

Akciğer Görüntü Verileri İle Hastalık Tahmin Modeli

1st Süleyman Talha Sarı
Bilişim Sistemleri Mühendisliği 3.sınıf

Kocaeli Üniversitesi

2nd Melih İyigören

Bilişim Sistemleri Mühendisliği

Kocaeli Üniversitesi

Özet— Bu çalışma, internetten toplanan akciğer görüntülerini işleyerek hastalık tahmini yapabilen bir yapay zeka modeli geliştirmeyi amaçlamaktadır. Görüntüler yeniden boyutlandırma, parlaklık ve kontrast ayarı ile veri artırma işlemlerinden geçirilmiştir. Python[2], Selenium[3], BeautifulSoup[4] ve OpenCV[5] kullanılarak veriler işlenmiş, model tahmin gücünü artırmak için ön işleme adımları uygulanmıştır. Bu proje, medikal tanı süreçlerinde otomasyonu desteklemeyi hedeflemektedir.

Anahtar Kelimeler—görüntü işleme, yapay zeka modeli, python, veri işleme.

GİRİŞ

Bu projede, akciğer hastalıklarının tahminine yönelik bir yapay zeka modeli geliştirmek amacıyla akciğer görüntüleri toplandı ve işlendi. Görüntüler, veri toplama kısıtlamalarına rağmen internetten elde edildi ve kullanılabilir olanlar seçildi. Verilerin standart hale getirilmesi için, Python ve OpenCV kütüphanesi kullanılarak boyut, parlaklık ve kontrast ayarlamaları gerçekleştirildi. Bu ön işleme adımları, modelin daha doğru tahmin yapabilmesi için görüntü verisinin kalitesini ve tutarlılığını artırmak amacıyla uygulanmıştır. Proje boyunca veri artırma teknikleri kullanılarak veri seti zenginleştirilmiş, böylece oluşturulacak olan modelin genelleme yeteneği geliştirilmiştir.

I. PROJE AŞAMALARI

A. Veri Kaynağı Olacak Website Seçimi

Bu projede akciğer hastalıklarının tespiti amacıyla kullanılacak görsellerin toplandığı web sitesi Radiopaedia ve OpenI olarak seçilmiştir. Radiopaedia, tıp alanında geniş bir görsel veri tabanına sahip, özellikle tıbbi görüntüleme konusunda zengin kaynaklar sunan bir platformdur. Akciğer hastalıklarıyla ilgili çok sayıda yüksek kaliteli tıbbi görsel barındıran bu site, projede kullanılacak veriler için ideal bir kaynaktır. Web sitesi, tıbbi görüntülerin doğru etiketlenmiş ve güvenilir kaynaklardan alınmasını sağlayarak, modelin doğruluğunu artırmaya yardımcı olacaktır. OpenI sitesi ise sağlıklı akciğer görüntülerinin temini için tercih edilmiştir. Tıbbi görüntüleme alanında bir başka

güvenilir kaynak olan bu platform, projeye çeşitlilik katmış ve modelin performansını artıracak nitelikte veri sağlamıştır.

B. Veri Çekme Teknolojileri ve Dil Seçimi

Veri çekme işlemi için Python programlama dili tercih edilmiştir. Python, veri işleme ve web scraping işlemleri için yaygın olarak kullanılan güçlü bir dil olup, geniş kütüphane desteği sunmaktadır. Veri çekme sürecinde, Selenium ve BeautifulSoup gibi popüler Python paketleri kullanılmıştır. Selenium, dinamik web sayfalarından veri çekebilmek için, web tarayıcılarını otomatikleştirme yeteneği sağlar ve JavaScript tarafından oluşturulan içeriği düzgün bir şekilde alabilmemize olanak tanır. BeautifulSoup ise, statik HTML içeriklerini parse ederek, verileri hızlı ve verimli bir şekilde çıkarmamıza yardımcı olmuştur. Bu araçlar, Radiopaedia gibi etkileşimli ve dinamik içeriklere sahip web sitelerinden görsel ve metin verilerini toplamak için ideal seçimlerdir. Python'un esnek yapısı ve bu araçların sağladığı güçlü özellikler, veri çekme işleminin güvenli, verimli ve doğru bir şekilde gerçekleştirilmesini sağlamıştır.

