T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

DÖNÜŞTÜRÜCÜ TABANLI ÇEKİŞMELİ ÜRETİCİ AĞLAR İLE YÜZ İFADELERİNDEKİ DEĞİŞİMİN ANALİZİ

Narmin YAGUBBAYLİ

YÜKSEK LİSANS TEZİ

Matematik Mühendisliği Anabilim Dalı

Matematik Mühendisliği Programı

Danışman

Dr. Öğr. Üyesi Mert BAL

Haziran, 2023

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

DÖNÜŞTÜRÜCÜ TABANLI ÇEKİŞMELİ ÜRETİCİ AĞLAR İLE YÜZ İFADELERİNDEKİ DEĞİŞİMİN ANALİZİ

Narmin YAGUBBAYLI tarafından hazırlanan tez çalışması 22.06.2023 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Matematik Mühendisliği Anabilim Dalı, Matematik Mühendisliği Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Dr. Öğr. Üye. Mert BAL
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Dr. Öğr. Üye. Mert BAL, Danışman
Yıldız Teknik Üniversitesi _____

Doç. Dr. Ramazan TEKERCİOĞLU, Üye
Yıldız Teknik Üniversitesi _____

Dr. Öğr. Üyesi Meryem UZUN PER, Üye
İstanbul Sağlık ve Teknoloji Üniversitesi _____

Danışmanım Dr. Öğr. Üyesi Mert BAL sorumluluğunda tarafımca hazırlanan "Dönüştürücü Tabanlı Çekişmeli Üretici Ağlar ile Yüz İfadelerindeki Değişimin Analizi" başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Narmin YAGUBBAYLİ

İmza

Aileme

ve

biricik eşime

TEŞEKKÜR

Ciddi bir emek ve özveri ile hazırladığım yüksek lisans tezimde fikirleri ve bilgi birikimi ile bana yol gösteren tez danışmanım Dr. Öğr. Üyesi Mert BAL'a teşekkürümü borç bilirim.

Daima beni destekleyen, her türlü fedakârlığı sağlayan, büyük sabır ve sevgi gösteren sevgili eşim Farid YAGUBBAYLİ'ye sonsuz teşekkür ederim. Yoğun geçen tez süreci ve çalışma tempomda bana inanan, güvenen, cesaretlendiren, beni bu günlere getiren başarılarımın arkasındaki asıl kahramanlara sevgili aileme çok teşekkür ederim. Sizi gururlandırmaya bir ömür devam edeceğim.

Narmin YAGUBBAYLİ

İÇİNDEKİLER

KISALTMA LİSTESİ vii
ŞEKİL LİSTESİ viii
TABLO LİSTESİ ix
ÖZET x
ABSTRACT xii
1 GİRİŞ 14
1.1 Literatür Özeti 14
1.2 Tezin Amacı 17
1.3 Hipotez 18
2 DERİN ÖĞRENME 19
2.1 Evrişimsel Sinir Ağları 19
2.2 Çekişmeli Üretici Ağlar 23
3 UYGULAMA DETAYLARI 43
3.1 Önceki Model: GANmut 43
3.2 Önerilen Model 44
3.3 Hiperparametre Seçiminin İncelenmesi 51
4 DENEY SONUÇLARI VE DEĞERLENDİRME 60
4.1 GanMut Modelinin Eğitilmesi 60
4.2 Modellerin Karşılaştırılması 61
4.3 Dönüştürücü Tabanlı Üreticinin Görsel Analizi 62
5 SONUÇ VE ÖNERİLER 65
KAYNAKÇA 67
TEZDEN ÜRETİLMİŞ YAYINLAR 71

KISALTMA LİSTESİ

BERT	Bidirectional Encoder Representations from Transformers
CGAN	Conditional Generative Adversarial Network (Koşullu Çekişmeli Üretici Ağlar)
CNN	Convolutional Neural Networks (Evrişimsel Sinir Ağları)
FFN	Feed Forward Network (İleri Beslemeli Ağ)
FID	Frechet Inception Distance (Fréchet Başlangıç Mesafesi)
GAN	Generative Adversarial Network (Çekişmeli Üretici Ağlar)
GPT	Generative Pre-trained Transformer
GPU	Graphics Processing Unit (Ekran kartı)
GRU	Gated Recurrent Units (Kapı Özyinelemeli Geçitler)
IDE	Integrated Development Environment (Tümleşik Geliştirme Ortamı)
LSTM	Long Short Term Memory (Uzun Kısa Dönemli Bellek)
ML	Machine Learning (Makine Öğrenmesi)
NLP	Natural Language Processing (Doğal Dil İşleme)
ReLU	Rectified Linear Unit (Doğrultulmuş Lineer Birim)
RNN	Recurrent Neural Network (Tekrarlayan Sinir Ağları)
VGG	Visual Geometry Group
VIT	Vision Transformer

ŞEKİL LİSTESİ

Şekil 2. 1 Üç kanallı bir RGB görüntü örneği ……...….20
Şekil 2.2 Evrişim uygulaması [34] …….....21
Şekil 2.3 VGG19 model mimarisi [35] ……..23
Şekil 2.4 Çekişmeli üretici ağ mimarisi [36] ……..23
Şekil 2.5 Koşullu çekişmeli üretici ağ mimarisi [36]……..27
Şekil 2.6 Dönüştürücü mimarisi [37]……..29
Şekil 2.7 Genel Vision Transformer mimarisi ……..….37
Şekil 2.8 Dikkat mekanizması [33] ……...……..…...38
Şekil 2.9 (a) Lineer Yoğun katmanı (b) Lineer Yoğun katmanın çıktıları…….….38
Şekil 2.10 Gömme vektörlerinin ağırlıklarla çarpılıp sorgu, anahtar, değer değerlerinin hesaplanması [39] …….....39
Şekil 2. 11 Dikkat puanlarının hesaplanması ……..……...40
Şekil 2.12 Hizalama vektörü ve ağırlık değerlerinin hesaplanması ……...…....40
Şekil 2.13 Multi-Head Attention mimarisi [17]…….....….…41
Şekil 3.1 GANMUT model mimarisi ……...…….43
Şekil 3.2 Dönüştürücü tabanlı Çekişmeli Üretici Ağ mimarisi ……..45
Şekil 3.3 Dönüştürücü tabanlı Çekişmeli Üretici Ağ mimarisinde üreticide CNN'nin Dönüştürücü ile değiştirilmesi ……46
Şekil 3.4 AffectNet veri seti örnekleri [20] ……...…….50
Şekil 3.5 HiperParametrelerin görselleştirilmesi …….……...56
Şekil 3.6 Modeller ve FID değerleri ……..……..57
Şekil 3.7 (a) Run_25 Üreticinin kayıp değer eğrisi (b) Run_25 Ayrıştırıcının kayıp değer eğrisi ……58
Şekil 3.8 (a) Run_52 Üreticinin kayıp değer eğrisi ……..…….....58
Şekil 4.1 RunID -52 kayıp eğrisi ……...….62
Şekil 4.2 Ganmut kayıp eğrisi ……..…….....…62
Şekil 4.3 Modelimizde üreticinin ürettiği görüntüler …….....63
Şekil 4.4 Mutluluk yoğunluk artışı ..……..……..…64

TABLO LİSTESİ

Tablo 3.1 AffectNet veriseti duygu dağılımı [20]……..49
Tablo 3.2 HiperParametre optimizasyonu…….…….53
Tablo 4.1 Modellerin karşılaştırılması ……...61

ÖZET

Dönüştürücü Tabanlı Çekişmeli Üretici Ağlar ile Yüz İfadelerindeki Değişimin Analizi

Narmin YAGUBBAYLİ

Matematik Mühendisliği Anabilim Dalı

Matematik Mühendisliği Programı

Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Mert BAL

Yüz ifadeleri insanların duygularını iletmesinde önemli bir yol olmakla beraber; yoğunluğu, nüansı ve belirsizliği açısından da benzersizdir. Yüz ifadeleri, sözlü olmayan iletişimde insan duygularını ifade etmede ve anlamada önemli bir unsurdur. Deneyimlenen duygu ile ifade arasındaki ilişki karmaşıktır ve henüz tam olarak anlaşılamamıştır. Farklı duygular bazen aynı ifadeyi bile oluşturabilmektedir, çünkü duygular yüz kaslarının hareketleri ile ifade edilmektedir. Bu durum, insan duygularını anlama konusundaki birçok sorunun yalnızca başlangıcıdır. Duygu analizi konusunda birçok çalışma yapılmıştır ve her bir çalışma farklı teknikler, algoritmalar ve yöntemler kullanmıştır.

Son yıllarda, dönüştürücü modeli, derin öğrenme alanındaki ilerlemelerin ana unsurlarından biri haline gelmiştir. 2017'deki ilk ortaya çıkışından sonra dönüştürücü mimarisi gelişerek, farklı türleri ortaya çıkmıştır. Genellikle doğal dil işlemede başarılı sonuçlar göstererek yaygınlaşmaya başlamış olsa da sonradan farklı alanlarda kullanılmaya başlanmıştır. Dönüştürücüler, dikkat mekanizmasına dayalı bir makine öğrenme modelidir. Bu tez kapsamında dönüştürücülerin duygu analizi alanında sentetik veri üretme üzerindeki başarısı analiz edilmiştir.

Önceki çalışmalarda çekişmeli üretici ağ(GAN) modelinin üretici bölümünde evrişimsel sinir ağları (CNN) kullanılarak duygu analizindeki başarısı ele alınmıştır. Bu tez çalışmasında ise üretici bölümünde CNN modülü yerine dönüştürücü kullanılarak yeni bir model olarak dönüştürücü tabanlı GAN oluşturulmuştur. Çalışmanın temel odak noktası; görsellerdeki insan duygularının değiştirilmesinde dönüştürücü tabanlı üreticinin başarısını ölçmektir. Geliştirilen GAN modelinde farklı duygu sınıflarına ait daha gerçekçi sentetik veriler üretmek için Vision Transformer kullanılmış ve CNN ile karşılaştırıldığında AffectNet veri seti üzerinde daha başarılı olduğu gösterilmiştir.

Anahtar Kelimeler: Derin Öğrenme, Çekişmeli Üretici Ağlar, Sentetik Görüntü Üretimi, Vision Transformer, Yüz ifadeleri.

ABSTRACT

Analysis of Change in Facial Expressions with Transformer-Based Generative Adversarial Networks

Narmin YAGUBBAYLİ

Department of Mathematical Engineering

Master of Science Thesis

Supervisor: Asst. Prof. Mert BAL

Facial expressions are an important means for conveying people's emotions. They are unique in terms of intensity, nuance, and ambiguity. Facial expressions play a significant role in nonverbal communication, expressing and interpreting human emotions. The relationship between experienced emotion and expression is complex and not yet fully understood. Different emotions can sometimes produce the same expression because emotions are expressed through the movement of facial muscles. This situation is only the beginning of many challenges in understanding human emotions. Numerous studies have been conducted on emotion analysis, each using different techniques, algorithms, and methods.

In recent years, the transformer model has become one of the common elements of progress in the field of deep learning. Since 2017, the transformer architecture has evolved and given rise to different types. While initially gaining popularity for successful results in natural language processing, transformers have been subsequently employed in various domains. Transformers are machine learning models based on the attention mechanism. In this thesis, the success of transformers in generating synthetic data for sentiment analysis has been analyzed.

Previous studies have explored the use of convolutional neural networks (CNN) in the generator part of the GAN model for sentiment analysis. However, in this thesis, a

transformer-based GAN model is created by using transformers instead of a CNN module in the generator part. The main focus of the study is to measure the success of the transformer-based generator in altering human emotions in images. The Vision Transformer is employed in the developed GAN model to generate more realistic synthetic data for different emotion classes, and it has been demonstrated to be more successful than CNN-based generator model on the AffectNet dataset.

Keywords: Deep Learning, Generative Adversarial Networks, Synthetic İmage Generation, Vision Transformer, Facial expressions

1
GİRİŞ

1.1 Literatür Özeti

Derin öğrenme [1], [2] ve yapay sinir ağları makine öğrenmesinin bir alt kümesidir. İnsan gibi çalışan ve analiz edebilen makine fikri 1900' lerin başına dayanır. Makinelerin öğrenip düşünebileceği fikrini ilk ortaya atan bilim adamlarından biri de Alan Turing'dir [3].

Günümüze kadar pek çok bilim insanı tarafından bu fikir gerçeğe dönüştürülmeye çalışılmış ve sonuçlar yeni elde edilmeye başlamıştır [8]. Google DeepMind'ın yaptığı testlerden sonra deneysel modellemenin önemi anlaşılmış, derin öğrenme başarısı makine öğrenmedeki diğer metotları geride bırakacak başarım göstermiştir [4]. Yeni tekniklerle derin öğrenme artık daha çok yaygınlaşmış, bu sayede hemen hemen her veri kümesine başarılı makine öğrenimi uygulanabilir olmuştur. Yapay sinir ağları artık diğer makine öğrenimi algoritmalarından çok daha iyi sonuçlar vermeye başlamıştır [4]. En sonunda, sinir ağları uzun kısa dönemli bellek (Long Short Term Memory- LSTM) [10], tekrarlayan sinir ağları (Recurrent Neural Network- RNN) [9] ve çekişmeli üretici ağlar (Generative Adverserial Network - GAN)[16] ile veri sınıflandırma dışında veri üretilmeye başlanmıştır.

Daha sonraki zamanlarda doğal dil işleme alanı için Google'ın yayınladığı makale [17] sayesinde artık yeni araştırmaların hemen hemen hepsinde bu mekanizma kullanılmaktadır. Dönüştürücüler doğal dil işleme (Natural Language Processing-NLP) görevlerinde uzun mesafe bağımlılıklarını ele alma konusunda tekrarlayan sinir ağlarının kısıtlamalarına karşı oluşturulmuştur. RNN'ler, sıralı girdi verilerini işleme yetenekleri nedeniyle günümüzde NLP'de geniş ölçüde kullanılmaktadır, ancak uzun mesafe bağımlılıkları olan girdilerde bazı zorluklar yaşanabilmektedir. 2017 de Vaswani vd. yaptıkları çalışmada [17], NLP çalışmaları için RNN'lere alternatif olarak Dönüştürücü mimarisini tanıttılar. Dönüştürücü mimarisi, girdi verilerini paralel olarak işleme yeteneği olan bir öz-dikkat mekanizması kullanır ki, bu da uzun mesafe bağımlılıklarını yakalamayı önemli ölçüde artırmaktadır.

Dönüştürücülerin en önemli avantajlarından biri, büyük miktarda veri ile çalışıp, öğrenebilme yetenekleridir. 2018' de Radford vd. tarafından geliştirilen Generative Pre-trained Transformer (GPT) modeli [5], dil üretimi ve metin tamamlama gibi çeşitli NLP görevleri için kullanılmıştır.

Son yıllarda Dönüştürücü mimarisi yüz tespit çalışmalarında da yaygınlaşmaya başlamıştır. 2021 yılında Lian vd. [6] ve 2022 yılında Chudasama [7] yaptıkları çalışmalarda konuşma verilerinden duygu algılama için Dönüştürücü tabanlı yaklaşım önermişlerdir.

1.1.1 Duygu Tanıma ve Önemi

Günümüz teknolojisi dünyasında, insan-makine etkileşimi talebi artmaktadır ve bu nedenle makine insanların el hareketlerini ve duygularını tanıyabilmelidir. Eğer makine insan duygularını doğru bir şekilde belirleyebilirse, insan davranışlarını daha iyi anlayabilir ve görevleri daha etkili bir şekilde yerine getirebilir. Farklı duygular bazen aynı ifadeyi bile yaratabilirler, çünkü yüz kaslarının hareketleriyle ifade edilirler. Bu, insan duygularını anlamadaki birçok sorunun yalnızca başlangıcıdır. Duygu tanıma, özellikle bir konu veya ürün hakkında insanların nasıl hissettiğini anlamaya yönelik olarak sosyal medya paylaşımlarını, müşteri geri bildirimlerini veya diğer türlerdeki iletişimleri analiz etmek için kullanılabilir. Duygular günlük görevlerde de örneğin, düşünme, öğrenme, dikkat, motivasyon, çözme, algılama, planlama, zihinsel işlemler, mantık yürütme vb. önemli rol oynamaktadır.

Gün geçtikçe duygu tanıma alanında büyük bir araştırma alanı oluşmaktadır ve duyguları tanımak için kullanılabilecek çeşitli metotlar, teknikler geliştirilmektedir. Bu teknikler ile veri üzerinde çeşitli özellikleri analiz edilir, duygular tanımlanıp, yorumlanır. Duygu tanıma, metin, ses, sözel ve yüz özellikleri gibi çeşitli veri türleri üzerinden gerçekleştirilebilir.

Duygu tanıma doğruluğunu artırmak için, çeşitli veri kaynaklarını dikkate almak ve veri analizinde çeşitli teknikler kullanmak önemlidir. Bu, duygu tanıma çalışmalarında sonuçların doğruluğunu artırır. Ayrıca, duygu tanıma için makine öğrenimi algoritmalarını eğitirken çeşitli ve büyük veri kümesine sahip olmak, modelin duyguları farklı bağlam ve durumlarda daha iyi anlamasına ve tanımasına yardımcı olabilir. Genel

olarak, duygu tanıma, çeşitli veri türlerinden insan duygularını tespit etme ve yorumlama işlemini içeren kompleks ve çok yönlü bir süreçtir.

Görüntülerden duygu tanıma, hareketsiz görüntüler veya video karelerinden insan duygularının algılanıp yorumlanmasını içerir. Bu, makine öğrenimi algoritmaları ve bilgisayarlı görü teknikleri kullanılarak yapılır. Görüntülerden duygu tanıma çalışmalarında, farklı duygusal ifadeler sergileyen insanların etiketlenmiş görüntülerinden oluşan büyük veri kümesi ile eğitilmiş makine öğrenimi algoritmaları kullanılmaktadır. Bu algoritmalar, mutluluk, üzüntü, öfke veya korku gibi belirli duyguların işaretçisi olarak düşünülen görüntülerdeki desenleri ve özellikleri tanıyabilir. Örneğin, makine öğrenimi algoritması, kalkık kaşları olan ve gülümseyen bir kişinin muhtemelen mutlu olduğunu tanıyabilecekken, kırışık kaşları, büzüşük dudakları, kırpılmış gözleri olan kişinin muhtemelen üzgün olduğunu tanıyabilir. Sonuç olarak, duygu tespit uygulamaları geniş bir potansiyel uygulama alanına sahiptir.

Teknolojinin ilerlemesi, müşterilerin daha talepkâr hale gelmesi ile kişiselleştirilmiş deneyim talebi, sanal ve artırılmış gerçeklik kullanımının artması, veri toplama ve analiz yöntemlerinin gelişmesi gibi birçok faktör nedeniyle duygu tespiti uygulamaları daha popüler hale gelmiştir. Teknolojinin hızlı ilerlemesi gerçek zamanlı olarak görüntü, ses analizi yoluyla duyguları doğru bir şekilde algılamayı ve tanımayı mümkün kılmıştır.

1.1.2 Duygu Tanıma Kullanım Alanları

Duygu tespit uygulamaları son yıllarda araştırma ve geliştirme konusunda popüler hale gelmiştir. Duygu tespiti pazarlama ve reklam, insan-bilgisayar etkileşimi, müşteri hizmetleri, oyunlar, eğitim gibi çeşitli alanlarda kullanılmaktadır.

Pazarlama ve reklam alanında, duygu tespiti reklam kampanyalarının etkililiğini ölçmek, müşterilerin duygularına dayalı pazarlama stratejisi hedeflemek için kullanılmaktadır. Bir ürün veya hizmet hakkında müşterilerin hangi fikre sahip olduğunu anlayarak, şirketler pazarlama çalışmalarını hedef kitlelerine daha uygun hale getirir, bu da satışların artmasına ve müşteri sadakatine yol açar.

İnsan-bilgisayar etkileşimi alanında, duygu tespiti sanal gerçeklik ve artırılmış gerçeklik uygulamalarında kullanıcının duygularını algılamak ve yanıtlamak için kullanılmaktadır. Kullanıcının duygusal durumunu anlayarak, sistem tercih edilecek bir seçenek sunar böylece işlemler ilgi ve memnuniyetin artması ile sonuçlanır.

Müşteri hizmetleri alanında, duygu tespiti çağrı merkezlerinde ve sohbet botlarında müşterilerin duygularını tanımlamak ve yanıtlamak için kullanılmaktadır. Müşterinin nasıl hissettiğini anlayarak, müşteri hizmet temsilcileri daha etkili bir yardım sağlamaktadır.

Oyun endüstrisinde, duygu tespiti oyuncunun duygularını algılayıp ve yanıtlayarak daha gerçekçi ve derinlemesine oyun deneyimi oluşturmak için kullanılmaktadır.

İntihara meyilli veya duygusal sorunları olan kişiler tespit etmek gibi kritik durumları erkenden tanımak ve dolayısıyla ciddi kayıpları önlemek için duygu tespiti yapan çeşitli yapay zekâ sistemleri geliştirilmektedir [43].

1.2 Tezin Amacı

Bilgisayarlı görü çalışmalarında dönüştürücü mimarilerin hızlı gelişimi, Çekişmeli Üretici Ağlarının potansiyelinin keşfedilmesine olanak sağlamaktadır.

Bu tez çalışmasında, GAN'lar içinde dönüştürücü tabanlı üretici mimarisi ile "Dönüştürücü tabanlı üreticiler, evrişimsel üreticilere göre daha iyi görüntüler üretebilir mi?" araştırma sorusu ele alınmaktadır.

