KOCAELİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

LİSANS TEZİ

VİSİON TRANSFORMER MODEL İLE AFLATOKSİN TESPİTİ

MERT KARA

180202038

KOCAELİ 2022

KOCAELİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BİTİRME PROJESİ

VİSİON TRANSFORMER MODEL İLE AFLOTOKSİN TESPİTİ

MERT KARA

Dr. Öğretim Üyesi Burak İNNER
Danışman, Kocaeli Üniv. .....

Prof.Dr. Yaşar BECERİKLİ
Jüri Üyesi, Kocaeli Üniv. .....

Prof.Dr. Kerem KÜÇÜK
Jüri Üyesi, Kocaeli Üniv. .....

Tezin Savunulduğu Tarih: 01.06.2022

ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışması, transformer model ile aflatoksin tespiti yapma amacıyla gerçekleştirilmiştir.

Tez çalışmamda desteğini esirgemeyen, çalışmalarıma yön veren, bana güvenen ve yüreklendiren danışmanım Dr. Öğretim Üyesi Burak İnner hocama sonsuz teşekkürlerimi sunarım.

Hayatım boyunca bana güç veren en büyük destekçilerim, her aşamada sıkıntılarımı ve mutluluklarımı paylaşan sevgili aileme teşekkürlerimi sunarım.

Haziran – 2022 Mert KARA

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR 3
İÇİNDEKİLER 4
ŞEKİLLER DİZİNİ 5
ÖZET 6
ABSTRACT 7
GİRİŞ 8
1. VERİ SETİ 11
1.1. Veri Setinin Etiketlenmesi 12
2. TRANSFORMER MODELLERDE TEMEL KAVRAMLAR 13
2.1. Transformer ve Attention kavramı 13
2.1.1 Self attention 15
2.1.2. Multi head self attention 18
2.2. Gaussian Error Linear Unit (GELU) 19
2.3. Group Normalization (Grup normalleştirmesi) 19
3. VİSİON TRANSFORMER 21
3.1 Parça Üretimi 22
3.2 Konumsal Gömme 23
3.2.1. Inductive bias 23
3.2.2. Class token 24
3.3 Transformer Kodlayıcı Blok 24
3.3.1. Çok katmanlı algılayıcı (MLP) 24
3.3.2. Normalization 24
4. YÖNTEM VE BULGULAR 25
5. SONUÇLAR 29
KAYNAKLAR 30

ŞEKİLLER DİZİNİ

Şekil 1.1 Görüntü elde etmek için kullanılan optik sistem……11
Şekil 1.2. Beyaz ışık altında örnek incir görüntüleri ve UV ışığı ile aydınlatılmış iki örnek, BGYF için bir pozitif ve bir negatif.....12
Şekil 2.1 Transformer Mimarisi 14
Şekil 2.2. Multi-Head Attention 15
Şekil 2.3. Giridi dizilerinin bağlam vektörüne dönüştürülmesi 15
Şekil 2.4. Eğitimde güncellenecek olan Anahtar, Sorgu ve Değer ağırlıkları 16
Şekil 2.5. Attention konseptinde nokta çarpım (Dot-Product) i'inci sorgu ile dizideki tüm Anahtarların nokta çarpımı …….17
Şekil 2.6. Attention bloklarının çıktısı olan bağlam vektörleri (C'ler) elde etmek……...…18
Şekil 2.7. GELU 19
Şekil 2.8. Parti boyutlarına göre ImageNet sınıflandırma hataları. 20
Şekil 3.1. Vision Transformer mimarisi.....21
Şekil 3.2. Parçalanmış görüntü……...24
Şekil 4.1. Yapılan ilk deneylerden bir tanesi…….25
Şekil 4.2. veriyi 32*32*3 piksel olarak aldığımız bir deney……..26
Şekil 4.3. veriyi 64*64*3 piksel olarak aldığımız bir deney……..26
Şekil 4.4. veriyi 128*128*3 piksel olarak aldığımız bir deney. 27
Şekil 4.5 Elde edilen en iyi accuracy değeri……...27
Şekil 4.6. Elde edilen en iyi loss değeri……..27
Şekil 4.7. VGG16 Mimarisi……28

VİSİON TRANSFORMER MODEL İLE AFLATOKSİN TESPİTİ

ÖZET

Bu çalışmanın amacı, daha önceden doğal dil işleme de kendini çokça kez kanıtlamış Transformer modellerin görüntü verileri üzerinden aflatoksin tespiti için evrişimli sinir ağları (CNN) kadar başarılı olabileceğini göstermektir.

Öncelikle, proje içerisinde kullanacağımız veri setini belirledik. Bu veri seti BGYF ışıması altındaki 200 kuru incir görüntüsünü içeren bir veri setidir. Veri setindeki başarıyı olumsuz olarak etkileyebileceği düşünülen tamamıyla siyah olan kısımlar kesilerek çıkarıldı.

Resimler n*n piksellik parçalara bölünerek doğal dil işleme de bir kelime ne ise burada da bu parçalara o şekilde davranılmıştır. Bu parçalar modele verilirken modelin hangi parçanın resmin hangi bölümüne ait olduğunun farkında olması için Pozison gömme (Position Embedding) işlemi uygulanmıştır.

Bu şekilde işlem uygulanan veriler Transformer Encoder olarak ifade edilen ve Multi-Head Attention ile Çok Katmanlı Algılayıcı (Multilayer perceptron) içeren kısıma verilmiştir. Bu kısımda öncelikle Multi-Head Attention ile görüntü içerisindeki bölümlerin birbiri ile olan yakınlıkları çıkartılıyor daha sonrasında ise Çok Katmanlı Algılayıcı kısmında klasik sınıflandırma işlemi uygulanıyor. Karşılaştırma yapmak için Transformer Learning (TL) yöntemi ile birçok state of art CNN modelini'de veri setimiz üzerinde eğitip tahminlerini kaydettik.

Deneyler esnasında TL yöntemi ile Inception, VGG, ResNet, EfficentNet, DenseNet dahil farklı mimarilerini içeren 25 farklı model 20 epoch olacak şekilde çalıştırldı. Daha sonra elde edilen en başarılı model üzerinde optimizer, learning rate gibi hiper parametrelerin optimum halleri belirlenmeye çalışıldı. Son olarak da modelin mimarisinde son katmana birkaç dense layer eklenerek başarısı gözlemlendi. VİT modelleri ile alakalı yapılan deneylerde ise patch size, learning rate, dropout değerleri ve modelin dense katmanlarındaki nöron sayıları üzerinde farklı değerler kullanılarak deneyler yapıldı.

Yapılan deneyler sonucunda en yüksek Doğruluk (Accuracy) değeri incelendiğinde önerilen VİT modelinin TL modellerine kıyasla daha iyi bir sonuç elde ettiğini gözlemledik ancak VİT modeli üzerinde deneyler tekrarlandığında bu başarının tekrar elde edilemediğini fark ettik. Bunun sebebinin modelin ağırlıklarının rastgele atanmasından kaynaklandığını düşünüyoruz. Buna önlem olarak modeli kaydedip istediğimiz zaman kaydedilen modeli yükleyip tahminleme yapmaya karar verdik.

Anahtar kelimeler: Vision Transformer, Transfer öğrenme, GELU, Dikkat

AFlATOKSİN DEDECTİON WİTH TRANSFORMER MODEL

ABSTRACT

The purpose of this study is to show that Transformer models, which have proven themselves many times in natural language processing, can be as successful as convolutional neural networks for aflatoxin detection on image data.

First, we determined the data set that we will use in the project. This dataset is a dataset containing 200 dried fig images under BGYF irradiation. The completely black parts, which were thought to adversely affect the success in the data set, were cut out.

By dividing the pictures into n*n pixel parts, these parts are treated in the same way as a word is in natural language processing. While these parts were given to the model, the position embedding process was applied to make the model aware of which part belongs to which part of the picture.

The data processed in this way was given to the part called Transformer Encoder and containing Multi-Head Attention and Multilayer perceptron. In this part, first, the proximity of the parts in the image is extracted with Multi-Head Attention, and then the classical classification process is applied in the Multi-Layer Perceptron part. For comparison, we trained many state of art CNN models on our dataset with the Transformer Learning (TL) method and saved their predictions.

During the experiments, 25 different models including different architectures including Inception, VGG, ResNet, EfficentNet, DenseNet were run with the TL method at 20 epochs. Then, on the most successful model obtained, the optimum states of hyper parameters such as optimizer and learning rate were tried to be determined. Finally, the success of the model was observed by adding a few dense layers to the last layer in the architecture of the model. In experiments related to VIT models, experiments were carried out using different values on patch size, learning rate, dropout values and the number of neurons in the dense layers of the model.

As a result of the experiments, when the highest Accuracy value was examined, we observed that the proposed VIT model achieved a better result compared to the TL models, but when the experiments were repeated on the VIT model, we realized that this success could not be achieved again. We think that this is due to the random assignment of the weights of the model. As a precaution, we decided to save the model and load the saved model whenever we want and make predictions.