II. WEB KAZIMA İLE RADIOPAEDIA'DAN VERİ ÇEKİMİ

Yazılan Python kodu, Radiopaedia sitesinden "lung" (akciğer) ile ilgili görselleri sitemap URL'leri üzerinden otomatik olarak indirir. Kod, Selenium kullanarak her bir sitemap sayfasını açar, XML yapısını işler, ve ilgili görsellerin URL'lerini alır. Belirli kriterlere uyan görseller requests ile indirilir ve bir klasörde saklanır. İndirme işlemi sırasında bağlantı veya yükleme hataları oluşursa, WebDriver yeniden başlatılarak işlem devam eder.

A. Sitemap İşleme Fonksiyonu

Python dilinde yazılmış olan bu fonksiyon bir sitemap URL'si alır, içindeki görselleri işler ve "lung" kelimesini içeren görselleri indirir. `driver.get(sitemap_url)`: Sayfa tarayıcıda açılır. `root = ET.fromstring(sitemap_content)`: XML verisini alır ve ElementTree kökünü oluşturur. `findall("./{*}image")`: Tüm image elemanlarını arar. `caption` ve `title`: Görsellerin açıklama ve başlıkları kontrol edilir. `download_image()`: image_url verilen dosya adına kaydedilir.

B. Sitemap URL Listesi

```
# Sitemap URL'leri
sitemap_urls = [
    "https://radiopaedia.org/sitemap-articles_1.xml",
    "https://radiopaedia.org/sitemap-articles_2.xml",
    "https://radiopaedia.org/sitemap-articles_3.xml",
    "https://radiopaedia.org/sitemap-articles_4.xml",
    "https://radiopaedia.org/sitemap-articles_5.xml",
```

Görsel 1. İşlenecek tüm sitemap URL'leri bir liste olarak tanımlanmıştır.

C. Sitemap URL 'lerin İşlenmesi

```
# URL'leri işleme alma
for sitemap_url in sitemap_urls:
    print(f"İşlem başlatılıyor: {sitemap_url}")
    process_sitemap(sitemap_url)
```

Görsel 2. process_sitemap() fonksiyonu her bir URL için çağrılır.

III. VERİ KAZIMA İLE OPENİ ÜZERİNDEN VERİ ÇEKME

OpenI sitesinden veri çekmek için Selenium ile bir otomasyon kodu geliştirilmiştir. Bu kod, tıbbi görüntülerin güvenilir bir şekilde toplanmasını sağlamak amacıyla optimize edilmiştir.

A. Sayfa Tarama

Selenium, her bir sayfayı ziyaret ederek görselleri otomatik olarak algılamıştır. Sayfa yüklenme süreleri dikkate alınarak kodda bekleme süreleri eklenmiştir.

```
# Sayfadaki görselleri bul
images = driver.find_elements(By.CSS_SELECTOR, value="img")
print(f"Sayfa {current_page_start // 100 + 1} - Bulunan görsel sayısı: {len(images)}")
```

Görsel 3.

B. Görsellerin Kaydedilmesi

Requests kütüphanesi kullanılarak her bir görsel, projede tanımlanan bir klasöre kaydedilmiştir. İndirme işlemi sırasında bağlantı hataları veya eksik veri durumları kontrol edilmiştir. Başarıyla kaydedilen görseller, belirli bir adlandırma şeması ile saklanmıştır. Bu işlem, veri organizasyonunu kolaylaştırmıştır.

IV. GÖRÜNTÜ VERİLERİNİ İŞLEME

Görüntü işleme için **OpenCV** (Open Source Computer Vision Library) kullanılmıştır. OpenCV, görüntü işleme ve bilgisayarla görme alanında yaygın olarak kullanılan açık kaynaklı bir kütüphanedir. Görüntüleri yeniden boyutlandırma, parlaklık ve kontrast ayarlamaları yapma gibi temel işlemler için güçlü fonksiyonlar sunar. Ayrıca, bu kütüphane, yüksek verimli ve hızlı işlem yapma kapasitesiyle büyük veri setleri üzerinde çalışmayı mümkün kılar. OpenCV, görüntüleri doğru formatta işleyerek, derin öğrenme modellerine uygun hale getirmek için ideal bir araçtır.

A. Python Kodlama Kısım

İlk olarak görseller üzerinde işlem yapabilmek için gerekli kütüphaneler aktarılmaktadır. Verilerin işlenmesi için kullanılacak olan parametreler tanımlanmıştır.

```
if not os.path.exists(output_folder):
    os.makedirs(output_folder)
```

Görsel 4.