Bilgisayarlı görü görevlerinde dönüştürücü mimarilerine ilginin artması ve başarılı sonuçlar elde edilmesi bu çalışmanın önemini vurgulamaktadır. Araştırma sorusunun test edilmesi amacıyla, her biri benzersiz insan duygusunu temsil eden yedi kategoriden oluşan görüntüler içeren AffectNet veri seti kullanılmıştır. Veri işlemeyi kolaylaştırmak için görüntüler -1 ve 1 arasında normalize edilmiştir. Aynı zamanda, veri setindeki duygu sınıflarına ait görsellerin sayı miktarı farklılık gösterdiğinden bazı duygu kategorilerini eğitmede zorluklar ile karşılaşılmıştır.

Çalışma kapsamında, mevcut GANmut [40] modelinden esinlenerek, evrişimsel üretici, dönüştürücü tabanlı üretici ile değiştirilmiştir. Sonraki aşamada, geniş çaplı hiper-parametre optimizasyonu, Fréchet Başlangıç Mesafesi değeri temelinde en iyi parametrenin seçilmesini sağlamış, böylece orijinal ve değiştirilmiş iki mimari arasında bir karşılaştırma analizi yapılmıştır.

Çalışmada elde edilen bulgular, dönüştürücü tabanlı üreticinin daha iyi görüntüler oluşturabildiğini ortaya koymuş ve bunu da orijinal mimarinin 9.7 olan FID değerine

karşı 9.1'lik daha düşük bir FID değeri ile desteklemiştir. Aynı zamanda, görsel analizler de modelin duygu tipi ve yoğunluğa bağlı olarak giriş görüntülerini değiştirme kapasitesini daha net göstermektedir.

Bu çalışma, Dönüştürücü tabanlı üretici mimarilerin, özellikle insan duygu değişikliği için GAN modellerinin üretici kısmında etkili bir şekilde kullanılabileceğini göstermiştir. Sonuçlar, dönüştürücü mimarilerin kullanımının başarısını göstermiş ve geleneksel evrişimsel mimarilere göre daha iyi bir performans elde edildiğini ispat etmiştir.

1.3 Hipotez

Bu çalışma kapsamında, farklı çalışma alanlarında kullanılan CNN tabanlı GAN modellerine kıyasla performans artışı sağlayan dönüştürücü tabanlı çekişmeli üretici ağ mimarisi geliştirilmiştir. Bunun için ilk aşamada CNN tabanlı GANmut mimarisi incelenmekte, daha sonra performans artışı sağlayan özgün mimari geliştirilerek literatüre katkı sunulmaktadır.

Çekişmeli üretici ağlar son yıllarda görüntüden görüntü dönüşümü yaparken bilgisayarlı görü alanında başarılar elde etmektedir. Bu tez çalışmasında bir duygu sınıfından başka bir duygu sınıfına ait görüntü dönüşümü ile ilgili GAN mimarisinin üretici bileşeninde dönüştürücü mimarisi kullanılarak, yeni model geliştirilmiştir.

2
DERİN ÖĞRENME

2.1 Evrişimsel Sinir Ağları

Evrişimsel sinir ağlarının (Convolutional Neural Networks- CNN) [19] görüntü sınıflandırması, nesne tespiti, bölütleme ve diğer bilgisayarlı görü görevlerinde kullanmanın çok etkili olduğu kanıtlamıştır. CNN'ler tipik olarak, her biri ağın genel işlevselliğinde belirli bir role sahip olan birkaç katmandan oluşur. Bu katmanlar üç ana kategoriye ayrılır: Evrişim katmanı, havuzlama katmanı ve tam bağlantılı katman.

Evrişim katmanı: Bu katman, bir CNN'nin temel yapı taşıdır. Girdi olarak alınan verilere çok boyutlu filtreler uygulanarak öznitelik haritaları oluşturulur. Öznitelik haritaları, kenarlar, köşeler ve dokular gibi girdi görüntüsünün farklı yönlerini yakalar. Bu katmanda bulunan filtreler ile öznitelik öğrenimi yapılmaktadır.

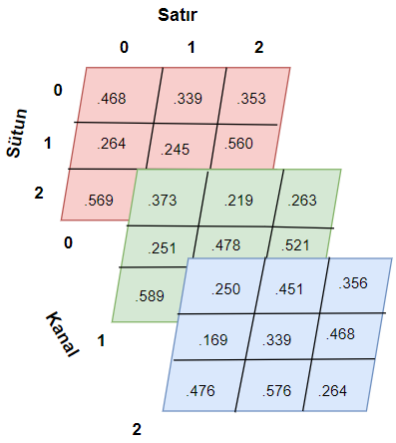
Havuzlama Katmanı: Havuzlama katmanı genellikle birden fazla evrişim katmanı arasında bulunur. Bu katmanda Evrişim katmanında yapılan uzamsal değişmezliğin alt örneklemi alınır. Havuzlama katmanının üç türü vardır: minimum havuzlama, maksimum havuzlama ve ortalama havuzlama. En yaygın türü her piksel havuzu içindeki maksimum değeri seçen maksimum havuzlamadır.

Tam Bağlı Katman: Tam bağlı katman, bir CNN'nin son ve en önemli katmanıdır. Önceki katmanlar tarafından oluşturulan öznitelikleri alır ve bir sınıf olasılıkları vektörü oluşturmak için bir dizi ağırlık uygular. Bu katmandaki nöronlar önceki katmandaki nöronlara tamamen bağlıdır. Tam bağlı katmanın çıktısı, girdi görüntüsü hakkında bir tahmin yapmak için kullanılır.

Evrişim katmanı CNN yapısının en önemli katmanıdır. Bu evrişim katmanı nöronların koleksiyonundan oluşan birden çok özellik haritası içerir. Tüm katmanlar arasında yerel bağlantılar bulunur. Eğitilebilir filtreler, evrişim katmanındaki nöronları önceki katmandaki nöronlara bağlar. Bu filtreler evrişim katmanının öznitelik haritası boyunca paylaşılır, CNN'lerin avantajlarından biri olan parametre (ağırlık) paylaşımı olarak adlandırılır.

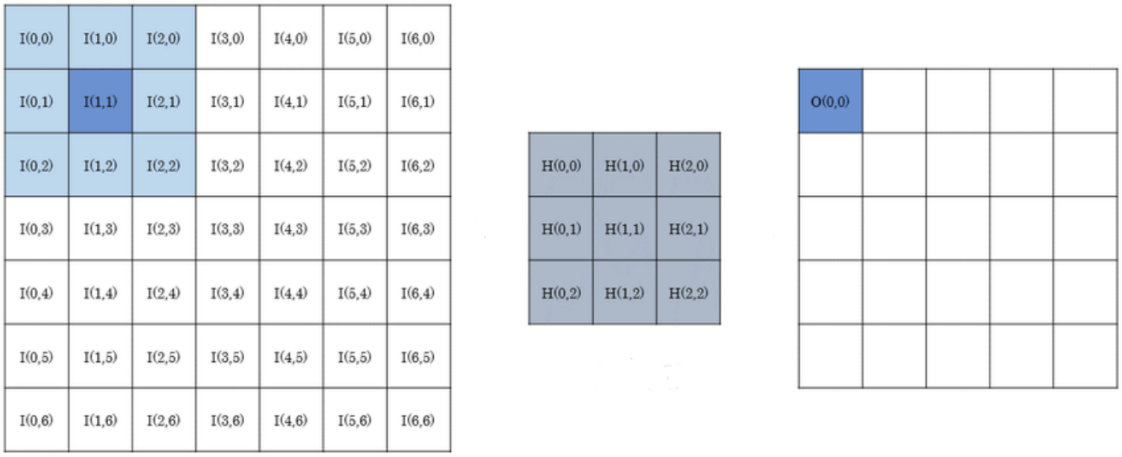
Görüntü, piksel değerlerinden oluşan bir matristir ve her görüntü bir piksel değerleri matrisi olarak temsil edilir. Kanal ise bu görüntünün bir bileşenine atıfta bulunmak için kullanılan bir terimdir.

Evrişimsel sinir ağlarının girdi katmanına verilen ham görüntüler tek kanallı yani gri tonlamalı ya da üç kanallı: kırmızı, yeşil, mavi (RGB) olabilir. 128x128 boyutundaki bir RGB görüntü 128x128x3 (Genişlik x Yükseklik x Kanal Sayısı) kadar piksel değeri içerir. 128x128 boyutunda üç kanallı bir girdi görüntüsü için toplam piksel sayısı 128x128x3 = 49.152 olur. Böylece 128x128 boyutuna sahip bir görüntü 49.152 elemandan oluşan bir vektör olarak temsil edilmiş olur. Her bir pikselin değeri 0 ile 255 arası tamsayı değerlerden oluşur. CNN'lerin, 3 boyutlu bir görüntünün normalize edilmiş piksel değerleri temsili olarak Şekil 2.1'de gösterilmiştir.



Şekil 2. 1 Üç kanallı bir RGB görüntü örneği

Filtre matrisinin farklı değerlerinin aynı giriş görüntüsü farklı öznitelik haritaları üretmektedir. Bu matrisin sayısal değerlerini değiştirerek keskinleştirme, kenar algılama ve bulanıklaştırma gibi işlemler gerçekleştirilir. Sonuç olarak, farklı filtrelerle bir görüntüden farklı özellikler çıkarılmaktadır. Bir öznitelik haritası oluştururken filtre, şekil 2.2'deki gibi giriş görüntüsünün üzerinde kaymaya başlar. Böylece aynı görüntü üzerinde farklı filtreler, farklı öznitelik haritaları çıkarırlar.



Şekil 2.2 Evrişim uygulaması [34]

Çıkış katmanında çıktı hacminin boyutunu etkileyen, filtre sayısı, adım boyutu ve sıfır dolgusu olmak üzere üç hiperparametre vardır. Burada adım boyutu, filtre matrisini girdi matrisi üzerinde kaydırdığımız piksel sayısıdır. Örneğin adım boyutu 2 olduğunda, filtreler her seferinde 2 piksel atlamış olur. Aslında büyük adım boyutu küçük öznitelik haritaları üretecektir. Sıfır dolgusu, filtreyi girdi görüntü matrisinin kenarlık elemanlarına uygulayabilmek için matrisin sınırlarını sıfırlarla doldurmaktır. Sıfır dolgusunun en önemli avantajı özellik haritalarının boyutunu kontrol etmemize izin vermesidir.

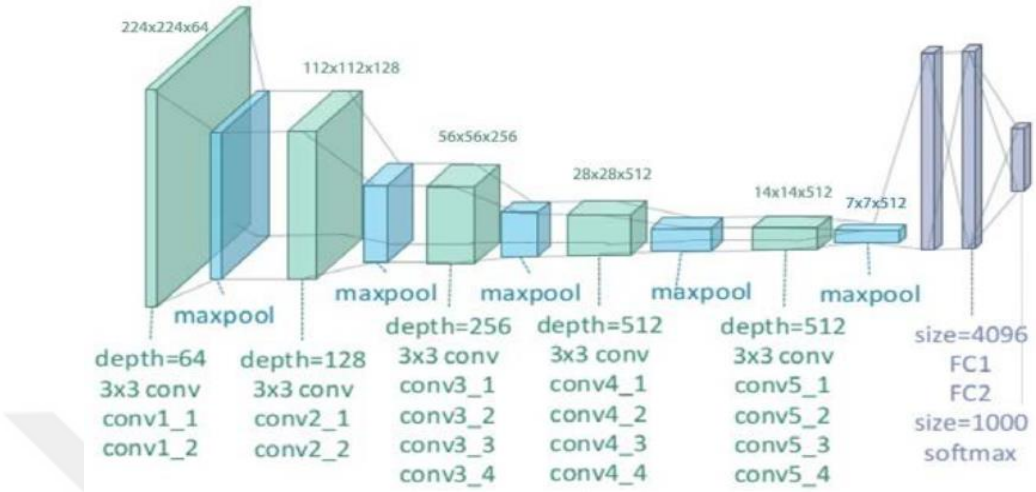
2.1.1 VGGNet

VGGNet İngiltere'nin Oxford Üniversitesi'ndeki Visual Geometry Group (VGG) tarafından tanıtılan bir evrişimsel sinir ağı mimarisidir. Karen Simonyan ve Andrew Zisserman tarafından yazılan makale [21] 2014 yılında Conference on Computer Vision and Pattern Recognition (CVPR) kongresinde yayınlanmıştır. VGG19, 16 yerine 19 katmanla genişletilmiş bir VGG16 mimarisidir ve görüntü sınıflandırma, nesne algılama gibi birçok çalışmada yüksek performans elde etmiştir.

VGG19, evrişimsel sinir ağları alanında kullanılan bir derin öğrenme modelidir ve ImageNet veri kümesindeki 1.000 farklı nesne sınıfını tanımlama görevi için tasarlanmıştır. O görevde modele girdi görüntü verilerek, görüntünün hangi nesne sınıfına ait olduğu tespiti gerçekleştirilmiştir.

Şekil 2.3'te gösterildiği gibi VGG19 mimarisi ardışık olarak birkaç evrişim katmanı, havuzlama katmanı ve tam bağlantılı katmanlardan oluşur. Evrişim katmanları görüntüdeki özellikleri çıkarırken, tam bağlı katmanlar evrişim katmanları tarafından elde

edilen özellikleri sınıflandırmak için kullanılır. VGG19 modelinin derinliğini artırmak için 3x3 evrişim katmanından oluşan bir mimari oluşturulmuştur.



Şekil 2.3 VGG19 model mimarisi [35]

VGG19 daha önceki CNN mimarilerine göre birçok avantaj ve geliştirme sunmaktadır:

Daha Derin Mimari: VGG19, 16 katmanlı öncüsü VGG16'nın aksine 19 katmanlı daha derin bir ağdır. Eklenen derinlik, ağın girdi görüntülerinde daha karmaşık özellikler ve desenler öğrenmesine olanak tanır.

Basit ve Tek Tip Mimari: VGG19'un tek tip tasarımı sayesinde tüm evrişim katmanları aynı filtre boyutuna ve adım boyutuna sahip olurken, tüm havuzlama katmanları aynı boyuta ve adım boyutuna sahiptir. Bu standartlılık, ağın anlaşılmasını ve uygulanmasını daha kolay hale getirir.

Öğrenme: VGG19, milyonlarca görüntüyü içeren ve binlerce farklı kategoriden oluşan büyük bir ImageNet veri kümesinde eğitilmiştir. Bu eğitim, ağın sınırlı eğitim verisi ile kullanıldığında genel görüntü özelliklerini öğrenmesine olanak tanır.

Performans: VGG19, görüntü sınıflandırma, nesne tespiti olmak üzere çeşitli bilgisayarlı görü görevlerinde iyi performans elde etmiştir. Bu nedenle, VGG19 bilgisayarlı görü araştırmalarında yaygın kullanılan CNN mimarilerinden biridir. ImageNet'in yanı sıra, CIFAR-10 ve CIFAR-100 gibi daha küçük veri setlerinde de iyi sonuçlar vermiştir. Genel olarak, VGG19'un bu veri kümelerindeki performansı, görüntülerdeki karmaşık özellikleri ve desenleri öğrenme kabiliyetini göstermektedir.

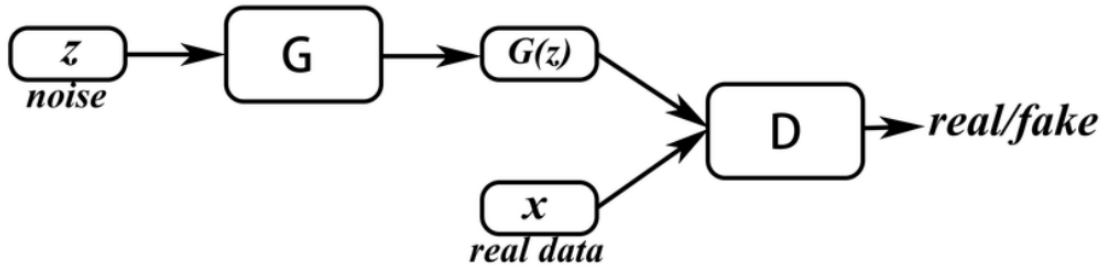
2.2 Çekişmeli Üretici Ağlar

Çekişmeli Üretici Ağlar [16] ilk defa 2014 yılında Goodfellow vd. tarafından önerilmiş ve o yıldan itibaren derin öğrenme alanında popüler konulardan biri haline gelmiştir. Bu ağ modeli hızla yaygınlaşmış ve kısa zamanda birçok varyasyonu ortaya çıkmıştır. Görüntü tamamlama, süper çözünürlük, görselden görsele geçiş, müzik üretimi gibi farklı alanlarda GAN uygulamalarını görmek mümkündür.

2.2.1 GAN Yapısı ve Çalışma Mekanizması

Çekişmeli Üretici Ağlar üretici ve ayrıştırıcı olmak üzere iki ağdan oluşmaktadır. Üretici ağ, herhangi bir görüntü, sinyal gibi bir veriyi alıp gerçeğine benzer sentetik veriyi üretmektedir. Bu sentetik verileri oluştururken üreticiye ayrıştırıcı tarafından geri bildirimler verilmektedir. Oluşturulan sentetik veriler daha sonra ayrıştırıcının eğitimi aşamasında kullanılır. Üretici modeller veri kümesindeki farklı sınıfların dağılımına odaklanır, veri noktalarının dağılımlarını ve desenlerini modellemeye çalışırlar. Üretici modeller modelleme yaparken ortak olasılık dağılımı $p(x,y)$'yi öğrenirler. Üretici sınıflandırıcılara Naive Bayes, Bayes ağları örnek gösterilebilir. Ayrıştırıcı modeller ise gözlemlediği verilerden faydalanarak sınıfları öğrenmeye çalışırlar. Bunun için koşullu olasılık dağılımı $p(y|x)$'i öğrenirler. Ayrıştırıcı sınıflandırıcılara örnek olarak, lojistik regresyon ve geleneksel yapay sinir ağları gösterilebilir.

Üretici modeller yeni örnekler üretebilirken, ayrıştırıcı modeller ise iki farklı örnek arasındaki farkı ayırt etmeyi hedeflerler. İki modelin de genel uygulama alanları, denetimli öğrenme problemleridir. Şekil 2.4 de standart bir Çekişmeli Üretici Ağ mimarisi gösterilmiştir:



Şekil 2.4 Çekişmeli üretici ağ mimarisi [36]

Eğitimin ilk aşamalarında üretici, sentetik veriler oluşturur ve ayrıştırıcı oluşturulan verilerin sahte olduğunu kolayca ayırt etmeyi başarır. Eğitimin ilerleyen aşamalarında ise üretici ayrıştırıcının ayırt edemeyeceği şekilde sahte veriler oluşturmaya başlar. Eğer üreticinin eğitimi başarısız olursa, ayrıştırıcı gerçek verileri oluşturulan sentetik verilerden kolayca ayırt edebilir. Ancak, üreticinin eğitimi başarılı biterse, üretici oluşturulan sentetik verileri gerçek verilerden ayırt etmekte zorlanır ve bu ayrıştırıcının sahte verileri gerçek veri olarak sınıflandırmasına neden olabilir. Bu durumda ayrıştırıcı ağın doğruluğu azalır.

G üretici, D ayrıştırıcı, z normal dağılımdan örneklenmiş bir vektör olmak üzere çekişmeli üretici ağların en genel amaç fonksiyonu denklem 2.1'de gösterilmektedir.

$$\begin{aligned} &\min_G \max_G \mathrm{V}(D,G) \\ &\quad = E_X \sim p_{data}(X)\,[\log D(X)] \\ &\quad + \; E_Z \sim p_z(Z)\left[\log\left(1 - D\left(G(Z)\right)\right)\right] \end{aligned} \tag{2.1}$$

Ayrıştırıcı ağ eğitimi için iki farklı veri örneği kullanılır. İlk örnek gerçek verilerdir ki, bu bizim çalışmamızda insanların farklı duygularını gösteren görsellerdir. İkinci örnek ise üretici tarafından üretilen sentetik verilerdir. Gerçek veriler, ayrıştırıcının eğitimi için pozitif örnekler, sentetik veriler ise ayrıştırıcının eğitimi için negatif örnekler olarak adlandırılır.

Ayrıştırıcı ağın eğitim işlemi aşağıdaki gibi sıralanabilir:

- Ayrıştırıcı ağı hem gerçek veri örneğini hem de üreticinin oluşturduğu sahte veriyi sınıflandırmaya çalışır.
- Ayrıştırıcı kayıp fonksiyonunu hem gerçek hem de sentetik veri örnekleri için hesaplar.
- Tüm ayrıştırıcı ağ kayıp fonksiyonunu göz önünde bulundurarak geri yayılım işlemi yapar ve ayrıştırıcı için en optimize parametreleri bulup, model parametrelerini günceller. Sonuç olarak ayrıştırıcı ağının her katmanındaki ağırlıklar geri yayılım ve optimizasyon işlemi ile güncellenir.

Üretici eğitimi için ise rastgele bir girdi veri örneği alınır. Buradaki en önemli nokta üreticinin eğitim için ayrıştırıcıya gerek olmasıdır.

Üretici ağın eğitim işlemi aşağıdaki gibi sıralanabilir:

- Rastgele bir girdi veri örneği seçilir.
- Üretici ağ bu girdiden bir çıktı oluşturur.
- Gerçek ve sentetik veriler ayrıştırıcı ağdan geçer ve gerçek/sahte veri tahminleri yapılır.
- Üretici ağ için kayıp fonksiyonu hesaplanır.
- Üretici ağın her katmanındaki ağırlıklar geri yayılım ve optimizasyon işlemi ile güncellenir.