Keywords: Vision Transformer, Transfer Learning, GELU, Attention

GİRİŞ

Aflatoksinler, Aspergillus mantar türleri tarafından üretilen en tehlikeli mikotoksindir [1]. Aflatoksinler karaciğer kanserine bağışıklık sisteminin zayıflamasına, tümör oluşumuna ve insanlarda enfeksiyon direncini azalmasına neden olabilir. [2,3]. Ek olarak, aflatoksinler bitkisel ve hayvansal üretimi olumsuz yönde etkiler. Ekonomik kayıplara neden olur [4]. Birleşmiş Milletler Gıda ve Tarım Örgütü (FAO) tarafından dünya mahsulünün %25'inin mikotoksinlerden etkilendiği ve bunun önemli ekonomik kayıplara neden olduğu ve önemli sağlık riskleri oluşturduğu tahmin edilmektedir [5]. Çeşitli Ülkeler ve kuruluşlar, aflatoksin bulaşmış ürünleri kontrol etmek ve ticaretini yasaklamak için katı düzenlemeler getirmiştir. İnsan tüketimi için güvenli aflatoksin limiti 4-30 µg/kg (ppb) aralığındadır [6]. İncir için kabul edilebilir maksimum aflatoksin miktarı ülkeler ve kuruluşlara göre değişir. Bu miktar ABD için 20 µg/kg (ppb) olarak belirlerken, Avrupa Birliği için 4 µg/kg (ppb) olarak belirlemiştir [7,8]. Kuru İnce tabaka kromatografisi (TLC), sıvı kromatografi-tandem kütle spektrometrisi (LC-MS/MS) ve yüksek performanslı sıvı kromatografisi (HPLC) gibi kromatografik yöntemler, aflatoksin tespiti için altın standart olarak kabul edilir. Aflatoksin için altın standart olarak belirlenen yöntemler iyi donanımlı bir laboratuvar ve deneyimli personel gerektirir. Bu yöntemler pahalıdır, zaman alıcıdır ve numunelerin parçalanmasını gerektirdiği için ürüne zarar verir. Aynı zamanda bu yöntemler akan bir banta uygulanamaz. Aflatoksin bulaşmış bazı gıda türleri ultraviyole (UV) ışıkla aydınlatıldığında, aflatoksin bulaşmış kısımlar floresan yayar. İncir de bu türlerden bir tanesidir. Floresansa Aflatoksin üreten Aspergillus mantarları tarafından sentezlenen kojik asit ve peroksidaz enzimlerinin etkisiyle oluşur ve Parlak Yeşilimsi Sarı Floresan (BGYF) olarak adlandırılır [9]. BGYF'nin varlığı sadece kojik asitin varlığından dolayı ortaya çıkabilir. Bu da BGYF'nin varlığının aflatoksinin varlığına kesin kanıt olmadığını gösterir. BGYF ve aflatoksin arasındaki güçlü ilişki nedeniyle, BGYF, aflatoksin içeren veya içerme olasılığı yüksek olan ürünleri tespit etmek için varsayımsal bir test yöntemi olarak kullanılır [10]. Bu nedenle incir işleme tesislerinde aflatoksin bulaşmış incirleri ayrıştırmak için BGYF kullanılmaktadır. UV

ışığı altında BGYF yayan incir çalışanlar tarafından manuel olarak tespit edilmektedir.

Bu çalışmanın amacı, aflatoksin bulaşmış incirlerin ayıklanmasında kullanılan BGYF tespiptinin insansız ve bu sayede insan sağlığına zarar vermeyecek şekilde yapılabilmesi için etkili ve hızlı olacak otomatik bir ayıklama sisteminin kurulmasına yardımcı olmaktır. Bu amaç için BGYF ışıması altındaki kuru incir görüntüleri üzerinde makine öğrenmesi ile sınıflandırma çalışması yapmaya karar verdik.

Transformer mimarisi, doğal dil işleme görevleri için fiili standart haline gelmiş olsa da bilgisayarlı görüye yönelik uygulamaları sınırlı kalmaktadır [11]. Transformer modellerin bilgisayarlı görüye yönelik kullanılması için oluşturulmuş modellere Vision Transformer (VİT) denir. Daha önce yüksek miktarda veriler üzerinde eğitilmiş Vision transformer modeller birçok orta ve küçük boyutlu benchmark (Kalite testi) veri setleri (ImageNet, CIFAR-100, VTAB, vs.) üzerinde denenmiş ve state-of-art CNN'lere karşı çok iyi sonuçlar elde etmiştir [11]. Ayrıca sınıflandırma dışında nesne tespiti [12-17], segmentasyon [18-19], görüntü geliştirme, [18, 20], görüntü üretimi [21] video işleme [22-23] ve 3D nokta bulutu işlemede kullanılmıştır [24].

Dikkat (Attention) genellikle ya evrişimli ağlarla birlikte olarak uygulanır ya da evrişimli ağların belirli bileşenlerinin yerine konulur. Bu yapılırken CNN'in genel yapısının korunmasına dikkat edilir [25]. Attention ham verileri (örneğin bir cümle içerisindeki kelimeler.) "Okumak" ve bunların her birini konumları ile alakalı bir öznitelik vektöründe, dağıtılmış temsillere dönüştürme işlemidir. Bu işlemde amaç her bir kelimenin konum bilgisini ve aynı zamanda her bir kelimenin birbiri ile olan benzerliklerini saklayan bir veri elde etmektir. Görüntü verilerinde bir görsel n*n boyutluk parçalara bölünmüştür ve her bir parçaya bir kelime muamelesi yapılarak bu işlem gerçekleştirilmiştir. Aynı zamanda Multi-Head Attention kavramı ile bu işlemi paralel olarak birden çok kez yapılarak kodun daha hızlı çalışması sağlanmıştır. Bir sonraki aşamada ise Multi-Head Attention'dan çıkan veriler Çok Katmanlı Algılayıcı'ya verilerek sınıflandırma işlemi yapılmıştır. Karşılaştırma yapmak için büyük miktarda veri ile eğitilmiş state of art CNN modellerini Transfer

Learning metodu ile kendi veri setimiz üzerinde eğiterek sonuçları kaydettik ve elde ettiğimiz Vision Transformer (VİT) sonuçları ile karşılaştırdık.

Deneyler esnasında TL yöntemi ile Inception, VGG, ResNet, EfficentNet, DenseNet dahil farklı mimarilerini içeren 25 farklı model 20 epoch olacak şekilde çalıştırldı. Daha sonra elde edilen en başarılı model üzerinde optimizer, learning rate gibi hiper parametrelerin optimum halleri belirlenmeye çalışıldı. Son olarak da modelin mimarisinde son katmana birkaç dense layer eklenerek başarısı gözlemlendi. VİT modelleri ile alakalı yapılan deneylerde ise patch size, learning rate, dropout değerleri ve modelin dense katmanlarındaki nöron sayıları üzerinde farklı değerler kullanılarak deneyler yapıldı.

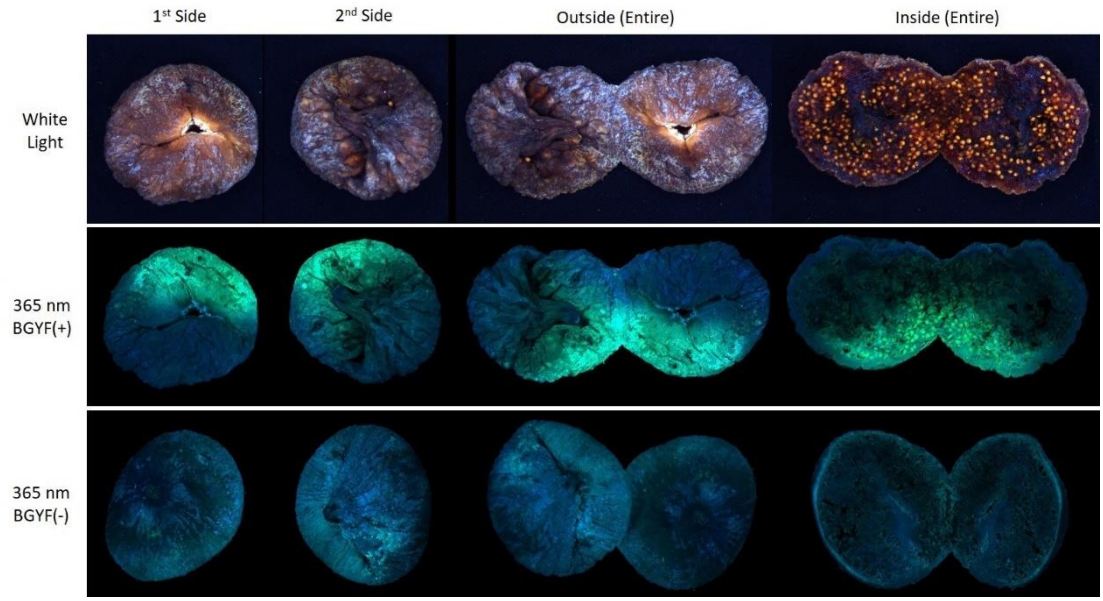
Yapılan deneyler sonucunda en yüksek Doğruluk (Accuracy) değeri incelendiğinde önerilen VİT modelinin TL modellerine kıyasla daha iyi bir sonuç elde ettiğini gözlemledik ancak VİT modeli üzerinde deneyler tekrarlandığında bu başarının tekrar elde edilemediğini fark ettik. Buna önlem olarak modeli kaydedip istediğimiz zaman kaydedilen modeli yükleyip tahminleme yapmaya karar verdik.

