



**ALZHEIMER HASTALIĞININ TEŞHİSİ İÇİN DEMXNET MODEL ÖNERİSİ
VE DERİN TRANSFER ÖĞRENME YÖNTEMLERİ İLE
KARŞILAŞTIRILMASI**

(Yüksek Lisans Tezi)

Naciye Nur ARSLAN

Kütahya - 2022

T.C.
KÜTAHYA DUMLUPINAR ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
Bilgisayar Mühendisliği Anabilim Dalı

YÜKSEK LİSANS TEZİ

**ALZHEIMER HASTALIĞININ TEŞHİSİ İÇİN DEMXNET
MODEL ÖNERİSİ VE DERİN TRANSFER ÖĞRENME
YÖNTEMLERİ İLE KARŞILAŞTIRILMASI**

Danışman:
Dr. Öğr. Üyesi Durmuş ÖZDEMİR

Hazırlayan:
Naciye Nur ARSLAN

Kütahya-2022

Kabul ve Onay

Lisansüstü Eğitim Enstitüsü Müdürlüğüne,

Bu çalışma jürimiz tarafından,

Bilgisayar Mühendisliği Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Tez Jürisi	İmza	
	Kabul	Ret
Dr. Öğr. Üyesi Durmuş ÖZDEMİR (Danışman)		
Prof. Dr. Doğan AYDIN		
Dr. Öğr. Üyesi Fırat AYDEMİR		

Onay

Doç. Dr. Arif KOLAY

Enstitü Müdürü

Bilimsel Etik Bildirimi

Yüksek Lisans tezi olarak hazırladığım “*Alzheimer Hastalığının Teşhisi için DEMXNET Model Önerisi ve Derin Transfer Öğrenme Yöntemleri ile Karşılaştırılması*” adlı çalışmanın öneri aşamasından sonuçlandığı aşamaya kadar geçen süreçte bilimsel etiğe ve akademik kurallara özenle uyduğumu, tez içindeki tüm bilgileri bilimsel ahlak ve gelenek çerçevesinde elde ettiğimi, tez yazım kurallarına uygun olarak hazırladığımı, bu çalışmamda doğrudan veya dolaylı olarak yaptığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu beyan ederim.

11/08/2022

Naciye Nur ARSLAN

Özgeçmiş

Lisans eğitimini Kütahya Dumlupınar Üniversitesi Bilgisayar Mühendisliği bölümünde 2018 yılında tamamladı. 3.sınıf güz döneminde ERASMUS programı aracılığıyla Portekiz Beira Interior Üniversitesinde eğitim aldı. 2019 yılında Kütahya Dumlupınar Üniversitesinde yüksek lisansa başladı. Üniversite lisans öğrenimi sırasında Havelsan A.Ş. ve Kütahya Belediyesinde stajlarını tamamladı. 2019'da Cevizsoft Teknoloji A.Ş.'de bilgisayar mühendisi olarak 2 yıl çalıştı. 2022 Mart ayından bu yana Kütahya Dumlupınar Üniversitesi Yazılım Mühendisliği bölümünde araştırma görevlisi olarak çalışmaktadır.



ÖZET

ALZHEIMER HASTALIĞININ TEŞHİSİ İÇİN DEMXNET MODEL ÖNERİSİ VE DERİN TRANSFER ÖĞRENME YÖNTEMLERİ İLE KARŞILAŞTIRILMASI

ARSLAN, Naciye Nur

Yüksek Lisans Tezi, Bilgisayar Mühendisliği Anabilim Dalı
Tez Danışmanı: Dr. Öğr. Üyesi Durmuş ÖZDEMİR
Ağustos, 2022, 84 sayfa

Demans, biyolojik yaşlanmanın olağan sonuçlarının ötesinde bilişsel işlevlerde bozulmanın görüldüğü bir sendromdur. Demansın en yaygın görülen türü Alzheimer hastalığıdır. Gözle görülür semptomlar ortaya çıkmadan şiddetlenir ve tedavisi mümkün olmayan beyin rahatsızlığına neden olur. Bu nedenle teşhis ne kadar erken olursa hasta için önleyici faaliyetler ve tedavi başlatılabilir. Mini-Zihinsel Durum Testleri Muayenesi gibi testler genellikle erken teşhis için kullanılsa da teşhis manyetik rezonans görüntüleme (MRI) beyin analizine dayanır. Bu tez çalışmasında, Alzheimer tipi demans hastalığının teşhisi ve sınıflandırılması için derin öğrenme modellerinin performansları karşılaştırılmış ve ayrıca DEMXNET adlı yeni bir derin öğrenme tabanlı model önerilmiştir. Alzheimer hastalığının 4 sınıfını içeren MRI görüntüleri veri seti Kaggle platformundan elde edilmiştir. Veri setindeki sınıflarda görüntü sayısının dengesiz dağılımından dolayı Sentetik Azınlık Aşırı Örnekleme Tekniği (SMOTE) kullanılmıştır. Modellerin tamamı SMOTE’li ve SMOTE’siz olarak iki senaryoda eğitilerek test edilmiştir. Tüm modellerde SMOTE tekniğinin doğruluk oranlarını önemli ölçüde artırdığı görülmüştür. Alzheimer tipi demans hastalığının sınıflandırılmasında kullanılan SMOTE tekniği uygulanan modellerin doğruluk (accuracy) oranı performans sonuçları VGG16 (%92.89), VGG19 (%95.94), InceptionV3 (%89.65), Xception (%91.05), DenseNet201(%95.47) ve önerilen modelimiz DEMXNET için (%96.48) olarak elde edilmiştir. Önerilen DEMXNET modeli karşılaştırılan diğer modellerden daha başarılı olmuştur. Önerilen DEMXNET modeli özellikle dengesiz veri setleri için verimli bir şekilde kullanılabilir ve farklı veri setlerine de uyarlanabilir.

Anahtar Kelimeler: Alzheimer, Demans, Derin Öğrenme, MRI, Transfer Öğrenme

ABSTRACT**MODEL PROPOSAL OF DEMXNET FOR THE DIAGNOSIS OF
ALZHEIMER'S DISEASE AND COMPARISON WITH DEEP TRANSFER
LEARNING METHODS****ARSLAN, Naciye Nur****Master Thesis, Computer Engineering Department****Thesis Advisor: Dr. Lecturer Durmuş ÖZDEMİR****August, 2022, 84 pages**

Dementia is a syndrome in which cognitive functions are impaired beyond the usual consequences of biological aging. The most common type of dementia is Alzheimer's disease. It exacerbates before visible symptoms occur and cause an incurable brain condition. Therefore, the sooner the diagnosis is made, the more preventive activities and treatment can be started for the patient. Diagnosis is based on magnetic resonance imaging (MRI) brain analysis, although tests such as the Mini-Mental Status Exam are often used for early diagnosis. In this thesis, the performances of deep learning models for the diagnosis and classification of Alzheimer's dementia disease are compared, and a new deep learning-based model called DEMXNET is proposed. The dataset of MRI images containing four classes of Alzheimer's disease was obtained from the Kaggle platform. Due to the uneven distribution of the number of images in the classes in the data set, Synthetic Minority Oversampling Technique (SMOTE) was used. All models were trained and tested in two scenarios, with and without SMOTE. It was observed that the accuracy rates of the SMOTE technique increased significantly in all models. Accuracy rate performance results of the models with the SMOTE technique used in the classification of Alzheimer's type dementia disease, VGG16 (92.89%), VGG19 (95.94%), InceptionV3 (89.65%), Xception (91.05%), DenseNet201 (95.47%) and our proposed model DEMXNET for (96.48%). The proposed DEMXNET model was more successful than the other models compared. The proposed DEMXNET model can be used efficiently, especially for unbalanced datasets, and can be adapted to different datasets.

Keywords: Alzheimer's, Dementia, Deep Learning, MRI, Transfer Learning

Ön Söz

Bu Yüksek Lisans Tez çalışma raporumda Alzheimer hastalığının teşhisi ve sınıflandırılması için MR görüntüleri kullanarak beş derin transfer öğrenme yöntemi ile önerdiğim DEMXNET modelini karşılaştırarak dikkatinize sunmaktayım.

Yüksek Lisans Tez çalışma sürecim boyunca yardımlarını esirgemeyen ve bana çok büyük katkılar sağlayan; Danışman Hocam Dr. Öğr. Üyesi Durmuş ÖZDEMİR'e sonsuz şükranlarımı sunarım.

Her zaman yanımda olan değerli eşim Sinan ARSLAN ve varlığıyla hayatımıza renk katan biricik kızım Eylül ARSLAN'a sonsuz teşekkür ederim. Son olarak eğitim hayatım boyunca dua ve destekleri ile yanımda olan babam ve annem; Ahmet ve Yasemin SARIBEY'e minnet ve şükranlarımı sunarım.

İÇİNDEKİLER

ÖZET	v
ABSTRACT	vi
SİMGELER VE KISALTMALAR	xi
TABLOLAR LİSTESİ.....	xii
ŞEKİLLER LİSTESİ.....	xiii
GİRİŞ	1

BİRİNCİ BÖLÜM YAPAY SİNİR AĞLARI

1.1. YAPAY SİNİR AĞLARININ TARİHÇESİ.....	9
1.2. YAPAY SİNİR AĞLARININ YAPISI.....	10
1.3. AŞIRI ÖĞRENME (OVERFITTING) VE EKSİK ÖĞRENME (UNDERFITTING) KAVRAMLARI	12
1.4. GERİ YAYILIM (BACKPROPAGATION) ALGORİTMASI.....	13
1.5. KAYIP (LOSS) FONKSİYONLARI.....	13
1.6. OPTİMİZASYON ALGORİTMALARI	13
1.7. YAPAY SİNİR AĞLARDA ÖĞRENME	15
1.7.1. Denetimli Öğrenme (Supervised Learning)	15
1.7.2 Denetimsiz Öğrenme (Unsupervised Learning).....	15
1.7.3 Yarı Denetimli Öğrenme (Semi-Supervised Learning).....	15
1.7.4 Takviyeli Öğrenme (Reinforcement Learning).....	15

İKİNCİ BÖLÜM DERİN ÖĞRENME

2.1. DERİN ÖĞRENME KAVRAMI.....	17
2.2. EVRİŞİMSEL SİNİR AĞLARI (CONVOLUTIONAL NEURAL NETWORKS – CNNs).....	18
2.3. EVRİŞİMSEL SİNİR AĞLARI KATMANLARI	19
2.3.1. Evrişim Katmanı (Convolution Layer)	19
2.3.2. Aktivasyon Katmanı (Activation Layer).....	20

2.3.3. Havuzlama (Pooling) Katmanı.....	20
2.3.4. Batch Normalization Katmanı.....	21
2.3.5. Bırakma Katmanı (Dropout Layer)	22
2.3.6. Düzleştirme Katmanı (Flatten Layer)	22
2.3.7. Tam Bağlantılı Katman (Fully Connected Layer/Dense).....	23
2.4. DERİN ÖĞRENMEDE HİPER-PARAMETRELER.....	24
2.5. TRANSFER ÖĞRENME.....	25
2.5.1. Transfer Öğrenme Modelleri.....	26
2.5.1.1. VGG16 ve VGG19	26
2.5.1.2. InceptionV3.....	27
2.5.1.3. Xception.....	28
2.5.1.4. DenseNet201	28

ÜÇÜNCÜ BÖLÜM

MATERYAL ve YÖNTEM

3.1. VERİ SETİ (DATASET).....	31
3.2. VERİ ARTIRMA (DATA AUGMENTATION).....	32
3.3. SENTETİK AZINLIK AŞIRI ÖRNEKLEME TEKNİĞİ (SMOTE- SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE)	32
3.4. VERİLERİN TRAIN, VALIDATION VE TEST SETLERİNE BÖLÜNMESİ	33
3.5. EĞİTİMDE KULLANILAN DONANIMLAR VE YAZILIMLAR	34
3.6. ÖNERİLEN YAKLAŞIM	34

DÖRDÜNCÜ BÖLÜM

BULGULAR VE DENEYSEL SONUÇLAR

4.1. DEĞERLENDİRME METRİKLERİ.....	39
4.2. MODELLERİN TEST SONUÇLARI.....	41
4.2.1. VGG16 Modeli.....	41

4.2.2. VGG19 Modeli.....	44
4.2.3. InceptionV3 Modeli	47
4.2.4. Xception Modeli.....	49
4.2.5. DenseNet201 Modeli.....	52
4.2.6. DEMXNET Modeli.....	55
4.3. MODELLERİN PERFORMANS ANALİZİ	57

BEŞİNCİ BÖLÜM

SONUÇLAR VE ÖNERİLER

5.1. SONUÇLAR VE ÖNERİLER	60
KAYNAKÇA	61
DİZİN	68

SİMGELER VE KISALTMALAR

AH	Alzheimer Hastalığı
YSA	Yapay Sinir Ağları
ANN	Artificial Neural Network
DL	Deep Learning
CNN	Convolutional Neural Network
SVM	Support Vector Machine
KNN	k-Nearest Neighborhood
NB	Naive Bayes
RF	Random Forest
DT	Decision Tree
BAG-DT	Bagged Decision Tree
LR	Lojistik Regresyon
MRI	Manyetik Rezonans Görüntüleme
CT	Bilgisayarlı Tomografi
SPECT	Tek Foton Emisyon Bilgisayarlı Tomografi
PET	Pozitron Emisyon Tomografi
MMSE	Mini-Zihinsel Durum Muayenesi
CDR	Clinical Dementia Rating
CDT	Clock Drawing Test
MID	Mild Demented
MOD	Moderate Demented
ND	Non Demented
VMD	Very Mild Demented
MAE	Mean Absolute Error
MSE	Mean Square Error
GD	Gradient Descent
SGD	Stochastic Gradient Descent
ADAM	Adaptive Moment Estimation
Adadelat	An Adaptive Learning Rate Method
SMOTE	Synthetic Minority Over-sampling Technique
ReLU	Rectified Linear Unit

TABLOLAR LİSTESİ

Sayfa

Tablo 3.1: Veri Setinin Dağılımı.....	31
Tablo 3.2: Veri Artırma Yöntemi için Kullanılan Parametreler	32
Tablo 3.3: Veri Setinin SMOTE Uygulandıktan Sonraki Dağılımı.....	32
Tablo 3.4: Hiper-Parametre Ayarları	35
Tablo 3.5: DEMXNET Model Detayları	37
Tablo 4.1: SMOTE’li VGG16 Modeli ile Test Verileri Kullanılarak Elde Edilen Sınıflandırma Raporu	43
Tablo 4.2: VGG16 Modelinin SMOTE’li ve SMOTE’siz Varyasyonlarının Karşılaştırılması	44
Tablo 4.3: SMOTE’li VGG19 Modeli ile Elde Edilen Sınıflandırma Raporu	46
Tablo 4.4: VGG19 Modelinin SMOTE’li ve SMOTE’siz Varyasyonlarının Karşılaştırılması	46
Tablo 4.5: SMOTE’li InceptionV3 Modeli ile Elde Edilen Sınıflandırma Raporu	49
Tablo 4.6: SMOTE’li ve SMOTE’siz Çalışılan InceptionV3 Modelinin Test Doğruluklarının Karşılaştırılması	49
Tablo 4.7: SMOTE’li Xception Modeli Test Sınıflandırma Raporu	52
Tablo 4.8: SMOTE’li ve SMOTE’siz Çalışılan Xception Modelinin Test Doğruluklarının Karşılaştırılması	52
Tablo 4.9: SMOTE’li DenseNet201 Modeli Test Sınıflandırma Raporu	54
Tablo 4.10: SMOTE’li ve SMOTE’siz Çalışılan DenseNet201 Modelinin Test Doğruluklarının Karşılaştırılması.....	54
Tablo 4.11: SMOTE’li DEMXNET Modeli Test Sınıflandırma Raporu	57
Tablo 4.12: SMOTE’li ve SMOTE’siz Çalışılan DEMXNET Modelinin Test Doğruluklarının Karşılaştırılması.....	57
Tablo 4.13: Modellerin Performans Analizi	58