GAN eğitimi için ayrıştırıcı ve üretici ağların eğitimi birlikte çalıştırılmalıdır. Çünkü bu iki sinir ağının eğitimi birbirlerine yoğun bir şekilde bağlantılı olduklarından, bir iterasyonda sırayla eğitilirler. Üretken modeller denetimsiz makine öğreniminin bir dalıdır, ancak üretken modellemeye ait olan GAN mimarisinin eğitimi denetimli makine öğrenimi olarak kabul edilir.

2.2.1.1 Çekişmeli Üretici Ağlarda Kayıp Fonksiyonları

Çekişmeli Üretici Ağlar (GAN)'da kayıp fonksiyonunun seçimi üretilen görüntülerin kalitesini doğrudan etkileyen eğitim sürecinin önemli bir parçasıdır. GAN'larda kullanılan kayıp fonksiyonları, ayrıştırıcının ayrım yapmaya çalıştığı gerçek verilerle üretilen sentetik veriler arasındaki farkı ölçmek için tasarlanmıştır. Üreticinin görevi, bu kayıp fonksiyonunu en aza indirgemekle gerçek verilere en benzer veriler üretmek şeklindedir. GAN'larda kullanılan birkaç tür kayıp fonksiyonu vardır. Bunlar ikili Çapraz-entropi kaybı (The binary cross-entropy loss), Wasserstein kaybı (Wasserstein loss) ve Menteşe kaybı (hinge loss) olan kayıp fonksiyonlarıdır.

İkili çapraz entropi kaybı, GAN'larda en yaygın kullanılan kayıp fonksiyonudur ve tahmin edilen ve gerçek etiketler arasındaki farkı ölçer. İkili çapraz entropi kaybı denklem 2.2'deki gibi tanımlanmıştır:

$$L(\hat{y}, y) = y \cdot log(\hat{y}) + (1 - y)\log(1 - \hat{y}) \qquad (2.2)$$

Wasserstein kaybı, gerçek ve sentetik veriler arasındaki farkı ölçmek için Wasserstein mesafesi kullanan ikili çapraz entropi kaybına alternatif bir kayıp fonksiyonudur. Bu kayıp fonksiyonu, daha pürüzsüz optimizasyon süreci sağlayarak GAN'ların kararlılığını artırmak için tasarlanmıştır. Özellikle görüntüler gibi yüksek boyutlu verilerin eğitiminde kullanıldığında yararlıdır. İki dağılım arasındaki mesafeyi ölçen Wasserstein kaybı fonksiyonu denklem 2.3'te gösterilmiştir.

$$W(P_r, P_\theta) = \sup ||f||_{L\leq 1} \left[E_{x\sim P_r}\,[f(x)] - E_{x\sim P_\theta}\,[f(x)] \right] \qquad (2.3)$$

Menteşe kaybı, ikili çapraz entropi kaybına göre daha güçlü değerlere dayanıklı olacak şekilde tasarlanan diğer bir kayıp fonksiyonudur.

$$L(y) = \max(0, 1 - t \cdot y) \qquad (2.4)$$

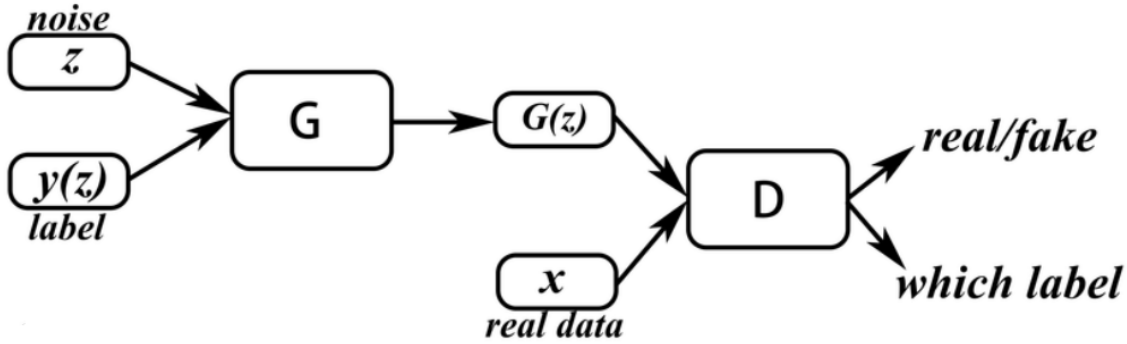
2.2.2 Koşullu Çekişmeli Üretici Ağlar

Çekişmeli Üretici Ağlar, gerçek veri dağılımını takip eden gerçekçi veriler üretmek için kullanılan bir tür sinir ağıdır. Ancak GAN'lar rastgele veriler üretilebilir ve üretilen çıktı üzerinde herhangi bir kontrol olmayabilir. Sınırlama işlemi yaparak istenilen şekilde veri üretilmesi için Koşullu Çekişmeli Üretici Ağlar (Conditional Generative Adversarial Network- CGAN) [41] geliştirilmiştir. Klasik GAN'lardan farklı olarak üretici ve ayrıştırıcı ağların girdisine ek olarak bir etiket verilir. Böylece ayrıştırıcı, girdideki görüntünün hangi sınıfa ait olduğuna karar verebilir. CGAN'lar, koşullu veri üretimi için güçlü bir araç sağlayarak görüntü tamamlama, görselden görsele geçiş de dahil olmak üzere çeşitli uygulamalarda başarıyla kullanılmaktadır.

Geleneksel GAN'larda, üretici rastgele görüntüyü girdi olarak alır ve bir çıktı üretir. Ayrıştırıcı ise üretilen çıktı ve gerçek veriyi girdi olarak alır, bu görüntünün gerçek veya sentetik veri olup olmadığını belirten ikili bir sınıflandırma üretir. Ancak, CGAN'larda üretici rastgele görüntüyü girdi olarak alırken, ayrıştırıcı üretilen çıktı ve ek etiketi gerçek girdi verileriyle birlikte girdi olarak alır. Koşullu çekişmeli ağlar için genel amaç fonksiyonu denklem 2.5'te gösterilmiştir.

$$\min_G \max_G V(D,G)$$
$$= E_X \sim p_{data}(X)\,[\log D(X|Y)]$$
$$+\ E_Z \sim p_z(Z)\left[\log\left(1 - D\left(G(Z|Y)\right)\right)\right] \quad (2.5)$$

Ek girdi bilgisi etiketler, özellikler, sınıflandırmalar veya kontrol işlemi için kullanılabilen bilgiler şeklinde olabilir. Örneğin, görselden görsele geçiş görevinde, ekstra etiket çeviri işlemi sırasında korunması gereken istenen çıktı görüntüsünün stil, renk veya diğer özellikleri olabilir. Bu ek etikete dayalı olarak üretici ağ koşullandırılarak, CGAN'lar daha belirli ve kontrol edilebilir çıktılar üretebilirler. Şekil 2.5'te Koşullu Çekişmeli Üretici Ağ mimarisi gösterilmiştir:



Şekil 2.5 Koşullu çekişmeli üretici ağ mimarisi [36]

2.2.3 Koşullu Çekişmeli Üretici Ağlarda Kayıp Fonksiyonu

CGAN'lerde kayıp fonksiyonu, normal GAN kaybına ek olarak üretilen çıktının, ek girdi bilgisine bağlı olarak istenen çıktıdan farkını ölçen ek kayıp terimi içerir. Ayrıştırıcı ek girdi bilgisine göre gerçek girdi verilerini ve üretilen çıktıları sınıflandırmak için eğitilirken, üretici de ayrıştırıcının bu sınıflandırmasında üretilen verileri gerçek sanacak şekilde çıktılar üretmek için eğitilir.

GAN kaybı, ayrıştırıcı çıktısı ve gerçek etiket arasındaki ikili çapraz entropi kaybı olarak tanımlanır. Gerçek etiket, gerçek veri için 1 ve üretilen veri için 0'dır. Ek kayıp terimi genellikle, ek girdi bilgisine bağlı olarak üretilen çıktı ile istenen çıktı arasındaki L_1veya L_2 mesafesidir. L_1 kaybı, üretilen çıktı ile istenen çıktı arasındaki mutlak farkı ölçerken, L_2 kaybı üretilen çıktı ile istenen çıktı arasındaki kare farkı ölçer.

$$L_1 LossFunction = \sum_{i=1}^{n} |y_{true} - y_{Predicted}| \qquad (2.6)$$

$$L_2 LossFunction = \sum_{i=1}^{n} (y_{true} - y_{predicted})^2 \qquad (2.7)$$

L_1 veya L_2 kaybının ek bir kayıp terimi olarak kullanımı, üretilen çıktı ile istenen çıktı arasındaki büyük farkları bularak üretilen çıktının kalitesini iyileştirmeye yardımcı olur. CGAN'lardaki kayıp fonksiyonu, üretecin ayrıştırıcıyı aldatmasını sağlayan klasik GAN kayıp fonksiyonunu içerir. GAN kaybı ve ek kayıp terimini birleştirerek, CGAN'lar ek etikete dayalı olarak özel çıktılar üretebilir.

2.2.4 Koşullu Çekişmeli Üretici Ağlarda Gizli Uzay

Gizli uzay GAN mimarisinde üretici ağın girdisi olarak kullanılan rastgele görüntü uzayını ifade eder. Buradaki gizli uzay, geleneksel GAN'lardaki gizli uzaydan farklıdır çünkü ek girdi bilgisi ile koşullandırılmıştır. Bu gizli uzaydaki farklı noktalar ek girdi bilgisi ve üretilen çıktının farklı kombinasyonlarına karşılık geldiği anlamına gelir.

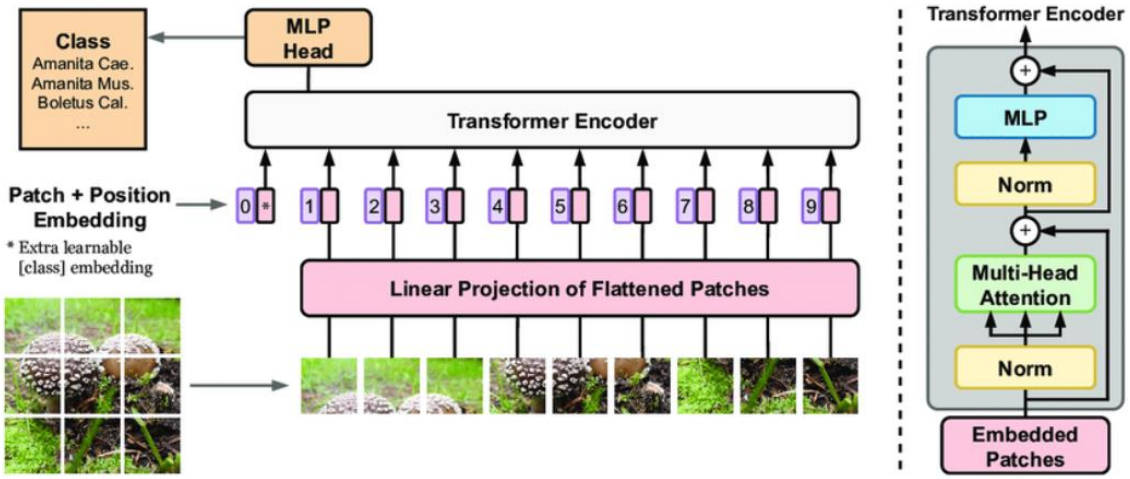
Geleneksel GAN'lar da üretici ağ rastgele görüntüyü girdi olarak alır ve bir çıktı üretir. Üretilen çıktının kalitesi, görüntünün kalitesine bağlıdır. Ancak CGAN'larda, üretici ağ görüntüye ek olarak ekstra girdi bilgisi de alır. Üretilen çıktının kalitesi hem görüntü hem de ek girdi bilgisine bağlıdır. Ekstra girdi bilgisi, üretici ağının izleyeceği bir rehber olarak düşünülebilir ve üreticin daha spesifik, kontrollü çıktılar üretmeye yönlendirmeye yardımcı olur. Üreticinin bu ek girdi bilgileri ile eşleştirerek gizli uzay daha yapılandırılmış hale gelir, üretilen çıktılar daha spesifik ve kontrol edilebilir olur.

CGAN'ların avantajlarından biri klasik GAN'lara kıyasla daha belirli ve kontrol edilebilir çıktılar üretebilmeleridir. Bu da CGAN'ların günümüzdeki gerçek problemlerin çözümü için daha avantajlı hale gelmesini sağlar. Örneğin, CGAN'lar belirli tasarım özelliklerine dayalı olarak giyim eşyalarının görüntülerini vs. oluşturmak için kullanılabilirler.

2.2.5 Dönüştürücü Mimarisi

Dönüştürücü mimarisi, özellikle doğal dil işleme alanında son yıllarda yaygın bir şekilde popülerlik kazanan derin öğrenme mimarisi türüdür. 2017 yılında Vaswani vd. tarafından yayınlanan makalede [17] tanıtılmış ve o zamandan beri yaygın kullanılan mimarilerden biri haline gelmiştir.

Dönüştürücü mimarisi, kodlayıcı ve çözücü olmak üzere iki ana bileşenden oluşur. Kodlayıcı bir girdi dizisi alır ve bir dizi üretir, bu dizi daha sonra çözücü tarafından çıktı dizisini oluşturmak için kullanılır. Kodlayıcının ve çözücünün her katmanı, Öz dikkat mekanizması ve ileri beslemeli sinir ağı olmak üzere iki temel bileşen içerir. Şekil 2.6'da Dönüştürücü mimarisi gösterilmiştir:



Şekil 2.6 Dönüştürücü mimarisi [37]

2.2.6 Dönüştürücü Mimarisinin Avantajları ve Uygulama Alanları

Dönüştürücü mimarisi, RNN'ler ve CNN'ler gibi geleneksel mimarilere göre birkaç avantaja sahiptir. Bunlardan biri uzun mesafeli bağımlılıkları yakalama kabiliyetidir.

Dönüştürücü mimarisi girdi dizisinin her bir öğesine dikkat mekanizması kullanarak, birbirinden uzak öğeler arasındaki bağımlılıkları da yakalamaktadır.

Dönüştürücü mimarisinin bir başka avantajı da işlemlerin paralelleştirilebilir olmasıdır. RNN'lerin sırayla işlem yapmak zorunda olmasının aksine, Dönüştürücü mimarisi dizideki tüm öğeleri paralel olarak işleyebilir ki, bu da onu daha hızlı ve verimli hale getirir.

Paralelleştirilebilirlik makine çevirisi ve dil modellemesi gibi büyük miktarda veri işleme gerektiren görevler için özellikle çok önemlidir. Dönüştürücü mimarisi ile bu görevler geleneksel mimarilere göre çok daha hızlı ve daha az hesaplama kaynağı kullanılarak tamamlanabilir.

Ayrıca, Dönüştürücü mimarisi geleneksel mimarilere göre daha açıklanabilir bir yapıya sahiptir. Öz-dikkat mekanizması ile model, girdi dizisindeki her öğeye atanan önemli

ağırlıkları görselleştirebilir ki, bu da araştırmacıların ve geliştiricilerin modelin tahminlerini nasıl yaptığını daha iyi anlamalarını sağlar.

Dönüştürücü mimarisi dil modelleme, duygu analizi ve metin sınıflandırması gibi farklı alanlarda kullanılmaya başlanmıştır. Dönüştürücü mimarisinin başarılı uygulamalarından biri makine çevirisinde kullanılmasıdır.

Dönüştürücüler önceki kelimeler verildiğinde bir cümledeki sonraki kelimeyi tahmin etmeyi içeren dil modelleme çalışmalarında da yaygınlaşmaya başlamıştır. Dil modellemesi önemli bir çalışmadır, çünkü makine çevirisi ve metin oluşturma gibi birçok alt görevin temelini oluşturur. Dönüştürücü mimarisi ayrıca metin sınıflandırması için de kullanılmıştır. Metin sınıflandırması, bir metnin önceden tanımlanmış kategorilerden birine sınıflandırılmasını içerir. Metin sınıflandırması, spam tespiti ve içerik düzenleme gibi çalışmalar için kullanılabilen önemli araştırma konularından biridir.

2.2.7 Dönüştürücü Türleri

Dönüştürücü mimarisi, doğal dil işleme alanında devrim yaratan güçlü bir derin öğrenme mimarisidir. Aynı zamanda Dönüştürücü mimarisi farklı türdeki problemler için kullanılan çeşitli türlere sahiptir. Örneğin;

Bidirectional Encoder Representations from Transformers (BERT) Google tarafından duygusal doğal dil işleme görevleri için geliştirilmiş Dönüştürücü tabanlı modeldir [23]. BERT çift yönlü bir dil işleme özelliğinin olması bakımından benzersizdir. Bu çift yönlü dil işleme özelliği BERT'in geleneksel tek yönlü modellere göre bağlam ve bağımlılıkları daha etkili bir şekilde yakalamasına olanak tanır.

Vision Transformer görüntü sınıflandırma, nesne algılama ve segmentasyon gibi problemler için tasarlanmış bir Dönüştürücü mimarisidir. ViT görüntü yamalarını eşzamanlı olarak işleyen Dönüştürücü kodlayıcılarını kullanarak geleneksel evrişim katmanlarını değiştirir, bu da modelin yerel ve küresel bağımlılıkları yakalamasına izin verir. ViT, geleneksel CNN tabanlı modellere kıyasla, görüntü sınıflandırma için birkaç veri kümesinde etkileyici performans sergilemiştir. ViT, ayrıca nesne algılama ve

segmentasyon gibi diğer bilgisayarlı görü çalışmalarında da uygulanmış ve umut verici sonuçlar göstermiştir.[22]

DEİT (Data – Efficient İmage Transformer DEİT) ViT'nin verimliliğini ve ölçeklenebilirliğini artırmak için tasarlanmış, Facebook tarafından geliştirilmiş bir Dönüştürücü tabanlı modeldir [24]. DEİT, bir bilgi damıtımı yaklaşımı kullanılarak eğitilir.

2.2.8 Dönüştürücü Bileşenleri

2.2.8.1 Dikkat Mekanizması

Dikkat mekanizması, doğal dil işleme ve derin öğrenme alanında önemli yer alan Dönüştürücü mimarisinin kritik bir bileşenidir. Dikkat mekanizması, 2014 yılında Bahdanau, Cho, K., ve Bengio, Y. tarafından yayımlanan makalede [25] tanıtılmış ve kaynak ve hedef dizilerini hizalamada son derece etkili olduğu kanıtlanmıştır. 2017 de, dikkat mekanizmasını temel bir bileşen olarak kullanan Dönüştürücü mimarisi tanıtılmış ve çeşitli NLP çalışmalarında iyi sonuçlar elde etmiştir [5][23].

Dikkat mekanizması girdi dizisindeki her öğe arasında dikkat değerlerinin hesaplanmasıyla çalışır. Bu değerler daha sonra, girdi dizisinin ağırlık toplamını hesaplamak için kullanılır. Dikkat mekanizmasının da farklı türleri bulunmaktadır ve her biri dikkat puanlarının hesaplanması açısından farklılık göstermektedir. En yaygın tür girdi dizisinde her öğe arasındaki dikkat değerlerini hesaplayan nokta çarpım dikkatidir [26]. Nokta çarpım dikkati, diğer dikkat mekanizmalarına göre birkaç avantaja sahiptir. İlk olarak, matris çarpımı kullanılarak kolayca uygulanabilmesi nedeniyle hesaplama açısından verimlidir. İkinci olarak, dizilerde uzun mesafe bağımlılıklarını yakalamada son derece etkili olduğu gözlemlenmiştir.

Dikkat mekanizmasının başka bir türü ise eklemeli dikkattir [25]. Bu dikkat mekanizması dikkat puanlarını girdi dizisindeki her öğenin ağırlık toplamı olarak hesaplar. Bu da modelin daha karmaşık problemleri öğrenmesine izin verir.

Temel dikkat mekanizmalarının yanı sıra, son yıllarda önerilen bir diğer tür de öz dikkat mekanizmasıdır [17]. Dönüştürücü mimarisi için özel olarak tasarlanmış bir dikkat mekanizması türüdür. Öz dikkat mekanizması modelin girdi dizisinde farklı ayrıntı düzeylerinde farklı parçalara dikkat etmesine olanak tanır, çünkü girdi dizisi içindeki farklı konumlar arasında dikkat puanları hesaplanarak elde edilir. Öncelikle üç farklı

matris kullanılarak hesaplanır, bunlar, Sorgu matrisi (query matrix), Anahtar matrisi (key matrix) ve Değer matrisidir (value matrix). Dikkat puanları sorgu matrisi ile anahtar matrisinin nokta çarpımı alınarak hesaplanır, ardından bir softmax işlemi uygulanır. Son olarak çıktı, dikkat puanlarını ağırlıklar olarak kullanarak değer matrisinin ağırlık toplamı olarak hesaplanır.

Öz dikkat, dil modellemesi, makine çevirisi ve soru cevaplama dahil birçok görevde başarılı olan Dönüştürücü mimarisinin önemli bir bileşenidir. Model, girdi dizisinin farklı parçalarına odaklanarak, farklı öğeler arasındaki karmaşık ilişkileri yakalamayı öğrenebilir. Bu, modellerin birçok çalışmada performansının artmasına sebep olmuştur.

