ yapısı oluşturulmalıdır.
- Bir tanesi 'Generator' yani görsel üretimi ile uğraşacak olan ağ yapısıdır. Aşağıdaki görselde, evrişimli ağ katmanları kullanılarak oluşturulmuş bir generator örneği bulunmaktadır.



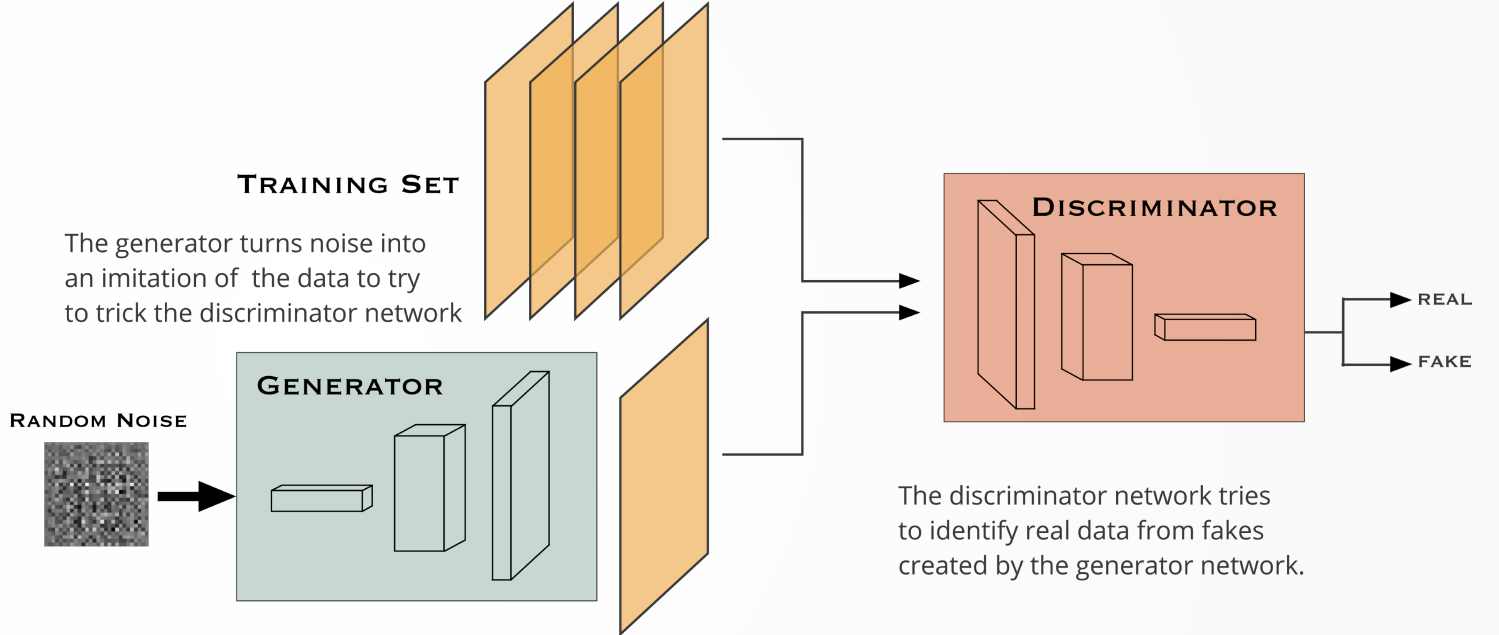
GAN - Discriminator

- İkincisi 'Discriminator' olarak adlandırılmakta ve Generator tarafından üretilen görsellerin gerçek mi yoksa sahte mi olduğunu ayırt edilmesi ile uğraşan ağ yapısıdır.
- Aşağıdaki görselde, yine evrişimli ağ katmanları kullanılarak oluşturulmuş bir Discriminator yapısı örneği bulunmaktadır.



GAN Mimarisini Anlamak

- ▶ Çalışmayı daha iyi anlayabilmek için GAN mekanizmasının bir bütün olarak görselleştirilmesi önemlidir.
- ▶ Yandaki görselde, rastgele bir gürültü ile piksellerin üretimesi, eğitim veri seti ile bu üretilen görüntünün discriminator tarafından değerlendirilmesi bir bütün olarak görülmektedir.