ŞEKİLLER LİSTESİ

Sayfa

Şekil 1: Yapay Zeka, Makine Öğrenimi ve Derin Öğrenme Arasındaki İlişki.....	4
Şekil 1.1: Biyolojik ve Yapay Sinir Hücresi.....	10
Şekil 1.2: Temel YSA Modeli.....	11
Şekil 1.3: Yapay Sinir Ağlarının Sınıflandırılması.....	12
Şekil 2.1: Makine Öğrenimi ve Derin Öğrenme Arasındaki Fark.....	17
Şekil 2.2: CNN'nin Genel Mimarisi.....	18
Şekil 2.3: 5x5x3 Boyuttaki Giriş Görüntüsüne 3x3'lük Filtrenin Uygulandığı Evrişim İşlemi.....	19
Şekil 2.4: Yaygın Olarak Kullanılan Aktivasyon Fonksiyonları: (a) Rectified Linear Unit (ReLU), (b) Sigmoid ve (c) Hyperbolic Tangent (tanh).....	20
Şekil 2.5: Max Pooling ve Average Pooling İşlemi (Stride: 2).....	21
Şekil 2.6: (a) 2 Gizli Katmana Sahip Sinir Ağı, (b) Soldaki ağın Dropout Uygulanmış Hali.....	22
Şekil 2.7: Düzleştirme İşlemi.....	22
Şekil 2.8: Tam Bağlantılı Katman.....	23
Şekil 2.9: Transfer Öğrenme Yaklaşımı.....	25
Şekil 2.10: VGG16 Modeli.....	26
Şekil 2.11: InceptionV3 Modeli.....	27
Şekil 2.12.: Xception Modeli.....	28
Şekil 2.13: DenseNet Modeli.....	29
Şekil 3.1: Veri Setinden Örnek Görüntüler.....	31
Şekil 3.2: Veri Setinin Eğitim, Test ve Doğrulama Setlerine Bölünmesi.....	33
Şekil 3.3: Önerilen Transfer Öğrenme Yaklaşımı.....	36
Şekil 3.4: Önerilen DEMXNET Modeli.....	37
Şekil 4.1: Hata (Confusion) Matrisi.....	39
Şekil 4.2: SMOTE'li VGG16 Modelinin Accuracy Grafiği.....	41
Şekil 4.3: SMOTE'li VGG16 Modelinin AUC Grafiği.....	42
Şekil 4.4: SMOTE'li VGG16 Modelinin Loss Grafiği.....	42
Şekil 4.5: SMOTE'li VGG16 Modelinin Hata Matrisi.....	43
Şekil 4.6: SMOTE'li VGG19 Modelinin Accuracy Grafiği.....	44

Şekil 4.7: SMOTE’li VGG19 Modelinin AUC Grafiği.....	45
Şekil 4.8: SMOTE’li VGG19 Modelinin Loss Grafiği.....	45
Şekil 4.9: SMOTE’li VGG19 Modelinin Hata Matrisi.....	46
Şekil 4.10: SMOTE’li InceptionV3 Modelinin Accuracy Grafiği.....	47
Şekil 4.11: SMOTE’li InceptionV3 Modelinin AUC Grafiği.....	47
Şekil 4.12: SMOTE’li InceptionV3 Modelinin Loss Grafiği.....	48
Şekil 4.13: SMOTE’li InceptionV3 Modelinin Hata Matrisi.....	48
Şekil 4.14: SMOTE’li Xception Modelinin Accuracy Grafiği.....	50
Şekil 4.15: SMOTE’li Xception Modelinin AUC Grafiği.....	50
Şekil 4.16: SMOTE’li Xception Modelinin Loss Grafiği.....	51
Şekil 4.17: SMOTE’li Xception Modelinin Hata Matrisi.....	51
Şekil 4.18: SMOTE’li DenseNet201 Modelinin Accuracy Grafiği.....	52
Şekil 4.19: SMOTE’li DenseNet201 Modelinin AUC Grafiği.....	53
Şekil 4.20: SMOTE’li DenseNet201 Modelinin Loss Grafiği.....	53
Şekil 4.21: SMOTE’li DenseNet201 Modelinin Hata Matrisi.....	54
Şekil 4.22: SMOTE’li DEMXNET Modelinin Accuracy Grafiği.....	55
Şekil 4.23: SMOTE’li DEMXNET Modelinin AUC Grafiği.....	55
Şekil 4.24: SMOTE’li DEMXNET Modelinin Loss Grafiği.....	56
Şekil 4.25: SMOTE’li DEMXNET Modelinin Hata Matrisi.....	56



TEZ METNİ

GİRİŞ

100 yıldan fazla bir süre önce Alois Alzheimer, Güneybatı Alman Psikiyatristler Derneği'nin bir toplantısında 55 yaşındaki hasta Auguste Deter'in hafıza kaybı, halüsinasyonlar yaşadığı olağandışı beyin hastalığının klinik ve patolojik özelliklerini tanımladı (Cipriani, vd., 2011). Bu histopatolojik ve klinik tanı ilk olarak Emil Kraepelin tarafından Alzheimer hastalığı ve daha sonra Alzheimer tipi demans olarak belirlenmiştir (Lazarczyk, vd., 2012).

Demans, latince mens (zihin) kelimesinden türemiş olup zihnin yitirilmesi anlamına gelmektedir. Demans bir hastalık değildir, sendromdur ve farklı çeşitleri vardır. Demansın türü olan ve en sık görüleni Alzheimer hastalığıdır. Tahminlere göre Türkiye'de 800.000'e yakın demanslı hasta bulunmaktadır ve bu hastaların 3'te 2'sini Alzheimer hastaları oluşturmaktadır (Bilgiç, vd., 2020).

65 yaşından önce başlangıç (erken başlangıç) olağandışıdır ve Alzheimer hastalarının %10'undan azında görülür. Şu ana kadar Alzheimer hastalığının bilinen bir tedavisi yoktur, ancak bazı semptomları iyileştirebilecek tedavi yöntemleri mevcuttur.

Uluslararası Alzheimer Hastalığı Derneği'nin yayınladığı rapora göre 2018 yılında dünya üzerinde 50 milyon civarında olan demanslı hasta sayısının eğer tedavi ve önlem konusunda büyük bir gelişme olmazsa 2030 yılında 82 milyona, 2050'de 152 milyona yükselmesi beklenmektedir. Demans dünyada en sık beşinci ölüm nedenidir. Demans hastalarına harcanan para yıllık 1 trilyon dolara yaklaşmaktadır (Bilgiç, vd., 2020).

Alzheimer hastalığı klinik ve otopsi serilerindeki tüm demansların %60-70'inden sorumlu olduğu saptanmıştır. Yetmiş beş yaşına kadar yaşayan bir kişinin Alzheimer hastası olma ihtimali %10-12'dir (Bilgiç, vd., 2020).

Demansın karakteristik semptomları, kişinin günlük aktivitelerini gerçekleştirme yeteneğini etkileyen hafıza, dil, problem çözme ve diğer bilişsel becerilerdeki zorluklardır (Knopman, vd., 2021). Hastalığın tanısı için ayırıcı tanı testi olmadığı için tanı için birçok test birlikte değerlendirilir. Hastanın ayrıntılı bir tıbbi öyküsü nörologlar tarafından alınır. Hastaların nörolojik fonksiyonlarını ölçen çeşitli tarama testleri, kan testleri ve gen taramaları yapılmaktadır. Ayrıca beyindeki doku kaybı

hakkında bilgi almak için Manyetik Rezonans Görüntüleme (MRI), Bilgisayarlı Tomografi (CT), Pozitron Emisyon Tomografi (PET) ve Foton Emisyon Bilgisayarlı Tomografi (SPECT) tekniklerinden elde edilen görüntüler kullanılmaktadır (Yigit & Isik, 2018).

Vakaların büyük çoğunluğunda hastalık, risk faktörlerinin bir kombinasyonu nedeniyle ortaya çıkar. Yaşlanma ana faktördür. Kardiyovasküler hastalıkların risk faktörlerinin de (hipertansiyon, yüksek kolesterol, obezite, diyabet vb.) gelişimine katkı sağladığı görülmektedir (Baumgart, vd., 2015). Bazı durumlarda enfeksiyon veya toksik ürünlere maruz kalmanın da rol oynaması mümkündür, ancak kesin bir kanıt elde edilmemiştir. Genetik faktörler de hastalığın başlangıcında önemli bir rol oynamaktadır. Bu nedenle, belirli genler, doğrudan hastalığın nedeni olmasalar da, bu hastalığa yakalanma riskini artırabilir. Genel popülasyonun yaklaşık %25'i, geç başlangıçlı Alzheimer hastalığı için en güçlü genetik risk faktörü olan Apolipoprotein E'nin (APOE ε4) en az bir ε4 alelini taşır (Gharbi-Meliani, vd., 2021). Bununla birlikte, birçok kişi bu genleri taşır ve asla hastalığa yakalanmaz ve tersine, bu genlere sahip olmayan bazı kişilerde hastalık gelişebilir. Hastalığın kalıtsal formları da vardır, ancak vakaların %5'inden azını oluşturur. Ailesel belirtiler erken, bazen 40 yaşından önce ortaya çıkar. Bununla birlikte, aynı ailenin birkaç üyesi bu hastalıktan etkilense de bu mutlaka kalıtsal bir form olduğu anlamına gelmez.

Alzheimer'lı demans genel olarak aşağıdaki tiplere ayrılmaktadır (Murugan, vd., 2021):

- Hafif Bilişsel Bozukluk: Günlük aktiviteleri yerine getirebilirler. Genel bir bellek bozukluğu görülmeye başlanmıştır. Fakat bu aşamada sadece nöropsikolojik değerlendirme ile hastalık saptanabilecek durumdadır.

- Hafif Demans: Belirtiler arasında hafıza eksikliği, belirsizlik, kişilik değişiklikleri, kaybolma ve rutin görevleri yerine getirmede zorluklar yer alır.

- Orta Derecede Demans: Hastanın ekstra bakım ve desteğe ihtiyaç duyduğu günlük yaşam tarzı çok karmaşık hale gelir. İnsanlar saçlarını taramak için bile daha fazla yardıma ihtiyaç duyabilirler. Ayrıca önemli kişilik değişiklikleri sergileyebilirler; örneğin, sebepsiz yere paranoyaklaşır veya sinirlenirler. Uyku bozukluklarının da ortaya çıkması muhtemeldir.

- Şiddetli Demans: Bu aşamada semptomlar kötüleşebilir. Bu hastalar iletişim kurma kapasitesinden yoksun olabilir ve kişi için tam zamanlı tedavi gerekebilir. Kişinin mesane kontrolü kaybolabilir ve başlarını normal pozisyonda tutmak ve sandalyede oturmak gibi küçük aktiviteleri bile gerçekleştirmeleri imkansızdır.

1975'te tanıtılan Mini-Zihinsel Durum Muayenesi (MMSE) sınavı, hastanın bilişsel yeteneğinin altı farklı yöntemini inceleyen en yaygın kullanılan ankettir. Özel bir eğitim veya ekipman gerektirmediği için yaygın olarak kullanılmaktadır ve AD ilerlemesinin uzunlamasına izlenmesi için güvenilir bir şekilde kullanılır. Dezavantajı ise demografik, yaş ve eğitim faktörlerinden etkilenmesi ve şiddetli AD durumunda ilerlemeyi ölçme duyarlılığından yoksun olmasıdır. Maksimum 30 puan ile sonuçlar normal kognisyon, hastalığın hafif, orta ve ağır evreleri olarak değerlendirilebilir (Yigit ve Isik, 2018).

1982 yılında tanıtılan Klinik Demans Derecelendirme (CDR – Clinical Dementia Rating), demansı ölçmek için altı farklı bilişsel ve davranışsal alanda elde edilen ve 0 ile 3 arasında bir değer elde etmek üzere birleştirilen puanlamaya dayalı alternatif yapılandırılmış bir görüşmedir. Puan değerlerinin anlamları şu şekildedir: 0-bilişsel normal, 0,5-çok hafif veya şüpheli demans, 1-hafif demans, 2-orta demans ve 3-ağır demans (Williams, Roe ve Morris, 2009). Erken veya orta dereceli AH'yi belirlemede duyarlılığı olmamasına rağmen, en çok kullanılan testlerden bir diğeri olan Saat Çizimi testidir (CDT- Clock Drawing test). Bu test ile uzun süreli bellek, zihinsel yoğunlaşma, soyut düşünme, görsel-mekansal işlevler ve planlama yetileri değerlendirilir (Yaman, 2015).

Alzheimer hastalığının teşhisinde ve sınıflandırılmasında MRI tekniği doktorlar tarafından yaygın olarak kullanılmaktadır. Hastalığın teşhisi ve sınıflandırılması uzman deneyimine ve bilgisine dayanmaktadır. Bu nedenle insan faktörüne dayalı hataların en aza indirilmesi amacıyla bilgisayar destekli bir yazılım uzman hekime karar verme sürecinde yardımcı olacaktır. Bu sebeple son yıllarda medikal görüntüler üzerine yapılan çalışmalar hız kazanmıştır.

Bu tezde kullanılan yöntemler derin öğrenme çatısı altında toplanmaktadır. Derin öğrenmenin kökleri Yapay Zeka kavramına dayanmaktadır. Yapay Zeka, minimum insan müdahalesi ile akıllı davranışı modellemek için bir bilgisayarın kullanımını ifade eden genel bir terimdir (Hamet ve Tremblay, 2017). Bilgisayar teknolojilerinin yeni

ortaya çıktığı dönemde, “Makineler düşünebilir mi?” sorusunun Mathison Turing tarafından sorulmasıyla yapay zekâ hakkında tartışmalar ortaya çıkmıştır (Turing, 1950). Fakat yapay zekâ kavramı resmi olarak ilk defa John McCarthy öncülüğünde 1956 yılında “Dartmouth College Artificial Intelligence” konferansında ortaya atılmış ve temelleri oluşturulmuştur (Andresen, 2002).

Şekil 1’de verilen hiyerarşiye göre yapay zekanın bir alt dalı olan Makine öğrenimi, insanların öğrenme şeklini taklit etmek için verilerin ve algoritmaların kullanımına dayanır. Makine öğreniminin bir alt kümesi olan DL, insan beyninde bulunan bilgi işleme modellerinden ilham almıştır. DL, çalışması için insan tarafından tasarlanmış herhangi bir kural gerektirmez; bunun yerine, verilen girdiyi belirli etiketlerle eşleştirmek için büyük miktarda veri kullanır. DL, her biri kendilerine beslenen verilerin farklı bir yorumunu sağlayan çok sayıda katman kullanılarak tasarlanmıştır (Alom, vd., 2019).

Şekil 1: Yapay Zeka, Makine Öğrenimi ve Derin Öğrenme Arasındaki İlişki



Kaynak: Alom vd., 2019

Bu tez çalışmasında, Alzheimer tipi demans hastalığının teşhisi ve sınıflandırılması için derin öğrenme modellerinin performansları karşılaştırılmış ve ayrıca yeni bir derin öğrenme modeli DEMXNET önerilmiş ve sonuçları diğer modeller ile kıyaslanmıştır. Bu amaçla güncel derin transfer öğrenme modellerinden olan VGG16, VGG19, InceptionV3, Xception ve DenseNet201 modelleri kullanılmıştır. Modeller, Kaggle'den alınan MRI veri seti üzerinde eğitilerek değerlendirildi. Veri seti, Hafif Demans (MID), Orta Demans (MOD), Demanssız (ND) ve Çok Hafif Demans (VMD) gibi dört tip demans

içermektedir. Veri setindeki sınıflarda görüntü sayısının dengesiz dağılımından dolayı Sentetik Azınlık Aşırı Örnekleme Tekniği (SMOTE) kullanılmıştır. Modellerin tamamı SMOTE'li ve SMOTE'siz olarak iki senaryoda eğitilerek test edilmiştir. Tüm modellerde SMOTE tekniğinin doğruluk oranlarını önemli ölçüde artırdığı görülmüştür.