1. VERİ SETİ

Veri seti olarak “BGYF Classification for Aflatoxin Detection in Dried Figs: A Deep Transfer Learning Approach” makalesinde kullanılan veri seti makale yazarlarının izni dahilinde kullanılmıştır. İncirlerin görüntülerini elde etmek için 250 nm ile 1000 nm arasında değişen farklı dalga boylarında ışık kaynakları, optik filtreler ve 15 megapiksel (mp) renkli CMOS kamera içeren bir sistem kullanılmıştır. Her bir incirin her görüntüsü, aynı optik koşullar ve parametreler altında elde edilmiştir. İncir örnekleri, dört taraftan 50° açıyla konumlandırılan LED ışık kaynakları ile aydınlatılan bir yüzeyin ortasına yerleştirilmiştir. Görüntüler, yüzeye 90°'lik bir açıyla yerleştirilmiş ve İstenmeyen ortam ışığını engellemek için cihazın kapakları kapatılarak karanlık bir ortam oluşturulmuştur. Görüntüleri elde etmek için optik sistem Şekil 1'de gösterilmiştir.

Şekil 1.1. Görüntü elde etmek için kullanılan optik sistem [26].

İlk olarak incir örneklerinin çift taraflı (ön ve arka) görüntüleri 365 nm (Nichia – 1450 mW) LED ışık kaynakları altında ayrı ayrı alınmıştır, çünkü aflatoksin kontaminasyonu incirin dışında herhangi bir pozisyonda bulunabilmektedir. Daha sonra incirler her seferinde alkolle temizlenen bir bıçakla ortadan ikiye kesilerek hem iç yüzeyi hem de tüm dış yüzeyi görüntülenmiştir. Sonuç olarak 100 adet incir numunesi için UV ışık kaynağı altında 400 adet görüntü elde edilmiştir. Alınan her bir incir numunesi görüntüsünün çözünürlüğü 2304x1644 pikseldir. Şekil 1.1, alınan örnek incir görüntülerini göstermektedir.



Şekil 1.2. Beyaz ışık altında örnek incir görüntüleri ve UV ışığı ile aydınlatılmış iki örnek, BGYF için bir pozitif ve bir negatif [26].

1.1. Veri Setinin Etiketlenmesi

İncir üretim tesisinin uzman çalışanları, incir örneklerini UV ışığı altında BGYF yayanlar ve yaymayanlar olarak sınıflandırmıştır. İncir örnekleri de kendi optik sistemimiz kullanılarak incelenmiştir. Sonuç olarak 400 görüntüden oluşan veri setimiz oluşturulmuş ve tüm incirler BGYF(+) veya BGYF(-) olarak etiketlenmiştir. Temsili bir örnek için, incir örneğinin floresan emisyonları incelenmiş ve saptanamayan, düşük, orta ve yüksek floresan yansıması olarak gruplara ayrılmıştır. Daha sonra her gruptan rastgele 25 adet incir seçilmiştir. Burada amaç veri setini dengeli olarak dağıtmaktır. Veri setinin dengeli olarak dağıtılması overfit probleminin önüne geçilmesi için önemli olmasının yanında fazla ışıma yapan incirlerin modelin aflatoksin tespitini kolaylaştırması az ışıma yapanların ise modelin tesppitini zorlaştırması sebebi ile ışıma olarak dengeli bir veri seti kullanılması önemlidir. Bu çalışmada elde edilen 100 verinin 365 nm boyundaki ışık altında üst ve alttan çekilen 200 görüntüsü kullanılmıştır. Bu görüntülerin etrafındaki siyah alanlar kesilerek atılmıştır.

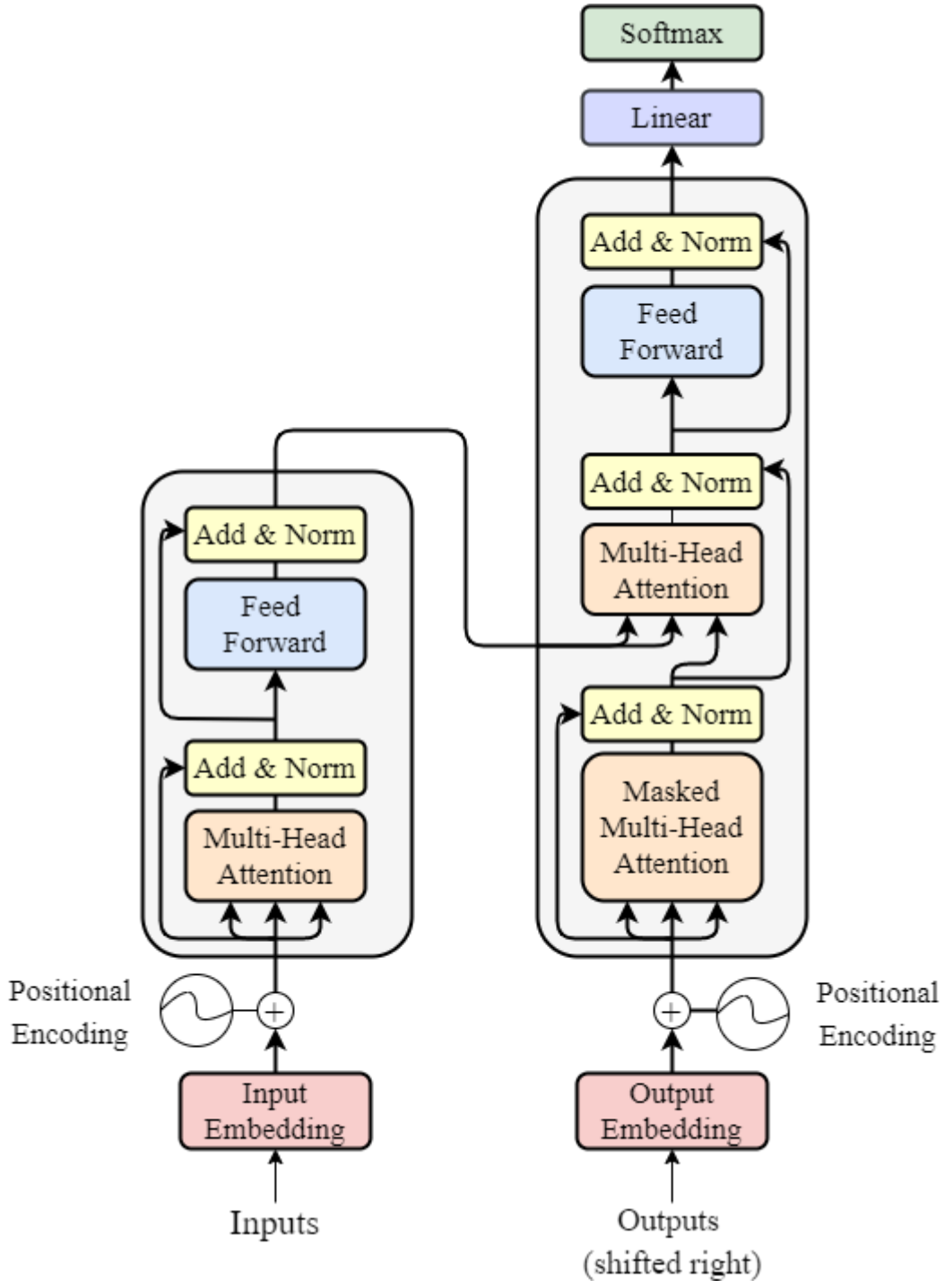
2. TRANSFORMER MODELLERDE TEMEL KAVRAMLAR

Transformer modeller giriş verilerinin her bir bölümünün önemini farklı şekilde ağırlıklandırır. Self attention mekanizmasını benimserler. Öncelikli olarak doğal dil işleme ve bilgisayarlı görü alanlarında kullanılır.

2.1. Transformer ve Attention kavramı

Transformer'ın temel fikri attention'ı tekrarlama (read RNN) olmadan kullanmaktır. Böylece transformer hala bir diziden diziye (Seq2Seq) modeldir ve kodlayıcı (encoder) kod çözücü (decoder) yapısını kullanır.

