Üretken Çekişmeli Ağlar

Oğuzhan Ercan
Bilgisayar Mühendisliği
Yıldız Teknik Üniversitesi
İstanbul, Türkiye
oguzhanercancs@gmail.com

***Özet*— Üretken Çekişmeli Ağlar (GANs) [1] 2014 yılında yayınlanmasından itibaren derin öğrenme alanındaki en popüler araştırma alanlarından olmuştur. Ian Goodfellow tarafından yayınlanan GANs günümüzde otonom araçlar, savunma sanayii, Snapchat filtreleri, deepfake, veri arttırma gibi bir çok alanda kullanılmaktadır.**

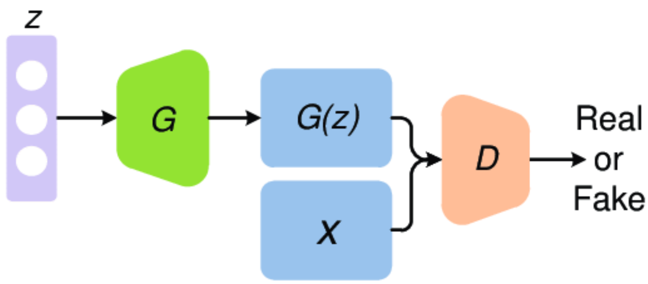
Anahtar Kelimeler—Üretken Çekişmeli Ağlar, Derin Öğrenme, Büyük Veri, Wasserstein, İkili Çapraz Entropi Hata Fonksiyonu , Konvolüsyon

I. GIRIS

Üretken çekişmeli ağlar doğal veriden sentetik veri üretmeyi hedefleyen bir derin öğrenme modelidir. Uniform bir dağılımı doğal verinin dağılımına yaklaştırarak sentetik verinleri doğal verilere benzetir. Üretken (Generative) ve ayırıcı (Discriminator) olarak iki farklı modelin çekişmesi sonucu üretken modelin çıktıları doğal verilere yaklaşır.

II. ÜRETKEN ÇEKİŞMELİ AĞLAR

A. Vanilla Üretken Çekişmeli Ağlar

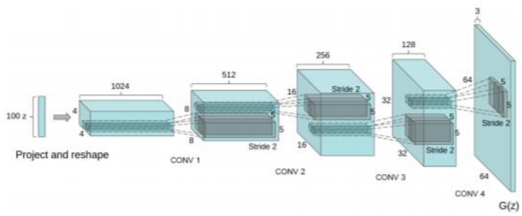


Resim 1

Vanilla Üretken Çekişmeli Ağlar [1] noise olarak adlandırılan uniform vektörü üretken modelde yukarı örnekleyerek doğal verimizin boyutlarına getirir. Ayırıcı model sentetik veri ve doğal veriyi sınıflandırması için kurgulanmıştır. Ayırıcı model doğal ve sentetik veriden gelen giriş değerlerini yapay sinir ağı aracılığıyla doğru şekilde sınıflandıracak şekilde parametrelerini geri yayılım algoritması ile optimize eder. Üretici model ayırıcı modeli kandırmayı hedeflemektedir ve ayırıcı modelden aldığı geri beslemeye göre geri yayılım algoritması ile parametrelerini günceller. Parametreler uniform dağılımını doğal verimizin dağılımına yaklaştırır. Üretken çekişmeli ağların orijinal makalesinde geçen implementasyonunu uyguladığımızda ürettiğimiz sonuçlar resim 2' de verilmiştir.