Alzheimer hastalığının teşhisi ve sınıflandırılmasına yönelik literatürde kullanılan yapay zeka yöntemleri incelenmiş ve üç grup altında sunulmuştur. Birinci grupta klasik makine öğrenmesi yöntemleri ile önerilen çözümler, ikinci grupta yapay sinir ağları kullanılarak önerilen yaklaşımlar ve son olarak bu çalışmanın da odağında olan derin transfer öğrenme yöntemleri ile sunulan yaklaşımlar verilmiştir.

Rabeh vd. (2016) çalışmalarında Alzheimer sınıflandırma problemi için SVM yöntemini uygulamışlardır. Hipokampus'u çıkarmak için frontal, Corpus Callosum'u analiz etmek için Sagital ve Cortex'in varyasyon özellikleriyle çalışmak için eksenel düzlemleri kullanmışlardır. Bu üç düzlemi ve düzlemlerin hepsinin de bulunduğu verilerin performansları kıyaslamışlardır. Verileri OASIS'ten (The Open Access Series of Imaging Studies) elde etmişlerdir. Verilerin 50'si normal, 17 Hafif Bilişsel Bozukluk (MCI), 8'i ise Alzheimer hastasına aittir. Önerdikleri sistem ile %90,66 doğruluk değerine ulaşmışlardır.

Mathew vd. (2018) çalışmalarında, ADNI veri tabanından elde edilen MRI görüntülerinden kontrast, homojenlik, korelasyon, enerji gibi farklı istatistiksel özellikler ve ayrıca şekil özellikleri çıkarılmıştır. Görüntüler Alzheimer Hastalığı, Hafif Bilişsel Bozukluk ve Normal olmak üzere üç sınıfta toplanmıştır. Olasılıksal Sinir Ağı (PNN), Destek Vektör Makineleri (SVM) ve K-En Yakın Komşu (KNN) yöntemleri karşılaştırılmış olup performanslar doğruluk, özgüllük ve duyarlılık açısından değerlendirilmiştir.

Yashodhar ve Kini (2021) çalışmalarında Destek Vektör Makinesi, Rastgele orman, Gradient Boost ve Çok Katmanlı algılayıcı algoritmaları olmak üzere dört farklı sınıflandırma algoritmasını karşılaştırarak performanslarını değerlendirmişlerdir. Veri setini ADNI'den elde etmişlerdir. Veri setinin eğitim görüntü sayısı 7814, doğrulama görüntü sayısı 2605 ve test görüntü sayısını 3473 olarak ayarlamışlardır. Çok Katmanlı Algılayıcı ile elde ettikleri başarı diğer yöntemleri geçmiştir.

Yigit ve Isik (2018), açık erişimli Alzheimer hastalığı görüntülerini boylamsal, kesitsel ve her ikisini de içerecek şekilde üç veri seti hazırlamışlardır. Sınıflandırma için

Yapay Sinir Ağları (YSA), (LR), k-Nearest Neighborhood (KNN) ve Karar Ağacı (KA) sınıflandırıcılarını kullandılar ve performanslarını üç veri setinde karşılaştırdılar. Testler sonucunda LR ve ANN modelleri Alzheimer hastalığının tespiti ve evrelerinin belirlenmesinde en iyi performansı göstermiştir.

Elfallah ve diğerleri (2021), AH'nin doğruluk oranlarını iyileştirmek için çeşitli görüntü normalleştirme ve sınıflandırma modelleri tekniklerini sunar ve karşılaştırır. Z-skor, Min-Maks, Ondalık Ölçekleme ve Standart sapma olmak üzere dört normalleştirme tekniği uygulanmıştır. Sınıflandırıcılar Naive Bayes, Lojistik regresyon, karar ağacı (DT), k-en yakın komşular (kNN), Yapay Sinir Ağı (ANN) ve destek vektör makinesidir (SVM). Analiz edilen veri seti, 150 kişiden MRG'yi ve CDR ölçeklerini içerir. Normalleştirilmiş veri seti Z-skoru kullanılarak analiz edildiğinde en iyi bulgular lojistik regresyondan (%91,2) elde edilmiştir.

Murugan ve diğerleri (2021), Kaggle'dan elde edilen veri setini kullanarak, önerilen DEMNET modelini SMOTE yöntemiyle ve SMOTE yöntemi olmadan karşılaştırmıştır. %95,23 doğruluk puanına ulaştılar. Önerilen DEMNET bloğu, ReLU aktivasyonu, toplu normalizasyon katmanı ve maksimum havuzlama katmanı ve 2 evrişim katmanından oluşan yığın katmanından oluşur.

Çalışmalarında Saratxaga ve diğerleri (2021), MRI tabanlı Alzheimer teşhisi için derin öğrenme ve görüntü işleme tekniklerine dayalı yeni bir yöntem önermişler ve önceki literatür çalışmaları ile karşılaştırmışlardır. Veri setleri OASIS'ten elde edilmiştir. Hem 2B hem de 3B yaklaşımlar için bilinen mimariler ve özel mimariler dikkate alınmıştır. Bilinen mimariler arasında ResNet ailesi Inception, Xception vb. ile değerlendirilmiş ve ResNet18 daha iyi sonuçlar vermiştir. Özel ağlar hem 2B evrişim hem de 3B evrişim yaklaşımlarında tasarlanmıştır. Önerilen yöntem, hastalığın görüntü tabanlı otomatik teşhisi için 0,93'e kadar bir denge doğruluğu (BAC) ve hastalık evresinin belirlenmesi için 0,88'lik bir BAC (sağlıklı doku, çok hafif ve şiddetli evre) elde etmiştir.

He ve diğerleri (2019), 300 Alzheimer'lı ve 300 Normal görüntüyü içeren ADNI MPRAGE-REPEAT darbe tarama dizisi veri kümesini kullandı. Onların makalesi, pozitif ve negatif örneklerin ağırlıklı kombinasyonu ile bir veri kümesi artış yöntemi ve az sayıda örnek için bir öğrenme yöntemi önerir ve bir 3D tam evrişimli DenseNet sınıflandırma modeli kurar.

Literatürde VGG16, VGG19, InceptionV3, Xception ve DenseNet201 modellerinin bir arada sunulduğu ve karşılaştırıldığı Alzheimer hastalığına yönelik çalışmanın bulunmadığı tespit edildi. Bu amaçla ImageNet yarışmasında başarılı olan derin öğrenme modelleri üzerinde fine-tuning yapılarak başarılı performanslar elde edilmiştir. Önerilen CNN tabanlı DEMXNET modeli ile önerilen transfer öğrenme modelleri kıyaslanmıştır. Karşılaştırılan derin öğrenme modelleri doğruluk, geri çağırma, özgüllük, kesinlik, F1 skoru gibi performans ölçütlerine dayalı olarak istatistiksel olarak değerlendirilmiştir. En başarılı performans gösteren model önerilen DEMXNET modeli olmuştur.





BİRİNCİ BÖLÜM
YAPAY SİNİR AĞLARI

1.1. YAPAY SİNİR AĞLARININ TARİHÇESİ

1943 yılında nörofizyolog Warren McCulloch ile matematikçi Walter Pitts elektrik devreleri ile basit bir sinir ağı modelleyerek yapay sinir ağlarına doğru ilk adımı atmışlardır (McCulloch ve Pitts, 1943). 1949'da Donald Hebb, “Organization of Behavior” adlı kitabında nöron kavramını pekiştirdi. Hebb kuralı sayesinde sinir ağının bağlantı sayısı ile öğrenilme ve uyum sağlayabilmenin ilişkili olduğu saptanmıştır (Hebb, 2002). 1950'lerde bilgisayarlar daha gelişmiş hale geldikçe, varsayımsal bir sinir ağını simüle etmek mümkün olmuştur. Buna yönelik ilk adım, IBM araştırma laboratuvarlarından Nathaniel Rochester tarafından atılmış ancak bu girişim başarısız olmuştur.

1956'da Yapay Zeka üzerine Dartmouth Yaz Araştırma Projesi, hem yapay zeka hem de sinir ağları için bir destek sağlamıştır. Bu, yapay zeka ve beynin çok daha düşük seviyeli sinirsel işleme bölümünde araştırmaları teşvik etmiştir. 1957'de John von Neumann, telgraf röleleri veya vakum tüpleri kullanarak basit nöron fonksiyonlarını taklit etmeyi önermiştir. 1958 yılında ise Frank Rosenblatt bugün tüm yapay sinir ağı modellerinin temelini oluşturan perceptronun yani algılayıcıların tanımını yapmıştır (Rosenblatt, 1958).

ADALINE (Adaptive Linear Neuron) ve MADALINE (Multiple Adaline) sinir ağı modelleri 1959 yılında Widrow ve Hoff tarafından Standford üniversitesinde geliştirilmiştir. ADALINE modeli, Rosenblatt'ın perceptron modeli ile aynı özelliklere sahiptir fakat ADALINE modelinin öğrenme algoritması daha gelişmiştir. MADALINE telefon hatlarındaki ses yankıları problemini çözmüştür ve gerçek problemlere uygulanmış ilk sinir ağıdır (Auer, Burgsteiner ve Maass, 2008).

1963 yılında Widrow'un geliştirdiği ters sarkaç denetleyicisi en önemli çalışmalardan birisidir. 1969 yılında Minsky XOR probleminin tek katmanlı bir ağ yapısıyla çözülmesinin mümkün olmadığını ortaya koyup, çok katmanlı yapay sinir ağı mimarilerinin temelini atmıştır (Minsky, Papert ve Bottou, 1969). Fukushima tarafından 1979 yılında önerilen Necognitron, çok katmanlı bir yapay sinir ağıdır. Japon el yazısı karakterlerini tanıma için kullanılmıştır. Evrişimli sinir ağları için ilham kaynağı olmuştur.

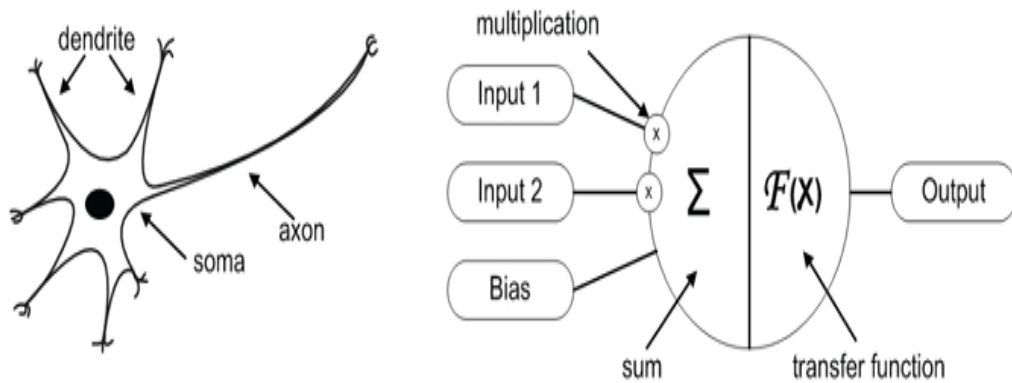
1987 yılında yapay sinir ağlarının edindiği bilgiyle yeni bilgileri ve kararları güncelleyebildiği, yani tahminleme ve öğrenme başarısını yüksek ölçüde artıran geriye

yayılım algoritması (back propagation algorithm) ortaya atılmıştır (Plaut ve Hinton, 1987). 1998'de Yann LeCun, Belge Tanımaya Uygulanan Gradyan Tabanlı Öğrenmeyi yayınlamıştır (LeCun, vd., 1998). 2006 yılında Hinton laboratuvarı, Derin Sinir Ağları için eğitim problemini çözmüştür. 2012 yılından günümüze kadar çeşitli derin öğrenme algoritmaları ortaya çıkmıştır. 2012 yılında ImageNet büyük ölçekli görsel tanımlama yarışmasında (ILSVRC- ImageNet Large Scale Visual Recognition Challenge) temel mimari olarak kabul edilen Evrimsel Sinir Ağı (CNN) ile kazanmışlardır (Krizhevsky, Sutskever ve Hinton, 2017). Bu yarışmadan sonra derin öğrenme ile ilgili yapılan çalışmalar bilim dünyasında çığır açmıştır.

1.2. YAPAY SİNİR AĞLARININ YAPISI

Yapay Sinir Ağları (YSA) insan beyninin en temel özelliği olan öğrenme fonksiyonunu örnekler yardımı ile gerçekleştiren bilgisayar sistemleridir (Krogh, 2008). YSA'lar biyolojik siniri yapısına benzer şekilde çalışır. Bir biyolojik sinirde dendritler gelen sinyalleri hücre gövdesine iletir daha sonra axonlar yardımıyla bu sinyaller diğer sinir hücrelerine aktarılır. Sinir hücreleri birbirleri ile sinapslar vasıtası ile iletişim kurarlar. Benzer şekilde YSA'lar da bilgileri bir toplama fonksiyonu ile toplar ve aktivasyon fonksiyonundan geçirerek çıktıyı üretip ağıın bağlantılarının üzerinden diğer katmanlara gönderir (Sadiq ve Rodriguez, 2004). Şekil 1.1'de biyolojik ve yapay sinir hücresinin tasarımı verilmiştir.

Şekil 1.1: Biyolojik ve Yapay Sinir Hücresi



Kaynak: Suzuki, 2011

Sisteme ilk olarak rastgele ağırlıklar verilir ve bu ağırlıklara göre bir çıktı oluşur. Bu çıktı gerçek değerle uyumlu ise diğer girdiler test edilir. Bütün girdiler mevcut ağırlıklar kullanılarak çıktılarda en az hata değerine ulaşınca kadar işleyiş devam eder. Matematiksel olarak ifadesi aşağıda verilmiştir.

$$y(k) = F\left(\sum_{i=0}^m w_i(k) \cdot x_i(k) + b\right)$$

y: x'in değerine bağlı olduğundan bağımlı değişkendir. Girdiye ait skoru verir.

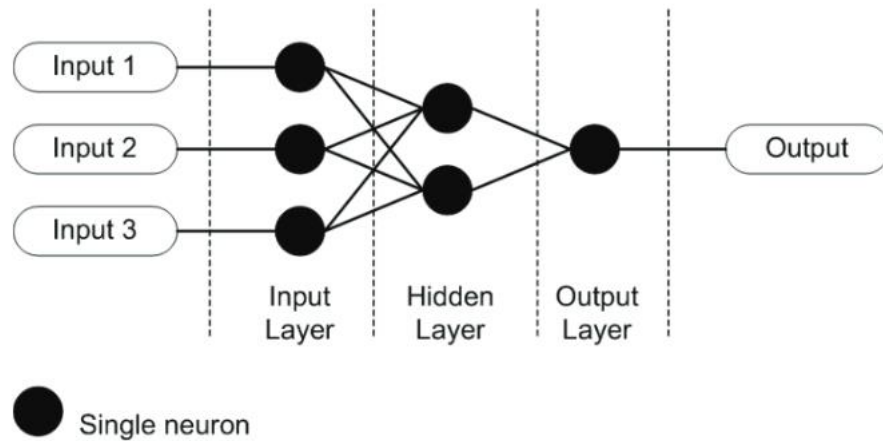
x: bağımsız değişken, girdi.

w: ağırlık parametresi

b: bias, aktivasyon fonksiyonunu sola veya sağa kaydırmak için kullanılır.

Girdiler ve ağırlık parametreleri ile çarpılıp bias eklenerek ağırlıklı toplam hesaplanır. Bulunan toplama sonucuna aktivasyon fonksiyonu uygulanarak sonuç elde edilir. YSA'lar girdi katmanı, ara katmanlar ve çıktı katmanı olmak üzere üç katmandan oluşur. Birer adet giriş ve çıkış bulunan modeller tek katmanlı, birden çok gizli katman bulunan modeller ise çok katmanlı model olarak adlandırılırlar. Şekil 1.2'de temel YSA modeli gösterilmiştir.