Doğal dil işlemedeki kullanımının yanı sıra, dikkat mekanizması, bilgisayarlı görü dahil olmak üzere derin öğrenmenin diğer alanlarına da uygulanmıştır. Bunların en önemlilerinden biri, görüntü sınıflandırma problemine Vision Transformer mimarisinin (ViT) uygulanmasıdır.

2.2.8.2 İleri Beslemeli Ağ

Dönüştürücü modeli son yıllarda devrim yaratmış ve makine çevirisi, soru cevaplama ve dil modellemesi gibi birçok NLP görevinde performans açısından yeni standartlar belirlemiştir. Dönüştürücü mimarisi, çıktı oluştururken girdinin farklı parçalarının önemini ölçebilme yeteneği sağlayan öz-dikkat mekanizmalarına dayanmaktadır. Öz-dikkatin yanı sıra, Dönüştürücü mimarisi İleri Beslemeli Ağ (Feed Forward Network-FFN) içerir ki, bu da modelin girdi vektörünü öğrenme işleminde kritik bir rol oynar.

Dönüştürücü mimarisindeki FFN, iki tam bağlantılı katmandan oluşur ve her biri bir doğrusal dönüşümün ardından bir doğrusal olmayan aktivasyon fonksiyonu ile takip edilir. İlk katman, gömülü girdi vektörü üzerinde doğrusal bir dönüşüm gerçekleştirirken, ikinci katman, birinci katmanın çıktısı üzerinde başka bir doğrusal dönüşüm gerçekleştirerek son çıktıyı oluşturur. FFN'de kullanılan aktivasyon fonksiyonu genellikle doğrusal olmayan ReLU'dur. Bu fonksiyon girdi vektörleri arasındaki karmaşık doğrusal olmayan ilişkileri öğrenme olanağı sağlar. FFN girdi vektöründeki her bir elemana aynı şekilde, aynı ağırlıklarla uygulanır ve dönüştürücü modelinin değişen uzunluklardaki dizileri paralel olarak işleyebilmesine olanak tanır. Bu özellik modelin uzun girdi dizileri içeren NLP görevlerinde verimli olmasını sağlar.

Girdi dizilerinin karmaşık temsillerini öğrenmek, doğal dil işleme alanındaki birçok görev için oldukça önemlidir. Bu konuda, FFN Dönüştürücü mimarisinde kritik bir bileşendir. FFN girdi dizileri arasındaki doğrusal olmayan ilişkileri modelleyerek, modelin girdileri paralel olarak işlemesi ve uzun girdi dizilerinin verimli bir şekilde ele alınması sağlar.

FFN girdi belirteçleri arasındaki ilişkileri modellemek için lineer olmayan fonksiyonlar kullanır. Bu sayede, girdi dizileri arasındaki karmaşık ilişkilerin öğrenilmesi mümkün hale gelir. Bu özelliği sayesinde FFN, RNN'ler ve CNN'ler gibi diğer modellerden ayrılır ve uzun girdi dizilerini işlemek için daha verimli hale gelir.

Girdi dizilerinin karmaşık temsillerini öğrenmek birçok görev için önemlidir. Bu amaçla, FFN gibi derin öğrenme yöntemleri kullanılır. Dönüştürücü mimarisi, FFN ve Multi-head attention bileşenlerini bir araya getirerek, iyi performans gösterir. FFN ayrıca Dönüştürücü modelinin değişken uzunluktaki girdi dizilerini paralel olarak işlemesine olanak tanır. Bu özellik, modeli uzun girdi dizileri içeren çalışmalar için son derece verimli kılan başka bir önemli özelliktir. Bu verimlilik, aynı ağırlıkların girdi dizisindeki her pozisyona uygulanarak elde edilir ve bu da modelin diziyi paralel olarak işlemesine olanak tanır.

2.2.8.3 Konumsal Kodlama

Dönüştürücü mimarisinde, dikkat mekanizması modelin farklı girdi dizisi parçalarına seçici bir şekilde odaklanmasına olanak tanır. Modelin girdi dizisi öğelerinin sırasını ve konumunu etkili bir şekilde yakalaması için konumsal kodlamanın modele dahil edilmesi gerekmektedir. Bunun için dönüştürücü mimarisi konumsal kodlama adı verilen bir teknik kullanır. Bu teknik, modelin girdi gömmeleriyle birlikte konumsal bilgini öğrenmesine olanak tanır. Bu konumsal gömmeler, girdi dizisindeki her bir öğenin konumu hakkında bilgi kodlar ve modelin farklı dizi öğeleri arasındaki konuma bağlı ilişkileri öğrenmesine izin verir.

Dönüştürücü modelinde kullanılan dikkat mekanizmasının girdi dizisi öğelerinin sırasını veya konumunu yakalamasında konumsal kodlamanın önemli rolü vardır. Dil modellemesi için kullanılan bir Dönüştürücü modelini inceleyerek konumsal kodlamanın önemini açıklamak gerekir. Bu kez girdi bir kelime dizisidir. Dönüştürücü modelindeki dikkat mekanizması modelin girdi dizisinin farklı parçalarına seçici bir şekilde odaklanmasına olanak tanımaktadır. Ancak modelin girdi dizisindeki kelime sırasını ve pozisyonunu doğru bir şekilde yakalaması için, modelin bir tür konumsal bilgiyi modele

dahil etmesi gerekmektedir. Bu, dikkat mekanizmasının tek başına girdi dizisindeki kelime sırasını ve pozisyonunu yakalayamadığından kaynaklanır. Konumsal kodlama bu sebeple dönüştürücü mimarisinde kullanılır. Konumsal kodlama modele konumsal gömmeler kümesi ekler. Bu konumsal gömmeler, girdi dizisindeki her kelimenin pozisyonu hakkında bilgi tutar ve modelin dizideki kelimeler arasındaki bağımlı ilişkileri öğrenmesine olanak tanır.

Dönüştürücü mimarisinde konumsal kodlama için yaygın bir yaklaşım, her öğenin pozisyonunu kodlamak için sinüzoidal bir fonksiyon kullanmaktır. 2017 yılında Vaswani vd. [17], konumsal kodlamanın denklemini sinüs ve kosinüs fonksiyonları ile denklem 2.3 ve denklem 2.4'teki gibi tanımlamışlardır.

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}}) \quad (2.3)$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}}) \quad (2.4)$$

Burada pos kelimenin cümledeki yeri, i gömmenin indisi, d_{model} ise modelin kullandığı kelime gömmesinin boyutu temsil etmektedir. Sinüs fonksiyonu konumsal gömmelerin pozisyon bilgisini sürekli ve türevlenebilir bir şekilde tutar. Dönüştürücü mimarisinde kullanılan sinüzoidal konumsal kodlamanın avantajlarından biri modelin farklı öğeler arasındaki konum bağımlı ilişkileri öğrenmesine olanak tanıması ve konum bilgilerini öğrenmesine izin vermesidir. Konumsal kodlama vektörleri sabit olup girdi dizisinin uzunluğuna bağlı olmadığı için, model farklı uzunluklarda girdi dizilerine genelleme yapabilir.

Sonuç olarak, Dönüştürücü mimarisinde konumsal kodlama, modelin girdi gömme vektörlerine konum bilgisi dahil etmek için kullanılan önemli bir tekniktir. Girdi gömmelerine sabit konumsal gömmeler ekleyerek, model girdi dizisindeki farklı öğeler arasındaki konuma bağlı ilişkileri öğrenebilir ve dizideki öğelerin sırasını etkili bir şekilde yakalayabilir.

Genel olarak, konumsal kodlama, doğal dil işleme, konuşma tanıma gibi çeşitli uygulamalarda Dönüştürücü modellerinin performansını ve esnekliğini artırmak için güçlü bir tekniktir.

2.2.8.4 Öğrenilen Gömme Kavramı

Dil çevirisi, dil modelleme ve duygu analizi gibi birçok görevde derin öğrenme modellerinin son derece etkili olduğu gözlemlenmiştir. Bu modellerin başarısı gösterimlerin kalitesine önemli bir şekilde bağlıdır.

NLP'deki "gömme" terimi, kelime veya ifadeleri bir kelime dağarcığından yüksek boyutlu bir vektör uzayına eşleme sürecine atıfta bulunur. Başka bir deyişle, gömme, her kelimenin veya ifadenin bağlamsal bilgisini yakalayarak, aralarındaki anlamsal ve sözdizimsel ilişkilerin hesaplanmasını mümkün kılar.

Dönüştürücü modelinde, girdi dizisi önce bir gömme katmanından geçirilir. Bu katman, dizideki her öğeyi sabit uzunluklu bir vektör gösterimine eşler. Bu eşleme, eğitim süreci sırasında öğrenilir ve farklı öğeler arasındaki anlamsal ve sözdizimsel ilişkileri yakalamak için optimize edilir. Son olarak, Dönüştürücü modelindeki öz dikkat mekanizması modelin farklı girdi dizisi parçalarına odaklanmasına izin verir ve farklı öğeler arasındaki uzun mesafeli bağımlılıkları yakalamaya yardımcı olur.

Dönüştürücülerdeki öğrenilmiş gömme yapıları, doğal dil işleme için birkaç önemli sonucu beraberinde getirmiştir. İlk olarak, kelime veya simgeler arasındaki anlamsal ve sözdizimsel ilişkilerin hesaplanabilmesini sağlarlar, bu da dil çevirisi ve duygu analizi gibi birçok doğal dil işleme görevi için temel gerekliliklerden biridir. İkinci olarak, gömme yapıları sınıflandırıcılar gibi diğer makine öğrenimi modellerinin girdi özellikleri olarak kullanılabilir ve gömme tarafından yakalanan bağlamsal bilgiyi kullanarak daha iyi sonuçlar elde edilebilir.

Öğrenilmiş gömme yapıları, RNN tabanlı modellere göre birkaç avantaja sahiptir. İlk olarak, öğrenilmiş gömme kullanan Dönüştürücü modeli, RNN tabanlı modellerle mümkün olmayan kelime ve simgeler arasındaki uzun mesafe bağımlılıklarını yakalayabilir. İkinci olarak, Dönüştürücü modeli, RNN tabanlı modellerin karşılaştığı kaybolan gradyan [28] problemiyle karşı karşıya kalmaz. Üçüncü olarak, Dönüştürücü modeli, yüksek derecede paralelleştirilebilir, bu da RNN tabanlı modellere kıyasla daha hızlı çalışmasını sağlar.

Word2Vec [27] ve GloVe [38] gibi NLP modellerine göre dönüştürücülerdeki öğrenilmiş gömme yapılarının birçok üstünlüğü bulunmaktadır. İlk olarak, Dönüştürücü modeli tahmine Word2Vec ve GloVe modelleri ile mümkün olmayan kelimelerin bağlamsal bilgisini yakalamaktadır. İkinci olarak, Dönüştürücü modeli Word2Vec ve GloVe modellerinin önemli bir kısıtlaması [29] olan kelime haznesinin dışındaki kelimeleri de

işleyebilmektedir. Son olarak, Dönüştürücü modeli Word2Vec ve GloVe gibi NLP modelleri ile mümkün olmayan çoklu dil desteği sağlamaktadır.
Dönüştürücülerdeki öğrenilmiş gömme katmanlarının başarısına rağmen, bu alandaki araştırmalar için hala birkaç zorluk ve gelecek yönelimi bulunmaktadır. Başlıca zorluk öğrenilmiş gömme katmanlarının yorumlanabilirliğidir. Gömme katmanları kelime bağlamlarını yakalamak için kullanılırken, gömme katmanlarının gerçekte kelimelerin altında yatan anlamı nasıl gösterdiğinin yorumlanması zordur. Bu da modellerin hata ayıklamasını zorlaştırır. Başka bir zorluk da Dönüştürücülerdeki öğrenilmiş gömme katmanlarının ölçeklenebilirliğidir. Dönüştürücü modeli, birçok çalışmada iyi sonuçlar (state-of-the-art) elde etmiş olsa da hesaplama maliyeti yüksektir.

2.2.9 Vision Transformer

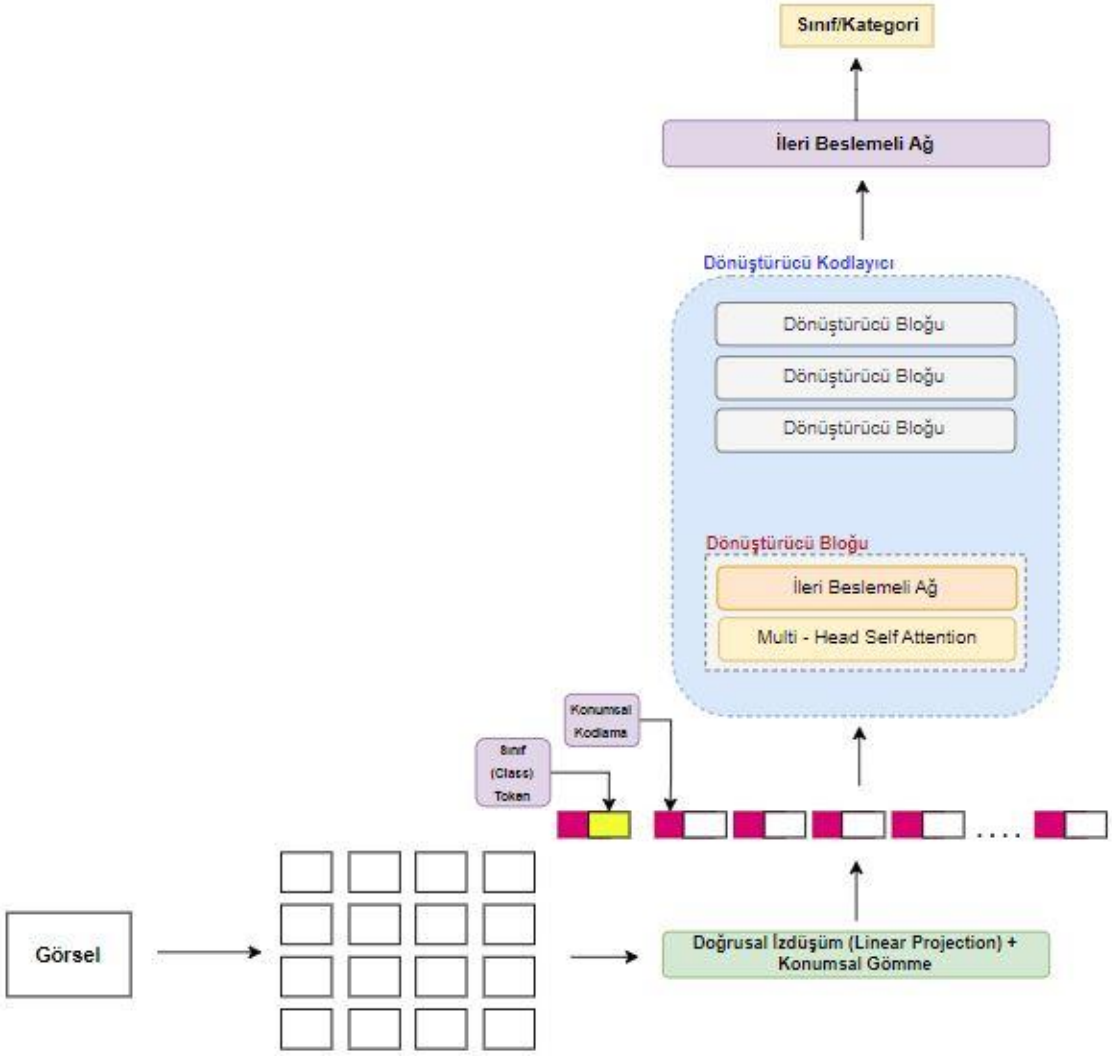
Dönüştürücü mimarisi, ilk olarak doğal dil işleme çalışmaları için geliştirilmiş, daha sonra bilgisayarlı görü çalışmalarında da kullanılmaya başlanmıştır. Bilgisayarlı görü çalışmalarında kullanılan dönüştürücü türlerinden biri de Vision Transformer (ViT)'tir.
Vision Transformer mimarisi geleneksel bilgisayarlı görü modellerindeki CNN'leri, sabit boyutlu yama parçalarına ayrılmış görüntüyü alarak bir dönüştürücü kodlayıcı ile değiştirir ve bu yama parçalarını paralel olarak işler. Öz dikkat mekanizması, modelin farklı yama parçaları arasındaki uzun mesafe bağımlılıkları yakalamasına olanak sağlarken, ileri beslemeli ağ, her yama parçasındaki yerel bilgileri yakalar. Vision Transformer mimarisi, yama gömme, konumsal kodlama, Dönüştürücü kodlayıcı ve çıktı katmanı olmak üzere dört bileşenden oluşur.

Yama gömme görüntüyü sabit boyutlu yamaların bir dizisine dönüştürürken, dönüştürücü kodlayıcı bu yamaları paralel olarak işleyerek görüntüdeki yerel ve küresel bağımlılıkları öğrenmeye yardımcı olur. Yama gömme, ViT mimarisinin ilk bileşenidir ve görüntüyü sabit boyutlu yama dizisine dönüştürmekten sorumludur. Yama boyutu, kullanıcı tarafından belirlenebilen bir hiperparametredir. Yamaların büyüklüğü arttıkça daha fazla bağlam yakalar ancak daha fazla hesaplama gerektirir. Küçük yamalar ise daha az bağlam yakalasa da çok hesaplama gerektirmez. ViT'de en yaygın yama boyutu 16x16'dır.

Görüntü yama dizisine dönüştürüldükten sonra, yamaların hangi sırayla geldiğini bilmek için bir konum numarası eklenir. Konum numarası, her yamanın görüntüdeki konumunu

kodlar ve modelin aynı içeriğe sahip yamalar arasındaki farkı ayırt etmesine yardımcı olur.

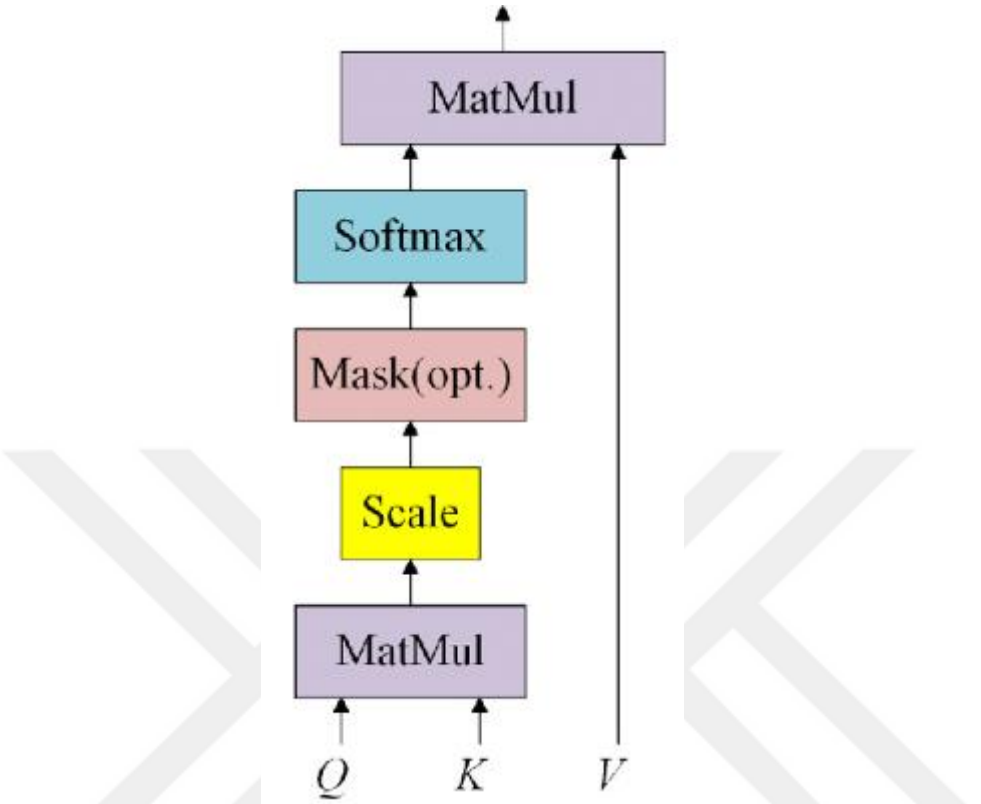
İkinci bileşen olan dönüştürücü kodlayıcı yama dizisini işleyerek yerel ve küresel bağımlılıkları öğrenmekten sorumludur. Genel ViT mimarisi (Şekil 2.7) dönüştürücü kodlayıcıdan, dönüştürücü kodlayıcıdaki dönüştürücü bloglarından oluşur ve her biri Multi-Head Attention katmanı ve ileri beslemeli ağ içerir.



Şekil 2.7 Genel Vision Transformer mimarisi

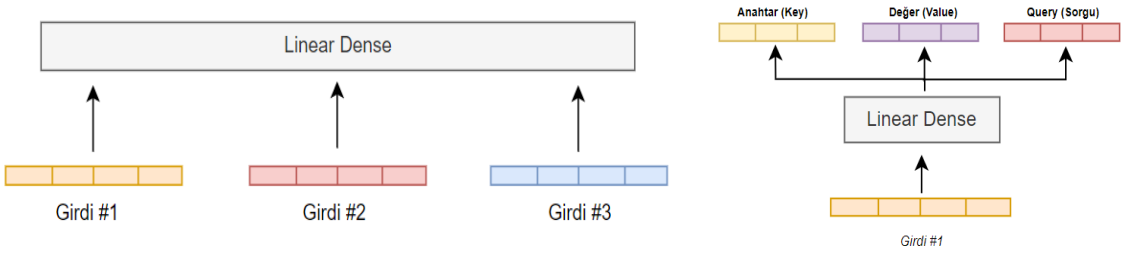
Multi-Head Attention katmanı, modelin girdi dizisinin farklı bölgelerine dikkat etmesine ve küresel bağımlılıkları yakalamasına olanak tanır. ViT'de öz dikkat doğal dil işleme çalışmalarında olduğu gibi tüm girdi dizisi üzerinde değil, numaralanmış yama dizisi üzerinde çalışır.