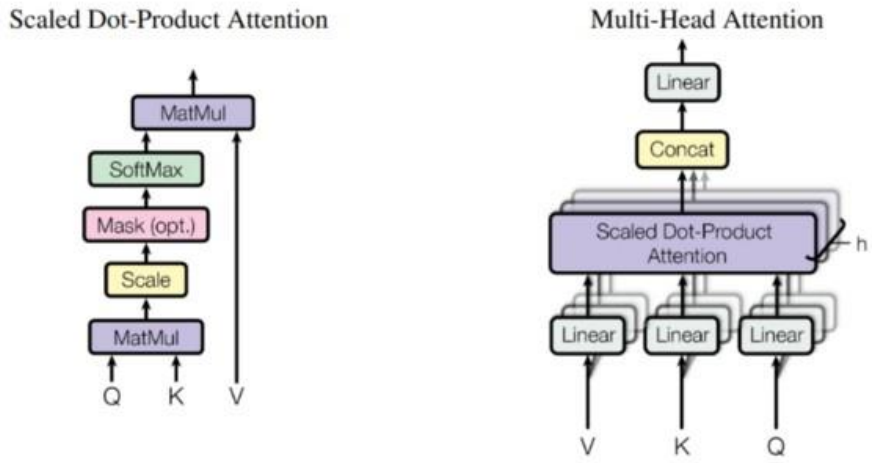
Kodlayıcı, sembol temsillerinin (x1, …, xn) bir giriş dizisini bir sürekli temsiller dizisine z = (z1, …, zn) eşler. z verildiğinde, kod çözücü daha sonra her seferinde bir öğe olmak üzere sembollerin bir çıktı dizisini (y1, …, ym) üretir. Model otomatik gerilemelidir (auto-regressive). Her adımda model, bir sonrakini oluştururken ek girdi olarak önceden oluşturulmuş sembolleri tüketir. [25]



Şekil 2.1. Transformer Mimarisi [11]

Kodlayıcı katman iki önemli parça içerir bunlar;

Multi-head self-attention blok ve Pozisyon olarak tam bağlantılı ileri beslemeli ağ.



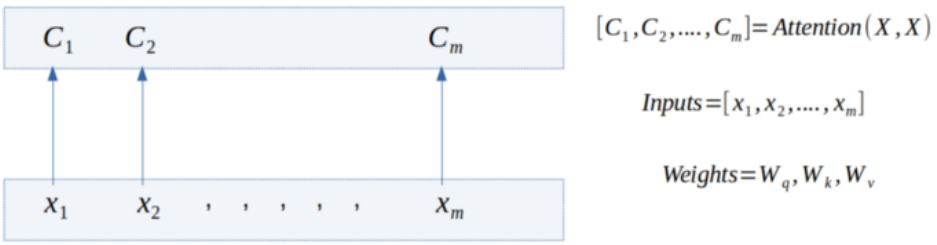
Şekil 2.2. Multi-Head Attention [11]

Şekil 2.1. deki diyagram içerisindeki 3 etiket Q, K, V Sorgu, Anahtar ve Değer vektörlerini ifade eder. Şimdilik bunu bilgi alma protokolümüzün bir parçası olarak düşünelim. Arama yaptığımızda (Sorgu) arama motoru sorgumuzu anahtarlar ile kıyaslar ve bize bir değer döndürür.

Kodlayıcı self attention katmanları içerir. Self attention katmanlarda tüm sorgu, anahtar ve değerler aynı yerden gelir. Bizim durumumuzda bu yer kodlayıcı içerisindeki bir önceki katman oluyor. Kodlayıcı içerisindeki her katman kendinden önceki katmanlara katılabilir.

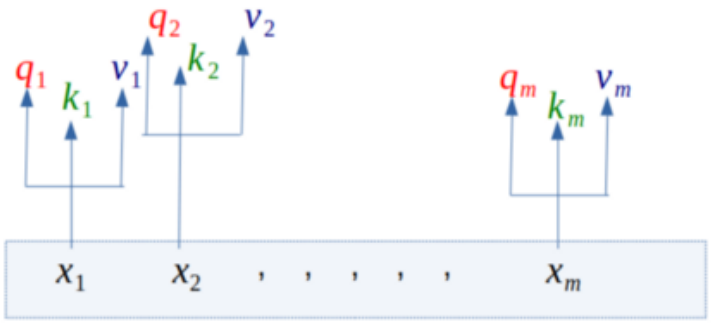
2.1.1 Self attention

Bir giriş dizisi düşünelim (x1, x2, …, xm) bu girdiye sahip olan self attention katmanının çıktısı, aynı uzunluktaki bir dizi bağlam vektörüdür. (C1, C2, …, Cm)



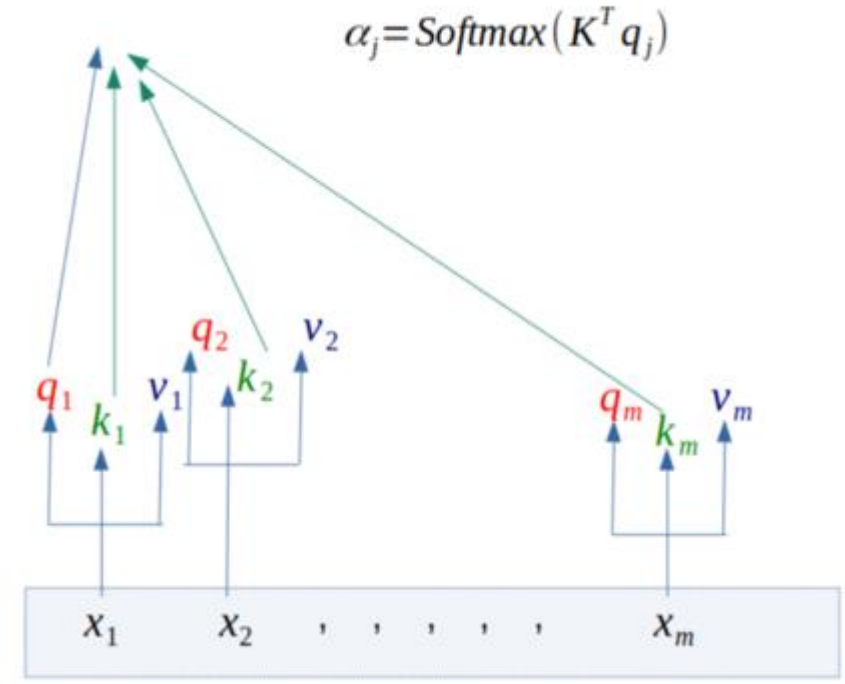
Şekil 2.3. Giridi dizilerinin bağlam vektörüne dönüştürülmesi [27].

Self attention'dan sonraki girdi dizisinin çıktıları bağlam vektörleridir. Bağlam vektörü C_i , X_i pozisyonundadır ancak C_i sadece X_i 'ye bağlı değildir tüm i lere bağlıdır. Şekil 1.3'te sorgu, anahtar ve değerler için W_q, W_k, W_v olarak eğitilecek ağırlıkları tanımlıyoruz.



Şekil 2.4. Eğitimde güncellenecek olan Anahtar, Sorgu ve Değer ağırlıkları [27].

X_i q_i, k_i, v_i olarak haritalanır. q_i, k_i ve v_i X_i'ye 3 set ağırlıkla bağlıdır. $q_i = W_q * X_i$; $k_i = W_q * X_i$; $v_i = W_v * X_i$. Bu ağırlıklar eğitim aşamasında eğitilir. İnput farketmeksizin aynı ağırlık kullanılır. Tüm i'lere aynı ağırlığın uygulanması önemli bir noktadır. Sorgu ve anahtarın boyutları d_k olarak alınır ve değerler için ise d_v olarak alınır. Örneğin 5 boyutlu bir X_i'ye sahip isek örneğin [0,1,1,2,3] ve sorgumuz 3 boyutlu ise o zaman Wq 5*3 boyutlu oluyor. Aynısı anahtar ve karşılık gelen ağırlıklar için de geçerlidir.



Şekil 2.5. Attention konseptinde nokta çarpım (Dot-Product) i'inci sorgu ile dizideki tüm Anahtarların nokta çarpımı [27].

α_i q_i'ye ve tüm K(k_i'ler) e bağımlıdır.

Nokta çarpım bir seçimdir ve artımlı da olabilir. Aslında birçok seçenek vardır konum tabanlı da olabilir içerik tabanlı da olabilir. Nokta çarpım attention hızlıdır ve çok fazla depolama alanı istemez. Bunun sebebi de yüksek oranda optimize edilmiş matris çarpma kodları kullanılarak kodlanabilmesidir. Her α_i X_i'nin pozisyonundadır ancak tüm X_i lere bağımlıdır. Önemli nokta ise α_i'nin hesaplanmasıdır.

$$c_1 = \alpha_{11} v_1 + ... + \alpha_{m1} v_m = V\ \alpha_1 = V\ Softmax(K^T q_1)$$

$$\vdots$$

$$c_j = \alpha_{1j} v_1 + ... + \alpha_{mj} v_m = V\ \alpha_j = V\ Softmax(K^T q_j)$$

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Şekil 2.6. Attention bloklarının çıktısı olan bağlam vektörleri (C'ler) elde etmek [27].