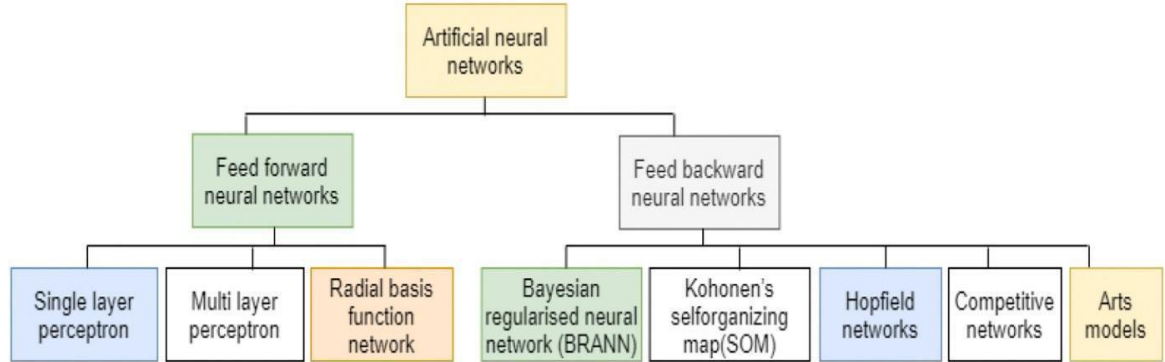
Şekil 1.2: Temel YSA Modeli



Kaynak: Suzuki, 2011

YSA'lar ileri ve geri beslemeli olarak ikiye ayrılırlar. Alt kategorileri ile genel görünüm Şekil 1.3'te sunulmuştur.

Şekil 1.3: Yapay Sinir Ağlarının Sınıflandırılması



Kaynak: Isaac Abiodun vd., 2018

İleri beslemeli YSA'lara gelen bilgiler sırasıyla giriş katmanı, ara katmanlar ve çıkış katmanında işlenir. Yani bilgi giriş katmanından çıkışa doğru tek yönde hareket eder ve döngü yoktur. Geri beslemeli YSA'larda ileri beslemeli olanların aksine bir hücrenin çıktısı sadece kendinden sonra gelen hücrenin katmanına girdi olarak verilmez. Kendinden önceki katmanda veya kendi katmanında bulunan herhangi bir hücreye de girdi olarak bağlanabilir.

1.3. AŞIRI ÖĞRENME (OVERFITTING) VE EKSİK ÖĞRENME (UNDERFITTING) KAVRAMLARI

Aşırı öğrenme (Overfitting):

Oluşturulan modelin veri seti üzerinde ezber yapmasına aşırı öğrenme denir. Model, aşırı öğrenmenin diğer bir deyişle ezberlemenin varlığından dolayı eğitim setinde mükemmel performans gösterirken, test setinde zayıf bir performans sergiler (Ying, 2019).

Eksik öğrenme (Underfitting):

Model hem eğitim hem de test veri setlerinde düşük performans gösteriyorsa model verilere yetersiz uyum sağlamıştır. Bir modelde öğrenme yetersiz kaldıysa modelin eğitim verilerine uygun ilerlemediği ve yeni verileri genelleştiremediği anlamına gelir.

1.4. GERİ YAYILIM (BACKPROPAGATION) ALGORİTMASI

Backpropagation, çok katmanlı yapay sinir ağlarında hataların geriye doğru yayılmasını sağlayan denetimli bir öğrenme algoritmasıdır. Bir sinir ağını gradyan iniş ile eğitirken, ağın tahminlerinin gerçek etiketlerden ne kadar uzak olduğunu gösteren bir kayıp fonksiyonu hesaplanır. Geri yayılım, ağın ağırlıklarının her birine göre kayıp fonksiyonunun gradyanını hesaplamamızı sağlar. Bu, birçok eğitim yinelemesinde kayıp işlevini kademeli olarak azaltmak için her ağırlığın ayrı ayrı güncellenmesini sağlar. Geri yayılım, son katmandan birinciye doğru ileri beslemeli ağ üzerinden geriye doğru ilerleyen gradyanın hesaplanmasını içerir (Werbos, 1990).

1.5. KAYIP (LOSS) FONKSİYONLARI

Kayıp fonksiyonları, tahmin edilen değer ve gerçek değer arasındaki hatayı en aza indirmek için geri yayılımı kullanmaktadır ve bu hata değerinin eğitim esnasında zamanla sifıra yaklaşması beklenmektedir. Kayıp fonksiyonlarının Cross-Entropy, Ortalama Mutlak Hata (MAE- Mean Absolute Error), Ortalama Karesel Hata (MSE- Mean Square Error), Hinge, Huber, Kullback-Leibler gibi çeşitleri bulunmaktadır. Bu tezde kategorik çapraz entropi fonksiyonu kullanılmıştır.

Kategorik Çapraz Entropi (Categorical Cross Entropy):

Bu fonksiyon genel olarak ikiden fazla sınıflı sınıflandırma modelleri için kullanılır ve olasılık dağılımlar arasındaki farklılığı ölçer.

1.6. OPTİMİZASYON ALGORİTMALARI

Optimizasyon algoritmaları, bir makinenin deneyimi yoluyla öğrenebileceği temeli oluşturur. Gradyanları hesaplarlar ve kayıp fonksiyonunu en aza indirmeye çalışırlar. Öğrenmenin farklı türde optimizasyon algoritmalarıyla uygulanmasının birkaç yolu vardır. Sık kullanılan optimizasyon algoritmaları aşağıda açıklanmıştır (Zaheer ve Shaziya, 2019; Seyyarer, vd., 2020):

Gradient Descent

Gradyan iniş (GD), belirli bir fonksiyonun yerel minimum/maksimumunu bulmak için kullanılan yinelemeli birinci dereceden bir optimizasyon algoritmasıdır. Bu yöntem, bir maliyet/kayıp işlevini en aza indirmek için makine öğreniminde ve derin öğrenmede yaygın olarak kullanılır.

Stochastic Gradient Descent - SGD (Olasılıksal Dereceli Azalma)

Stokastik gradyan inişi, veri kümesi içindeki her örnek için bir eğitim dönemi çalıştırır ve her eğitim örneğinin parametrelerini birer birer günceller. Yalnızca bir eğitim örneğini tutmanız gerektiğinden, bunları bellekte saklamak daha kolaydır. Bu sık güncellemeler daha fazla ayrıntı ve hız sunabilse de toplu gradyan düşüşüne kıyasla hesaplama verimliliğinde kayıplara neden olabilir. Sık güncellemeleri gürültülü gradyanlara neden olabilir, ancak bu aynı zamanda lokal minimumdan kaçmada ve global olanı bulmada yardımcı olabilir.

AdaGrad

AdaGrad, öğrenme oranını parametrelere uyarlayan stokastik bir optimizasyon yöntemidir. Sık oluşan özelliklerle ilişkili parametreler için daha küçük güncellemeler ve seyrek olarak meydana gelen özelliklerle ilişkili parametreler için daha büyük güncellemeler gerçekleştirir.

Adadelata - An Adaptive Learning Rate Method

Bu optimizasyon yöntemi öğrenme problemini çözmeye yönelik bir Adagrad versiyonudur. Adadelata yönteminde, Adagrad ve RMSProp yöntemlerinden farklı olarak öğrenme katsayısı seçme zorunluluğu yoktur. Öğrenme katsayısı yerine, geçerli ağırlıklar ile güncellenen ağırlıklar arasındaki farkı ifade eden delta değerlerinin karelerinin momentumlu toplamaları kullanılmaktadır.

RMSProp

Adagrad'da olduğu gibi sabit öğrenme katsayısı problemini çözmek için önerilmiştir. Aralarındaki fark ise, AdaGrad yöntemindeki gradyanların karelerini almak yerine momentumlu gradyanların karelerini almaktadır.

Adam – Adaptive Moment Estimation

RMSprop gibi öğrenme hızını ölçeklendirmek için kare gradyanları kullanır ve momentum ile SGD gibi gradyan yerine gradyanın hareketli ortalamasını kullanarak momentumdan yararlanır.

1.7. YAPAY SİNİR AĞLARDA ÖĞRENME

1.7.1. Denetimli Öğrenme (Supervised Learning)

Bu öğrenme türü, yeni bilgiler edinmek için geçmiş deneyimlerden yararlanan insanların öğrenmesine benzer ancak bilgisayarların deneyimleri olmadığından gerçek dünya uygulamalarından elde edilen verilerden öğrenir (Bkassiny, Li ve Jayaweera, 2011). Denetimli öğrenmede veri seti girdi ve çıktılardan oluşur. Algoritmanın var olan verilerden öğrenip kendine bir fonksiyon oluşturması ve yeni gelen veriyi bu fonksiyona tabi tutarak tahminde bulunması istenir (Akgözlüoğlu ve Özçift, 2021).

Destek Vektör Makinesi (SVM), Yapay Sinir Ağları (YSA), Lojistik Regresyon (LOGREG), Basit Bayes (NB), k-En Yakın Komşu (KNN), Rastgele Orman (RF), Torbalama ağaçları (BAG-DT) ve Karar Ağaçları (DT) algoritmaları denetimli öğrenme modeli oluşturulurken kullanılan yöntemlerdendir (Caruana, 2006).

1.7.2 Denetimsiz Öğrenme (Unsupervised Learning)

Eğitim verileri hakkında bir fikre sahip olunmadığında makinenin anlamlı sonuçlar üretmesine denetimsiz öğrenme denir. Bu öğrenme tipinde çok fazla veriden kümeleme oluşturabilmek için birtakım algoritmalar kullanılır çıktı olarak birbiriyle daha yakın kümeler oluşturulur. Bu kümelemelere bakılarak verilerin nerelerde toplandığı anlaşılmaya çalışılır. Denetimsiz öğrenme, temel olarak kümeleme ve ilişkilendirme algoritmaları için kullanışlıdır (Dike, vd., 2018).

1.7.3 Yarı Denetimli Öğrenme (Semi-Supervised Learning)

Verileri etiketleme, ilgili konuda deneyimli insanların çabalarını gerektirdiğinden, elde edilmesi genellikle zor, pahalı veya zaman alıcıdır. Etiketsiz verilerin toplanması daha kolaydır. Bu problemin çözümü için etiketlenmemiş veri sayısı etiketli verilerden çok daha fazla olduğu durumlarda bu öğrenme tipi kullanılır (Zhu Xiaojin, 2008).

1.7.4 Takviyeli Öğrenme (Reinforcement Learning)

Bu öğrenme tipinde sistem, ödül ve ceza yöntemi ile öğrenme süreci boyunca yönlendirilir. Denetimli öğrenmeden farkı doğru girdi ve çıktı eşleşmelerinin verilmemesidir. Bu yöntem robotik, oyun programlama, metin oluşturma, doğal dil işleme, akıllı ulaşım sistemleri, işletme yönetimi gibi alanlarda sıklıkla kullanılır (Li, 2018).



İKİNCİ BÖLÜM

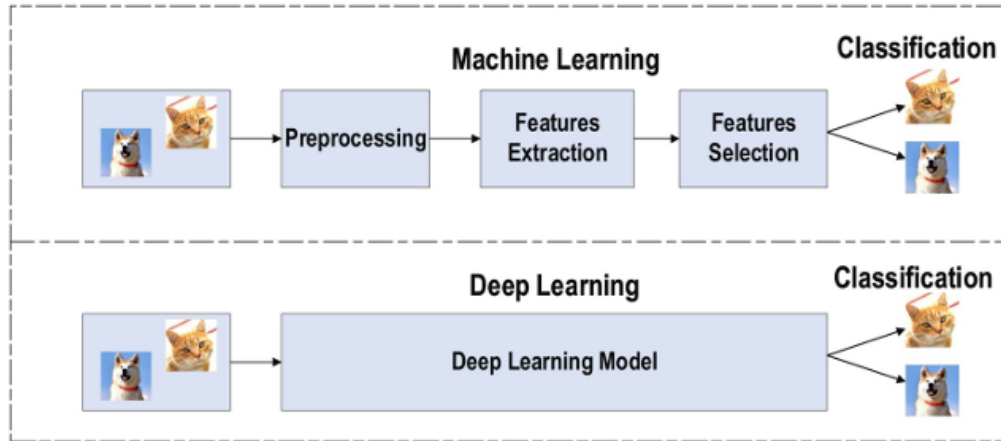
DERİN ÖĞRENME

2.1. DERİN ÖĞRENME KAVRAMI

Son yıllarda, derin öğrenme ve evrimsel ağlardaki gelişmeler nedeniyle nesne sınıflandırma ve algılama başarısı önemli ölçüde artmıştır (Szegedy, vd., 2015). Derin sinir ağları, modelin girdi verilerinin daha karmaşık temsillerini öğrenmesini sağlayan birden çok gizli katmana sahip standart sinir ağının uzantısıdır (Wang, Zhao ve Pourpanah, 2020). Derin öğrenme modelleri özel bilgi işlem donanımı üzerinde çalışabilme yetenekleri sayesinde büyük veri kümelerine ölçeklenir ve daha fazla veri ile gelişmeye devam ederek birçok klasik makine öğrenimi yaklaşımından daha iyi performans göstermektedir (Esteva, vd., 2019).

Derin öğrenme modellerinde geleneksel makine öğrenimi modellerinin aksine manuel segmentasyona ve öznetelik çıkarmaya gerek yoktur. Bu nedenle derin öğrenme modelleri daha popülerdir ve ikili veya çok sınıflı sınıflandırma görevini tamamen otomatikleştirir. Şekil 2.1’de makine öğrenimi ve derin öğrenme arasındaki fark gösterilmiştir.

Şekil 2. 1: Makine Öğrenimi ve Derin Öğrenme Arasındaki Fark



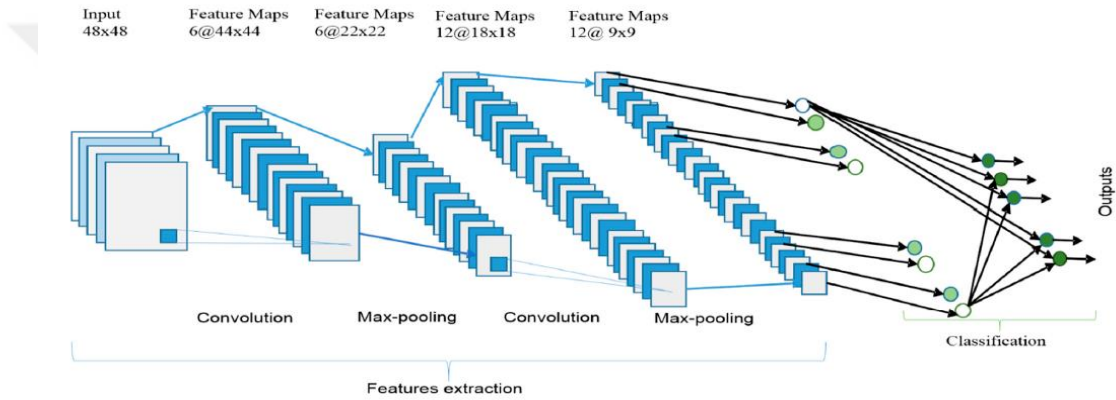
Kaynak: Alzubaidi vd., 2021

Yaygın olarak kullanılan derin sinir ağı mimarilerine örnek olarak Evrimsel Sinir Ağları, Autoencoder, Restricted Boltzman Machine, Long Short-Term Memory verilebilir (Shrestha ve Mahmood, 2019). Bu tez çalışmasının temelini CNN oluşturduğundan bölüm 2.2’de açıklanmıştır.

2.2. EVRİŞİMSSEL SİNİR AĞLARI (CONVOLUTIONAL NEURAL NETWORKS – CNNs)

Derin öğrenme yöntemlerinden biri olan CNN mimarileri görüntü sınıflandırma, nesne tanıma gibi alanlarda yaygın olarak kullanılmaktadır. CNN'nin yapısı ilk olarak 1988'de Fukushima tarafından önerildi (Fukushima, 1988). Ağı eğitmek için donanımlardaki hesaplama gücüne ihtiyaç duyulduğundan yaygın olarak kullanılmadı. 1990'larda LeCun ve arkadaşları CNN'lere gradyan tabanlı bir öğrenme algoritması uygulamış ve el yazısıyla yazılmış rakam sınıflandırma problemi için başarılı sonuçlar elde etmiştir.