Multi-Head Attention yapısını anlamak için önce dikkat mekanizmasını göz önüne almak gerekir. Şekil 2.8'te dikkat mekanizması gösterilmiştir.



Şekil 2.8 Dikkat mekanizması [17]

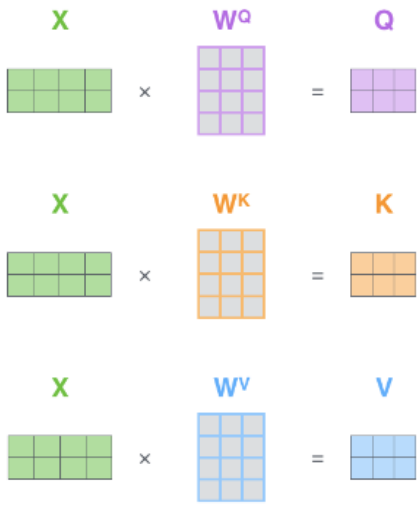
Örneğin, boyutları 4 olan üç girdi değerimiz (Şekil 2.9 (a)) olsun Lineer Yoğun katmanında her girdi değeri için üç temsil elde edilir. Şekil 2.9 (b) de ilk girdi için görselleştirme yapılmıştır.



Şekil 2.9 (a) Lineer Yoğun katmanı **(b)** Lineer Yoğun katmanın çıktıları

Bu temsiller anahtar (key), sorgu (query), değer (value) olarak isimlendirilir. Bu örnek için temsillerin boyutunun üç olduğunu varsayalım. Tüm girdilerin boyutu 4 olduğundan

her bir ağırlık kümesinin boyutu 4x3 şeklinde olmalıdır. Şekil 2.10'da ağırlıklar matrisleri gösterilmiştir.

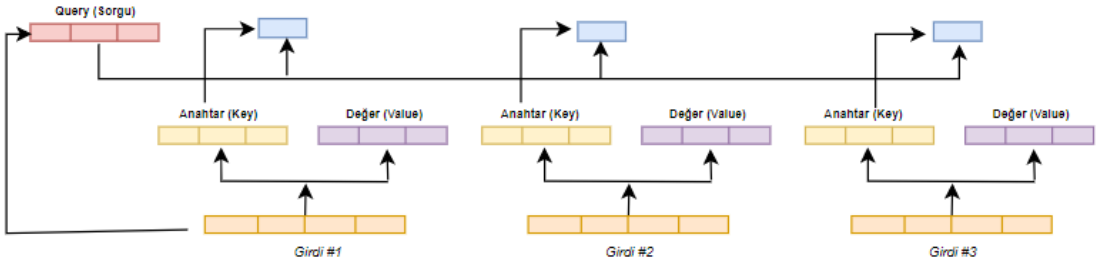


Şekil 2.10 Gömme vektörlerinin ağırlıklarla çarpılıp sorgu, anahtar, değer değerlerinin hesaplanması [39]

Yapılan matris çarpımlarından sonra Sorgu, Anahtar ve Değer değerleri elde edilir. Ardından matris çarpımı (Mat. Multiplication) katmanından geçen matrisler sorgunun anahtarın transpozu ile çarpımından (denklem 2.5) hesaplanarak ham matris oluşturur. Çıktı filtre matrisi kelimelerin birbiri ile ilişkilerini içerir.

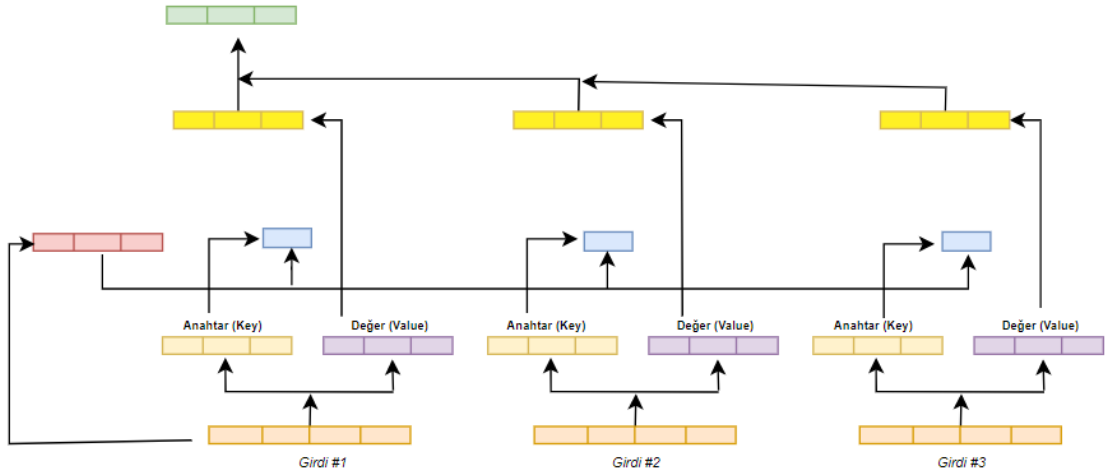
$$Sorgu * Anahtar^T = Çıktı\ Filtre \tag{2.5}$$

Bu katmanın ardından ölçekleme ve softmax işlemleri sıra ile uygulanmaktadır. Ölçekleme aşamasında dikkat puanları hesaplanmaktadır. Birinci girdi için dikkat puanı hesaplarken kendisi dahil tüm anahtarlar ile nokta çarpımı yapılmaktadır. 3 temel temsil (sorgu, anahtar, değer) olduğu için 3 dikkat puanı elde ederiz. Dikkat puanları şekil 2.11'de mavi kutucukla gösterilmiştir. Bu işlem sadece ilk sorgu için yapılmıştır aynı adım diğer sorgular içinde yapılmalıdır. Dikkat puanları bize her kelimenin diğerleri ile olan bağlantısını gösterir.



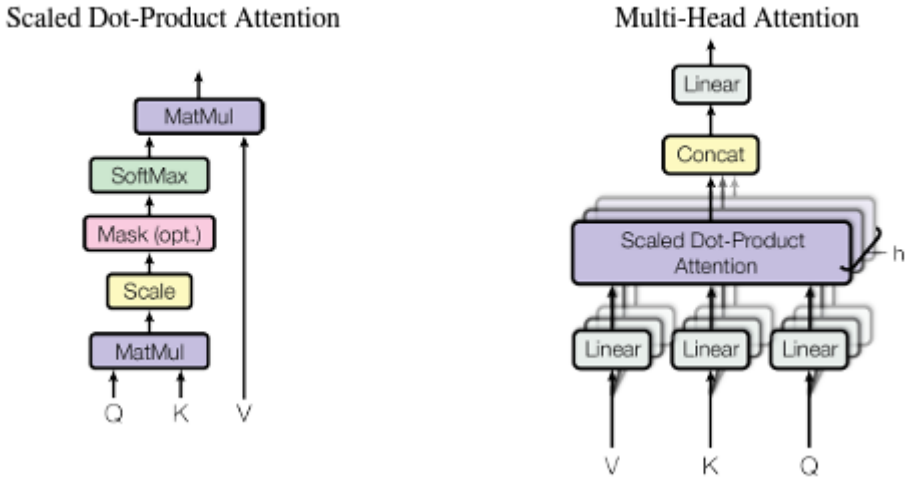
Şekil 2. 11 Dikkat puanlarının hesaplanması

Daha sonra softmax işlemine geçilir. Her dikkat puanı girdi değerleri ile çarpılarak hizalama vektörü oluşturur. Üç girdi olduğundan 3 hizalama vektörü, bu vektörlerin toplamından ise ağırlıklı değerler elde edilir. Şekil 2.12 de hizalama vektörleri sarı renkli kutucukla, ağırlıklı değer ise yeşil kutucukla gösterilmiştir.



Şekil 2.12 Hizalama vektörü ve ağırlık değerlerinin hesaplanması

Bu filtrelerin birden çok kullanımı modele farklı özellikleri öğrenme imkânı sağlar. Dönüştürücü mimarileri de bu filtreleri birleştirerek farklı özellikleri öğrenirler. Multi-Head Attention Şekil 2.13'te gösterildiği gibi bu filtrelerin birleşiminden oluşur.



Şekil 2.13 Multi-Head Attention mimarisi [17]

Her dönüştürücü bloğundaki ileri beslemeli ağ, öz-dikkat katmanının çıktısını işlemek ve bloğun nihai çıktısını üretmekten sorumludur. Son dönüştürücü bloğunun çıktısı ise nihai sınıflandırma çıktısını elde etmek için doğrusal bir katmana geçirilir.

2.2.9.1 Vision Transformer Uygulama Alanları

Vision Transformer'ın en yaygın uygulaması görüntü sınıflandırmasıdır. ViT geleneksel CNN tabanlı modelleri geride bırakarak birkaç veri seti üzerinde etkileyici performans sergilemiştir. ViT ayrıca görüntüde nesne tespiti görevlerine de uygulanmış, bu çalışmalarda da başarı göstermiştir. ViT'den alınan yama görüntüleri, önerilen bölge için özellikler olarak kullanılmış ve bu özelliklerin üzerine ayrı bir sınıflandırıcı eğitilmiştir. Bu şekilde ViT nesne sınıfını ve konumunu tahmin etmiştir [31]. Ancak, nesne tespiti için hala nispeten yeni bir yaklaşım olduğundan, potansiyelini tam olarak keşfetmek için daha fazla araştırma yapılması gerekmektedir.

ViT'nin bir diğer uygulaması, modelin görüntünün her pikseline bir semantik etiket ataması için eğitildiği semantik bölütlemedir. ViT semantik bölütleme çalışmalarında öncelikle girdi görüntüsü için özellik çıkarıcı olarak kullanılmakta, ardından ayrı bir ağın tahminleri ile elde edilen özellikleri birleştirerek nihai bölütleme çıktısı elde edilmektedir. Zhang vd. 2022'deki çalışmasında SegViT'in önceki yaklaşımlara göre daha az parametre gerektirerek COCO-Stuff ve ADE20K veri kümelerinde iyi sonuçlar verdiğini göstermişlerdir [32].

2.2.9.2 Vision Transformer Avantajları ve Performansı

Vision Transformer son yıllarda görüntü tanıma ve bilgisayarlı görü görevleri için popülerlik kazanan bir derin öğrenme mimarisidir. ViT resimleri işlemek için öz dikkat mekanizmalarını kullanan bir tür sinir ağıdır. Resimlerden özellik çıkarmak için evrişim katmanlarına dayanan geleneksel evrişimsel sinir ağlarının aksine, görsel bilgiyi işlemek için bir dönüştürücü tabanlı mimari kullanır.

Vision Transformer'ın ana avantajı, görüntülerdeki uzun mesafe bağımlılıkları öğrenebilme yetenekleridir. Görüntünün farklı bölgelerine dikkat etmeyi ve farklı ölçeklerdeki özellikler arasındaki ilişkileri yakalamayı öğrenebilmektedirler. Bu özellikleri Vision Transformer'ların değişen boyutlarda ve çözünürlüklerde olan çeşitli görüntüleri işleyebilmelerini sağlar.

Vision Transformer diğer bir avantajı, ardışık ve görüntü olmayan verileri de işleyebilme yetenekleridir. Dönüştürücüler doğal dil işleme görevleri için tasarlanmış ve geleneksel tekrarlayan sinir ağları, evrişimsel sinir ağları ile karşılaştırıldığında daha iyi performans gösterdikleri kanıtlanmıştır. Performans açısından, ViT görüntü sınıflandırma ve nesne algılama için birkaç veri kümesinde en son teknolojik başarılar göstermiştir. Örneğin, Dosovitskiy vd. tarafından 2021 yılında tanıtılan ViT modeli, ImageNet veri kümesinde %90,6'lık iyi sonuç elde ederek CNN'leri geride bırakmıştır [12]. Touvren 2021 yılında yayınladığı makalesinde [13] iyi sonuçlar elde etmiştir.

Ancak, Vision Transformer'ların kullanımı ile ilgili bazı sınırlamalar ve zorluklar da vardır. En önemlisi hesaplama ve bellek gereksinimleridir. Dönüştürücüler genellikle CNN'lere göre daha büyük ölçekli veri kümeleri ile kullanılır ve daha çok hesaplama yoğundur. Bu da ölçeklenebilirliklerini sınırlayabilir. Diğer bir zorluk, eğitim verilerinin kalitesi ve çeşitliliğine olan hassasiyetleridir. Vision Transformer'lar görsel kavramların temsillerini öğrenmek için büyük, çeşitli veri kümelerine ihtiyaç duyarlar. Böyle veri kümelerinin hazırlanması ve etiketlenmesi zaman alıcı ve maliyetli olabilir.

Sonuç olarak, görüntü tanıma ve bilgisayarlı görü görevleri için Vision Transformer'lar umut verici bir yaklaşımı temsil eder. Uzun mesafe bağımlılıklarını öğrenme yetenekleri ve büyük boyutlu verileri işleme yetenekleri birçok uygulamalar için önemli rol oynamaktadır. Ancak Vision Transformer uygulanırken, hesaplama ve veri gereksinimleri, eğitim verisi kalitesi, çeşitliliğine duyarlılıkları dikkatlice değerlendirilmelidir.

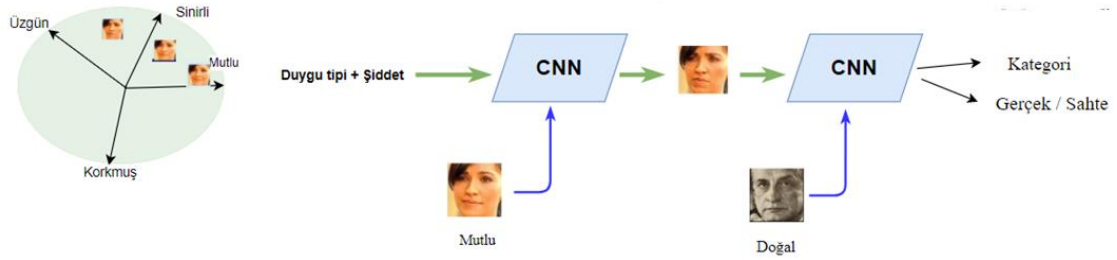
3
UYGULAMA DETAYLARI

3.1 Önceki Model: GANmut

Farklı duygulara karşılık gelen yüz ifadelerini tanıyabilme ve üretebilme yeteneği, insan iletişiminin önemli bir yönüdür. Son yıllarda, bu görevleri otomatik olarak gerçekleştirebilen makine öğrenimi modellerini geliştirmeye yönelik büyük bir ilgi bulunmaktadır. Özellikle, GAN'lar gibi üretken modeller belirli özelliklere sahip gerçekçi ve çeşitli görüntüler oluşturma konusunda umut verici sonuçlar vermiştir. 2021 yılında Kim vd. "GANmut: Learning Interpretable Conditional Space for Gamut of Emotions" [40] başlıklı makalede, yüz ifadeleri üretmek için koşullu GAN (CGAN) mimarisi kullanan bir yaklaşım önermişlerdir.

Koşullu çekişmeli üretici ağlar, görüntü vektörüne ek olarak koşullandırma bilgisini girdi olarak alır. Yüz ifadeleri oluşturmak için mevcut yaklaşımların birkaç kısıtlaması vardır. Bunlardan biri, genellikle karmaşık ön işleme adımlarına dayanarak girdi görüntülerinden duyguyla ilgili özellikleri çıkarmak için yapılan işlemlerdir. Diğer bir kısıtlama ise genellikle çeşitlilik eksikliği olan veya yorumlanamayan görüntüler oluşturmasıdır.

GANmut model mimarisi, geniş bir duygu yelpazesine karşılık gelen yüz ifadeleri oluşturmak için tasarlanmıştır. Bu modelde, bir milyondan fazla yüz görüntüsü, öfke, iğrenme, korku, mutluluk, üzüntü, şaşkınlık veya nötr olarak etiketlenmiş yedi farklı duygu sınıfı içeren AffectNet veri kümesi kullanılmıştır. GANmut model mimarisi Şekil 3.1'de gösterilmiştir.



Şekil 3.1 GANMUT model mimarisi

Yapılan çalışmada Koşullu Çekişmeli Üretici Ağ mimarisi kullanılarak, üretici ağa görüntü vektörü ve koşullandırma vektörü girdi olarak verilmiş ve verilen duyguya karşılık gelen bir görüntü oluşturulmuştur.

GANmut model mimarisinde üreticiden sonra ayrıştırıcı ağ görüntüyü ve koşullandırma vektörünü girdi olarak alır, verilen koşullandırma bilgisine uygun olup olmadığına bağlı olarak görüntüyü gerçek veya sahte (sentetik) olarak sınıflandırır. Bu yaklaşım, yüz ifadeleri gibi belirli özelliklere sahip görüntülerin oluşturulmasına izin verir.

Genel olarak, GANmut model mimarisi, geniş bir duygu yelpazesini temsil eden yüksek kaliteli yüz ifadeleri üretmek için oluşturulmuş bir mimaridir. Bu mimaride üreticinin çıktılarının değerlendirilmesi için Fréchet Başlangıç Mesafesi (Frechet Inception Distance – FID) değerlendirme ölçütü kullanılmıştır. FID üretilen görüntülerin dağılımı ile gerçek görüntülerin dağılımı arasındaki mesafeyi ölçerek görüntülerin kalitesini ölçer. GANmut modeli, bilgisayarlı görü ve yapay zekâ gibi çeşitli alanlarda örneğin, daha gerçekçi sanal avatarlar oluşturmak, robotların duygusal ifade zenginliğini artırmak vb. kullanılabilir. Ancak modelin daha çeşitli yüz görselleri üzerinde daha fazla değerlendirme gerekliliği gibi sınırlaması bulunmaktadır.

3.2 Önerilen Model

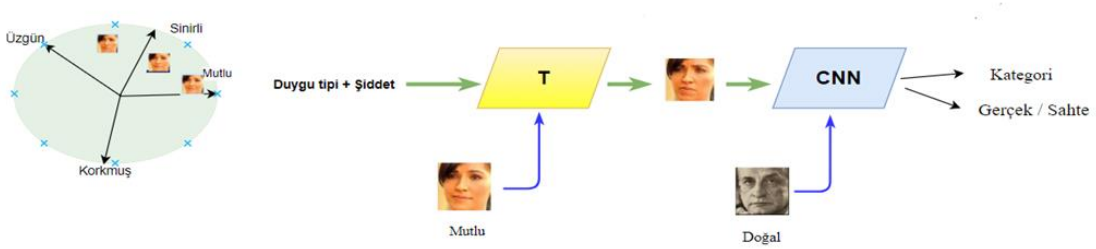
3.2.1 Dönüştürücü Tabanlı GAN Mimarisi

Yüz ifade analizi psikoloji, insan-bilgisayar etkileşimi ve eğlence gibi birçok uygulama alanı olan önemli bir araştırma konusudur. Günümüzde gerçekçi yüz ifadeleri oluşturmak için farklı teknikler üretilmekte, araştırmalar yapılmaktadır.

AffectNet veri kümesi ile eğitilmiş ve literatüre girmiş GANmut mimarisinden esinlenerek bu tez kapsamında yeni bir GAN mimarisi tasarlanmıştır. Önerilen GAN mimarisinde, üretici kısmındaki geleneksel CNN modeli Dönüştürücü ile değiştirilmiş ve Dönüştürücü tabanlı GAN modeli oluşturulmuştur.

Bu çalışmada anlamlı ve yorumlanabilir sonuçlar elde etmek düşüncesiyle farklı duygu sınıflarına karşılık gelen görüntüler üretmek için koşullu GAN kullanarak, Dönüştürücü tabanlı GAN mimarisi önerilmiştir. Koşullu GAN'lar özellikle belirli kategoriler veya etiketlere karşılık gelen görüntüleri oluşturmak için uygun bir şekilde tasarlanmıştır.

Önerilen GAN mimarisi, üretici bileşeninde Dönüştürücü (vision transform), ayrıştırıcı bileşeninde ise VGG19 sinir ağından oluşmaktadır. Şekil 3.2 de Dönüştürücü tabanlı Çekişmeli Üretici Ağ mimarisi gösterilmiştir.