Tüm girdiler birbirleri ile etkileşimdedir ('Self' terimi buradan geliyor.) ve nereye daha çok dikkat edilmesi gerektiğinin bulurlar. Genelde katmanlarımızı tüm model boyunca eşit varyansa sahip olması amacıyla başlatırız. Ancak Q ve K vektörlerine σ^2 varyans ile nokta çarpım uyguladığımızda bu ölçekleyicinin d kat daha fazla varyansa sahip olması ile sonuçlanır. d_k Q ve K'nın boyutudur ve V nin boyutu da d_v dir.

$$q_i \sim N(0, \sigma^2),\ ki \sim N(0, \sigma^2), \rightarrow Var\left(\sum_{i=1}^{d_k} q_i\ .\ k_i\right) = \sigma^2\ .\ d_k$$

Normal olarak dağıtılmış sorgular ve anahtarlar. Nokta çarpımdan sonraki toplam varyans d_k kat daha fazladır. Varyansı σ^2'ye geri ölçeklendirmezsek logitlerin üzerindeki softmax, rastgele bir eleman için 1'e ve diğerleri için 0'a doymuş (saturated) olurdu. Softmax üzerinden gradyanlar sıfıra yakın olacak, böylece parametreleri uygun şekilde öğrenemeyeceğiz. Ancak şekil 2.1'deki nokta çarpımı yapısı sayesinde bu sorunu ortadan kaldırabiliyoruz.

2.1.2. Multi head self attention

Single-head self attention'un basit bir eklentisidir. Multi-head self-attention da h adet single head self-attention (katmanlar) bulunur. Single-head self attention'da

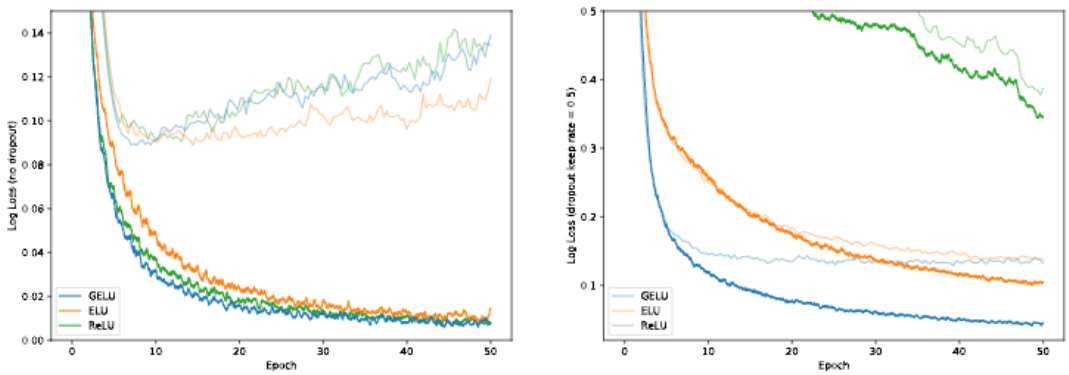
eğitilebilir parametreler Wq, Wk, Wv'dir. 'h' adet single-head self attention katmanı parametrelerini paylaşmaz bu sayede 3h parametremiz oluşur. Her bir single-head self attention katmanı bir adet bağlam vektörü çıktısı verir. Bu bağlam vektörleri birleştirilir. Eğer Single-head self attention çıktı olarak çok boyutlu bir vektör verirse örneğin her bir C_i d * 1 boyutlu. Bu durumda verilen h katmanlı Single-head self attention'dan oluşan multi-head çıktısı hd * 1 boyutlu vektör olur.

Multi-head self attention'un önemi şudur ki;

Multi-head self attention modelin farklı pozisyonlardaki farklı temsili alt uzaylardan gelen ortaklaşa bilgilerine katılmasını sağlar [25].

2.2. Gaussian Error Linear Unit (GELU)

GELU Yüksek performanslı bir Sinir ağı (Neural Network) aktivasyon fonksiyonudur. GELU aktivasyon fonksiyonu xΦ(x)'dir. xΦ(x)'in standart Gauss kümülatif dağılımı olduğu noktada. GELU, ReLU'larda olduğu gibi girişleri işaretlerine göre yönlendirmek yerine, doğrusal olmayan girdileri değerlerine göre ağırlıklandırır [28].

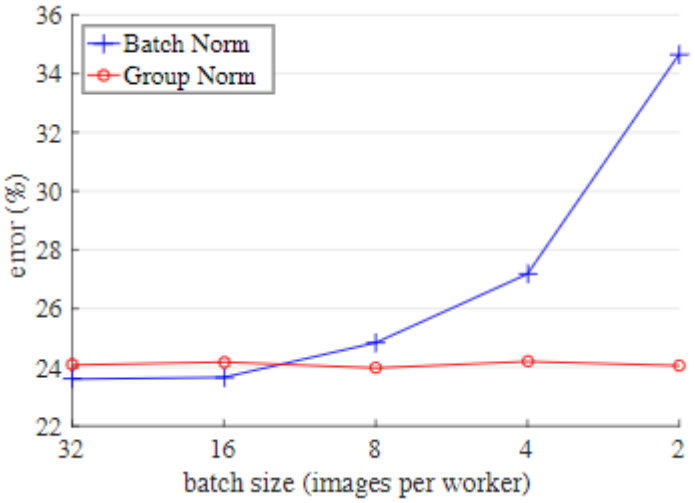


Şekil 2.7. GELU [28]

2.3. Group Normalization (Grup normalleştirmesi)

Batch Noarmalization (BN), derin öğrenmenin geliştirilmesinde bir kilometre taşı tekniğidir. Şu ana kadar çeşitli ağların eğitilmesine olanak sağladı. Bununla birlikte, parti boyutu (batch dimension) boyunca normalleştirme bazı problemlere yol açtı.

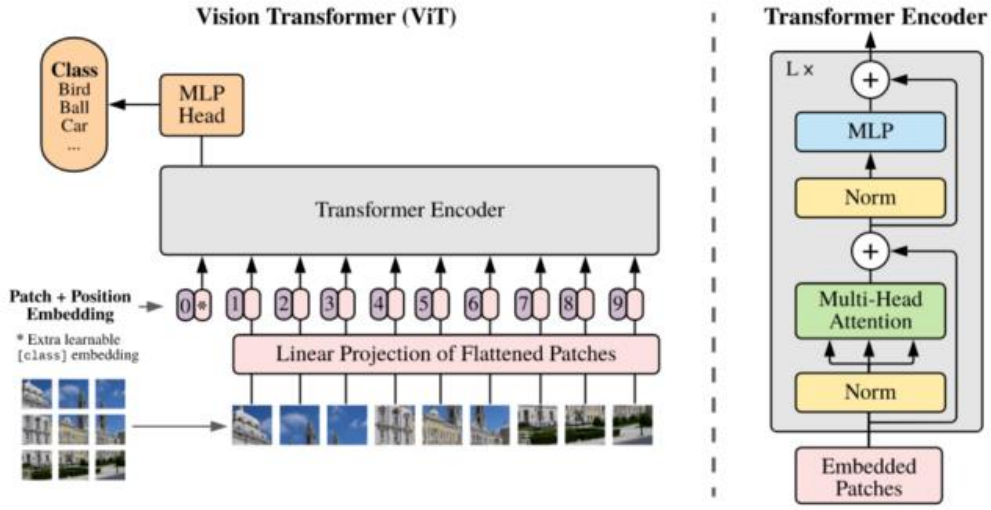
Parti boyutu küçüldükçe BN'in hatası hızla artar. Bu sorun hatalı toplu istatistik tahmininden kaynaklanır. Bu, BN'i daha büyük modelleri eğitmekte ve özellikleri bilgisayarlı görü görevlerine öznitelikleri aktarmak için kullanımında sınırlamaktadır. Örneğin tespit (dedection), segmentasyon ve video işleme gibi küçük partiler gerektiren bellek tüketimiyle sınırlandırılmış işlemlerde. Group Normalization (GN) BN'in bir alternatifi olarak ortaya atılmıştır. GN kanalları gruplara böler ve her grup içinde normalizasyon için ortalama ve varyansı hesaplar. GN'nin hesaplaması parti boyutlarından bağımsızdır. Doğruluğu, çok çeşitli parti boyutlarında stabil olduğu gözlemlenmiştir [29].



Şekil 2.8. Parti boyutlarına göre ImageNet sınıflandırma hataları. [29]

3. VİSİON TRANSFORMER

Vision Transformer modelini kurarken “An Image is Worth 16 X 16 Words” adlı makaleden yararlandık.



Şekil 3.1. Vision Transformer mimarisi [11]

Bir VİT sınıflandırıcı modeli yapısında öncelikle görüntülerden parçalar (patches) oluşturulur. Bu parçalara konumsal gömme eklenir. Konumsal gömme eklenen parçalar Transformer kodlayıcı bloğa girdi olarak verilir. Bu girdilerden ise vektörler elde edilir. Bu vektörler multilayer perceptron sınıflandırıcısı için girdi olarak verilir. Bu sınıflandırıcı verileri sınıflandırır.