Şekil 2.2: CNN'nin Genel Mimarisi



Kaynak: Alom vd., 2019

Şekil 2.2’de CNN’nin genel mimarisi verilmiş olup bir giriş katmanı, çoklu evrişim ve maksimum havuzlama katmanları, bir tam bağlantılı katman ve bir sınıflandırma katmanını içerir.

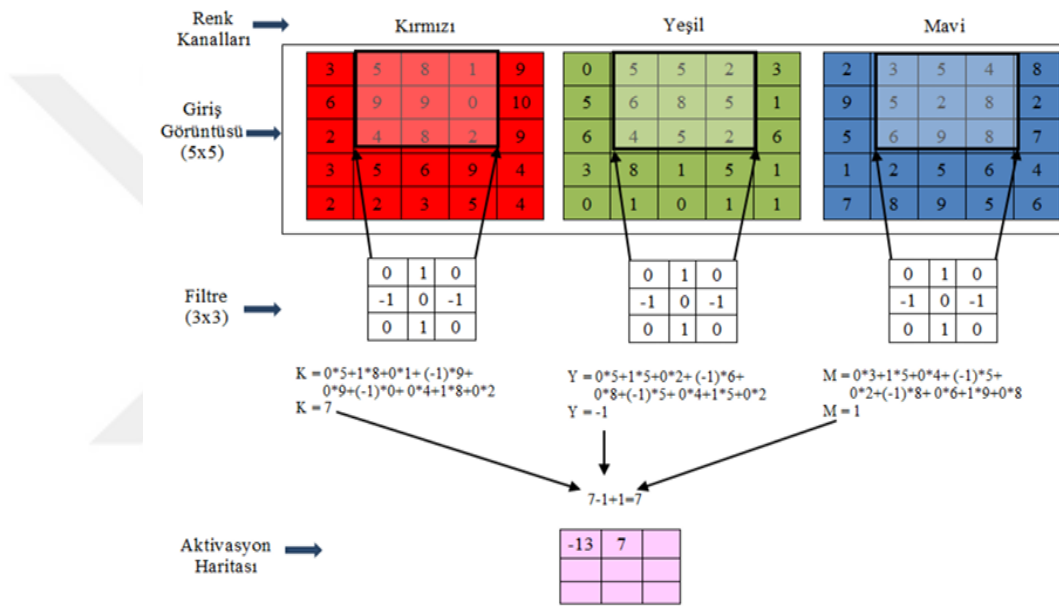
CNN, ileri beslemeli bir sinir ağı olup iki boyutlu bir görüntüden özellikler çıkarır ve ağ yapısını optimize etmek, ağıdaki bilinmeyen parametreleri çözmek için geri yayılım (backpropagation) algoritmasını kullanır (Zhu Anwen, vd., 2017). Örnekler ağıya verilir ve gerekli özellikler ön işleme ile çıkarılır. Daha sonra çıktıyı elde etmek için sınıflandırma veya regresyon yapılır.

2.3. EVRİŞİMSEL SİNİR AĞLARI KATMANLARI

2.3.1. Evrişim Katmanı (Convolution Layer)

Belirli görsel özellikleri algılamaya yönelik belirli boyuttaki filtreler giriş görseline uygulanarak filtre özelliklerinin çarpımından oluşan bir matris elde edilir. Her bir filtre karakteristik görsel özellikleri ortaya çıkarır. Ağın ilk katmanlarındaki filtreler çok daha basitken ilerleyen katmanlarda uygulanan filtreler çok daha anlamlı olmaktadır (İnik Özkan ve Ülker Erkan, 2017).

Şekil 2.3: 5x5x3 Boyuttaki Giriş Görüntüsüne 3x3'lük Filtrenin Uygulandığı Evrişim İşlemi



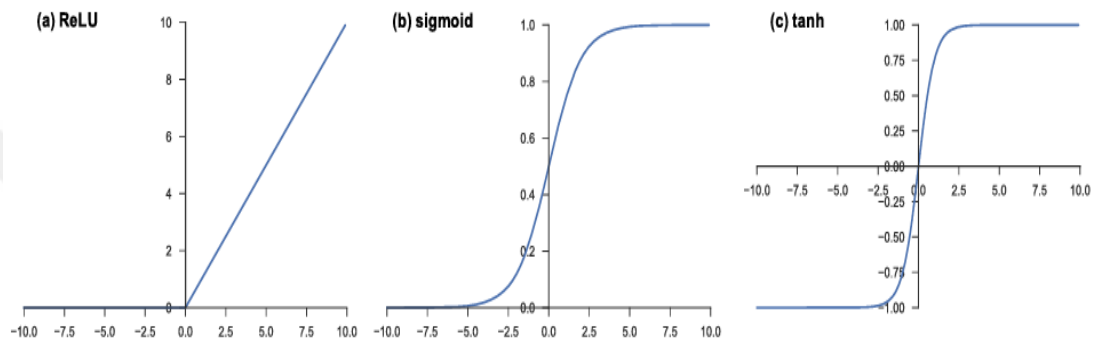
Kaynak: İnik Özkan ve Ülker Erkan, 2017

Şekil 2.3'te, 5x5 boyutundaki girdi verisi üzerinde evrişim filtresi gezdirilerek 3x3 evrişim operatörünün uygulanışı ve çıkış görüntüsünün piksel değerleri gösterilmiştir. Evrişim işlemi sonucunda yatay kenarlar, dikey kenarlar, açısall kenarlar, yumuşatılmış ve keskinleştirilmiş imge gibi orijinal görüntüyle aynı boyutta görüntü türevleri elde edilmektedir (İnik Özkan ve Ülker Erkan, 2017).

2.3.2. Aktivasyon Katmanı (Activation Layer)

Konvolüsyon katmanından sonra aktivasyon aşaması gelir. Aktivasyon işlevi, derin sinir ağlarının eğitiminin başarısında önemli bir rol oynar. Doğrusal ilerleyen matris hesaplarını doğrusal olmayan bir yapıya dönüştürür. Sık kullanılan aktivasyon fonksiyonlarından ReLU, sigmoid ve hiperbolik tanjant Şekil 2.4'te gösterilmiştir.

Şekil 2.4: Yaygın Olarak Kullanılan Aktivasyon Fonksiyonları: (a) Rectified Linear Unit (ReLU), (b) Sigmoid, (c) Hyperbolic Tangent (tanh)



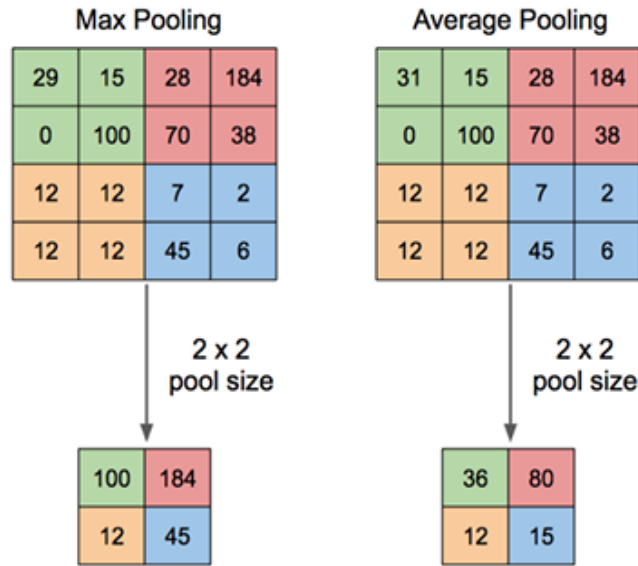
Kaynak: Yamashita vd., 2018

Şu anda en başarılı ve yaygın olarak kullanılan aktivasyon fonksiyonu Rektifiyeli Doğrusal Birimdir (ReLU – Rectified Linear Unit). $f(x) = \max(x, 0)$ şeklinde ifade edilir yani görüntüdeki negatif değerleri eler (Isaac Abiodun vd., 2018). Sigmoid fonksiyonu 0 ile 1 arasında değerler üretir. Hiperbolik Tanjant fonksiyonu ise değerleri -1 ile 1 aralığına toplar (Sibi, Jones ve Siddarth, 2013).

2.3.3. Havuzlama (Pooling) Katmanı

Evrişim katmanından gelen verilerin ana karakteristiklerini koruyarak boyutunu azaltır. Böylelikle işlenecek veri sayısı azalır. Daha az veri ile yoğunlaştırılmış çıktılar elde edilir. Sistemin ezberlemesinin önüne geçer.

Şekil 2.5: Max Pooling ve Average Pooling İşlemi (Stride/Adım: 2)



Kaynak: Tammina, 2019

Bu katmanda da belirli filtreler kullanılır ve belli adımlarla matris üzerinde gezdirilir piksellerin maksimum değerlerini (max-pooling) ya da değerlerin ortalaması (average-pooling) alınarak işlem yapılmaktadır (Boureau, vd., 2010). Şekil 2.5'te bu iki yöntem verilmiştir. Maksimum havuzlama yöntemi genellikle daha iyi sonuç verdiği için tercih edilmektedir.

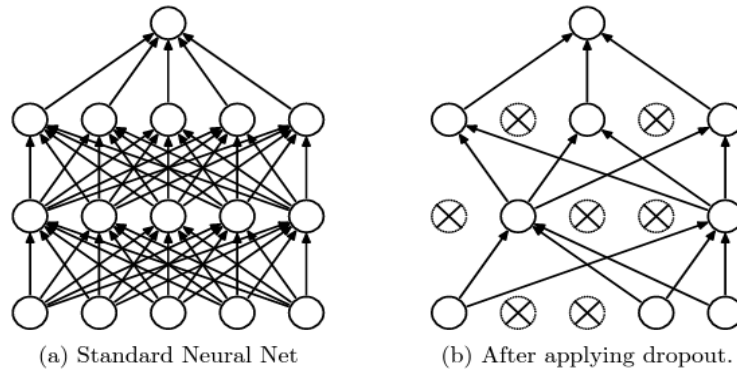
2.3.4. Batch Normalization Katmanı

Girdileri normalleştiren katmandır. Toplu normalleştirme, ortalama çıktıyı 0'a ve çıktı standart sapmasını 1'e yakın tutan bir dönüşüm uygular. Derin Sinir Ağlarının eğitimi önceki katmanların parametreleri değiştikçe eğitim sırasında her katmanın girdilerinin dağılımının değişmesi gerçeğiyle karmaşıktır. Bu, daha düşük öğrenme hızları ve dikkatli parametre başlatma gerektirerek eğitimi yavaşlatır ve doygun doğrusal olmayan modellere sahip modelleri eğitmeyi zorlaştırır. Bu duruma dahili ortak değişken kayması (internal covariate shift) denilmektedir ve sorun katman girdilerini normalleştirerek çözülür (Ioffe ve Szegedy, 2015).

2.3.5. Bırakma Katmanı (Dropout Layer)

CNN’de eğitim aşamasında fazla eğitimden dolayı ağ ezberleme (overfitting) yapabilir. Ağı ezber yapmasının önüne geçmek için bu katman kullanılır.

Şekil 2.6: (a) 2 Gizli Katmana Sahip Sinir Ağı (b) Soldaki Ağı Dropout Uygulanmış Hali



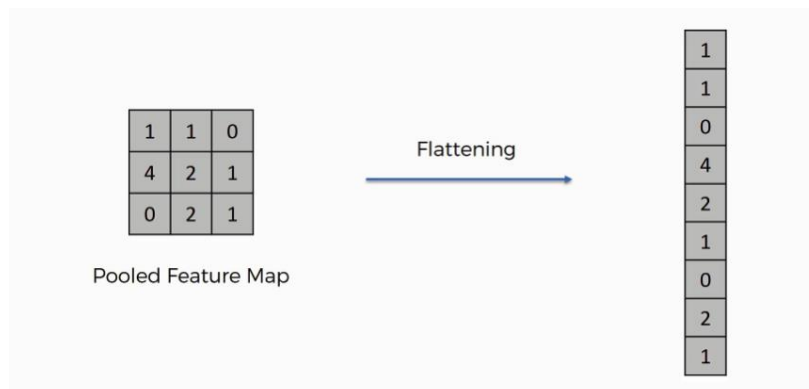
Kaynak: Srivastava vd., 2014

Bir sinir ağındaki birimleri rastgele olarak tüm gelen ve giden bağlantılarla birlikte ağdan geçici olarak çıkarmaya dropout işlemi denir (Srivastava, vd., 2014). Bu işlem Şekil 2.6’da resmedilmiştir.

2.3.6. Düzleştirme Katmanı (Flatten Layer)

Flatten katmanının amacı son katman olan tam bağlantılı katmanın giriş verilerini hazırlamaktır.

Şekil 2.7: Düzleştirme İşlemi

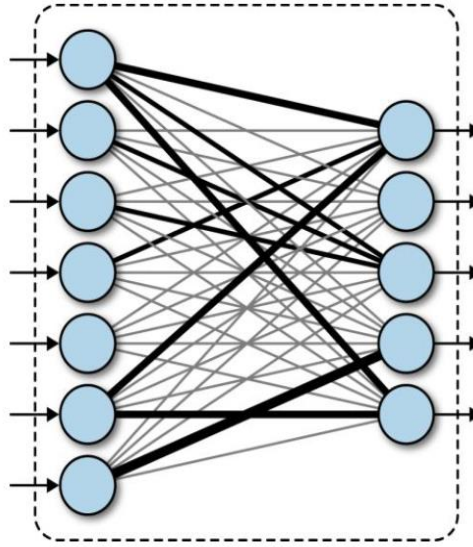


Boyutsallık problemini çözmek için önceki katmanlardan gelen matrisleri tek boyutlu bir diziye çevirir (Chlap, vd., 2021). Şekil 2.7’de düzleştirme işlemi gösterilmiştir.

2.3.7. Tam Bağlantılı Katman (Fully Connected Layer/Dense)

Tam bağlantılı katman ya da Dense Katman olarak da bilinen CNN'in son ve en önemli katmanıdır. Modelden öğrenilen, girdilerden üst düzey öznitelikler çıkararak girdileri farklı kategorilere sınıflandırmaktan tam bağlantılı bir katman sorumludur.

Şekil 2.8: Tam Bağlantılı Katman



Kaynak: Tammina, 2019

Tam bağlı katman (FC), Şekil 2.8'deki gibi her girişin tüm nöronlara bağlı olduğu bir giriş üzerinde çalışır. Her tam bağlı katman (Dense layer) bir aktivasyon fonksiyonundan (örneğin tanh veya ReLu) geçirilir, ancak son dense katman çıktısı Softmax'tan geçirilir. Softmax çok sınıflı sınıflandırmada kullanılan kayıp fonksiyonunun adı Çapraz Entropidir.

Softmax Fonksiyonu

Softmax fonksiyonun çok sınıflı bir problemde her sınıfa ondalık olasılıklar atamaktadır. Bu ondalık olasılıkların toplamı 0 ve 1 değer aralığında olmalıdır. Softmax fonksiyonunun çıktısı, N-boyutlu bir vektördür; burada N, CNN'nin seçmesi gereken sınıf sayısıdır. Bu N boyutlu vektördeki her sayı, görüntünün her bir belirli sınıfa ait olma olasılığını temsil eder.

2.4. DERİN ÖĞRENMEDE HİPER-PARAMETRELER

Veri Seti Boyutu

Veri setinin büyüklüğü ve çeşitliliği modelin eğitimi aşamasında önem arz etmektedir. Veri setinin yeterli olmadığı durumlarda veri arttırma yöntemleri kullanılabilir (Chlap, vd., 2021).

Batch Boyutu

Modelin aynı anda kaç veriyi işleyebileceğini belirler. Her iterasyonda geriye yayılım (backpropagation) ve gradyan hesaplaması yapılarak ağırlıklar güncelleneceğinden zaman ve bellek açısından veri setindeki tüm verileri aynı anda işlemek maliyetli olacaktır. Bu problemin çözümü için veri seti küçük gruplara ayrılır ve öğrenme işlemi gerçekleştirilir. Batch boyutu genellikle 2'nin katları şeklinde seçilir. Veriler kendi içerisinde gruplaşmış olabileceğinden eğitim başlamadan önce veri seti karıştırılarak (shuffle) ezberlemenin (overfitting) önüne geçilmelidir (Kandel ve Castelli, 2020).