Şekil 3.2 Dönüştürücü tabanlı Çekişmeli Üretici Ağ mimarisi

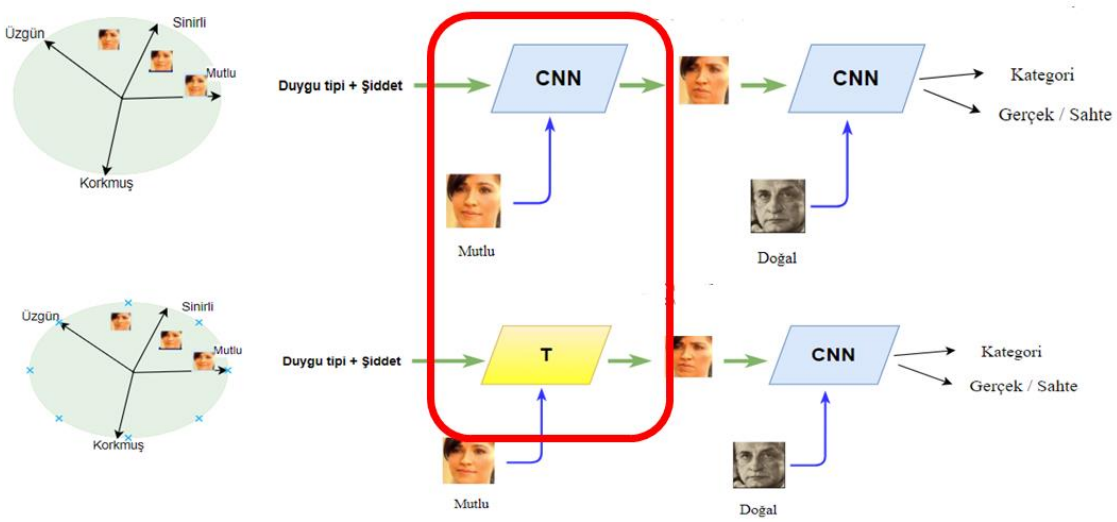
Bu çalışmada modelin performansını değerlendirmek için AffectNet veri kümesinde deneyler yapılmıştır. Bu çalışmanın ana motivasyonu, farklı duygu sınıflarına ait mevcut yöntemlerden daha gerçekçi sentetik veriler oluşturmaktır.

Bu çalışma kapsamında, gerçek insan görselleri üzerinde duygu analizi, bir duygudan yedi farklı duygu tipine ait görsel oluşturma üzerine bir araştırma yapılmaktadır. AffectNet veri seti kullanılarak geleneksel derin öğrenme modellerinden farklı yeni bir model önerilmektedir.

Çekişmeli üretici ağlar ve Dönüştürücü sentezi ile oluşturulan model, görsellerden duygu algılama, yeni duygu kategorisine ait görsel oluşturmak için kullanılabilir. Çalışma kapsamında, bu geliştirilen model ile daha gerçekçi görseller oluşturma metodolojisi önerilmiştir.

GANmut modelinde üreticinin performansını değerlendirmek için FID değeri kullanılmış olup bu çalışma kapsamında geliştirilen modelin performansını değerlendirmek için de FID değeri kullanılmıştır. FID değeri oluşturulan görsellerin dağılımı ile veri seti referans görsellerin dağılımı arasında benzerlik değeridir.

Üretici de geleneksel CNN yerine dönüştürücü kullanılması (Şekil 3.3), FID değeri ölçüsüne göre modelin performansını artırmıştır. Deneylerin sonuçları, yeni modelin mevcut yöntemlerden daha gerçekçi yüz ifadeleri oluşturabildiğini ve daha geniş bir duygu yelpazesi oluşturabildiğini göstermiştir.



Şekil 3.3 Dönüştürücü tabanlı Çekişmeli Üretici Ağ mimarisinde üreticide CNN'nin dönüştürücü ile değiştirilmesi

3.2.2 Kullanılan Teknolojiler

Veri ön işlemesi, modellerin geliştirilmesi, test edilmesi gibi uçtan uca tüm geliştirme süreçlerinde Python programlama dili ve Pytorch kütüphanesi kullanılmıştır. PyTorch 2017 yılında Facebook tarafından Torch'a bağlı Python tabanlı ve açık kaynaklı makine öğrenmesi kütüphanesi olarak yayınlanmıştır. Grafik işlem birimlerini kullanarak daha hızlı bir deneyim sunması onu diğer kütüphanelere göre daha fazla tercih edilir kılmaktadır.

AffectNet verisinin işlenmesi için "Google Cloud" ortamındaki yüksek nitelikli sanal sunucular üzerinde çalışma yürütülmüştür. AffectNet veri setinin büyük olması sebebiyle ve üzerinde çalıştırmayı hedeflediğimiz algoritmaların performanslarını en yüksek seviyede ve kısa sürede gözlemleyebilmek adına Google Cloud ortamında sanal makine (Virtual Machine – VM) oluşturulmuştur. Çalışmamız için gereken ortam ve ürünler Nvidia A100 Gpu, 85 GB RAM ve 12 A2 CPU'ya sahip Ubuntu 20.04 işletim sistemli bir sanal makineye kurulmuştur.

NVIDIA A100, yüksek performanslı hesaplama, derin öğrenme ve yapay zekâ çalışmaları için özel olarak NVIDIA tarafından tasarlanmış bir en son teknoloji GPU hızlandırıcısıdır. A100'ün temel avantajlarından biri, NVIDIA Ampere GPU mikro mimarisine dayalı bir mimaridir. Bu mimari, AI yükleri için eşi benzeri görülmemiş performans sunan üçüncü nesil Tensor Cores özelliklerini içerir ve önceki nesil GPU'lara

göre 20 kata kadar daha hızlı AI eğitimi ve 10 kata kadar daha hızlı çıkarım sağlamaktadır. Bulut üzerinde A100 GPU'larına erişerek, kullanıcılar derin öğrenme modellerini hızlandırabilir ve eğitim, çıkarım yüklerini daha büyük veri kümelerine, daha karmaşık modellere ölçeklendirebilirler.

Çalışmadaki tüm deneyler Python programlama dili ve açık kaynak kodlu derin ögrenme kütüphanesi olan Pytorch ile gerçekleştirilmiştir. PyTorch, Makine öğrenimi ve Derin Öğrenme modellerini destekleyen, CPU ve GPU kullanan açık kaynaklı bir derin öğrenme kütüphanesidir. Arka planında kullandığı CPU ve GPU sayesinde tensör hesaplamalarında daha yüksek hız ve performans sunma becerisine sahiptir. Yüksek sayıdaki veri model performansını artırırken yüksek hesap gücü gerektirmektedir. Bu tezdeki çalışmada büyük veri seti kullandığı için hesaplama gücü ve model eğitim süresi öncelikli kriterler olarak belirlenmiştir.

Kod yapılandırma, kod düzenleme için Visual Studio Code ürünü, model çıktılarının görselleştirilmesi için ise Weights & Biases (W&B) platformu kullanılmıştır. Visual Studio Code Linux, Mac OS ve Windows işletim sistemlerinde çalışan IDE ortamıdır.

Weights & Biases (W&B), makine öğrenimi (Machine Learning- ML) modellerini takip etmek, görselleştirmek ve hata ayıklamak için araçlar sunan bir ML platformudur. 2017 yılında, geliştiricilerin ve veri bilimcilerin ML modelleri oluşturmayı ve optimize etmeyi kolaylaştırmak amacıyla kurulmuştur. Temelinde, W&B deney takibi ve yönetimi için bir platformdur. Kullanıcılar hiper parametreleri, metrikleri, kod sürümlerini ve veri seti sürümleri gibi ML deneylerinin çeşitli yönlerini kaydedebilir ve bir gösterge tablosunda görselleştirebilir. Bu da birden fazla deneyi takip etmeyi, karşılaştırmayı ve yeniden üretmeyi daha kolay hale getirir. Deney takibinin yanı sıra, W&B aynı zamanda model görselleştirme ve hata ayıklama için araçlar da sağlar. Kullanıcılar modelin performansını zaman içinde görselleştirebilir, nasıl tahminler yaptığını görebilir ve iyileştirme alanlarını belirleyebilir. W&B, farklı endüstrilerdeki girişimlerden büyük kuruluşlara kadar geniş bir kullanıcı kitlesi tarafından kullanılmaktadır. Kolay kullanımı, esnekliği ve ML geliştirme verimliliğini ve etkililiğini artırma yeteneği nedeniyle, makine öğrenmesinde uygulama geliştirmenin popüler bir aracı haline gelmiştir.

Uygulama detayları bölümünde, Şekil 3.1 ve Şekil 3.2'de bahsedilen model mimarilerinin geliştirilmesi, testleri yukarıda açıklanan teknolojiler ve donanımsal alt yapı kullanılarak gerçekleştirilmiştir.

3.2.3 Değerlendirme Metrikleri

GAN, gerçek görüntüler kümesinin bir kısmı ile eğitim yaparak yeni görüntüler üretmeyi öğrenen derin öğrenme modelleridir. FID değeri, Çekişmeli Üretici Ağlar tarafından üretilen görüntülerin kalitesini değerlendirmek için kullanılan popüler bir ölçüttür.

FID değeri, üretilen sentetik görüntüler ve gerçek görüntüler arasındaki benzerliğin bir ölçüsüdür. İlk kez 2017 yılında Martin Heusel vd. tarafından yayınlanan makalede [14] önerilmiştir. FID değeri (Denklem 3.1), üretilen ve gerçek görüntülerden çıkarılan özelliklerin üzerine uygulanan Inception-v3 adlı önceden eğitilmiş bir sinir ağı kullanılarak iki Gaussian dağılımı arasındaki mesafeyi ölçer.

$$FID = |\mu - \mu_{\omega}|^2 + tr\left(\sum + \sum_{\omega} - 2\left(\sum \sum_{\omega}\right)^{1/2}\right) \qquad (3.1)$$

Inception-v3, başlangıçta görüntü sınıflandırma görevleri için tasarlanmış bir derin evrişimsel sinir ağıdır. Gerçek dünya görüntüleri üzerinde önceden eğitilmiş ve görüntülerden anlamlı özellikler çıkarma yeteneğine sahiptir. FID değeri bağlamında, Inception-v3 hem üretilen görüntülerden hem de gerçek görüntülerden özellikler çıkarmak için kullanılır. Bu özellikler daha sonra FID değerini hesaplamak için kullanılmaktadır. İlk olarak, üretilen ve gerçek görüntülerin özellik vektörlerinin ortalaması ve kovaryans matrisi hesaplanır. Daha sonra, bu iki Gaussian dağılımı arasındaki Fréchet mesafesi hesaplanır. Fréchet mesafesi, iki olasılık dağılımı arasındaki mesafenin bir ölçüsüdür. FID değeri ne kadar düşük olursa, üretilen görüntüler gerçek görüntülere o kadar benzer.

GAN'ların değerlendirilmesi için ölçüt olarak FID değeri kullanmanın avantajı, görüntü kalitesi üzerine insan değerlendirmesi ile iyi bir şekilde ilişkili olmasıdır. Başka bir deyişle, daha düşük bir FID değeri, üretilen görüntülerin gerçek görüntülere daha benzer olduğu anlamına gelir.

3.2.3 AffectNet Veri Seti

AffectNet veri seti; üç büyük arama motoru (Google, Yahoo, Bing) üzerinde altı farklı dilde (İngilizce, İspanyolca, Portekizce, Almanca, Arapça ve Farsça) 1250 duygu ile anahtar kelimeler kullanılarak arama motorlarından sorgulanarak elde edilmiştir [20].

Temelde yedi duygu (nötr, mutlu, üzgün, öfkeli, korkmuş, iğrenmiş, şaşırmış) etiketli olarak, 1 milyondan fazla insan yüzü resmini içerir. Tablo 3.1 de bu veri setindeki görsellerin duygu sınıfları ve görsel veri sayısı gösterilmiştir.

Tablo 3.1 AffectNet veriseti duygu dağılımı [20]

Duygu	Yüz Görüntü Sayısı
Nötr (Neutral)	80276
Mutlu (Happy)	146198
Üzgün (Sad)	29487
Şaşırmış (Suprise)	16288
Korkmuş (Fear)	8191
İğrenme (Disgust)	5264
Kızmak (Anger)	28130
Hor Görmek (Contempt)	5135
İfadesiz (None)	35322
Belirsiz (Uncertain)	13163
Hiçbir yüz (Non-face)	88895

AffectNet veri kümesi geniş bir yelpazede kullanım alanları olan, özellikle de psikoloji, insan-bilgisayar etkileşimi ve pazarlama araştırmaları gibi alanlarda yüz ifadelerinden duyguları tanımak sıklıkla kullanılır. AffectNet'deki görüntüler, çevrimiçi fotoğraf paylaşım web siteleri ve arama motorları gibi çeşitli kaynaklardan toplandı. On iki uzman tarafından elde edilen veri kümesinin yaklaşık yarısı yedi farklı yüz ifadesi etiketi ile manuel olarak etiketlenmiştir [20]. Farklı aydınlatma koşulları altında, farklı ayarlarda, farklı yaş ve cinsiyete sahip çeşitli insan resimlerini içermektedir. AffectNet veri kümesi, Carnegie Mellon Üniversitesi ve Pittsburgh Üniversitesi'ndeki araştırmacıların iş birliğiyle oluşturulmuştur. Genel olarak, AffectNet veri kümesi, duygu tanıma ve ilgili teknolojiler üzerinde çalışan araştırmacılar ve geliştirmeciler için önemli bir kaynaktır.

Şekil 3.4'te AffectNet veri setinden örnek veriler gösterilmiştir.



Şekil 3.4 AffectNet veri seti örnekleri [20]

Büyük boyutu ve çeşitli görüntüleri ile makine öğrenimi modellerini eğitmek ve değerlendirmek için değerli bir veri kümesi haline gelmiştir. AffectNet veri kümesi birçok derin öğrenme mimarisi ile değerlendirilmiş, bunlar arasında evrişimsel sinir ağları, tekrarlayan sinir ağları ve bu mimarilerin kombinasyonları yer almaktadır.

CNN'ler, görüntü sınıflandırma görevlerinde geniş çapta kullanılmış ve AffectNet veri kümesini kullanarak yüz ifadesi tanıma konusunda da uygulanmıştır. Veri kümesinde değerlendirilen bazı CNN mimarileri arasında VGGNet, ResNet ve Inception yer almaktadır. RNN'ler, özellikle görüntü dizilerindeki zamansal bağımlılıkları yakalamak için yüz ifadesi tanıma görevlerinde kullanılmıştır. Veri kümesinde değerlendirilen bazı RNN mimarileri arasında LSTM ağları ve kapı özyinelemeli geçitler (Gated Recurrent Units - GRU) yer almaktadır. Bu mimarilerin yanı sıra bazı araştırmacılar, CNN ve RNN modellerini birleştirerek yüz ifadesi tanıma görevlerinde hem sınıflandırma hem de zamansal özellikleri yakalamaya çalışmışlardır. AffectNet verisetiden farklı olarak Extended Cohn-Kanade(CK+), FER-2013, MS-Celeb-1M veri setleri de yüz ifade analizi çalışmalarında kullanılmaktadır. 2016 yılında hazırlanan Ms-Celeb-1M veri seti toplamda 10 milyon resim, 100.000 ünlü isim ve onların her birinin yaklaşık 100 fotoğrafını içermektedir. AffectNet veri setinden sonra yaygın kullanılan FER-2013 veri seti yaklaşık 30000 etiketli görüntüden oluşmaktadır. Bu veri setinde yedi duygu sınıfına ait veriler içermektedir ve en yaygın duygu sınıfı mutluluktur [42].

Genel olarak, bu mimarilerin AffectNet veri setindeki performansı kullanılan mimari, veriye uygulanan ön işleme teknikleri ve eğitim parametreleri gibi faktörlere bağlı olarak değişmektedir. Bununla birlikte, bu mimarilerin birçoğu veri kümesinde en iyi (state-of-the-art) performansı elde etmiş ve yüz ifadesi tanıma alanını ilerletmeye yardımcı olmuştur.

3.3 Hiperparametre Seçiminin İncelenmesi

Gerekli hesaplama kaynakları göz önüne alındığında büyük öneme sahip hiper parametreler, hiperparametre optimizasyonu sürecinde öncelikli olarak tercih edilmektedir. Eğitim sırasında ağırlıklar üzerinde daha güçlü etkisi olan hiperparametreler sinir ağı eğitimi için önemlidir [15].

3.3.1 Bayes Yöntemi

Bayes yöntemleri çıkarım, tahmin ve karar verme için kullanılan istatistiksel tekniklerin bir kümesidir. Bu yöntemler model hakkındaki düşüncelerimizi güncellemenin bir yolunu sağlayan Bayes teoremine dayanmaktadır [30]. Bayes yöntemleri, olasılıksal tahminler sağlama ve belirsizlikle başa çıkma yetenekleri nedeniyle, özellikle son yıllarda makine öğrenimi ve veri bilimi alanlarında giderek daha popüler hale gelmiştir.

Bayes yöntemlerin temel fikri, belirsizliği modellemek için olasılık teorisini kullanmaktır. Bu çerçevede, olaylara veya hipotezlere olasılıklar atanır ve bu olasılıklar, yeni veriler ortaya çıktıkça güncellenir. Bu güncelleme işlemi, Bayes teoremi kullanılarak yapılır.

Bayes yöntemler, parametre tahmini, model seçimi ve tahmin gibi çeşitli uygulamalarda kullanılabilir. Parametre tahmininde, Bayes yöntemleri, parametrelerin sonsal olasılık kullanarak model parametrelerini olasılıksal bir şekilde tahmin etme yolunu sağlar [33]. Model seçiminde Bayes yöntemleri, olasılıkları temel alarak farklı modelleri karşılaştırmak için kullanılabilir. Tahminde ise, Bayes yöntemleri, model parametrelerindeki belirsizliği entegre ederek, gelecekteki bir sonucun olasılıksal tahminini sağlayabilir.

Bayes yöntemlerinin temel avantajlarından biri, belirsizlikleri ele alabilme yeteneğidir. Klasik istatistik yöntemleri, parametrelerin veya tahminlerin nokta tahminlerini

sağlarken, Bayes yöntemleri modeldeki belirsizliği yakalayan bu nicelikler üzerinde bir dağılım sağlar. Bu daha güvenli karar verme imkânı yaratır.

Özetlemek gerekirse, Bayes yöntemler, makine öğrenimi ve veri bilimi alanlarında yaygın olarak kullanılan güçlü bir istatistiksel teknikler kümesidir. Belirsizlikleri modelleme ve önceden bilgiyi modele dahil etme imkânı sağlarlar. Bayes yöntemleri, olasılıksal tahminler sağlayabilme ve belirsizlikleri ele alabilme yetenekleri nedeniyle giderek daha popüler hale gelmiştir, bu da daha güvenli kararlar verme ve yeni verilere daha iyi genelleme yapma imkânı sağlayabilir.

3.4.2 HiperParametre Seçimi

Bu bölümde, önerilen dönüştürücü tabanlı GAN modelinin hiperparametrelerini optimize etme süreci açıklanmaktadır. Hiperparametre optimizasyonu, en uygun konfigürasyonu bulmaya yardımcı olduğundan, modelin mümkün olan en iyi performansını sağlamak için çok önemlidir.

Hesaplama açısından maliyetli olan GridSearch yöntemini kullanmak yerine, farklı hiperparametre kombinasyonlarını sistematik olarak keşfetmek için Bayes optimizasyon yöntemi kullanılmıştır. Her çalışmada (run) oluşturulan görüntülerin kalitesini değerlendirmek için FID değeri kullanılmıştır. FID değeri, gerçek ve oluşturulmuş görüntülerin özellik dağılımları arasındaki benzerliği ölçtüğü için model performansının güvenilir bir göstergesidir. FID değeri ne kadar düşükse, modelin performansı o kadar iyidir.

Çalışmamızda modelimizi 50 farklı hiperparametre konfigürasyonu ile analiz ettikten sonra, en düşük FID değerlerine sahip olan 2 hiperparametre kombinasyonu daha fazla adımda eğitilmek için seçilmiştir. Seçilen bu iki hiperparametre kombinasyonuyla dönüştürücü tabanlı GAN modeli en iyi performansı elde etmek amacı ile toplam 200.000 adımda eğitilmiştir.

Hiperparametre optimizasyonu sonrasında, son eğitim için en iyi parametreli, FID değeri 14,94 olan run-25 ve FID değeri 16,21 olan run-52 isimli iki çalışma seçilmiştir. Teorik olarak, en düşük FID değerine sahip ilk çalışmayı seçmek en iyi modeli elde etmek için yeterli olmalıydı. Ancak, hiperparametre optimizasyonu sırasında modeller sadece 36.000 adımda eğitilmiştir. Sonuç olarak, bir ağın düşük bir FID değerine hızlı bir şekilde ulaşabilmesine rağmen, 36.000 adımdan sonra gelişmeme olasılığı da vardı.

Tablo 3.2 ‘deki Hiperparametre tablosunun görselleştirilmiş versiyonu Şekil 3.5’te gösterilmiştir.