Model Katmanları İncelemesi - Generator

- ➔ Yanda görüleceği üzere GANlarda CNN mimarisi kullanımı diğer derin öğrenme uygulamalarında olduğu gibidir.
- ➔ Görsel boyutları eşleştirecek şekilde başlangıçta belirlenir ve Generator bu boyutlara göre görseller üretmek üzere farklı gürültü değerleri kullanarak pikseller üretir.
- ➔ Aktivasyon fonksiyonları olarak Keras üzerindeki çeşitli fonksiyonlar probleme uygun olacak şekilde belirlenmelidir.
- ➔ Dikkat edilmesi gereken nokta Transpoze yani normal evrişimli ağa göre zıt yönde olacak şekilde işlem yapan katmanın kullanılmasıdır. Deconvolution olarak adlandırılır. Generator üretmeye çalıştığından dolayı zıt yönde bir işlem yapmaktadır.

```
generator = keras.models.Sequential([
    keras.layers.Dense(7 * 7 * 128, input_shape =
[num_features]),
    keras.layers.Reshape([7, 7, 128]),
    keras.layers.BatchNormalization(),
    keras.layers.Conv2DTranspose(
        64, (5, 5), (2, 2), padding = "same",
activation = "selu"),
    keras.layers.BatchNormalization(),
    keras.layers.Conv2DTranspose(
        1, (5, 5), (2, 2), padding = "same",
activation = "tanh"),
])
```

Model Katmanları İncelemesi - Discriminator

- ➡ Discriminator katmanlarında ise yine keras ağ yapılarından katmanlar kullanılmıştır.
- ➡ Bu kez normal Conv2D kullanılmıştır, bu algoritma standart bir evrişimli ağdaki gibi görsel girişi almaktadır (Generator tarafından üretilen input ile veri setinden gelen inputlar) ve bunu gerçek mi sahte mi şeklinde değerlendirmektedir.
- ➡ Ara katmanlar olarak ReLu, DropOut katmanları eklenebilir ve en sonda düzleştirme işlemi için bir Flatten katmanı kullanılmıştır.
- ➡ Bu bir tür binary işlem olduğundan dolayı(real-fake) çıkış katmanında 'sigmoid' kullanılmıştır.
- ➡ Model mimarisi dışında kalan kısımlar standart CNN projelerindeki gibidir

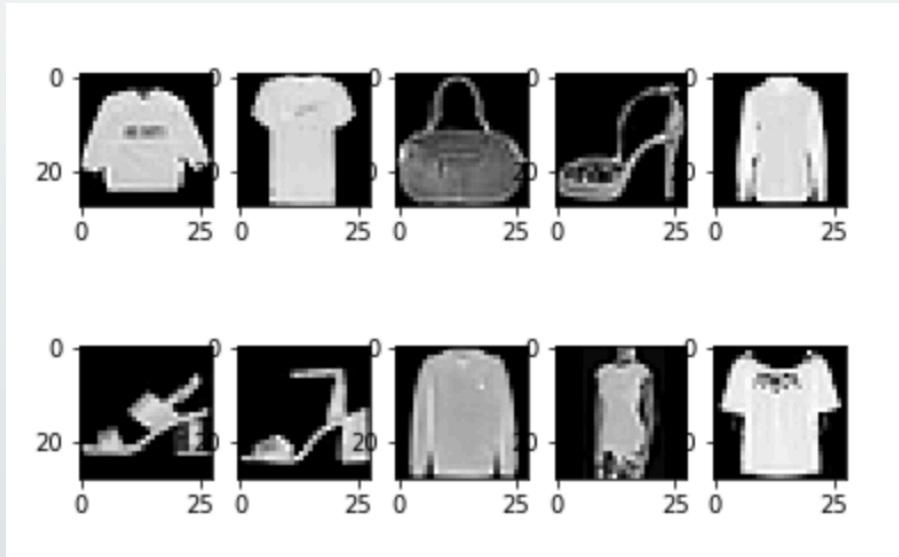
```
discriminator = keras.models.Sequential([  
    keras.layers.Conv2D(64, (5, 5), (2, 2),  
        padding = "same", input_shape = [28, 28, 1]),  
    keras.layers.LeakyReLU(0.2),  
    keras.layers.Dropout(0.3),  
    keras.layers.Conv2D(128, (5, 5), (2, 2),  
        padding = "same"),  
    keras.layers.LeakyReLU(0.2),  
    keras.layers.Dropout(0.3),  
    keras.layers.Flatten(),  
    keras.layers.Dense(1, activation = 'sigmoid')  
])
```