Bu blok, çıktı görsellerinin kaydedileceği klasörün var olup olmadığını kontrol eder. Eğer klasör mevcut değilse, os.makedirs() fonksiyonu ile belirtilen output_folder adıyla yeni bir klasör oluşturulur.

```
def normalize_brightness_contrast(image, target_brightness, target_contrast):
    # Görüntüyü normalize etmek için ortalama ve standart sapmasını hesapla
    mean, stddev = cv2.meanStdDev(image)

    # Parlaklık ayarı
    brightness_adjusted = image - mean[0][0] + target_brightness

    # Kontrast ayarı
    contrast_factor = target_contrast / (stddev[0][0] + 1e-5) # Sıfıra bölünmeyi önlemek için küçük bir sayı ekledik
    normalized_image = cv2.convertScaleAbs(brightness_adjusted, alpha=contrast_factor, beta=0)

    return normalized_image
```

Görsel 5.

Bu fonksiyon, verilen görüntünün parlaklık ve kontrastını normalize etmek için kullanılır:

```
# Görüntüleri işleme
for filename in os.listdir(input_folder):
    img_path = os.path.join(input_folder, filename)
    img = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE) # Görselleri gri tonlamada açıyoruz

    if img is not None:
        # Yeniden boyutlandır
        resized_img = cv2.resize(img, target_size)

        # Parlaklık ve kontrastı normalize et
        normalized_img = normalize_brightness_contrast(resized_img, standard_brightness, standard_contrast)

        # Çıktı dosyasını kaydet
        output_path = os.path.join(output_folder, filename)
        cv2.imwrite(output_path, normalized_img)
        print(f"{filename} başarıyla işlendi ve kaydedildi.")
    else:
        print(f"{filename} yüklenemedi.")
```

Görsel 6. Görüntülerin işlenmesini ve işlenen görüntülerin kaydedilmesini sağlayan kod parçası.

V. MODEL EĞİTİMİ VE PERFORMANS ANALİZİ

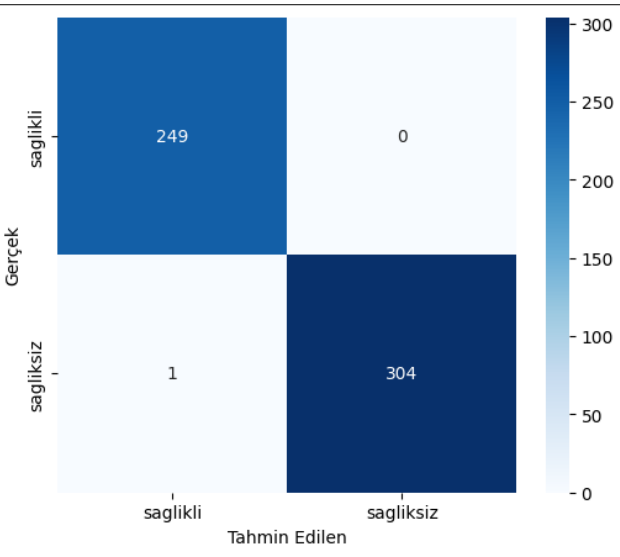
A. ViT Modeli ile Eğitim ve Performans Analizi

Bu çalışmada, akciğer MR görüntülerini kullanarak sağlıklı ve sağlıklısız ayrımı yapmak amacıyla Vision Transformer (ViT) modeli eğitilmiştir. Transformer tabanlı bir model olan ViT, görüntüleri bölümlere (patch) ayırarak, her bir bölümü bir diziye dönüştürüp self-attention mekanizması ile işleyerek sınıflandırma yapmaktadır. Model eğitimi dört epoch boyunca gerçekleştirilmiş ve her epoch sonunda loss değerlerinde önemli bir azalma gözlemlenmiştir.

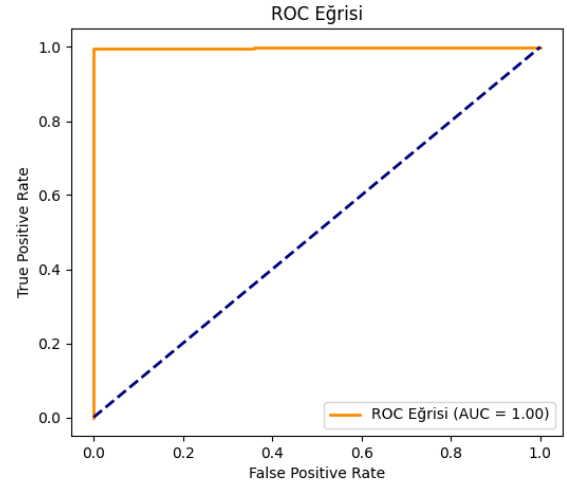
Eğitim sürecinde loss değerlerinin başlangıçta 0.6492 olduğu, ancak ilerleyen adımlarda kayda değer şekilde düştüğü görülmüştür. İlk epoch sonunda loss değeri 0.0716 seviyesine inerken, ikinci epoch sonunda bu değer 0.0062'ye kadar gerilemiştir. Üçüncü ve dördüncü epoch'larda loss değerleri daha da azalarak sırasıyla 0.0041 ve 0.0030 seviyelerine ulaşmıştır. Eğitim tamamlandıktan sonra modelin doğruluk ve hata analizleri için bir konfizyon matrisi oluşturulmuştur.