Tablo 3.2 HiperParametre optimizasyonu

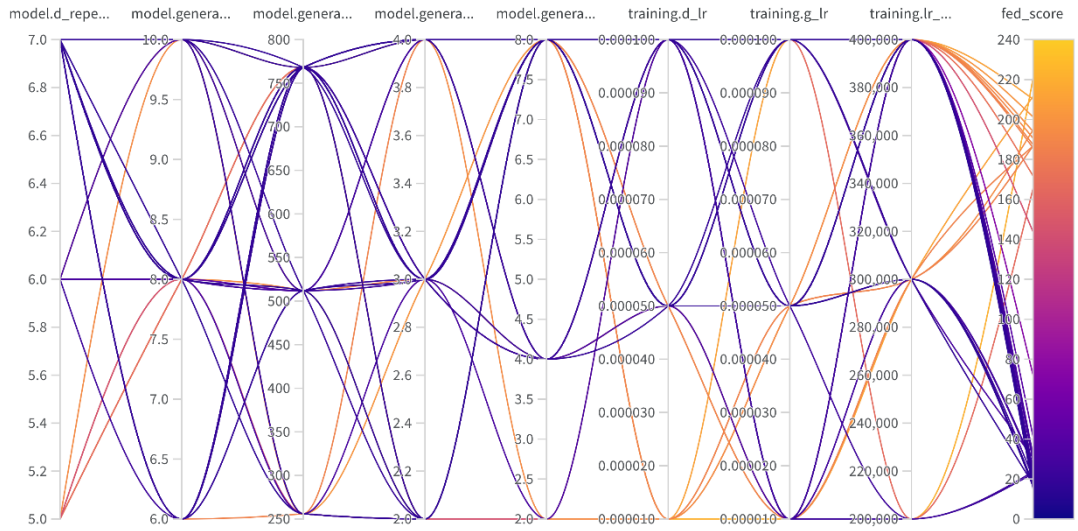
Run ID	Ayrıştırıcının Derinliği	Üreticinin Derinliği	Üretici Gömme boyutu	Üreticinin MLP oranı	Üretici head sayısı (number of heads)	Ayrıştırıcının öğrenme oranı	Üreticinin öğrenme oranı	Öğrenme oranı azalma adımı	FID Değerleri
52	7	8	768	4	8	$5e^{-5}$	$5e^{-5}$	300k	16.21
51	7	6	768	3	8	$1e^{-4}$	$1e^{-5}$	400k	23.69
50	7	6	512	3	8	$5e^{-5}$	$5e^{-5}$	400k	17.58
49	7	8	768	3	8	$1e^{-4}$	$1e^{-5}$	300k	27.62
48	7	8	768	4	4	$1e^{-4}$	$5e^{-5}$	400k	20.41
47	7	8	768	3	8	$1e^{-4}$	$5e^{-5}$	400k	30.71
46	7	6	768	4	8	$5e^{-5}$	$5e^{-5}$	400k	19.13
45	7	6	768	4	8	$5e^{-5}$	$1e^{-5}$	300k	211.57
44	7	8	768	4	8	$1e^{-4}$	$1e^{-4}$	400k	17.58
43	7	6	768	3	8	$1e^{-4}$	$1e^{-5}$	400k	24.31
42	7	8	768	3	8	$1e^{-4}$	$5e^{-5}$	400k	17.24
41	7	6	768	4	8	$5e^{-5}$	$5e^{-5}$	400k	22.34
40	7	8	768	4	8	$5e^{-5}$	$1e^{-5}$	300k	186.72
39	7	10	768	4	8	$5e^{-5}$	$5e^{-5}$	400k	16.89

Tablo 3.2 HiperParametre optimizasyonu (devamı)

38	7	6	512	3	4	$1e^{-4}$	$5e^{-5}$	300k	20.73
37	7	6	768	4	8	$1e^{-4}$	$5e^{-5}$	400k	20.11
36	7	8	768	2	8	$1e^{-5}$	$1e^{-4}$	400k	210.17
35	7	6	768	4	8	$5e^{-5}$	$5e^{-5}$	300k	17.68
34	7	8	768	3	8	$5e^{-5}$	$5e^{-5}$	400k	16.82
33	7	10	768	3	8	$5e^{-5}$	$1e^{-4}$	300k	17.46
32	7	6	768	3	4	$1e^{-4}$	$5e^{-5}$	400k	19.43
31	7	8	768	3	8	$5e^{-5}$	$5e^{-5}$	400k	18.48
30	7	8	512	4	8	$1e^{-4}$	$1e^{-5}$	400k	23.64
29	7	6	768	3	4	$1e^{-4}$	$1e^{-5}$	300k	26.66
28	7	8	768	4	4	$5e^{-5}$	$1e^{-4}$	400k	18.73
27	5	10	768	4	2	$1e^{-5}$	$5e^{-5}$	400k	194.02
26	5	8	256	2	2	$1e^{-4}$	$5e^{-5}$	400k	143.72
25	7	8	512	2	8	$1e^{-4}$	$1e^{-4}$	400k	14.94
24	7	8	768	2	8	$5e^{-5}$	$5e^{-5}$	400k	18.27
23	7	10	256	2	8	$1e^{-4}$	$1e^{-4}$	400k	171.73
22	7	10	256	2	8	$1e^{-4}$	$1e^{-4}$	400k	18.46
21	7	8	768	4	4	$1e^{-4}$	$1e^{-5}$	400k	185.81
20	7	8	512	4	4	$1e^{-4}$	$1e^{-5}$	400k	71.98

Tablo 3.2 HiperParametre optimizasyonu (devamı)

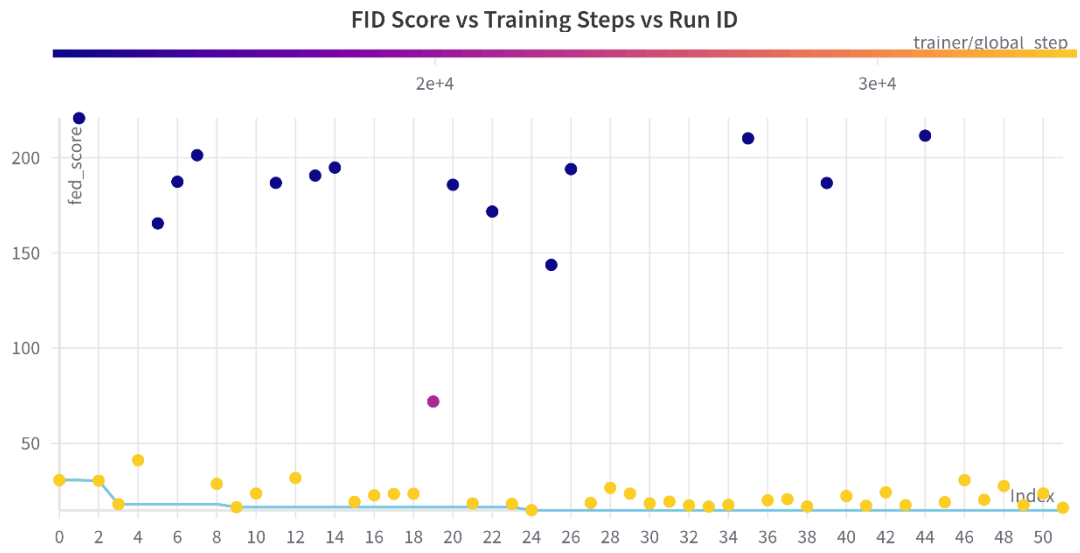
19	7	6	768	4	8	$1e^{-4}$	$1e^{-5}$	400k	23.55
18	7	10	512	3	8	$1e^{-4}$	$1e^{-4}$	400k	23.47
17	7	6	768	3	4	$1e^{-4}$	$1e^{-5}$	200k	22.71
16	7	10	512	2	8	$5e^{-5}$	$1e^{-4}$	400k	19.27
15	5	8	512	2	2	$1e^{-5}$	$5e^{-5}$	300k	194.82
14	5	8	768	2	8	$1e^{-4}$	$1e^{-5}$	400k	190.62
13	6	8	768	4	4	$5e^{-5}$	$1e^{-5}$	400k	31.87
12	7	6	256	4	8	$1e^{-5}$	$5e^{-5}$	300k	186.79
11	6	6	768	3	4	$1e^{-4}$	$5e^{-5}$	200k	23.66
10	7	8	256	2	8	$5e^{-5}$	$1e^{-4}$	400k	16.53
9	7	8	768	3	8	$1e^{-4}$	$1e^{-5}$	400k	28.77
8	6	10	256	3	8	$1e^{-5}$	$5e^{-5}$	400k	201.33
7	7	10	512	3	4	$5e^{-5}$	$1e^{-5}$	400k	187.38
6	5	8	768	3	4	$1e^{-4}$	$1e^{-4}$	200k	165.52
5	6	8	512	4	8	$1e^{-4}$	$1e^{-5}$	400k	41.14
4	6	8	256	3	2	$1e^{-4}$	$5e^{-5}$	400k	30.42
3	7	6	768	2	8	$1e^{-4}$	$5e^{-5}$	400k	18.09
2	5	8	768	2	8	$1e^{-5}$	$1e^{-5}$	200k	220.74
1	6	10	256	2	8	$5e^{-5}$	$1e^{-4}$	300k	30.70



Şekil 3.5 HiperParametrelerin görselleştirilmesi

Paralel Koordinat grafiği, makine öğreniminde hiperparametrelerle model performansı arasındaki ilişkileri anlamak için çok boyutlu bir görselleştirme tekniğidir [44]. Bu grafikte dikey çizgiler hiperparametreleri ve değerlerini simgeler. Geleneksel dağılım grafikleri karmaşık etkileşimleri temsil etmekte başarısız olduğu için, bu görselleştirme yüksek boyutlu hiperparametre alanları için önemlidir [45]. Bu grafikler, araştırmacıların hiperparametre etkilerini daha iyi anlamasını sağlayarak hiperparametre ayarlama sürecini optimize etmelerine yardımcı olur [46].

Şekil 3.5 ve Tablo 3.2 analizi sonucunda, uygulanan arama yöntemiyle parametre kombinasyonlarının kapsamlı bir şekilde araştırıldığını gösteren yoğun bağlantı gözlemlenmektedir. Bu durum, modelin en iyi parametre arama sürecinde hiper parametre alanını detaylı bir şekilde incelediğini göstermektedir. Dahası, bu parametre kombinasyonlarının çoğunluğu, sağ taraftaki grafik segmentine doğru yoğunlaşan bağlantıların gösterdiği gibi (mavi renk yoğunluğu), daha düşük FID değerlerine karşılık gelmektedir. Bu durum, arama sürecinde optimal çözümlere doğru bir yakınsama olduğunu göstermektedir. Sonuç olarak, bu gözlemlerin sentezlenmesiyle, Bayes arama yönteminin parametre çeşitliliğini kapsamlı bir şekilde içerdiği, aynı zamanda optimal yapılandırmaya yakın bölgeleri etkin bir şekilde araştırdığı sonucuna varılabilir.



Şekil 3.6 Modeller ve FID değerleri

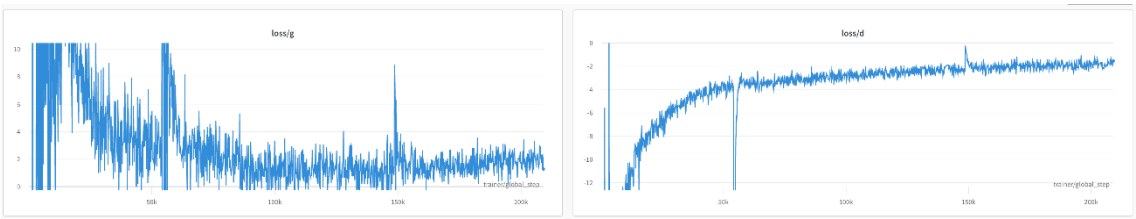
Şekil 3.6 da hiperparametre optimizasyonu sırasındaki model id'leri ve FID değerleri görselleştirilmiştir. Bu işlem sırasında modelin FID değeri 15.000 adımda hala 80 ve üzeri ise çalışmanın durdurulması ve yeni parametrelerle tekrardan başlaması şeklinde ayarlanmıştır. Grafikte ID numarası 14 olan modelin sonuçlarına bakacak olursak FID değeri 190 olarak hesaplanmış ve 15.000 adımda FID değeri belli bir eşik değerinin altında olmadığı için eğitim orada durdurulmuştur. Mor renkle işaretlenmiş noktalar 15.000 adımdaki şartı sağlayamamış eğitim çıktılarıdır.

15.000 adıma geçip FID değeri 80 ve altında olan çalışmalar ise sarı renkle gösterilmiştir. Örneğin, ID numarası 30 olan çalışmada FID değeri 23,64 olarak hesaplanmış ve sarı renkle işaretlenmiştir.

En düşük FID değerine sahip ağın 36.000 adımdan sonra gelişmeme olasılığını düşünerek, ikinci en iyi model (run – 52) de seçilerek 200.000 adımlık bir eğitim ile eğitilmeye devam edilmiştir. Böylece, modellerin daha uzun süre eğitildiğinde FID değerlerdeki değişimi görmemize olanak sağlamıştır. Bu modellerin eğitilmesi ile hiperparametre optimizasyonu sonuçlarının doğrulanması ve soruna çözüm olacak en iyi model yapılandırmasının seçilmesi hedeflenmektedir.

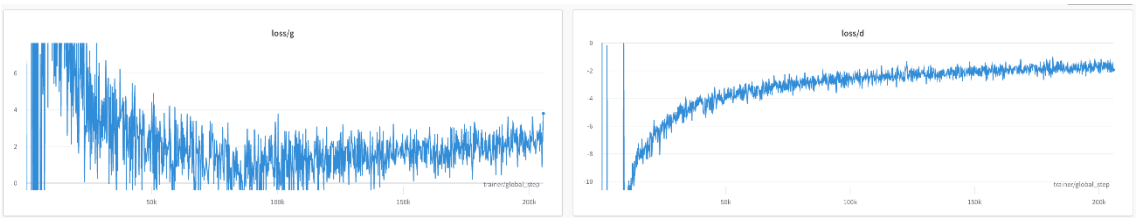
Her iki modelin eğitiminde öğrenme oranı, 10.000 adıma kadar doğrusal şekilde artırılıp, 10.000'den 400.000 adıma kadar ise ters kosinüs fonksiyonu ile her eğitim aşamasında azalacak şekilde ayarlanmıştır. Dönüştürücü tabanlı modellerin öğrenme oranında

warmup olmadan yakınsamadığını ve öğrenme oranı düşüşünü kullanmanın eğitimin sonunda model performansını artırmak için önemli olduğu deneysel olarak gösterilmiştir.



Şekil 3.7 (a) Run_25 Üreticinin kayıp değer eğrisi (b) Run_25 Ayrıştırıcının kayıp değer eğrisi

Grafikler, eğitim süreci ilerledikçe üretici ve ayrıştırıcının nasıl geliştiğini göstermektedir. Şekil 3.7 (a) de üreticinin kayıp değer eğrisi, şekil 3.7(b) ise ayrıştırıcının kayıp değer eğrisi gösterilmiştir. Şekil 3.7(b) ayrıştırıcının kayıp değerlerinin zamanla daha iyi olduğu görülse de, gelişme hızı yavaşlamaktadır. Aynı zamanda, üretecinin performansının önce iyileştiği ve sonra biraz kötüleştiği gözükmektedir ki, bu da normal bir durumdur. Ayrıştırıcının gelişimi önemlidir çünkü ne kadar iyi olursa, üreticiye o kadar iyi geri bildirim sağlayabilir ve onun da gelişmesine yardımcı olur. Üreticide gözlemlediğimiz desen beklenendir, performansı çok iyi ise ayrıştırıcıyı kandırabilir ve üreticinin kayıp değerini düşürür. Ayrıştırıcı geliştikçe, üretecinin de gelişmesine yardımcı olur. Bununla birlikte, üretecinin kayıp değeri kötüleşiyor gibi görünse de, bu ürettiği görüntülerin kötü olduğu anlamına gelmez, çünkü ayrıştırıcı da güçlü hale gelir ve daha zorlu geri bildirimler gönderir.



Şekil 3.8 (a) Run_52 Üreticinin kayıp değer eğrisi

(b) Run_52 Ayrıştırıcının kayıp değer eğrisi

RunId_52 çalışmasında üretecinin ve ayrıştırıcının kayıp eğrileri daha az dalgalanma göstermektedir. Bu dalgalanmanın azalması, daha istikrarlı bir eğitim sürecinin işareti olarak kabul edilir [16]. Eğitimdeki istikrar, sağlam ve güvenilir bir model performansı için önemli bir faktördür [47]. Ayrıca, kayıp eğrilerindeki hafif dalgalanmalar, eğitim sürecinin güvenilir bir şekilde yakınsadığını gösterir ve model optimizasyonu için önemli

bir özelliktir [11]. Dolayısıyla, şekil 3.8‘de Run_52 parametre seçiminin kararlı ve güvenilir bir yakınsama elde ettiği tespit edilmiştir.

Her iki modeli eğittikten sonra, en iyi model kaydedilmiş ve FID değerleri karşılaştırılmıştır. Başlangıçtaki düşüncelerimiz, RunID-25'in FID değerinin 9.7 olması ve RunID-52'nin 9.1 ile daha düşük FID değeri elde etmesiyle doğrulandı.

Grafiklerde gösterildiği gibi RunID-25'in kayıp eğrilerinde birkaç dalgalanma olsa da, RunID-52'de dalgalanma bulunmamaktadır, bu da daha istikrarlı bir eğitim sürecinin olduğunu gösterir (Şekil 3.8). GAN modelleri genelinde, kayıp eğrilerinde dalgalanmaların olması sorun yaratır, çünkü dengedeki üretici ve ayrıştırıcı arasında salınımlara işaret edebilir. Bu sorunlar daha düşük kaliteli görüntülerin üretilmesine yol açabilir, bu nedenle RunId-52'deki gibi istikrarlı bir eğitim sürecine sahip bir model seçmek önemlidir.

4
DENEY SONUÇLARI VE DEĞERLENDİRME

4.1 GanMut Modelinin Eğitilmesi

Bu bölümde, ilgili bir çalışma modeli olan GANmut modelinin seçilen veri kümesi üzerindeki eğitim süreci tartışılmaktadır. GANmut uygulamasına https://github.com/stefanodapolito/GANmut adresinden erişilebilir. Önerilen model de olduğu gibi, GANmut modeli de 200.000 adımda eğitilmiştir. Tez kapsamında kullanılan veri kümesi, orijinal veri kümesinin bir alt kümesi olup ve GANmut geliştiricileri tarafından kullanılan veri kümesinden farklı olduğundan, modelin kendi veri kümemizde yeniden eğitilmesi gerekmiştir.

Veri kümesi üzerinde modelin eğitilmesi için GANmut makalesinde bildirilen hiper parametreler kullanılmıştır. Eğitim sürecinde, projenin kaynak kodundaki yollar ayarlanmış ve eğitim sürecini takip etmek için W&B görselleştirme ürünü eklenmiştir. Orijinal uygulamada başka herhangi bir değişiklik yapılmamıştır.

GANmut, tez kapsamında kullanılan veri kümesinde 200.000 adımda eğitildikten sonra, model 9.7 FID değerini elde etmiştir. Bu sonuç, optimizasyon adımından seçilen hiper parametrelerle eğitilen modellerin değerlerine (FID 9.7 ile RunID-25 ve FID 9.1 ile RunID-52) benzemektedir.

Çalışmanın ardından GANmut'un, dönüştürücü tabanlı modelimize kıyasla daha hızlı bir şekilde yakınsadığı tespit edilmiştir. Bu, Dönüştürücü modellerinin eğitiminin genellikle CNN tabanlı modellere göre daha yavaş olması nedeniyle beklenen bir durumdur. Bu farkın nedeni, dönüştürücülerin her katman da kendine dikkat mekanizmaları için matris çarpımlarına ihtiyaç duymaları nedeniyle daha yüksek bir hesaplama işlemlerine sahip olmalarıdır [17]. Bunun aksine, CNN'ler, hesaplama açısından daha az maliyetli olan evrişimler aracılığıyla yerel işlemler gerçekleştirir [18]. GANmut gibi CNN tabanlı modeller, dönüştürücü tabanlı modellere göre daha hızlı yakınsama eğilimindedirler.

GANmut'un daha hızlı yakınsaması bir avantaj olmasına rağmen, daha bilinçli bir karar vermek için modellerin genel performansı, istikrarı açısından FID değerlerinin ve kayıp eğrilerinin analiz edilmesi önemlidir.

4.2 Modellerin Karşılaştırılması

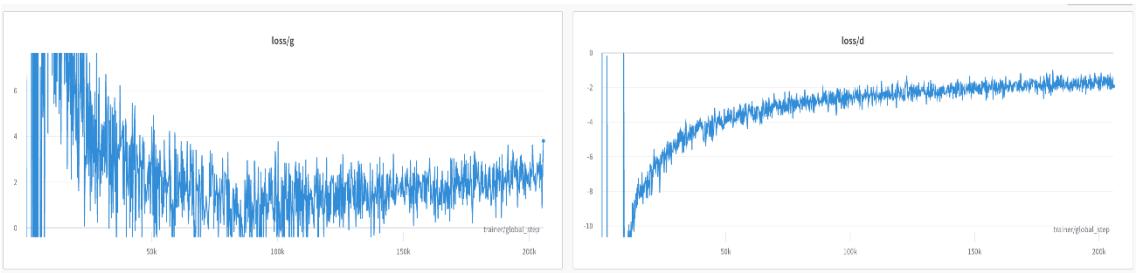
Bu bölümde, CNN'lere dayalı GANmut modeli ile dönüştürücü tabanlı GAN modelimizin (run-52) performansı FID değerleri ve model kayıp eğrileri (loss curve) metrikleriyle karşılaştırılmaktadır. Her iki model 200.000 adımda AffectNet veri seti ile eğitilmiştir.

FID Değeri: Dönüştürücü tabanlı GAN modelimizin (run-52) FID değeri 9.1 iken, GANmut modeli 9.7 FID değeri elde etmiştir. Sonuç olarak, modelimizin FID değeri GANmut'un FID değerinden daha düşüktür, bu da modelimizin ilişkili çalışma modeline göre daha yüksek bir görüntü kalitesinde performans sergilediğini göstermektedir. Tablo 4.1'de modeller karşılaştırılmıştır.