Çıktılar – Sonuçların Değerlendirilmesi

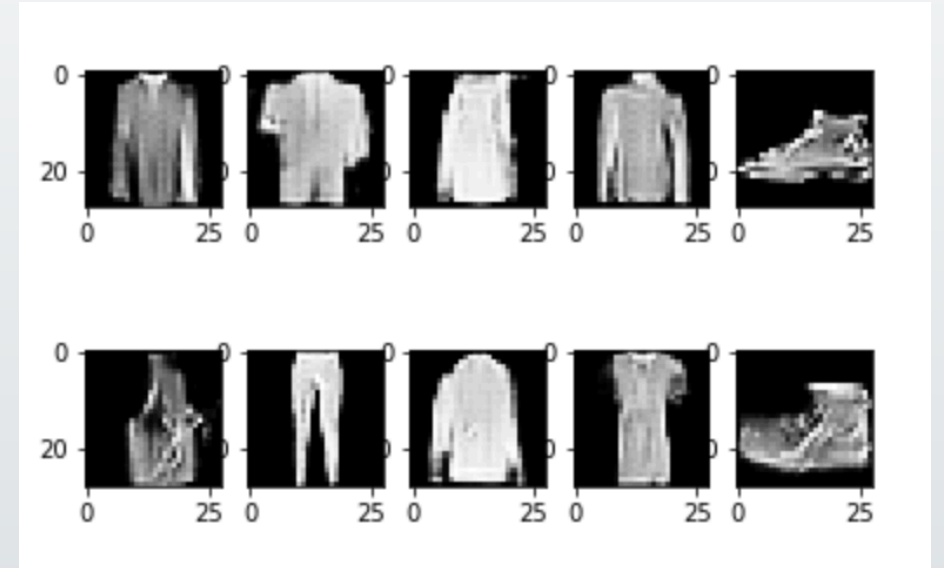
- Çalışmanın çıktıları olarak discriminator değerlendirmelerine göre modelin %82 gibi bir başarımlı değeri (accuracy) ulaşabildiği görülmüştür.
- Kullanılan veri setinin renk skalasının sade olmasından dolayı bir önceki slaytlarda da gösterilen ağ mimarisinin çok fazla kompleks bir yapıda olmamasına rağmen 80 ve üzeri başarımlı değerlerini yakalayabildiği görülmektedir.
- GAN eğitimlerinde, geliştiriciler genelde belli miktarda çıktıyı her epoch sonunda görselleştirmektedirler. Böylelikle modelin gelişmesi baştan sonra gözlemlenebilmektedir.
- Bu çalışmadaki çıktıları adım adım incelediğimde farkettiğim şey; modelin, kıyafetin genel hatlarını çoğunlukla başarılı bir şekilde oluşturabildiği fakat görselin bazı piksel gruplarında kararsız/başarısız kaldığı oldu. Bu durumun sebebi görsellerin çok küçük boyutlarda olması olabilir ya da renk skalasının rgb gibi geniş olmamasından dolayı bazı detayları kaçırmış olabilir.
- Genel anlamda modelin oluşturduğu görseller gerçeğe çok yakın görünmektedir. Sonuç olarak, oluşturulan görselin bir ayakkabı ya da elbise olduğu rahatlıkla anlaşılmaktadır.

Model Çıktıları – Real/Fake Karşılaştırması

Gerçek Görseller



Üretilen Görseller



Kaynaklar

- [Veri Seti](#)
- [Transpoze Evrişimli Sinir Ağı Katmanı](#)
- [DCGAN – Deep Convolutional Gan with Keras](#)
- [Introduction to GANs](#)
- [Generative Advertiserial Networks Nedir? – GAN Nedir?](#)
- [Deep Convolutional Generative Adversarial Networks](#)

Enes Çavuş
Derin Öğrenme ve Evrişimli Sinir Ağları - Proje