Bir Vision Transformer sınıflandırıcı örneğinde örneğin bir resim alalım 256*256*3. Bu resmi daha küçük parçalara bölünür. Örneğin 16*16*3. Bu parçaları doğrusal hale getirilir. Örneğin 768*1. Bu veriler transformer kodlayıcı’ya girdi olarak verilir. Transformer kodlayıcı hd * 1 boyutlu bir vektörü çıktı olarak verir. Bu vektör multilayer perceptron’a verilerek Sınıflandırma yapılır.

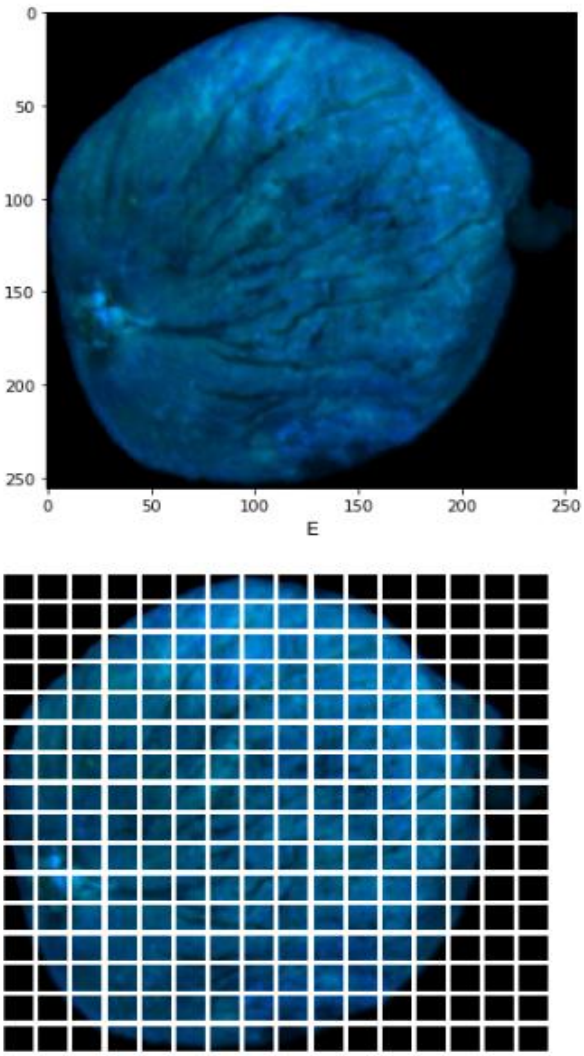
3.1 Parça Üretimi

Bir resmi $x \in R(H \times W \times C)$ olarak alalım ve bunu parçalara bölelim.

x_p $p \in R(N \times P \times P \times C)$ H ve W orijinal resimin boyu ve genişliğidir. C ise kanal sayısıdır. (P, P) her resim parçasının çözünürlüğüdür. $N=HW/P^2$ ise elde edilen parça sayısıdır. Aynı zamanda Transformer için girdi boyutu (input size) değeridir.

Kod içerisinde parça oluşturmak için Keras kütüphanesinden convulational layer adlı class çağrılarak bu class'ın üzerine yazarak yeni oluşturulan class kullanılmıştır. Bu fonksiyonun filtre sayısı 'hidden_dimension' adlı değişkene atanmıştır. Hidden dimension sorgu ve anahtar boyutudur. Daha önceden d_k olarak ifade ediyorduk. Fonksiyonun oluşturduğu çıktının boyutu ise (batch_size, parça sayısı, hidden_dim)'dır. Aynı boyuttaki bir öğrenilebilir konumsal gömme katmanı girdi olarak eklenmiştir.

Hidden dimension'ı kodlayıcı blokta multi head attention katmanına ihtiyacımız olduğunda kullanacağız. Yani bu kod konumsal olarak gömülmüş ve düzleştirilmiş (Flattened) parçaları transformer'a girdi olarak verilebilir hale getiriyor.



Şekil 3.2. Parçalanmış görüntü.

3.2 Konumsal Gömme

Bu kısımda kod içerisinde konumsal gömme yapabilmek için 'tf.keras.Layer'

Class'ını genişleterek (extend) ağırlıkları rastgele başlatacağız. Konumsal kodlamayla ilgili bazı önemli noktalar Inductive Bias ve Class Token'dir.

3.2.1. Inductive bias

Bir görsel parçalara bölündüğünde girdi'nin yapısını kaybederiz konumsal gömme modelin girdisi olan resim'in yapısını öğrenmesi konusunda yardımcı olur. Bu

konumsal yerleştirmeler öğrenilebilir ve modelin görüntü yapısı hakkında kendi başına ne kadar öğrenebileceğini gözler önüne serer. 2D gömme, performansı fazla değiştirmez. Transformer'lar ile CNN'ler arasındaki en büyük fark ise CNN'de çekirdekler (Kernels) 2D komşuluk yapılarını öğrenmemizi sağlar. Ancak transformer modeller MLP (Multi Layer Perceptron) katmanı dışında bu lokal 2D yapısı kullanılmaz. Başlatma zamanındaki konumsal gömmeler, parçaların 2D konumu hakkında hiçbir bilgi taşımaz ve parçalar arasındaki tüm uzamsal ilişkiler sıfırdan öğrenilir [11].

3.2.2. Class token

Class token orijinal transformer mimarisini olabildiğince benzetmek için yapılmıştır. Daha önce yapılan çalışmalarda Araştırmacılar ayrıca yalnızca görüntü parça yerleştirmelerini, küresel ortalama havuzlamayı (GAP) ve devamında ise doğrusal bir sınıflandırıcı kullanmayı denediler [11]. Devamında elde edilen düşük performans token'in olmadığından değil de öğrenme katsayısı'nın optimum olmamasından kaynaklandığını buldular [11].

Tüm bunlar göz önüne alındığında kodu yazarken biz class token kullanmadık.

3.3 Transformer Kodlayıcı Blok

Transformer kodlayıcı blok iki parçadan oluşur.

3.3.1. Çok katmanlı algılayıcı (MLP)

Çok katmanlı algılayıcı (MPL) GELU içerir. MLP boyutları örnek alınan makalede verilmiş ancak bu boyutlardaki modeler çok fazla veriye aç oldukları için elimizdeki veri sayısınıda göz önünde bulundurarak daha küçük boyutlardaki bir versiyonunu kullanmayı tercih ettik. Kodlayıcı bloğu tekrar ettiğinden dolayı. 'Dense' katmanlarındaki birim sayısına dikkat etmeliyiz. Bunun sebebi çıktı boyutu bir sonraki multi head attention katmanı'nın girdisi ile uyumlu olması gerektiğidir.

3.3.2. Normalization

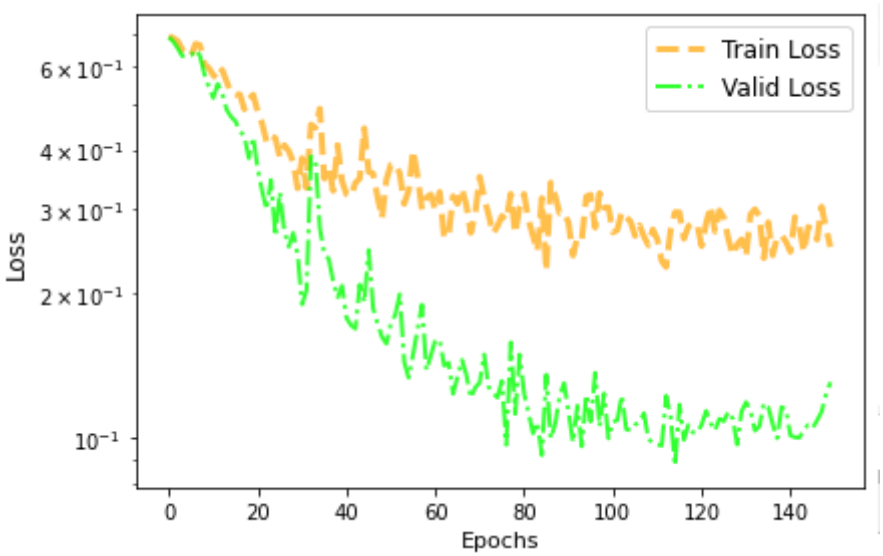
Boyutu (N, C, H, W) olan bir girdi tensor'unu düşünelim. Ortalamayı ve varyansı hesaplayalım (μ_i, σ_i). Bunu (C, H, W) boyutları ile birlikte yapalım. Bu bir batch içerisindeki tüm özniteliklerin diğerinden bağımsız olmasını sağlamak için yapılır.

4. YÖNTEM VE BULGULAR

Kullandığımız modelleri python programlama dilinde yazdık. Jupyter notebook ide olarak kullanıldı. Veri seti %80'e %20 olacak şekilde rasgele ayırdık. Verileri sahte veri üretilerek çoğalttık.

VİT modelleri ile alakalı yapılan deneylerde patch size, learning rate, dropout değerleri ve modelin dense katmanlarındaki nöron sayıları üzerinde farklı değerler kullanılarak deneyler yapıldı. Verilerin boyutları transfer learning modelleri için 244*244*3 olacak şekilde kullanıldı. Bu transformer modellerde 256*256*3 128*128*3 64*64*3 ve 32*32*3 olarak değişti. Patch size (parça boyutu) olarak ise 4*4*3 8*8*3 ve 16*16*3 olarak deneyler yapıldı.