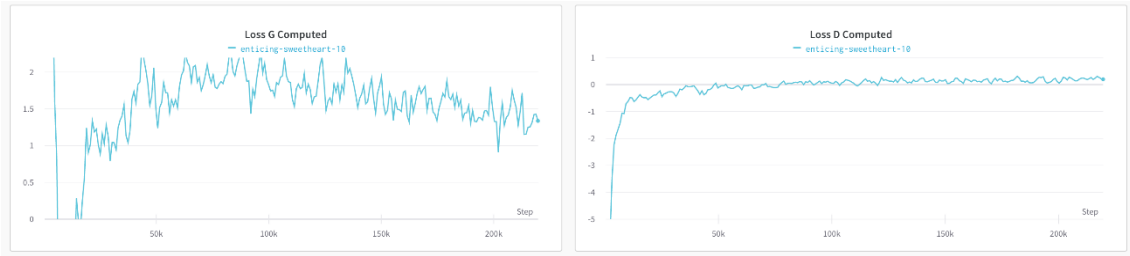
Tablo 4.1 Modellerin karşılaştırılması

	Önerilen Model	GANMUT
Üretici	Vision Tranformer	CNN
Üreticinin Öğrenme Oranı	5e^(-5)	1e^(-4)
Ayrıştırıcı	CNN	CNN
Ayrıştırıcının Öğrenme Oranı	5e^(-5)	1e^(-4)
Öğrenme Adımı	200.000	200.000
Optimizer Türü	Adam	Adam
Kullanılan Veriseti	AffectNet	AffectNet
Eğitim Verisi	230119	230119
Doğrulama veri seti	28762	28762
Test veri seti	28770	28770
FID Değeri	9.1	9.7

Kayıp Eğrisi: Dönüştürücü tabanlı GAN modelimizin ve GANmut modelinin kayıp eğrileri modellerin eğitim istikrarı ve yakınsaması hakkında bilgi almamızı sağlar. Şekil 4.1 de Dönüştürücü tabanlı GAN modelimizin (run-52) dalgalanmalar olmadan düzgün bir kayıp eğrisine sahip olduğu gösterilmiştir, bu da istikrarlı bir eğitim süreci olduğunu göstermektedir. GANmut modeli, dönüştürücü tabanlı Gan modeline göre daha hızlı bir yakınsama sergilemiştir (Şekil 4.2), bu beklenen bir durumdur [17][18]. Ancak, daha hızlı yakınsama, FID değerleri ile kanıtlandığı gibi daha iyi görüntü kalitesinde performans sergilediğini göstermez.



Şekil 4.1 RunID -52 kayıp eğrisi



Şekil 4.2 GANmut kayıp eğrisi

GANmut'un daha hızlı bir şekilde yakınsaması bir avantaj olsa da, karar vermek için modellerin FID değerleri ve kayıp eğrileri açısından genel performansını ve istikrarını dikkate almak önemlidir. Dönüştürücü tabanlı GAN modelimiz, GANmut modelinin 9.7 FID değerine kıyasla 9.1 FID değeri ile daha iyi performans göstermiştir. Ayrıca, modelimizin kayıp eğrisi düzgün olup, herhangi bir sıçrama göstermemiştir, bu da istikrarlı bir eğitim süreci olduğunu göstermektedir.

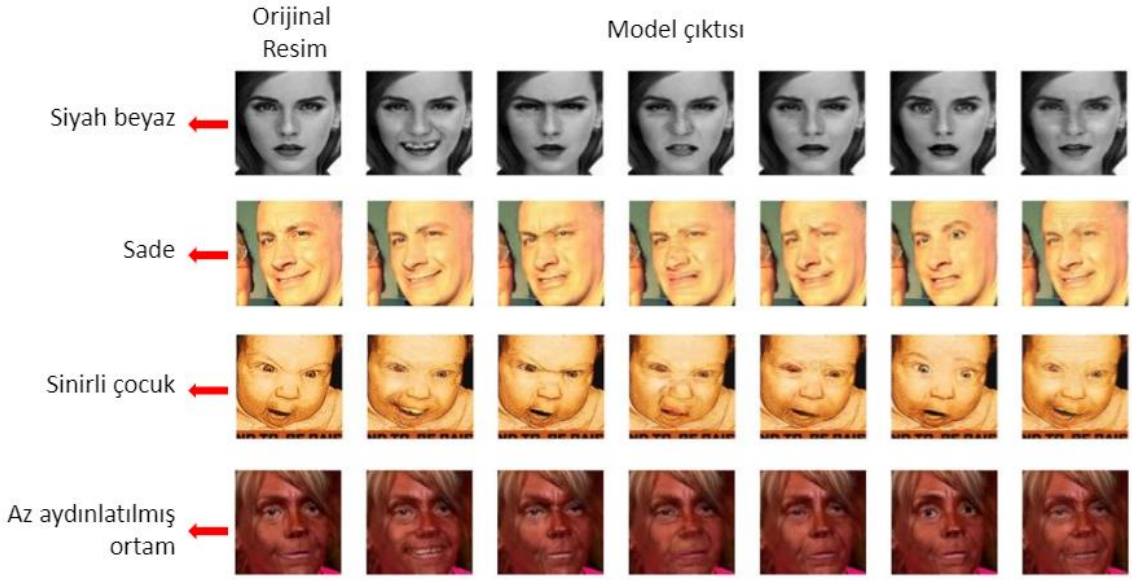
Sonuç olarak, görüntü kalitesi ve eğitim istikrarı öncelikli olan uygulamalar için Dönüştürücü tabanlı GAN modelimiz, CNN tabanlı modelden biraz daha yavaş yakınsamasına rağmen daha iyi performans göstermektedir. Bu tür uygulamalarda GANmut modeline göre daha uygun bir seçim olduğu gözlemlenmiştir.

4.3 Dönüştürücü Tabanlı Üreticinin Görsel Analizi

Önceki bölümlerde bahsedilen sayısal sonuçların yanı sıra, Koşullu GAN'ların performansını değerlendirirken nitel sonuçları dikkate almak da önemlidir. FID değerleri gibi sayısal ölçütler modelin performansının genel bir değerlendirmesini sağlayabilir, ancak farklı koşullar altında üretilen görüntülerin inceliklerini tam olarak yakalamayacağı için görsel analiz yapmak önemlidir. Bu bölümde, dönüştürücü tabanlı GAN modelinin nitel sonuçları gösterilmiş, giriş parametreleri ve belirtilen çeşitli koşullar ile duygu sınıfı ve duygu yoğunluğuna ait görüntü üretme yeteneği vurgulanmıştır.

4.3.1 Değişen Duygu Sınıfı

Şekil 4.3'teki görseller yüksek yoğunluğu ifade eden 0.95 sabit duygu yoğunluğu uygulanarak üretilmiştir. Geliştirilen dönüştürücü tabanlı modelimiz, farklı duygu türlerine uyum sağlama ve belirtilen duyguyu doğru bir şekilde yansıtan görüntüler üretme yeteneğini sergilemiştir.

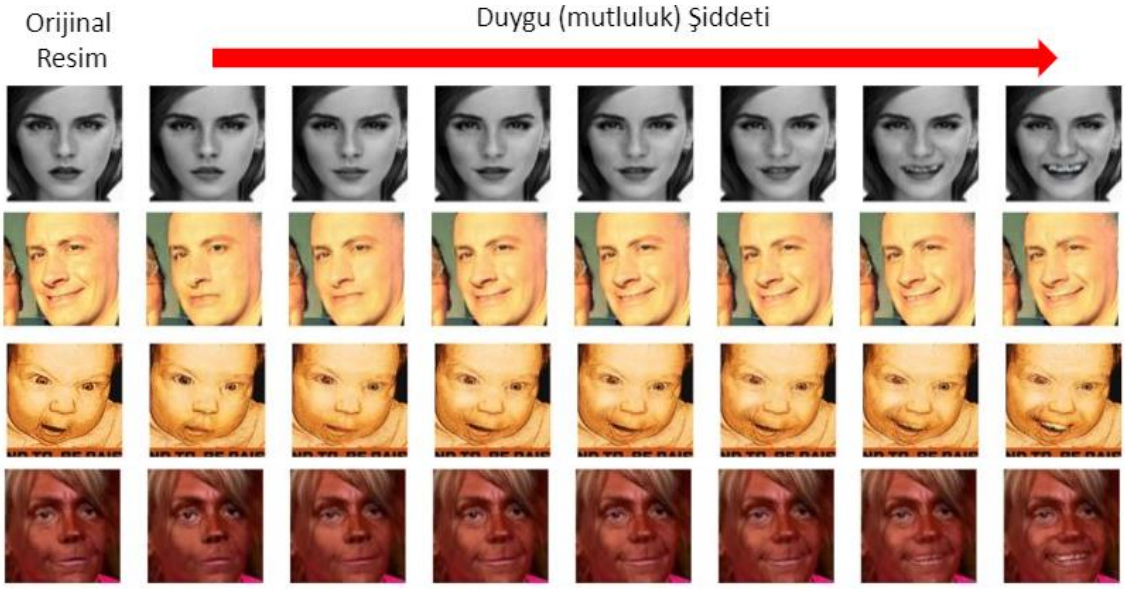


Şekil 4.3 Modelimizde üreticinin ürettiği görüntüler

Şekil 4.3'te en sol kısımda üreticiye verilen görseller, sağ tarafta ise üreticinin ürettiği görseller gösterilmiştir. Verilen fotoğrafın sınıfından bağımsız, modelin her zaman istenilen sınıfa ait fotoğraf ürettiği gözükmektedir. Örneğin, ilk çıktı tüm örnekler için mutlu, ikinci görsel sinirli vb. Buradaki önemli husus görsellerin farklı şartlı ortamlarda olmasına rağmen üreticinin başarıyla yedi duygu kategorisine ait görüntü üretmesidir. Dönüştürücü tabanlı GAN modelimiz, siyah-beyaz görüntüler, zayıf ışık koşulları ve çocuk içeren görüntüler gibi zorlu koşullar altında bile güçlü bir performans sergilemiştir. Bu ortamlarda olan görüntülerin az olmasına rağmen model hepsini işleyebilmektedir ki, bu da modelin genelleme kapasitesini gösterir. Bu özellik, çeşitli duygusal ifadelerle görüntülerin sentezini gerektiren uygulamalar için önemlidir.

4.3.2 Değişen Duygu Yoğunluğu

Bu analizde, duygu tipi sabit tutularak duygu yoğunluğu 0 (düşük yoğunluk) ile 1 (yüksek yoğunluk) arasında değiştirilmiştir. Modelimiz, duygusal yoğunluğun farklı derecelerine sahip görüntüler üretmeyi başararak, üretilen görüntülerde duygunun gücünü kontrol etme yeteneğini sergilemiştir.



Şekil 4.4 Mutluluk yoğunluk artışı

Şekil 4.4'te yoğunluğa bağlı görsellerdeki duygu değişimi gösterilmektedir. Üreticinin girdi olarak aldığı görseller üzerinde yoğunluk artırılarak çıktılar görselleştirilmiştir. Örneğin, ilk fotoğraftaki kadın nötral yüz ifadesine sahipken yoğunluk arttıkça mutluluk oranı da artmaktadır. Girdi görüntüsündeki yüz ifadesinden, duygu sınıfından bağımsız olarak, modelimizin tüm fotoğrafları başarıyla dönüştürebildiğini gözlemlenmektedir. Bu esneklik, sentezlenen görüntülerin duygusal içeriği üzerinde ince ayar yapmayı gerektiren uygulamalar için önemlidir.

5
SONUÇ VE ÖNERİLER

Bu tez çalışmasında geliştirilen model; farklı duygu türlerine ve yoğunluklarına uyum sağlayabilme becerisine sahip olmakla beraber, zorlu koşullar altında başarılı performans göstermiş, aynı zamanda çeşitli ve karmaşık duygusal içeriğe sahip çalışmalar için uygunluk sağlamıştır. Nitel bulgular nicel sonuçlar ile birleştirilerek, geliştirilen dönüştürücü tabanlı modelin, ilgili çalışma modeli GANmut'a göre üstünlüğü tespit edilmiştir.

Dönüştürücü tabanlı üreticinin performansını değerlendirmek için Fréchet Başlangıç Mesafesi değeri kullanılmıştır. Daha düşük FID değeri daha iyi performansı göstermektedir. Kapsamlı hiper parametre optimizasyonu sonrasında, 9.1 FID değeri elde edilmiş ve bu önceki mimarinin 9.7 değerine kıyasla üstün performans göstermiştir. Bu sonuçlar, tez kapsamında geliştirilen dönüştürücü tabanlı modelin istenen duygu değişikliklerini doğru bir şekilde yansıtan yüksek kaliteli görüntüler üretme yeteneğini göstermektedir. Ayrıca, geliştirilen dönüştürücü tabanlı model, istenen duygu tipi ve yoğunluğuna bağlı iki parametre temelinde girdi görüntülerini başarılı bir şekilde değiştirebilme yeteneği sergilemiştir. Dönüştürücü tabanlı üretici model, farklı duygu tiplerine ve yoğunluklarına uyum sağlamış, insan duygularının karmaşıklığını ve çeşitliliğini yakalama konusunda sağlamlığını ve esnekliğini göstermiştir.

Sonuçlar, dönüştürücü tabanlı bir üreticinin insan duygu değişimi için etkili olduğunu göstermektedir. Nicel ve nitel sonuçlar, belirtilen duygu tipine göre modelin üstün performansını ve esnekliğini doğru bir şekilde sergilemektedir. Tez kapsamında yapılan çalışmalar, bilgisayarlı görü alanlarında dönüştürücü mimarileri üzerine literatüre katkı sağlayarak, duygu yoğunluğu ve değişimiyle ilgili çeşitli alanlarda dönüştürücü tabanlı modellerin uygulanmasını keşfetmek için yeni olanaklar sunacaktır.

Çalışmamızın odak noktası olarak GAN'ların üretici bileşeni ele alınmıştır. Ayrıştırıcı bileşeninin potansiyel değişikliğini konusunda yapılacak araştırmalar, gelecek çalışmaların konusunu oluşturmaktadır. Bu çalışmanın gelecek adımlarında duygu tespiti amacıyla uçtan uca dönüştürücü tabanlı GAN modeli geliştirilebilir. Bunun yanı sıra, veri seti sayısı ve yapılan analizlerin kapsam ve çeşitliliği de gelecek çalışmalarda

arttırılabilir. Dönüştürücü tabanlı GAN modeli, duygu tespiti için geliştirilen birçok modelin performansları ve sonuçları ile karşılaştırılabilir. Gelecekte yeni dönüştürücü tabanlı GAN modelinin, video kayıtlar üzerindeki başarısı gözlemlenebilir. Tez kapsamında elde edilen sonuçlar, dönüştürücü mimarileri kullanarak duygu değişikliğini tespit etmek için başarılı bir temel sağlamakta, aynı zamanda bilgisayarlı görü ve GAN mimarisine önemli ölçüde katkıda bulunmaktadır.

KAYNAKÇA

[1] J. Brownlee, “Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions”. Machine Learning Mastery, pp. 3–14, 2018.

[2] M. A. Nielsen, "Neural Networks and Deep Learning," Determination Press, San Francisco, CA, vol. 25, pp. 15-24, 2015.

[3] A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem. A correction" Proceedings of the London, pp. 10–340, 1938.

[4] K. Zhang, W. Zuo, Y. Chen, D. Meng and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," in IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 3142-3155, 2017.

[5] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training", pp. 3–8, 2018.

[6] Z. Lian, B. Liu, and J. Tao, "CTNet: Conversational transformer network for emotion recognition," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 985-1000, 2021.

[7] V. Chudasama, P. Kar, A. Gudmalwar, N. Shah, P. Wasnik, and N. Onoe, "M2FNet: Multi-modal fusion network for emotion recognition in conversation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4652-4661, 2022.

[8] A. L. Beam, "Deep learning 101-part 1: History and background," Accessed, vol. 3, 2018.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation" California Univ San Diego La Jolla Inst for Cognitive Science, pp. 8-31, 1985.

[10] S. Hochreiter, and J. Schmidhuber, "Long short-term memory" in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[11] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning" in SIAM Review, vol. 60, no. 2, pp. 223-311, 2018.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner et al., "An image is worth 16x16 words: Transformers for image recognition at scale" arXiv preprint arXiv:2010.11929, 2020.

[13] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles and H. Jégou, "Training data-efficient image transformers & distillation through attention" in International Conference on Machine Learning, pp. 10347-10357, 2021.

[14] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium" in Advances in Neural Information Processing Systems, vol. 30, pp. 1-38 , 2017.

[15] A. Ng, "Improving deep neural networks: Hyperparameter tuning, regularization and optimization" Deeplearning.ai on Coursera, 2017.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, and Y. Bengio, "Generative adversarial nets" in Advances in Neural Information Processing Systems, pp. 2672-2680, 2014.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez et al., "Attention is all you need" in Advances in Neural Information Processing Systems, pp. 5998-6008, 2017.

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haner, "Gradient-Based Learning Applied to Document Recognition" pp. 1-44, 1998.

[19] K. O'Shea and R. Nash, "An introduction to convolutional neural networks" arXiv preprint arXiv:1511.08458, pp. 1-9, 2015.

[20] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild" in IEEE Transactions on Affective Computing, vol. 10, no. 1, pp. 18-31, 2017.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition" arXiv preprint arXiv:1409.1556, pp. 1-14, 2014.

[22] W. Chen, X. Du, F. Yang, L. Beyer, X. Zhai, T. Y. Lin et al., "A Simple Single-Scale Vision Transformer for Object Detection and Instance Segmentation," in Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X, pp. 711-727. Cham: Springer Nature Switzerland, 2022.

[23] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding" arXiv preprint arXiv:1810.04805, pp. 4171-4186,2018.

[24] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention" in International Conference on Machine Learning, pp. 10347-10357, 2021.

[25] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate" arXiv preprint arXiv:1409.0473, pp. 1- 15,2014.

[26] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation" arXiv preprint arXiv:1508.04025, pp. 1-9, 2015.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space" arXiv preprint arXiv:1301.3781, pp. 1-10, 2013.

[28] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks" in International Conference on Machine Learning, pp. 1310-1318, 2013.

[29] A. Arseniev-Koehler, "Theoretical foundations and limits of word embeddings: what types of meaning can they capture?" in Sociological Methods & Research, 00491241221140142, 2021.

[30] J. O. Berger, L. R. Pericchi, J. K. Ghosh, T. Samanta, F. De Santis, J. O. Berger, and L. R. Pericchi, "Objective Bayesian methods for model selection: Introduction and comparison" in Lecture Notes-Monograph Series, pp. 135-207, 2001.

[31] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012-10022, 2021.

[32] B. Zhang, Z. Tian, Q. Tang, X. Chu, X. Wei, and C. Shen, "SegViT: Semantic segmentation with plain vision transformers" in Advances in Neural Information Processing Systems, vol. 35, pp. 4971-4982, 2022.

[33] M. D. Richard, and R. P. Lippmann, "Neural network classifiers estimate Bayesian a posteriori probabilities" in Neural Computation, vol. 3, no. 4, pp. 461-483, 1991.

[34] C. Baskin, N. Liss, E. Zheltonozhskii, A. M. Bronstein, and A. Mendelson, "Streaming architecture for large-scale quantized neural networks on an FPGA-based dataflow platform" in 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 162-169, 2018.

[35] Y. Zheng, C. Yang, and A. Merkulov, "Breast cancer screening using convolutional neural network and follow-up digital mammography" in Computational Imaging III, Vol. 10669, p. 1066905, SPIE, May 2018.

[36] F. Han, H. Zhang, S. Chatterjee, Q. Guo, and S. Wan, "A modified generative adversarial nets integrated with stochastic approach for realizing super-resolution reservoir simulation" in IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 2, pp. 1325-1336, 2019.

[37] L. Picek, M. Šulc, J. Matas, J. Heilmann-Clausen, T. S. Jeppesen, and E. Lind, "Automatic fungi recognition: Deep learning meets mycology" in Sensors, vol. 22, no. 2, p. 633, 2022.

[38] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation" in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543, Oct. 2014.

[39] K. Slechten, "A deep dive into transformers" pp. 3-6, 2023.

[40] S. d'Apolito, D. P. Paudel, Z. Huang, A. Romero, and L. Van Gool, "GANmut: Learning interpretable conditional space for gamut of emotions" in Proceedings of the

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 568-577, 2021.

[41] M. Mirza and S. Osindero, "Conditional generative adversarial nets" arXiv preprint arXiv:1411.1784, pp. 1-6, 2014.

[42] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner et al., "Challenges in representation learning: A report on three machine learning contests" in Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III 20, pp. 117-124, Springer, Berlin, Heidelberg, 2013.

[43] G. Coppersmith, R. Leary, P. Crutchley, and A. Fine, "Natural language processing of social media as screening for suicide risk" in Biomedical Informatics Insights, vol. 10, 1178222618792860, pp. 1-11,2018.

[44] A. Inselberg and B. Dimsdale, "Parallel coordinates: a tool for visualizing multi-dimensional geometry" in Proceedings of the 1st Conference on Visualization'90, pp. 361-378, IEEE Computer Society Press, 1990.

[45] Y. H. Fua, M. O. Ward, and E. A. Rundensteiner, "Hierarchical parallel coordinates for exploration of large datasets" in Proceedings of the Conference on Visualization'99: Celebrating Ten Years, pp. 43-50, IEEE Computer Society Press, 1999.

[46] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization" Journal of Machine Learning Research, vol. 13, Feb, pp. 281-305, 2012.
[47] O. Bousquet, S. Boucheron, and G. Lugosi, "Stability and generalization" Journal of Machine Learning Research, vol. 3, Mar, pp. 499-526, 2002.

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirileri

1. N.Yagubbayli, M.Bal “Transformer Bazlı GAN ile Duygu Analizi” , INSAC International Researches Congress on Natural and Engineering Sciences (INSAC-IRNES-2023), 18-19 March 2023.