Resim 2

B. Derin Konvolüsyonel Üretken Çekişmeli Ağlar



Resim 3

Derin konvolüsyonel üretken çekişmeli ağlar [2] üretken çekişmeli ağlar için bir dönüm noktası oldu. Üretken çekişmeli ağlar üzerine yapılan çalışmaların büyük bir kısmı görüntü verileri üzerinde. Konvolüsyonel ağlar görüntü verileri üzerinde lokal örüntüleri yakalama ve bunları işlemedeki başarısını üretken çekişmeli ağlarda yakalamak için hem yukarı örnekleme hem de doğal veriler ile semtetik verileri ayırmak için 2016 yılında kullanılmaya başlandı. Sonrasında yayınlanan bir çok çalışmada konvolüsyonel üretici ağlar model başarılarını kıyaslamada insiyatif olarak kabul edildi. Üretken çekişmeli ağlar mimarisindendeki üretken modelde fractional strided convolution kullanılarak yukarı örnekleme, konvolüsyon kullanılarak aşağı örnekleme ve batch normalizasyon ile modellerin mimarilerinde değişiklik yapılarak derin konvolüsyonel üretken çekişmeli ağlar sunuldu. Derin konvolüsyonel üretken çekişmeli ağların sonuçları resim 4'de verilmiştir.

Resim 3

Üretken Çekişmeli Ağlar implementasyonu: https://github.com/Oguzhanercan/GANs/blob/master/tutorial/GANs_Türkçe/GANs.ipynb

Derin Konvolüsyonel Üretken Çekişmeli Ağlar implementasyonu: https://github.com/Oguzhanercan/GANs/blob/master/tutorial/GANs_Türkçe/DCGANs.ipynb

C. *Wasserstein Üretken Çekişmeli Ağlar*

İkili çapraz entropi fonksiyonu Kullback-Leibler ıraksamasını temel alır ve bu ıraksama dağılımların ayrık olmadığını var sayar. Görüntü verileri gibi yüksek boyuttaki verilere uniform bir vektör dağılımı ile yaklaşmaya çalıştığımızda ayırıcı model üretken modele karşı baskınlık sağlar. Wasserstein üretken çekişmeli ağlar [3] bu sebeple III. bölümde değineceğimiz earth mover distance önerir.

III. ÜRETKEN ÇEKİŞMELİ AĞLAR'IN HATA FONKSIYONLARI

A. *İkili Çapraz Entropi Hata Fonksiyonu*

2014 yılında yayınlanan vanilla üretken çekişmeli ağlar [1] ikili çapraz entropi hata fonksiyonunu kullanıyordu.

$$-\frac{1}{n}\sum_{1}^{n} E_{x\sim p_{data}(x)}[\log D(x) + E_{x\sim p_z(z)}[\log(1-D(G(x)))] \tag{1}$$

Denklem 1'de belirtilen hata fonksiyonumuzun çalışma prensibi doğal veriden beslenen 1. komponentte doğal verimizin etiketi 1 olarak kurgulandığı için ayırıcı doğru tahmin ettiğinde 0 değeri gelecektir. Sentetik veriden beslenen 2. komponentte ise üretici modelin sonucunu en uygun parametreler ile ayırıcı model 0 olarak tahmin edecek ve 2. komponentin değeri 0 gelecektir. Ayırıcı model denklem 1'in sonucunu 0'a yaklaştıracak şekilde parametrelerini günceller. Üretken model ise $\log\left[1 - D\big(G(x)\big)\right)]$ değerini maksimize ederek ayırıcı modeli yanlış yapmaya zorlar.

B. *Wasserstein Hata Fonksiyonu*

$$W(p_r, p_g) = \frac{1}{K} sup_{||f||\leq K} E_{x\sim p_r}[f(x)] - E_{x\sim p_g}[f(x)] \tag{2}$$

Üretken çekişmeli ağlarda ikili çapraz entropi hata fonksiyonu kullanılması sürekli olan 2 dağılımın süreksiz bir hata fonksiyonu ile birbirine yaklaştırmada başarısız sonuçlanır. Dağılımlar ayrık olduğu taktirde Kullback Leibler ıraksaması sonsuz değer döndürür ve gradient descent uygulaması başarısız sonuçlanır. Wasserstein hata fonksiyonu kısaca iki dağılımı birbirine yaklaştırmak için harcamamız gereken enerjiyi temel alır, bir diğer değişle iki dağılım arasındaki mesafeyi ölçer. Denklem 2'deki Wasserstein hata fonksiyonu sürekli olduğu ve hata miktarının gerçek sayı değeri alması gradient descent uygulanmasına olanak sağlar. Oluşacak tüm olası ayrık dağılımlar arası hataların en büyüğünü alır. Yukarıdaki durumlar üretici modele güç sağlar [4].