Sonuçlara göre model 249 sağlıklı örneği doğru sınıflandırmıştır ve bu kategoride hata yapmamıştır. Model, 305 hasta vakasından 304'ünü doğru tahmin ederken yalnızca 1 hatalı sınıflandırma yapmıştır. Bu sonuçlar, modelin yüksek doğruluk oranına sahip olduğunu ve başarılı bir şekilde sınıflandırma yaptığını göstermektedir. ViT modeli, eğitim süresi açısından diğer transformer tabanlı modellere kıyasla daha uzun bir zaman gerektirmiştir. Özellikle CvT (Convolutional Vision Transformer) modeli ile karşılaştırıldığında, ViT'nin eğitim süresinin oldukça fazla olduğu gözlemlenmiştir. Bunun temel sebebi, ViT'nin tüm görüntüyü doğrudan self-attention mekanizmasıyla işlemesi ve bu işlemin yüksek hesaplama maliyeti gerektirmesidir.

Sonuç olarak, ViT modeli akciğer MR görüntüleri üzerinde yüksek doğrulukla sınıflandırma yapabilen güçlü bir model olarak öne çıkmaktadır. Ancak, uzun eğitim süresi ve yüksek hesaplama maliyeti nedeniyle pratik uygulamalarda daha optimize çözümlerin değerlendirilmesi gerekebilir.



Görsel 7. ViT modelinin doğruluk matrisi.



Görsel 8. ViT modelinin ROC eğrisi

B. CvT Modeli ile Eğitim ve Performans Analizi

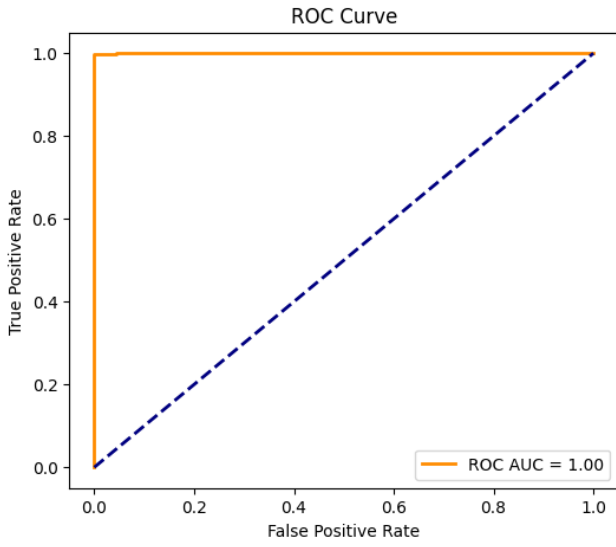
Convolutional vision Transformer (CvT), geleneksel evrişimsel sinir ağları (CNN) ile transformer mimarisinin birleşiminden oluşan bir modeldir. CvT, görüntüleri işlerken önce evrişim katmanlarını kullanarak düşük seviyeli özellikleri öğrenir ve ardından transformer yapısı ile uzun menzilli bağımlılıkları modelleyerek daha iyi genelleme yapmayı hedefler. Bu modelde, 3 adet evrişim katmanı kullanılarak görüntülerin özellikleri çıkarılmış ve ardından tam bağlı bir katman (FC layer) kullanılarak sınıflandırma gerçekleştirilmiştir.

Model, 224x224 boyutlarına yeniden boyutlandırılmış görüntülerle eğitilmiş ve iki sınıflı (sağlıklı/hastalıklı) sınıflandırma yapacak şekilde tasarlanmıştır. Eğitim sürecinde Adam optimizasyon algoritması ve Çapraz Entropi (CrossEntropyLoss) kayıp fonksiyonu kullanılmıştır. Modelin öğrenme oranı $1e-4$ olarak belirlenmiş ve 4 epoch boyunca eğitilmiştir.