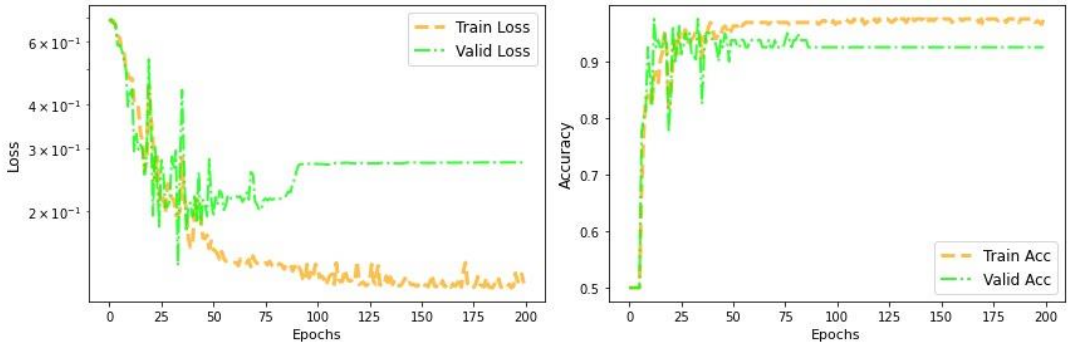
Deneyler esnasında TL yöntemi ile Inception, VGG, ResNet, EfficentNet, DenseNet dahil farklı mimarilerini içeren 26 farklı model 20 epoch olacak şekilde çalıştırldı. Daha sonra elde edilen en başarılı model üzerinde optimizer, learning rate gibi hiperparametrelerin optimum halleri belirlenmeye çalışıldı. Son olarak da modelin mimarisinde son katmana birkaç dense layer eklenerek başarısı gözlemlendi.



Şekil 4.1. Yapılan ilk deneylerden bir tanesi.

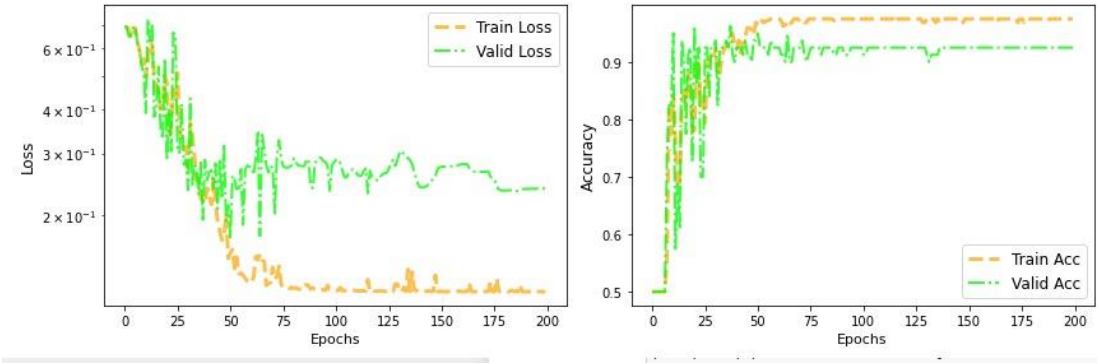
Yapılan ilk deneylerde validation loss değerinin train loss değerinden daha düşük çıkması gibi bir problemle karşılaştık. Bu problemi çözmek için yaptığımız

araştırmalar sonucunda bu problemin modeldeki dropout değerlerinin çok yüksek olması ile alakalı olabileceğini düşündük. Dropout değerlerini düşürüldükten sonra bu problemi aştık.



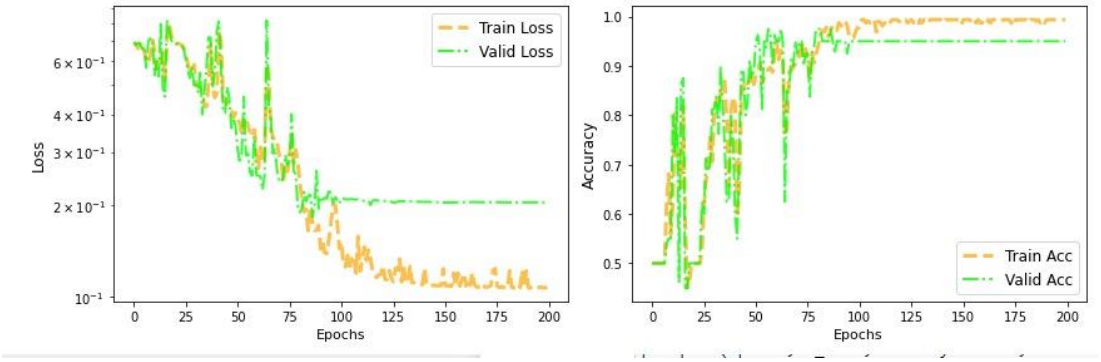
Şekil 4.2. veriyi 32*32*3 piksel olarak aldığımız bir deney.

32*32*3 piksel olarak alınan veriler ile 200 epochs'ta yapılan deneylerde validasyon başarı oranı %92,5 olmuştur.



Şekil 4.3. veriyi 64*64*3 piksel olarak aldığımız bir deney.

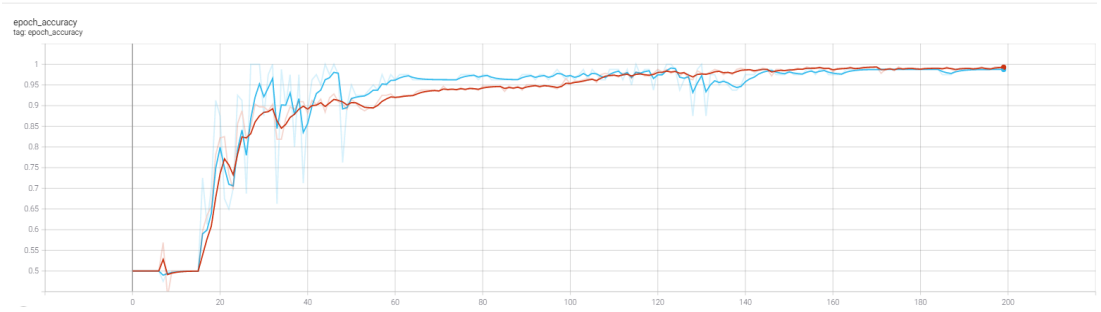
64*64*3 piksel olarak alınan veriler ile yapılan deneylerde validasyon başarı oranı %92,5 olmuştur.



Şekil 4.4. veriyi 128*128*3 piksel olarak aldığımız bir deney.

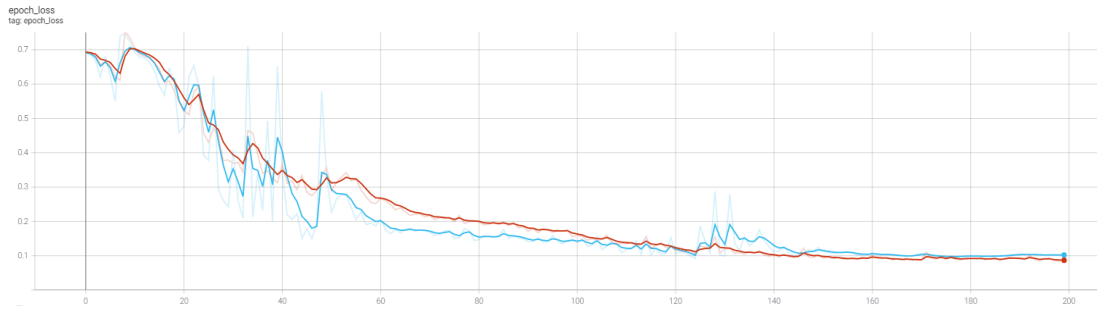
128*128*3 piksel olarak alınan veriler ile yapılan deneylerde validasyon başarı oranı %95,0 olmuştur.

Veri boyutu olarak 32, 64 ve 128 ile yapılan deneylerde en iyi valıdasyon başarısı olarak %96 değeri bulunurken Yapılan tüm çalışmalarda en başarılı model resimlerin 256*256*3 piksel olarak alındığı ve patch size'ın 16*16*3 olarak seçildiği veri setinin ise rasgele olarak dağıtıldığı bir VİT modelinde elde edildi.



Şekil 4.5 Elde edilen en iyi accuracy değeri.