C. *Parametre Kırpma ile Wasserstein Hata Fonksiyonu*

Denklem 2'de belirtilen f fonksiyonu wasserstein hata fonksiyonunca K – lipschitz sürekli kabul edilir. K- lipschitz süreklilik gradyan normlarının sınırlı değerler arasında olacağını garanti eder [5]. K-lipschitz bir fonksiyonun türevi fonksiyonun her noktasında [-K,K] aralığındadır. Yazarlar [3] bunu sağlamak için parametre değerleri sınır dışında ise değerleri sınır değerlerine yuvarlamıştır. Bu durum modelin yaklaşabileceği fonksiyon uzayını daraltır. Bu durum modelin başarısını olumsuz etkiler. *Gradyan Cezalandırma ile Wasserstein Hata fonksiyonu* bölümünde bu duruma önerilen çözümü inceleyeceğiz.

D. *Gradyan Cezalandırma ile Wasserstein Hata Fonksiyonu*

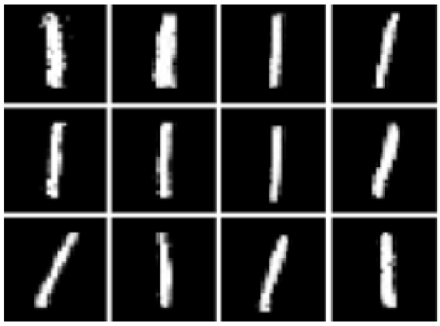
$$W(p_r, p_g) = \frac{1}{K} sup_{||f||\leq K} E_{x\sim p_r}[f(x)] - E_{x\sim p_g}[f(x)] + E_{\hat{x}\sim P_{\hat{x}}}[||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2 \tag{3}$$

Geliştirilmiş Wasserstein üretken çekişmeli ağlarda [6] parametre kırpmanın fonksiyon uzayını kısıtlaması üzerine 1-lipschitz sürekliliği sağlaması için parametre kırpma yerine gradyan cezalandırma önerilmiştir. Denklem 3 denklem 2'ye gradyan cezalandırmanın eklenmiş formudur. Paramametrelerin gradyan normlarından lipschitz sürekliliği derecesi çıkarılarak karesi alınır. Bu cezalandırma ile hata fonksiyonumuzun türevine göre yapacağımız parametre güncellemesi için hesapladığımız gradyanlar lipschitz süreklilik derecesi değeri ve negatif değeri arasına yakınsar fakat bu aralıkta olacağını garantilemez.

IV. ÜRETKEN ÇEKİŞMELİ AĞLAR'DA MODEL STABİLİZASYON PROBLEMLERİ

Üretken çekişmeli ağlarda modelin stabilizasyonunu sağlamak en çok araştırma yapılan alanlardan biridir. Mode çöküşü, Nash dengesi, hata fonksiyonunun minimum değere yaklaşamaması gibi problemler üretken çekişmeli ağları eğitmede zorluk çıkaran başlıklardır.

A. *Nash Dengesi*



Resim 4

Nash dengesi üretken ve ayırıcı modelin çekişmesi sırasında birbirine üstün gelmediği durumdur. Bu dengenin sağlanamadığı durumlarda modellerden biri diğerine üstün gelir. Bir modelin gelişimi diğer modele bağlı olduğu için eğer Nash dengesi sağlanamaz ise iki modelin de gelişimi durur. Nash dengesi bozulduğunda gradyanların 0' a yaklaşması durumu gerçekleşir. Bir sonraki başlıkta inceleyeceğimiz mode çöküşü Nash dengesinin sağlanamadığı durumda meydana gelen bir problemdir.