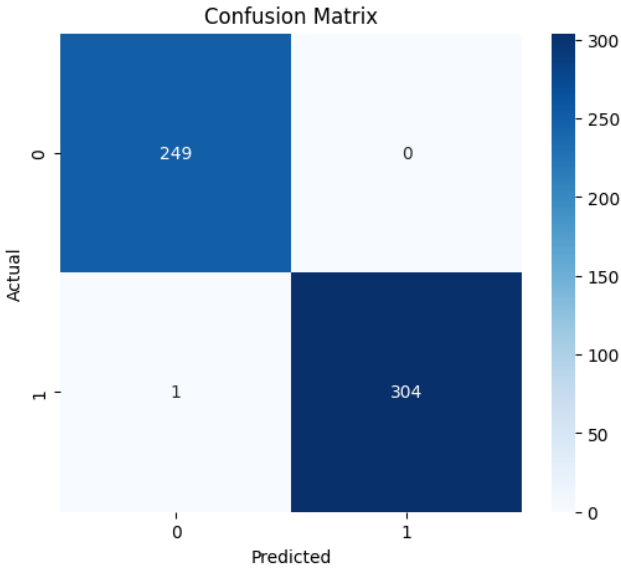
Eğitim sırasında, modelin kayıp değeri hızla düşerek kısa sürede düşük hata oranına ulaşmıştır. İlk epoch'ta kayıp değeri 0.68 seviyesinden 0.06 seviyesine kadar düşerken, ikinci epoch'tan itibaren neredeyse sıfıra yaklaşmıştır. Bu, modelin hızlı şekilde eğitim verisini ezberleyebildiğini ve muhtemel bir aşırı öğrenme (overfitting) riskini gösterdiği için dikkat edilmesi gereken bir durumdur.

Modelin değerlendirilmesi sırasında, doğruluk (accuracy), hassasiyet (recall/sensitivity), özgüllük (specificity), F1 skoru ve ROC eğrisi (AUC) gibi metrikler kullanılmıştır. Sonuçlara bakıldığında, modelin doğruluk oranı oldukça yüksek çıkmış ve AUC değeri 1.00 olarak hesaplanmıştır. Karmaşıklık matrisi de modelin tüm test verilerini neredeyse hatasız bir şekilde sınıflandırdığını göstermektedir.

Ancak, modelin doğrulama verisi üzerindeki başarısı dikkatle incelenmelidir. Eğitim kaybının hızla sıfıra yaklaşması, modelin aşırı öğrenmeye meyilli olabileceğini düşündürmektedir. Bu nedenle, veri artırma (data augmentation) yöntemleriyle modelin genelleştirme kapasitesini artırmak ve aşırı öğrenme riskini azaltmak gerekmektedir.



Görsel 9 CvT modelinin ROC eğrisi.



Görsel 10. CvT modelinin doğruluk matrisi.

C. Swin Modeli ile Eğitim ve Performans Analizi

Swin Transformer, geleneksel Vision Transformer (ViT) modellerinin yüksek hesaplama maliyetine çözüm olarak geliştirilmiş, hiyerarşik yapı ve kayan pencere (shifted window) mekanizması kullanan bir transformatör modelidir. Modelin en önemli avantajlarından biri, lokal ve global ilişkileri verimli bir şekilde öğrenebilmesi ve daha büyük görüntülerde performansını koruyabilmesidir.

Kayan pencere mekanizması, görüntüyü küçük bölgelere ayırarak işlem yapar ve daha sonra bu bölgeleri kaydırarak farklı perspektiflerden bağlamsal bilgiler yakalar. Böylece, model hem hesaplama maliyetini azaltır hem de görüntüdeki farklı ölçeklerdeki detayları öğrenebilir. Swin Transformer'ın bu özellikleri, özellikle tıbbi görüntü işleme ve sınıflandırma gibi yüksek çözünürlüklü veri gerektiren görevlerde büyük avantaj sağlamaktadır.