Eğitim Tur Sayısı (Epoch Size)

Model eğitilirken batch boyutuna göre parçalar halinde eğitilir. İlk parça eğitildikten sonra modelin başarımı test edilir başarıya göre geriye yayılım ile ağırlıklar güncellenir. Daha sonra yeni eğitim kümesi ile model tekrar eğitilip tekrar ağırlıklar güncellenir. Bu eğitim adımlarının her birine epoch denir. Epoch sayısı modele ve veri setine göre değişiklik gösterebilir (Brownlee, 2018).

Öğrenme Oranı (Learning Rate)

Öğrenme oranı, bir modelin minimum kayıp fonksiyonuna ulaşması için adım boyutunu kontrol eder. Öğrenme oranı yüksek bir değer verilirse modelde hızlı öğrenme gerçekleşir fakat minimum kayıp fonksiyonunu kaçırabilir. Öğrenme oranı düşük bir değer verilirse minimum kayıp fonksiyonuna ulaşmak daha kolay olabilir fakat modelde yavaş öğrenme gerçekleşir. Öğrenme oranı için başlangıçta büyük değerler verilip sonlara doğru küçük değerler verilecek şekilde ayarlanırsa sınıflandırma başarısı artacaktır (Fang, Luo ve Tang, 2005).

Optimizasyon Algoritması

Derin öğrenmede kayıp değeri (loss value) modelin o anda ne kadar kötü performans verdiğini açıklar. Bu kaybı en aza indirmek modelin daha iyi performans göstereceği anlamına gelir. Herhangi bir matematiksel ifadeyi minimize etme (veya maksimize etme) işlemine optimizasyon denir. Optimize ediciler, kayıpları azaltmak için sinir ağının ağırlıklar ve öğrenme oranı gibi özelliklerini değiştirmek için kullanılan algoritmalar veya yöntemlerdir. Optimizasyon algoritmalarına örnek olarak Gradient Descent, Stochastic Gradient Descent (SGD), Adaptive Gradient (AdaGrad), AdaDelta, RMSprop, Adam verilebilir (Sun, 2020; Zaheer ve Shaziya, 2019)

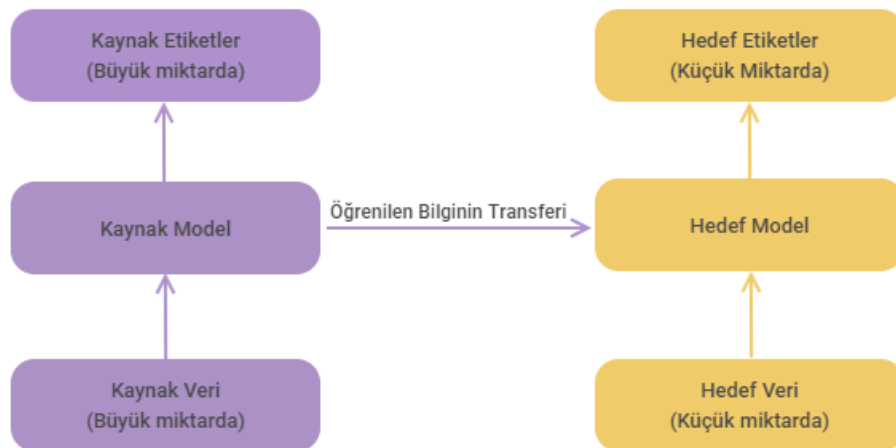
Aktivasyon Fonksiyonu

Derin öğrenme yöntemlerinde doğrusal olmayan problemlerin çözümü daha başarılı olduğundan gizli katmanlarda gerçekleşen matris işlemleri doğrusal bir yapıdan doğrusal olmayan bir yapıya aktivasyon fonksiyonları aracılığıyla dönüştürülür. Aktivasyon fonksiyonlarına örnek olarak sigmoid, tanjant, relu ve leaky-relu verilebilir (Sibi, Jones ve Siddarth, 2013). Probleme bağlı olarak aktivasyon fonksiyonunun seçimi yapılır. Örneğin negatif değerler önem arz ediyorsa Leaky-relu aktivasyon fonksiyonu olarak seçilebilir.

2.5. TRANSFER ÖĞRENME

Transfer öğrenme, yapay öğrenme sistemlerinin eğitim aşamasıyla öğrendiği bilgiyi farklı veya benzer problem çözümlerinde kullanılmasını inceleyen öğrenme yaklaşımıdır (Weiss, Khoshgoftaar ve Wang, 2016).

Şekil 2.9: Transfer Öğrenme Yaklaşımı



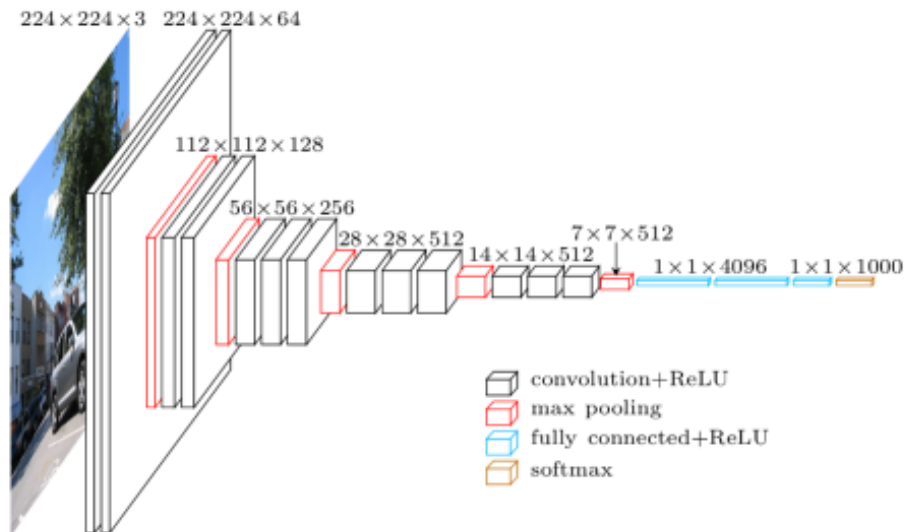
Elde edilen eğitim veri setinin küçük olduğu durumlarda, maliyet fonksiyonu yerel bir minimumda takılıp kalabilir ve bu da aşırı öğrenme (overfitting)/eksik öğrenme (underfitting) ile sonuçlanabilir (Shorten ve Khoshgoftaar, 2019). Küçük boyutlu veri setlerinin eğitimi için önceden eğitilmiş büyük veri kümesinden elde edilen ağırlıklar kullanılarak daha başarılı sonuçlar sunan transfer öğrenimi iyi bir alternatiftir. Şekil 2.9’da bu yaklaşım gösterilmektedir. Yalnızca kullanılan transfer öğrenme modelinin son tam bağlı katmanı, hedef eğitim verileri ile yeniden eğitilir. Bu sayede önceden eğitilmiş sağlam bir ağırlık seti sağlamakta kalmaz, aynı zamanda problemimizde kanıtlanmış ağ mimarilerini kullanma fırsatı verir.

2.5.1. Transfer Öğrenme Modelleri

2.5.1.1. VGG16 ve VGG19

VGG16 modeli Oxford Üniversitesi Görsel Geometri Grubundan (Visual Geometry Group – VGG) K.Simonyan ve A. Zisserman tarafından “Very Deep Convolutional Networks for Large-Scale Image Recognition” makalesinde önerilmiştir (Özdemir ve Arslan, 2021). 13 evrişimli, 5 maksimum havuzlama ve 3 yoğun katman (dense layer) olmak üzere toplam 21 katmandan oluşmasına rağmen sadece 16 ağırlık katmanı vardır.

Şekil 2.10: VGG16 Modeli



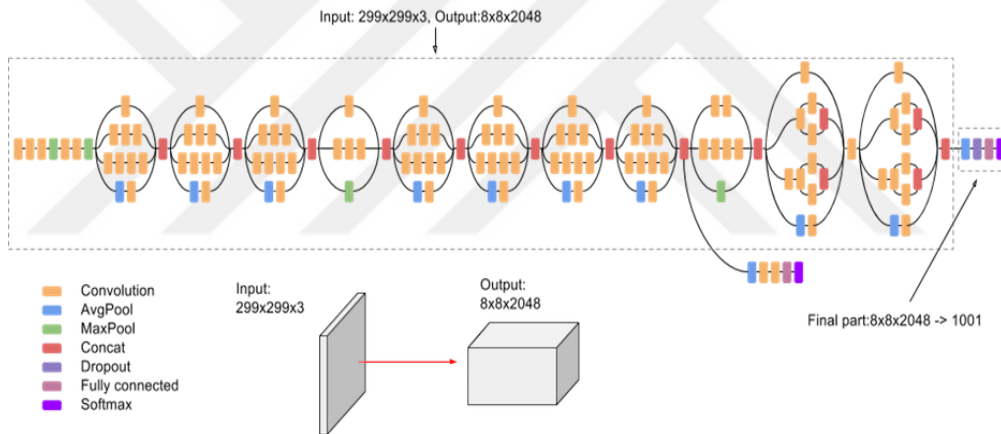
Kaynak: Muneeb ul Hassan, 2018

Katmanlar; evrişimli katmanlar, maksimum ortaklama katmanları, aktivasyon katmanları ve tam bağı katmanlardan oluşur. Şekil 2.10'daki örnekte görüldüğü üzere giriş katmanında $224 \times 224 \times 3$ boyutunda görüntü yer alır ve tam bağlantı katmanında $7 \times 7 \times 512 = 4096$ nöronlu bir öznitelik vektörüne dönüştürülür (Akgözlüoğlu & Özçift, 2021). İki tam bağlantı katmanı çıkışında 1000 sınıflı softmax ile başarımları oranı hesaplanır. VGG19 modeli ise VGG16 modelinin geliştirilmiş bir versiyonu olup 16 evrişim katmanı, 3 tam bağı katmandan oluşur.

2.5.1.2. InceptionV3

Szegedy ve arkadaşları (2015) tarafından “Rethinking the Inception Architecture for Computer Vision” isimli makalede öne sürülen bir modeldir. Şekil 2.11’de çalışmada önerilen modelin genel yapısı sunulmuştur.

Şekil 2.11: InceptionV3 Modeli



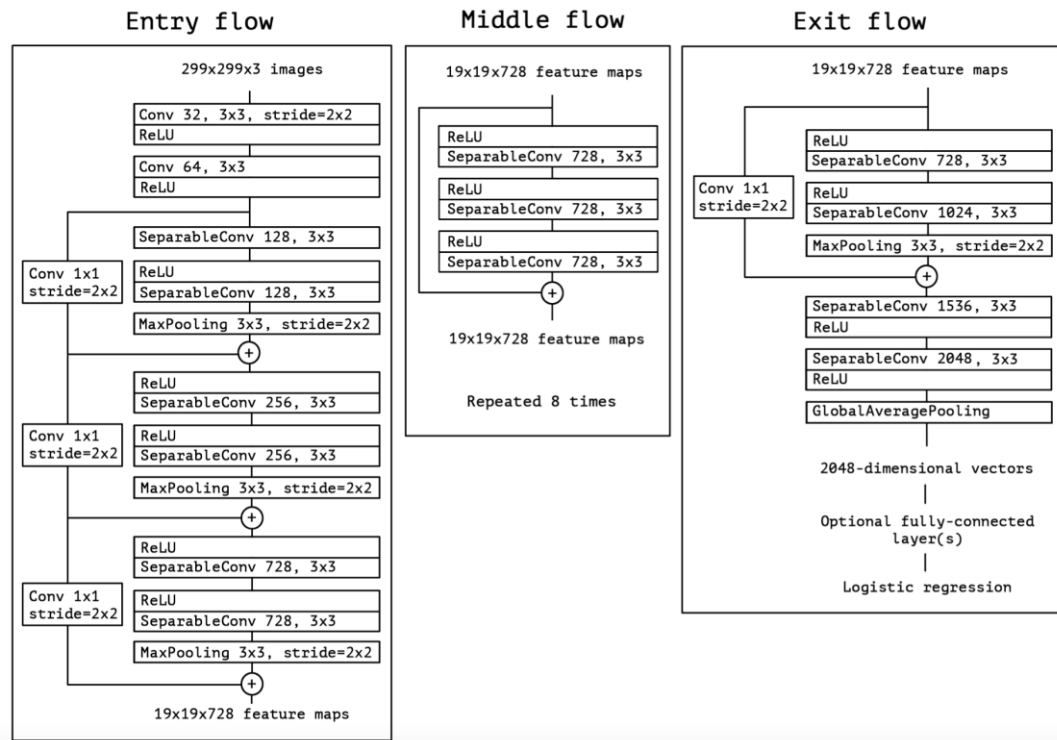
Kaynak: Islam, vd., 2018

2014 yılında, ImageNet büyük ölçekli nesne sınıflandırma yarışmasının galibi, Inception veya GoogleNet olarak bilinen ağ mimarisiydi. 22 katmana sahiptir ve 9 başlangıç modülünden oluşur. Farklı boyutlardaki evrişim ve maksimum havuzlama işlemleri her modüle dahil edilmiştir (Szegedy, vd., 2015).

2.5.1.3. Xception

Tamamen derinlemesine ayrılabilir evrişim katmanlarına dayanan bir evrişimli sinir ağı mimarisidir. Konvolüsyonel sinir ağlarının özellik haritalarında kanallar arası korelasyonların ve uzamsal korelasyonların haritalanması tamamen ayrıştırılabilir. Bu hipotez, Inception mimarisinin altında yatan hipotezin daha güçlü bir versiyonu olduğu için, önerilen mimariye “Extreme Inception” anlamına gelen Xception adını verilmiştir (Chollet, 2017).

Şekil 2.12: Xception Modeli



Kaynak: Chollet, 2017

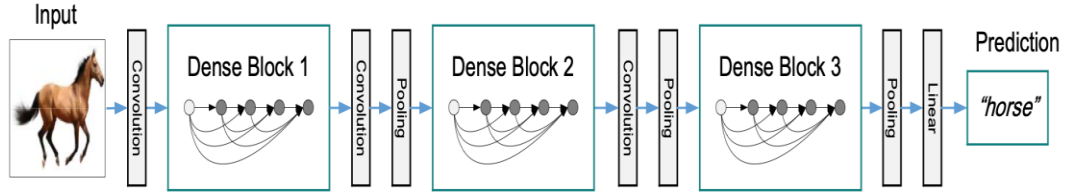
Xception mimarisi, ağıın özellik çıkarma tabanını oluşturan 36 evrişim katmanına sahiptir. 36 evrişim katmanı, ilk ve son modüller hariç, hepsinin çevresinde doğrusal artık bağlantılara sahip olan 14 modül halinde yapılandırılmıştır. Şekil 2.12’de gösterildiği gibi veriler önce giriş akışından, ardından sekiz kez tekrarlanan orta akıştan ve son olarak çıkış akışından geçer (Chollet, 2017).

2.5.1.4. DenseNet201

DenseNet, tüm katmanları (eşleşen özellik haritası boyutlarıyla) doğrudan birbirine bağladığımız Yoğun Bloklar (Dense Block) aracılığıyla katmanlar arasındaki

yoğun bağlantıları kullanan bir tür evrişimli sinir ağıdır (Huang, Liu, van der Maaten, & Weinberger, 2017).

Şekil 2.13: DenseNet Modeli



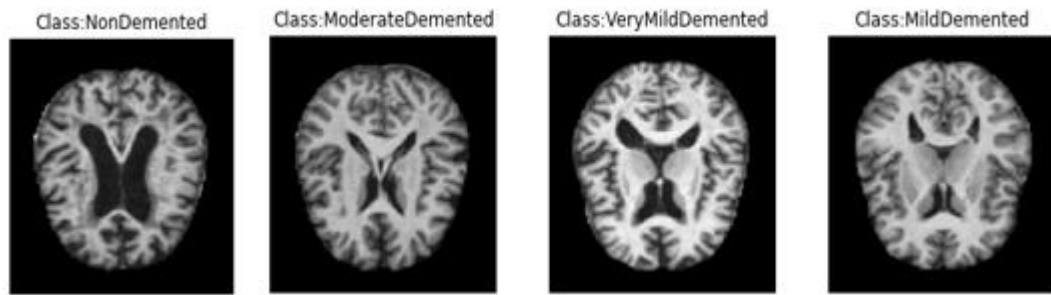
Şekil 2.13'te gösterildiği gibi ileri beslemeli yapıyı korumak için, her katman önceki tüm katmanlardan ek girdiler alır ve kendi özellik haritalarını sonraki tüm katmanlara iletir. DenseNet-201, 201 katman derinliğine sahip evrişimli bir sinir ağıdır. Ağ, 224'e 224'lük bir görüntü giriş boyutuna sahiptir.