B. *Mode Çöküşü*

Resim 3'de gösterilen mode çöküşü üretici modelin sınırlı sayıda örnek üretebilmesidir. Üretken ağın gradyanlarının 0'a yaklaşması ve modelin öğrenememesi bu duruma sebep olur. Üretken ağın gradyanlarının 0'a yaklaşması ayırıcı modelin görevinin çok daha basit olması sebebiyle eğitim sürecinin başında üretken ağa üstün gelmesidir.

C. Hata Fonksiyonunun Minimuma Yakınsamaması

Hata fonksiyonları üretken çekişmeleri ağların başarısındaki en önemli komponentlerden biridir. 2 farklı modeli bir hata fonksiyonu üzerinden eğitmek beraberinde bir çok zorluk getirmektedir. Bunlardan biri de hata fonksiyonunun minimum değere yakınsamamasıdır. İkili çapraz entropi ve Wasserstein hata fonksiyonlarını incelemiştik. İkili çapraz entropi hata fonksiyonunun bir çok uygulamada yakınsamadığını sunulmuştur. Bunun yanında Wasserstein hata fonksiyonu da vanilla haliyle uygulandığında yakınsamamaktadır. Hata fonksiyonunun minimum değere yakınsamaması durumunda model parametreleri optimale yakın şekilde güncellenmez. Dağılımlar birbirine yaklaşmaz. Bu durumda üretici ağdan istediğimiz sentetik veri girdilerinin çıktıları beklentimiz olan doğal verilere benzemez.

V. HANGİ ÜRETKEN ÇEKİŞMELİ AĞ EĞİTME METHODU YAKINSAR ?

Method	Local convergence (a.c. case)	Local convergence (general case)
unregularized (Goodfellow et al., 2014)	✓	✗
WGAN (Arjovsky et al., 2017)	✗	✗
WGAN-GP (Gulrajani et al., 2017)	✗	✗
DRAGAN (Kodali et al., 2017)	✓	✗
Instance noise (Sønderby et al., 2016)	✓	✓
ConOpt (Mescheder et al., 2017)	✓	✓
Gradient penalties (Roth et al., 2017)	✓	✓
Gradient penalty on real data only	✓	✓
Gradient penalty on fake data only	✓	✓

Tablo 1

"Aslında hangi üretken çekişmeli ağlar yakınsar ? " [7] makalesine ait Tablo 1'de görüldüğü üzere incelediğimiz üretken çekişmeli ağ modellerin hiçbiri yakınsamamaktadır. Bu modelleri incelememizin sebebi üretken çekişmeli ağlar için dönüm noktaları olması ve ardından gelen çalışmalarca ilham kaynağı olarak görülerek en çok atıf alan çalışmalar olmasıdır.

TEŞEKKÜR

SKY LAB: Yıldız Teknik Üniversitesi Bilgisayar Bilimleri Kulübü AIR (Artificial Intelligence Research) ekibine derin öğrenme yolculuğumda verdiği destekler için teşekkür ederim.

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.

[2] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR, abs/1511.06434, 2015.

[3]] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. arXiv preprint arXiv:1701.07875, 2017K.

[4] Thomas Pinetz, Daniel Soukup, Thomas Pock. On the estimation of the Wasserstein distance in generative models. arXiv preprint arXiv: 1910.00888, 2019K.

[5] Kanglin Liu, Guoping Qiu. Lipschitz Constrained GANs via Boundedness and Continuity. arXiv preprint arXiv: 1803.06107, 2020K.

[6] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville. Improved Training of Wasserstein GANs. arXiv preprint arXiv: 1803.06107, 2017K.

[7] Lars Mescheder, Andreas Geiger, Sebastian Nowozin. Which Training Methods for GANs do actually Converge? arXiv preprint arXiv: 1801.04406, 2018K.