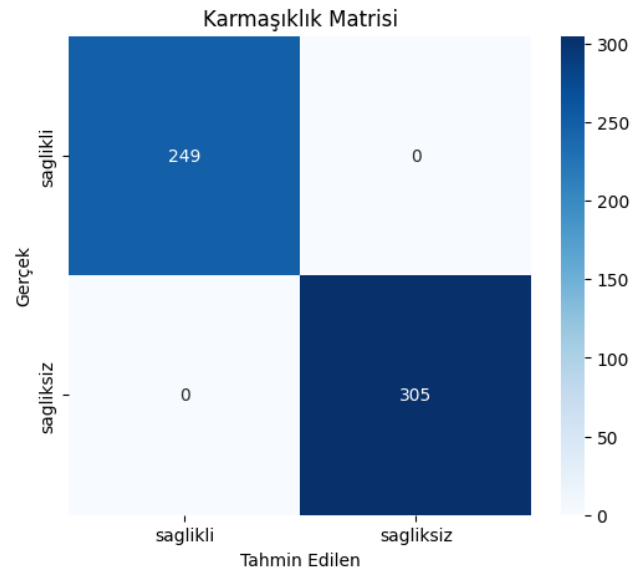
Model, akciğer MR görüntülerini sağlıklı ve sağlıklı olmayan olarak iki sınıfa ayırmak amacıyla eğitilmiştir. Eğitim sürecinde ImageNet ile önceden eğitilmiş "swin-tiny-patch4-window7-224" modeli kullanılmış ve son katman iki sınıflı sınıflandırmaya uygun hale getirilmiştir.

Eğitimde kullanılan optimizasyon algoritması, AdamW olup, $1e-5$ gibi düşük bir öğrenme oranı ile modelin parametreleri güncellenmiştir. Eğitim süreci boyunca, modelin kaybı (loss) her epoch sonunda hızlı bir şekilde azalmış ve 10. epochta kayıp değeri neredeyse sıfıra düşmüştür. Bu hızlı ve etkili öğrenme, modelin veri setine yüksek uyum sağladığını göstermektedir. Ancak bu durum, aşırı öğrenme (overfitting) riski taşıyabilir, çünkü model çok iyi bir şekilde eğitim verisine uyum sağlamıştır.

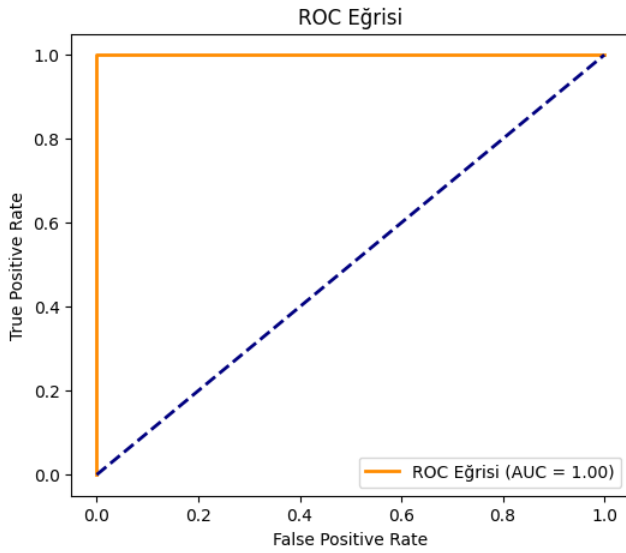
Eğitim süreci tamamlandıktan sonra, model doğrulama (validation) veri seti üzerinde test edilmiştir. Modelin sınıflandırma performansı oldukça yüksek olmuştur. Gerçek ve tahmin edilen sınıflar arasındaki uyum çok iyi seviyede bulunmuş ve modelin doğruluğu (%100) sağlanmıştır. Özellikle sağlıklı ve sağlıklı olmayan sınıfları arasında her iki sınıf için de precision, recall ve F1 skorları 1.00 olarak hesaplanmıştır. Bu, modelin her iki sınıfı da doğru bir şekilde sınıflandırdığını ve hiçbir hata yapmadığını göstermektedir.

Modelin AUC (Area Under Curve) değeri de 1.0 olarak hesaplanmıştır. Bu, modelin sınıflandırma performansının mükemmel olduğunu ve doğrusal olmayan sınırları çok iyi şekilde ayırabildiğini ortaya koymaktadır.

Swin Transformer, akciğer MR görüntülerinin sınıflandırılmasında olağanüstü bir performans sergilemiştir. Modelin doğruluk oranı %100'e ulaşmış ve tüm metriklerde maksimum başarı sağlanmıştır. Ancak, bu tür mükemmel sonuçlar genellikle aşırı öğrenme (overfitting) riski taşır. Modelin eğitildiği veri setine aşırı uyum sağlamış olması, farklı veri kümelerinde genelleme başarısı test edilmelidir.



Görsel 11. Swin modelinin doğruluk matrisi.



Görsel 12. Swin modelinin ROC eğrisi.

Sınıf	Precision	Recall	F1-Score	Destek (Support)
Sağlıklı	1.00	1.00	1.00	249
Sağlıksız	1.00	1.00	1.00	305
Genel	1.00	1.00	1.00	554

Tablo 1. Swin modelinin sınıflandırma performansı.