ÜÇÜNCÜ BÖLÜM
MATERYAL ve YÖNTEM

3.1. VERİ SETİ (DATASET)

Alzheimer hastalığı veri seti, Çok Hafif Dementli (VMD – Very Mild Demented), Hafif Dementli (MID – Mild Demented), Orta Dementli (MOD – Moderate Demented), Demanssız (ND – Non Demented) olmak üzere dört sınıfın 6400 MR Görüntüsünden oluşan açık kaynaklı platform olan Kaggle'dan toplanmıştır (Sarvesh, 2019). Veri seti 176 x 208 görüntü boyutuna sahiptir. Görüntüler 224 x 224 olarak yeniden boyutlandırılmıştır. Dört sınıftan alınan örnek görüntüler Şekil 3.1’de gösterilmiştir.

Şekil 3.1: Veri Setinden Örnek Görüntüler



Kaynak: Sarvesh, 2019

Tablo 3.1, Kaggle platformundan elde edilen veri kümesindeki sınıfların dağılımını gösterir ve sınıflar arasında görüntü sayısı bakımından dengesizlik problemi görülmektedir.

Tablo 3.1: Veri Setinin Dağılımı

Class	Görüntü Sayısı
Demanssız	2240
Çok Hafif Demans	64
Hafif Demans	896
Orta Demans	3200

3.2. VERİ ARTIRMA (DATA AUGMENTATION)

Derin öğrenme çalışmalarında model performansının başarısı büyük veri setlerine dayanmaktadır. Veri setinin büyük olmadığı durumlarda modelde overfitting problemi görülebilir. Bu durumu en aza indirmek için veri artırma yöntemleri kullanılır (Shorten ve Khoshgoftaar, 2019). Tablo 3.2’de veri artırma yöntemlerini uygulamak için kullanılan parametreler verilmiştir.

Tablo 3.2: Veri Artırma Yöntemi için Kullanılan Parametreler

İsim	Açıklama	Değer
Rescale	Piksel değerlerini yeniden ölçeklendirmek	1./255
ZOOM	Görüntü yakınlaştırma	[.99, 1.01]
BRIGHT_RANGE	Parlaklık ayarı	[0.8, 1.2]
HORZ_FLIP	Görüntüyü yatay yönde döndürme	True
FILL_MODE	Girişin sınırları dışındaki noktalar verilen moda göre doldurulur	constant

3.3. SENTETİK AZINLIK AŞIRI ÖRNEKLEME TEKNİĞİ (SMOTE-SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE)

Sınıf dengesizliği (Class imbalance), Sentetik Azınlık Aşırı Örneklemme Tekniği (Synthetic Minority Over-Sampling Technique- SMOTE) kullanılarak giderilir. SMOTE, veri setindeki görüntü sayısı az olan sınıfları, sayıca çok olan sınıflarla eşleştirmek için rastgele çoğaltarak sınıf dengesizliği sorununu çözmek için veri setine uygulanır (Chawla, vd., 2002). Verilerin en yakın k komşusunu baz alarak yapay veriler üretir.

Tablo 3.3: Veri Setinin SMOTE Uygulandıktan Sonraki Dağılımı

Class	Görüntü Sayısı
Demanssız	3200
Çok Hafif Demans	3200
Hafif Demans	3200
Orta Demans	3200

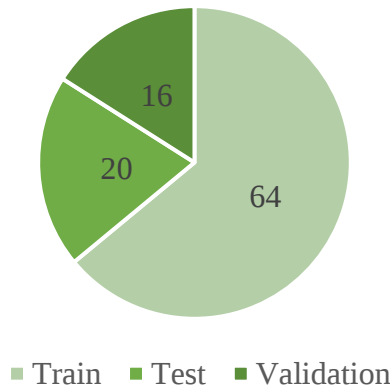
Random_state parametresi 42 olarak belirlenerek azınlık sınıfları SMOTE tekniği kullanılarak aşırı örneklendi. SMOTE aşırı öğrenmeyi (overfitting) engellemek ve model performansını arttırmak için kullanılmıştır. Tablo 3.3, SMOTE tekniği ile veri setinin

12800 görüntüye yükseltilmesi ile sınıflar arası dağılımı gösterir, her sınıf 3200 görüntü içerir.

3.4. VERİLERİN TRAIN, VALIDATION VE TEST SETLERİNE BÖLÜNMESİ

Eğitim verileri, makine öğrenimi modellerini eğitmek için kullanılan ilk verilerdir. Eğitim verileri, onlara nasıl tahminde bulunacaklarını veya istenen bir görevi nasıl gerçekleştireceklerini öğretmek için makine öğrenimi algoritmalarına verilir. Validation verileri, eğitim sırasında model performansımızı doğrulamak için kullanılan eğitim setinden ayrı bir veri setidir. Bu doğrulama süreci, modelin hiper-parametrelerini ve konfigürasyonlarını buna göre ayarlamamıza yardımcı olan bilgiler verir. Eğitimin doğru yönde ilerleyip ilerlemediğini bize söyleyen bir eleştirmen gibidir. Model, eğitim setinde eğitilir ve eş zamanlı olarak, model değerlendirmesi her epoch'tan sonra doğrulama setinde gerçekleştirilir. Veri kümesini bir doğrulama kümesine bölmenin ana fikri, modeldeki aşırı öğrenmeyi önlemektir. Yani model, eğitim kümesindeki örnekleri sınıflandırmada gerçekten iyi olur, ancak daha önce görmediği veriler üzerinde genelleştiremez ve doğru sınıflandırmalar yapamaz. Eğitim verileri ile oluşturulan modeli test etmek için de verilere ihtiyaç vardır. Bu verilere test verileri denir. Eğitimde kullanılan algoritmaların performansını değerlendirmek ve optimize etmek için kullanılır (Shrestha ve Mahmood, 2019).

Şekil 3.2: Veri Setinin Eğitim, Test ve Doğrulama Setlerine Bölünmesi



Önerilen yöntemde hold-out yöntemi kullanılarak veri setleri oluşturulmuştur. Eğitilen derin öğrenme modelinin, aynı zamanda eğitimini doğrulaması için gereken validasyon seti de sayısal olarak yeterli olmalıdır. SMOTE uygulanarak aşırı örnekleme

yapıldıktan sonraki 12800 görüntünün %64'ü eğitim, %20'si test ve %16'sı validation setlerine ayrılmıştır, şekil 3.2'de bu yüzdeler sunulmuştur.

3.5. EĞİTİMDE KULLANILAN DONANIMLAR VE YAZILIMLAR

Python Programlama Dili

Python, yaygın olarak kullanılan genel amaçlı, üst düzey bir programlama dilidir. İlk olarak 1991 yılında Guido van Rossum tarafından tasarlandı ve Python Yazılım Vakfı tarafından geliştirildi. Esas olarak kodun okunabilirliğine vurgu yapmak için geliştirilmiştir ve sözdizimi, programcıların kavramları daha az kod satırında ifade etmelerine olanak tanır. Python, bilişim teknolojilerinde üst düzey konular olan yapay zekâ, makine öğrenmesi ve derin öğrenme konularında kendini ispatlamış bir programlama dilidir.

Tensorflow, Keras Kütüphaneleri ve Python ile Derin Öğrenme

TensorFlow, makine öğrenimi için uçtan uca bir açık kaynak platformudur. Araştırmacıların makine öğrenimi alanındaki en son teknolojiyi kullanmalarına ve geliştiricilerin makine öğrenimi destekli uygulamaları kolayca oluşturup dağıtmalarına olanak tanıyan kapsamlı, esnek bir araç, kitaplık ve topluluk kaynakları ekosistemine sahiptir. Keras, Python ile yazılmış, makine öğrenimi platformu TensorFlow'un üzerinde çalışan bir derin öğrenme API'sidir. Hızlı denemeyi mümkün kılmaya odaklanılarak geliştirilmiştir.

3.6. ÖNERİLEN YAKLAŞIM

Bu tez çalışmasında veri setinin eğitilmesi için beş farklı derin transfer öğrenme modeli kullanılmıştır ve CNN tabanlı DEMXNET modeli önerilmiştir. Bütün modellerin başarısı karşılaştırılmıştır. Tüm modeller için aynı eğitim, doğrulama ve test verileri kullanılmıştır. Eğitim, test ve doğrulama veri sayısı bütün modeller için aynı oranda belirlenmiştir. Veri setinin tüm modeller için aynı ön işleme adımlarından geçirilmesi sağlanmıştır. Böylelikle eğitim sonrası test verileri için modellerin performanslarının doğru bir şekilde karşılaştırılması yapılmıştır. Veri setindeki görüntüler 224x224 olarak yeniden boyutlandırılmıştır. Tüm modeller için eğitim döngü sayısı (epoch) 50 olarak belirlenmiştir. Tek seferde modele verilecek olan örnek sayısı (batch size) tüm modeller için 32 olarak belirlenmiştir. Tüm modeller için RMSProp optimizasyon yöntemi

kullanılmıştır. Modellerin maliyetini ölçmek amacıyla kayıp fonksiyonu hesaplamasında çapraz doğrulama fonksiyonu kullanılmıştır.

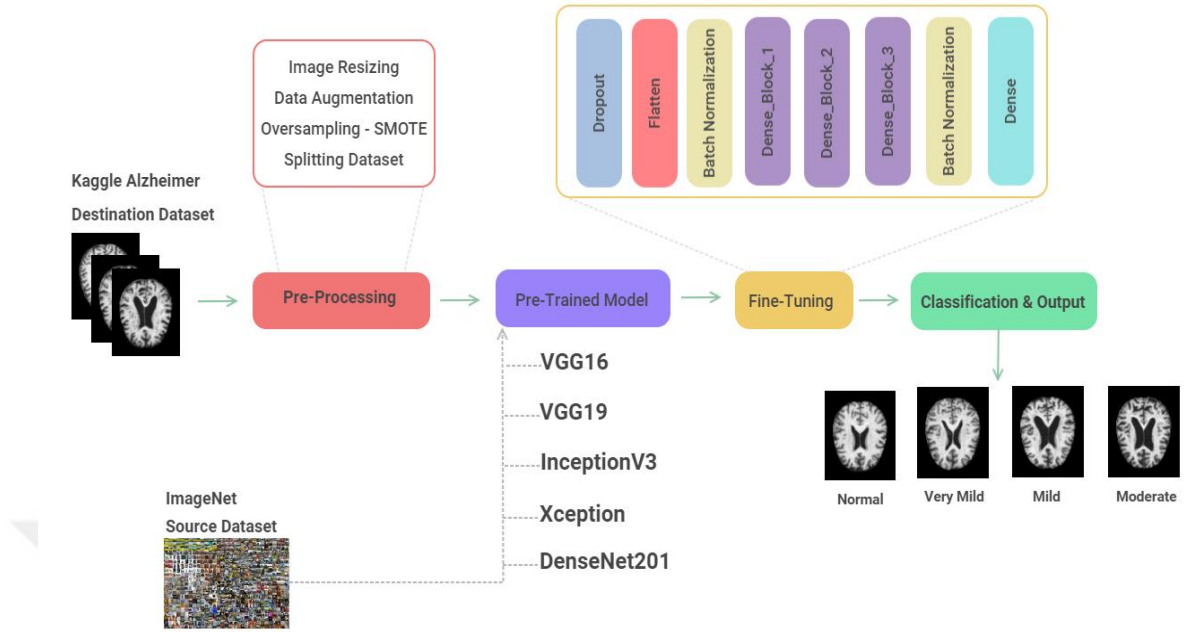
Tablo 3.4: Hiper-Parametre Ayarları

İsim	Değer
Epoch	50
Batch Size	32
Loss	CategoricalCrossentropy
Optimizer	RMSProp

Alzheimer hastalığının teşhis ve sınıflandırılmasında önerilen Derin transfer öğrenme modelleri çalışmasının ayrıntılı mimarisi şekil 3.3'te verilmiştir. Ön işleme adımları görüntü yeniden boyutlandırma, veri seti küçük olduğundan veri artırma, verilerdeki sınıf dengesizliği problemi için SMOTE uygulanması ve veri setinin eğitim, doğrulama ve test setlerine bölünmesidir. Daha sonra model eğitimi için kaynak olarak IMAGENET veri seti üzerinde önceden eğitilmiş ağırlıklar hedef veri setine aktarılmıştır. IMAGENET Large-Scale Visual Recognition Challenge veri seti 1,2 milyon görüntüden oluşur ve bu resimler 1000 kategoride sınıflanmaktadır. Kullanılan transfer öğrenme modelinin son tam bağlı katmanı, hedef eğitim verileri ile yeniden eğitildi.

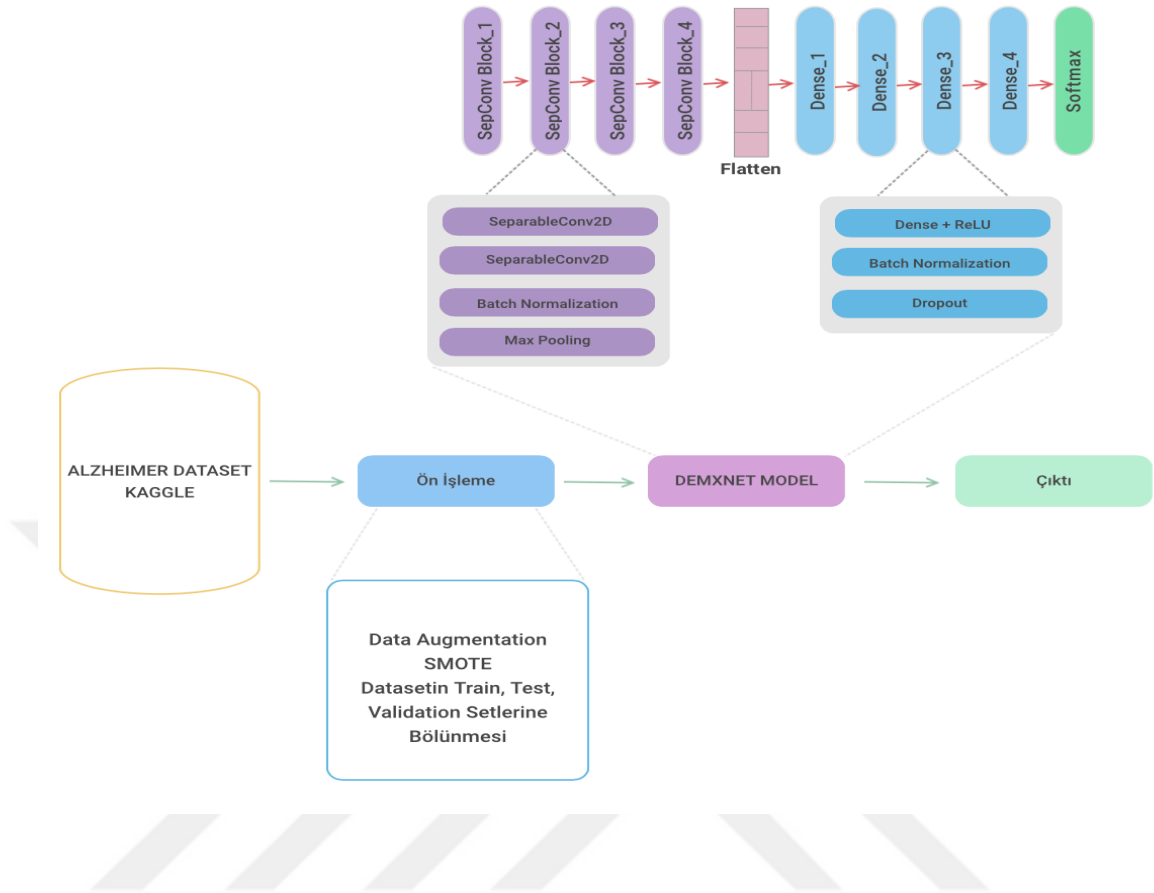
Önceden eğitilmiş modele fine-tuning yapılarak sırasıyla Dropout, Flatten, Batch Normalization, Dense Block_1, Dense_Block_2, Dense_Block_3, Batch Normalization ve Dense katmanları uygulanır. Dense Bloklar Dense Layer, Batch Normalization ve Max Pooling katmanlarından oluşur. Dense bloklardaki filtreler sırasıyla 512, 256, 128 ve Softmax sınıflandırma katmanındaki filtre, sınıf sayısı 4 olarak belirlenmiştir. Bütün modellerin fine-tuning (ince ayar) işlemi aynıdır. Bir CNN'i bir dizi önceden eğitilmiş ağırlıktan eğitmeye fine-tuning denir ve çeşitli uygulamalarda başarıyla kullanılmıştır. Yaygın bir uygulama, önceden eğitilmiş CNN'in son tam olarak bağlı katmanını, yeni hedef uygulamadaki sınıf sayısı kadar nörona sahip yeni bir tam bağlantılı katmanla değiştirmektir.