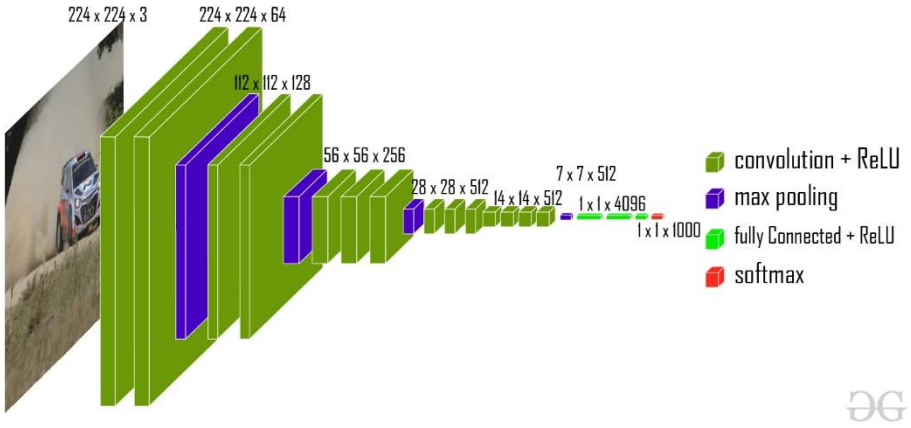
Yapılan bu deneyde elde edilen accuracy score train için 99.37 iken test için accuracy score ise %98.75 oldu.



Şekil 4.6. Elde edilen en iyi loss değeri.

Aynı deneyde elde edilen en düşük loss değeri ise train için 0.086 iken test için 0.1021'dir.

Transfer learning kullanılarak çalıştırılan modellerde ise öncelikle tüm modeler denendi burada en düşük loss değerine sahip olan VGG16 modeli seçildi. İlerleyen aşamalarda ise bu model üzerinde tuning yapılarak başarısı arttırılmaya çalışıldı.



Şekil 4.7. VGG16 Mimarisi.

İlk aşamada modelin hangi optimizer ile en başarılı sonucu elde ettiğini bulmak için "Adam","SGD","RMSprop","Adadelta","Adagrad","Adamax","Nadam","Ftrl" adlı optimizerlar denendi. Çıkan sonuçta en iyi optmizer "Adamax" olarak belirelendi. Daha sonra ise bu optimizer ile en başarılı öğrenme katsayısını bulmaya çalıştık. Burada da yapılan deneyler sonucu 0.0001 değeri en başarılı sonucu verdi. Başarı ölçütümüz loss değeri idi ilk etapta 0.40 loss ile VGG16 belirlendi İkinci olarak 0.24 ile Adamax belirlendi en son olarak'da 0.22 ile öğrenme katsayısı belirlendi.

5. SONUÇLAR

Bu çalışmada kuru incir görüntüleri üzerinde aflatoksin tespiti için VİT modeli ile sınıflandırma yapmak önerilmiştir. Sınıflandırma yapabilmek için dengeli bir veri seti oluşturulmuştur. Yapılan çalışmalar gösteriyor ki elde edilen sonuçlara göre VİT modelin sahip olduğu validasyon başarısı TL yöntemi ile kullanılan konvülasyon içeren modellerden daha iyidir. Ancak VİT modelindne elde edilen sonuçlar stabil değildir deneyler farklı train ve test verileri ile tekrarlandığında farklı başarı sonuçları elde ediliyor. Bu bize modellerin potansiyelleri ile ilgili fikir verse de modellerin kesin olarak karşılaştırılması için bu çalışmalar daha fazla veri ile ve daha farklı veri setleri ile tekrarlanmalıdır. Eğer bu rasgelelikten kurtulunmak isteniyorsa VİT modelleri kaydedilip daha sonrasında yüklenerek tahmin yapılabilir. Yöntemimiz, kuru incir üretim tesislerinde kullanılan konveyör bant sistemlerinde hızlı hareket eden incirler arasında dahi BGYF emisyonlarını otomatik olarak tespit etmek için etkin bir şekilde kullanılabilir.

KAYNAKLAR

[1] Valencia-Quintana R, Milić M, Jakšić D, Klarić MŠ, Tenorio-Arvide MG, Pérez-Flores GA, et al. Environment changes, aflatoxins, and health issues, a review. Vol. 17, International Journal of Environmental Research and Public Health. MDPI AG; 2020. p. 1–10.

[2] Gimeno A, Martins ML. Mycotoxins and Mycotoxicosis in Animals and Humans. Miami, USA: Special Nutrients, Inc.; 2006

[3] Kumar P, Mahato DK, Kamle M, Mohanta TK, Kang SG. Aflatoxins: A global concern for food safety, human health and their management. Front Microbiol. 2017;7(JAN):1–10.

[4] Mitchell NJ, Bowers E, Hurburgh C, Wu F. Potential economic losses to the US corn industry from aflatoxin contamination. Food Addit Contam - Part A Chem Anal Control Expo Risk Assess [Internet]. 2016 Mar 3 [cited 2020 Dec 5];33(3):540–50. Available from: https://www.tandfonline.com/doi/abs/10.1080/19440049.2016.1138545

[5] Janik E, Niemcewicz M, Ceremuga M, Stela M, Saluk-Bijak J, Siadkowski A, et al. Molecular Aspects of Mycotoxins-A Serious Problem for Human Health. Vol. 21, International journal of molecular sciences. NLM (Medline); 2020.

[6] Udomkun P, Wiredu AN, Nagle M, Müller J, Vanlauwe B, Bandyopadhyay R. Innovative technologies to manage aflatoxins in foods and feeds and the profitability of application – A review. Vol. 76, Food Control. Elsevier Ltd; 2017. p. 127–38.

[7] European Commission-EC. Commission regulation (EU) no 165/2010 of 26 February 2010, amending regulation (EC) no 1881/2006 setting maximum levels for certain contaminants in foodstuffs as regards aflatoxin. Off J Eur Union. 2010; L 50:8–12.

[8] Van Egmond HP, Schothorst RC, Jonker MA. Regulations relating to mycotoxins in food: Perspectives in a global and European context. Vol. 389, Analytical and Bioanalytical Chemistry. 2007. p. 147–57.

[9] Marsh PB, Simpson ME, Ferretti RJ, Merola G V., Donoso J, Craig GO, et al. Mechanism of Formation of a Fluorescence in Cotton Fiber Associated with Aflatoxins in the Seeds at Harvest. J Agric Food Chem. 1969;17(3):468–72.

[10] Hruska Z, Yao H, Kincaid R, Brown RL, Bhatnagar D, Cleveland TE. Temporal effects on internal fluorescence emissions associated with aflatoxin contamination from corn kernel cross-sections inoculated with toxigenic and atoxigenic Aspergillus flavus. Front Microbiol. 2017;8(SEP):1–10.

[11] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", arXiv.org, 2022. [Online]. Available: https://arxiv.org/abs/2010.11929. [Accessed: 15- May- 2022].

[12] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. arXiv preprint arXiv:2005.12872, 2020.

[13] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159, 2020

[14] Z. Dai, B. Cai, Y. Lin, and J. Chen. Up-detr: Unsupervised pre-training for object detection with transformers. arXivpreprint arXiv:2011.09094, 2020.

[15] Z. Sun, S. Cao, Y. Yang, and K. Kitani. Rethinkingtransformer-based set prediction for object detection. arXivpreprint arXiv:2011.10881, 2020

[16] M. Zheng, P. Gao, X. Wang, H. Li, and H. Dong. End-to-end object detection with adaptive clustering transformer. ArXiv preprint arXiv:2011.09315, 2020.

[17] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, and W. Gao. Pre-trained image processing

[18] Y. Wang, Z. Xu, X. Wang, C. Shen, B. Cheng, H. Shen, and H. Xia. End-to-end video instance segmentation with transformers. arXiv preprint arXiv:2011.14503, 2020.

[19] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo. Learning texture transformer network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5791–5800, 2020.

[20] N. Parmar, A. Vaswani, J. Uszkoreit, Ł. Kaiser, N. Shazeer, A. Ku, and D. Tran. Image transformer. arXiv preprint arXiv:1802.05751, 2018.

[21] L. Zhou, Y. Zhou, J. J. Corso, R. Socher, and C. Xiong. End-to-end dense video captioning with masked transformer. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8739–8748, 2018.

[22] Y. Zeng, J. Fu, and H. Chao. Learning joint spatial-temporal transformations for video inpainting. In European Conference on Computer Vision, pages 528–543. Springer, 2020.

[23] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun. Point transformer. arXiv preprint arXiv:2012.09164, 2020.

[24] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on

Computer Vision and Pattern Recognition, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[25] A. Vaswani et al., "Attention Is All You Need", *arXiv.org*, 2017. [Online]. Available: https://arxiv.org/abs/1706.03762. [Accessed: 15- May- 2022].

[26] C. Kılıç and B. İnner An Effective "BGYF Classification for Aflatoxin Detection in Dried Figs: A Deep Transfer Learning Approach"

[27] https://github.com/suvoooo/Learn TensorFlow/blob/master/ViT_TensorFlow/Understand%26Implement_VIT_TensorFlow.ipynb (Ziyaret Tarihi: 25 Mayıs 2022)

[28] Hendrycks, Dan, and Kevin Gimpel. "Gaussian error linear units (gelus)." *arXiv preprint arXiv:1606.08415* (2016).

[29] Wu, Yuxin, and Kaiming He. "Group normalization." *Proceedings of the European conference on computer vision (ECCV)*. 2018.