D. BEiT Modeli ile Eğitim ve Performans Analizi

Model, doğal dil işleme (NLP) alanındaki BERT modelinin görsel veri üzerinde uygulanmış bir versiyonudur ve resimlerdeki özellikleri anlamak için dönüştürme (transformer) mimarisini kullanır. BEiT, özellikle dil modellerinde kullanılan maskelenmiş dil modelleme (Masked Image Modeling, MIM) yaklaşımını kullanarak, görsel veriler üzerinde de etkili sonuçlar elde etmeyi amaçlar.

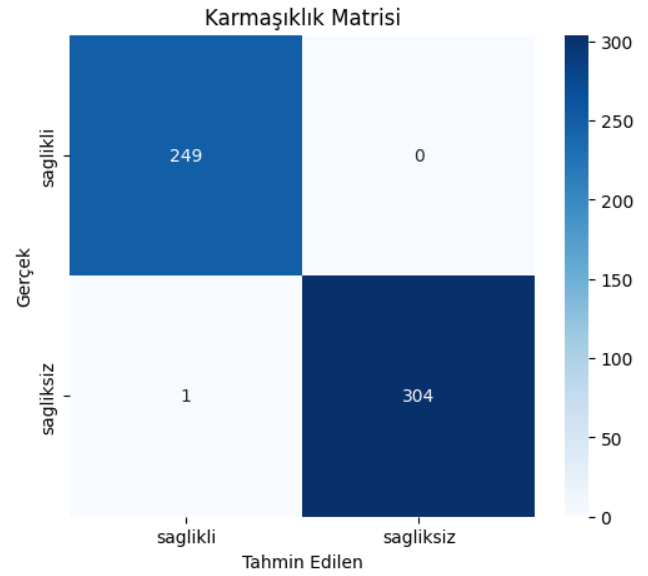
BEiT modeli, microsoft/beit-base-patch16-224 önceden eğitilmiş ağı kullanarak sınıflandırma yapılmıştır. BEiT, geleneksel konvolüsyonel ağlar (CNN) yerine Transformer tabanlı bir yapı kullanır. Görsel veriler üzerinde yapılan her bir model eğitimi, görseli sabit boyutlu parçalara ayırarak, her bir parçayı dil modellerinde olduğu gibi maskeler ve ardından bu maskelenmiş parçalar üzerinde anlamlı temsiller öğrenmeye çalışır. Bu yaklaşım, görsel verileri anlamak için daha zengin özellikler çıkarılmasını sağlar. BEiT'in özelleştirilmiş yapısı, görsel özellikleri bağlamsal bir şekilde öğrenmesini ve bu sayede çok daha güçlü sınıflandırmalar yapabilmesini sağlar.

BEiT modelinin eğitiminde, AdamW optimizör ve CrossEntropyLoss kayıp fonksiyonu kullanılmıştır. Model, 10 epoch boyunca eğitilmiş ve her epoch sonunda modelin doğruluğu ile kaybı izlenmiştir. Eğitim sürecinde, modelin doğruluk oranı hızla arttı ve eğitim sonrasında %100'e yakın bir doğruluk oranına ulaşılmıştır.

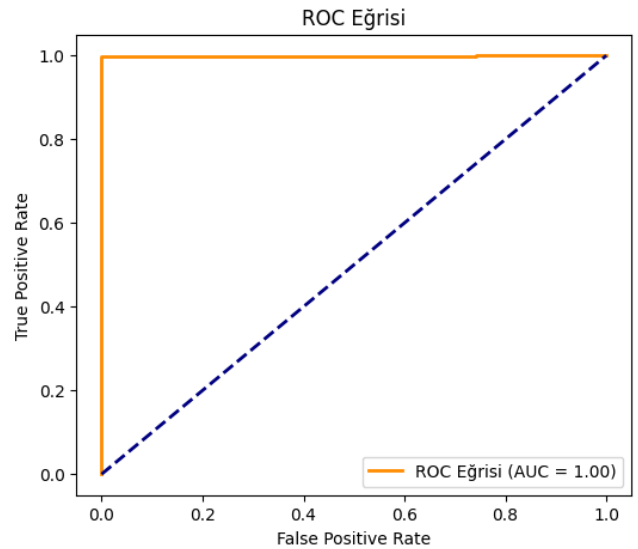
Her bir batch işleminde model, görsellerin sınıflandırılması için çıktılar üretmiş ve bu çıktılar üzerinden kayıp fonksiyonu hesaplanmıştır. Eğitim boyunca modelin çıktısı ile gerçek etiketler arasındaki fark, arka yayılım (backpropagation) yöntemiyle minimize edilmiştir.

BEiT modeli, eğitim sürecinde oldukça başarılı bir performans sergilemiş, %100 doğruluk oranına ulaşmıştır. Modelin kullandığı Transformer tabanlı mimari, görsel verilerdeki bağlamsal anlamada büyük bir avantaj sağlamış, bu da yüksek başarı oranlarına yansımıştır. Model, doğrulama setinde yaptığı sınıflandırmalarla, görsel özellikleri çok iyi bir şekilde öğrenmiş ve doğru sınıflandırmalar yapmıştır.

BEiT, geleneksel CNN yaklaşımlarına kıyasla çok daha güçlü bağlamsal özellik çıkarımları yapabilen bir modeldir. Bu nedenle, görsel sınıflandırma gibi görevlerde daha derinlemesine ve anlamlı temsiller öğrenmesi sayesinde yüksek başarı göstermektedir.



Görsel 13. BEiT modelinin doğruluk matrisi.



Görsel 14. BEiT modelinin ROC eğrisi.

VI. SONUÇ

Bu çalışmada, Radiopaedia ve OpenI sitelerinden akciğerle ilgili görsellerin otomatik olarak toplanması, ön işlenmesi ve veri setinin oluşturulması başarıyla gerçekleştirilmiştir. OpenI sitesinden sağlıklı akciğer görüntülerinin temininde kritik rol oynamıştır. Kullanılan Python kodları, Selenium ve requests kütüphaneleri aracılığıyla belirlenen bağlantıları tarayarak uygun görselleri indirip parlaklık, kontrast ayarı ve yeniden boyutlandırma gibi işlemlerle standardize etmiştir. Bu görseller, makine öğrenmesi modeline uygun hale getirilerek akciğer hastalıklarının tespit edilmesi üzerine eğitilecek bir veri kümesi oluşturulmuştur. Çalışmanın sonucunda elde edilen veri seti, modelin performansını artıracak nitelikte temiz ve homojen hale getirilmiştir.

Bu çalışmada, Swin, CvT, ViT ve BEiT modellerinin görsel sınıflandırma görevlerindeki performansları karşılaştırıldı. Her biri Transformer tabanlı yaklaşımlar kullanırken, farklı mimari özellikler gösteriyor.

Swin Transformer, yerel ve global özellikleri birleştirerek büyük görsellerde başarılı performans sergileyebilir, ancak eğitim süresi uzundur ve overfitting riski oluşturabilir.

CvT, CNN ve Transformer birleşimiyle düşük seviyeli özellikleri hızla öğrenir, ancak büyük veri kümesiyle eğitilmezse overfitting yaşanabilir. ViT, büyük veri kümesi gereksinimi duyar, ancak küçük veri setlerinde overfitting problemi yaşayabilir. BEiT, ViT'in yapısını geliştirerek görsel verilerde bağlamsal anlamayı iyileştirir, ancak yine büyük veri kümesine ihtiyaç duyar ve overfitting riski taşır.

Her model, belirli bir başarı gösterdi, ancak daha büyük ve dengeli veri setlerinde daha verimli sonuçlar elde edilebilir. Genel olarak, Transformer tabanlı modeller büyük veri setlerine ve doğru düzenleme tekniklerine ihtiyaç duyar. Overfitting problemi çoğu modelde görülebilir, ancak her modelin güçlü ve zayıf yönleri vardır. Eğitim sürecinde dikkatli optimizasyon ve veri artırma yöntemleri bu sorunu azaltabilir.

VII. Kaynakça

- [1] Dosovitskiy, A., et al. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR. <https://arxiv.org/abs/2010.11929>
- [2] Liu, Z., et al. (2021). Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. ICCV. <https://arxiv.org/abs/2103.14030>
- [3] Wu, H., et al. (2021). CvT: Introducing Convolutions to Vision Transformers. ICCV. <https://arxiv.org/abs/2103.15808>
- [4] Wang, W., et al. (2021). Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. ICCV. <https://arxiv.org/abs/2102.12122>
- [5] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. arXiv preprint arXiv:2103.14030.
- [6] Bao, H., Dong, L., Piao, S., & Wei, F. (2021). BEiT: BERT Pre-Training of Image Transformers. arXiv preprint arXiv:2106.08254.
- [7] Huynh, E. (2022). Vision Transformers in 2022: An Update on Tiny ImageNet. arXiv preprint arXiv:2205.10660.