Şekil 3.3. Önerilen Transfer Öğrenme Yaklaşımı



Önerilen DEMXNET modeli içinde diğer transfer öğrenme modelleri için uygulanan ön işleme teknikleri, optimizasyon tekniği, loss fonksiyonu aynı seçilmiştir. Model 4 adet SepConv blok, Flatten katmanı, 4 Dense Blok ve Softmax sınıflandırma katmanlarından oluşmaktadır. SepConv Blok 2 adet Seperable Convolution katmanı, Batch Normalization ve Max Pooling katmanlarından oluşur. Bu bloklardaki filtreler sırasıyla 32, 64, 128 ve 256 olarak seçilmiştir. Önerilen modelde Seperable Convolution işlemi tercih edilmiştir. Giriş görüntüsünün her kanalında ayrı bir evrişim işlemi gerçekleştirmektedir. Derinlemesine ayrılabilir evrişim kullanımı, standart evrişime kıyasla eğitilebilir parametre sayısını azaltmaktadır. Bu da aşırı öğrenme problemini önleyebilmektedir. Ayrıca daha az hesaplama nedeniyle işlem maliyetini azaltmaktadır. Dense Bloklar aktivasyon fonksiyonu ReLu olan Dense katman, Batch normalization ve Dropout katmanlarından oluşur. Dense bloklardaki filtre boyutları da sırasıyla 512, 256, 128 ve 64'tür. Tablo 3.5'te DEMXNET modelinin detaylı mimarisi verilmiştir. Softmax fonksiyonu ile 4 sınıflı Alzheimer sınıflandırması yapılmıştır. Önerilen DEMXNET modeli Şekil 3.4'te sunulmuştur.

Şekil 3.4: Önerilen DEMXNET Modeli



Tablo 3.5: DEMXNET Model Detayları

Layer Type	Output Shape	Parameters
DEMXNET_Block_1	(None, 112, 112, 32)	1627
DEMXNET_Block_2	(None, 56, 56, 64)	7392
DEMXNET_Block_3	(None, 28, 28, 128)	27072
DEMXNET_Block_4	(None, 14, 14, 256)	103296
Flatten	(None, 50176)	0
DENSE_1	(None, 512)	25692672
DENSE_2	(None, 256)	132352
DENSE_3	(None, 128)	33408
DENSE_4	(None, 64)	8512
DENSE_SOFTMAX	(None, 4)	260

DÖRDÜNCÜ BÖLÜM
BULGULAR VE DENEYSEL SONUÇLAR

4.1. DEĞERLENDİRME METRİKLERİ

Genellikle sınıflandırıcı performans değerlendirmesi olarak adlandırılan değerlendirme metrikleri, bir sınıflandırma modelinin oluşturulmasında ve seçilmesinde esastır. Sınıflandırma algoritmalarının performansını değerlendirmek için birçok değerlendirme metriği geliştirilmiştir. Bunun nedeni, sınıflandırıcının bir metrikte güçlü, ancak diğerinde zayıf performans göstermesidir. Bu tez çalışması, modelin performansını doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 puanı olmak üzere 4 değerlendirme ölçütü kullanarak değerlendirir.

Şekil 4.1: Hata (Confusion) Matrisi

		Gerçek	
		Pozitif	Negatif
Tahmin	Pozitif	TP	FP
	Negatif	FN	TN

Şekil 4.1’de sunulan hata matrisi (confusion matrix), belirli bir sınıflandırma ağının, gerçek değerlerin zaten bilindiği belirli bir test veri kümesinde nasıl performans sergilediğini göstermek için kullanılan bir çizelgedir.

True Positive - TP

Bu değer, sınıflandırıcının pozitif bir sınıfı pozitif olarak doğru bir şekilde tahmin ettiği tahminlerin sayısını ifade eder.

True Negative - TN

Bu değer, sınıflandırıcının negatif sınıfı negatif olarak doğru bir şekilde tahmin ettiği tahmin edilen sayıyı ifade eder.

False Positive - FP

FP değeri, sınıflandırıcının negatif bir sınıfı pozitif olarak yanlış tahmin ettiği tahminlerin sayısını ifade eder.

False Negative - FN

FN değeri, sınıflandırıcının pozitif bir sınıfı negatif olarak yanlış tahmin ettiği tahminlerin sayısını ifade eder.

TP, FP, TN ve FN değerleri kullanılarak accuracy, recall, precision, specificity, F1-skor değerleri sırasıyla Eşitlik (1), Eşitlik (2), Eşitlik (3), Eşitlik (4) ve Eşitlik (5)'teki formüllerle bulunur.

Doğruluk (Accuracy) değeri, doğru tahmin edilen örneklerin toplam örneklere oranıdır.

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \quad (1)$$

Duyarlılık (recall), Pozitif olarak tahmin edilmesi gereken örneklerin ne kadarının pozitif olarak tahmin edildiğini hesaplar.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

Kesinlik (precision), pozitif olarak tahmin edilen değerlerin gerçekte kaçının pozitif olduğunu gösteren metriktir.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

Özgüllük (Specificity) (gerçek negatif oran), gerçekten negatif olması şartıyla negatif bir testin olasılığını ifade eder.

$$Specificity = \frac{TN}{TN+FP} \quad (4)$$

F1 Skoru, Precision ve Recall değerlerinin harmonik ortalamasıdır.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

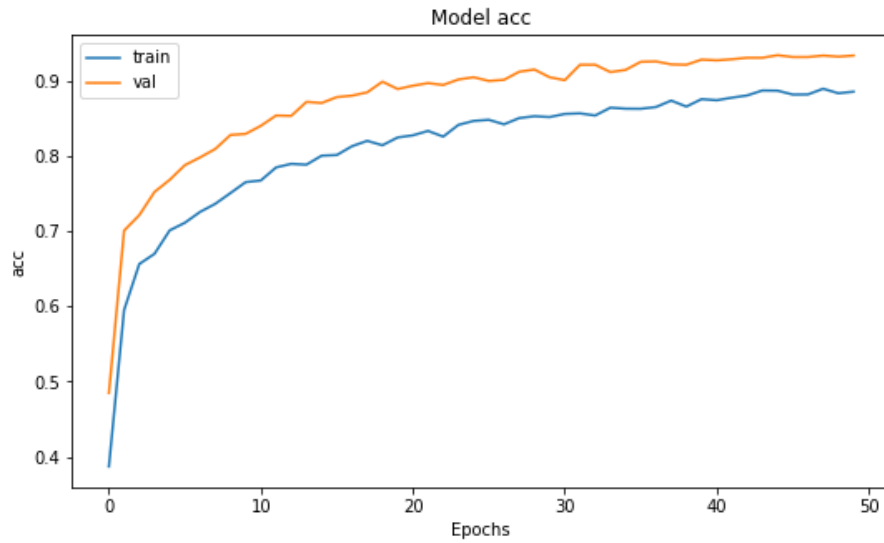
4.2. MODELLERİN TEST SONUÇLARI

4.2.1. VGG16 Modeli

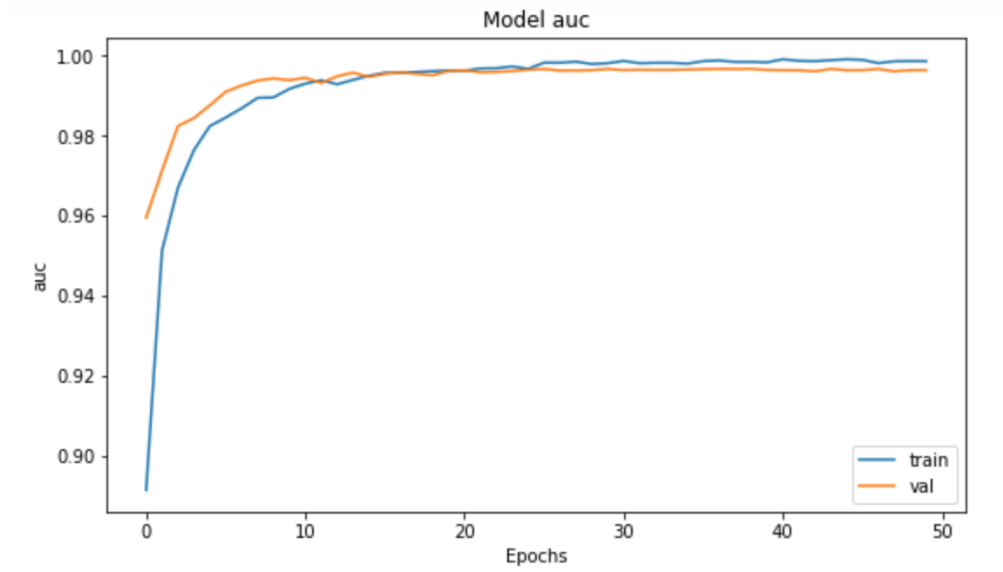
VGG16 modeli transfer öğrenme başlığı altında açıklanmıştır. Bu modelin Kaggle’da bulunan Alzheimer veri seti üzerinde başarısını görebilmek için transfer öğrenme yöntemi kullanılmıştır. Eğitim için epoch sayısı 50 ve batch size 32 olarak belirlenmiştir. Optimizasyon yöntemi olarak RMSProp kullanılmış olup kayıp fonksiyonu Categorical Cross Entropy seçilmiştir. Fine-Tuning uygulanmıştır. SMOTE’li ve SMOTE’siz iki ayrı senaryoda eğitim işlemi gerçekleştirilmiştir.

VGG16 modelinin SMOTE tekniği kullanılarak yapılan eğitim süreci ile alakalı Şekil 4.2, 4.3 ve 4.4’de sırasıyla Accuracy, AUC (The Area Under Curve) ve Loss grafikleri verilmiştir. SMOTE tekniğine sahip model, %88.90 genel eğitim doğruluğuna ve %93.31 doğrulama doğruluğuna ulaşır.

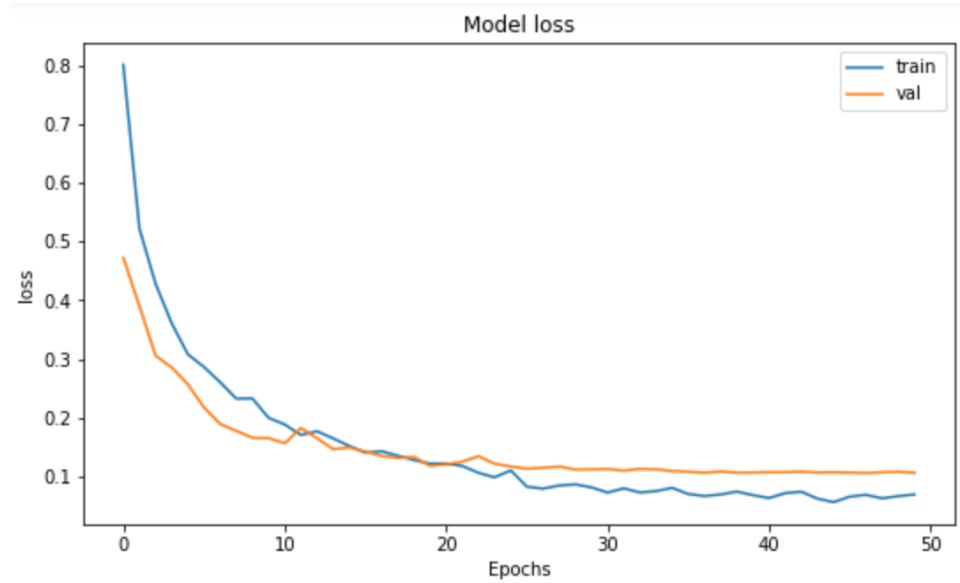
Şekil 4.2: SMOTE’li VGG16 Modelinin Accuracy Grafiği



Şekil 4.3: SMOTE’li VGG16 Modelinin AUC Grafiği

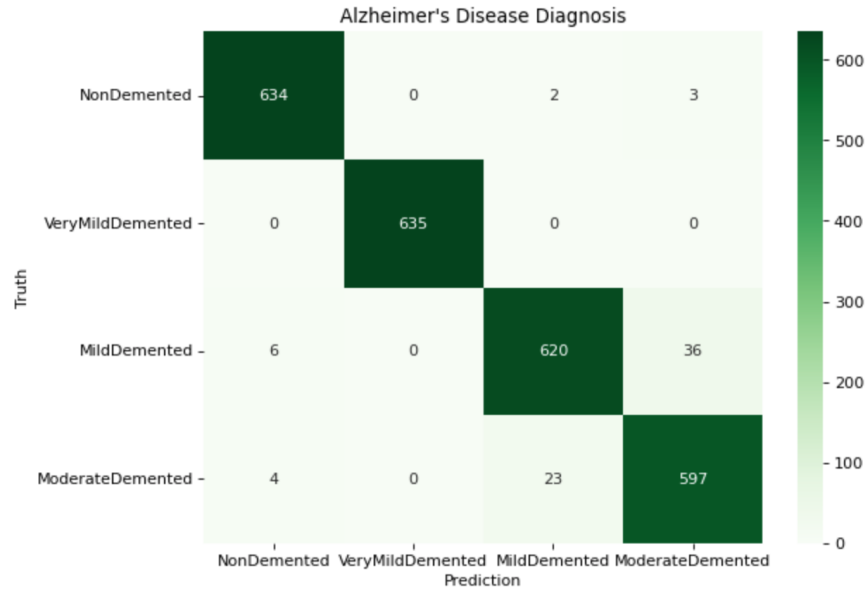


Şekil 4.4: SMOTE’li VGG16 Modelinin Loss Grafiği



Şekil 4.5’te sunulan hata matrisi, test veri kümesinden elde edilen model performansını göstermektedir. Hesaplama, ND’ye ait 639 görüntü, VMD’ye ait 635 görüntü, MD’ye ait 662 görüntü ve MOD’a ait 624 görüntü için yapılmıştır.

Şekil 4.5: SMOTE’li VGG16 Modelinin Hata Matrisi



Bireysel sınıf metrikleri, karışıklık matrisinden hesaplanır ve Tablo 4.1’de gösterilmiştir. Test veri setinden ND, VMD, MD ve MOD’un precision, recall ve F1-skor değeri verilmiştir.

Tablo 4.1: SMOTE’li VGG16 Modeli ile Test Verileri Kullanılarak Elde Edilen Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.96	0.97	0.97	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.89	0.87	0.88	662
Orta	0.87	0.87	0.87	624

VGG16 modeli kullanılarak Test veri seti üzerinden elde edilen sonuçlar Tablo 4.2’de gösterilmiştir. VGG16 modeli, SMOTE yöntemi ile %92,89 ve SMOTE yöntemi olmadan %84,38 test doğruluğuna ulaşmıştır.

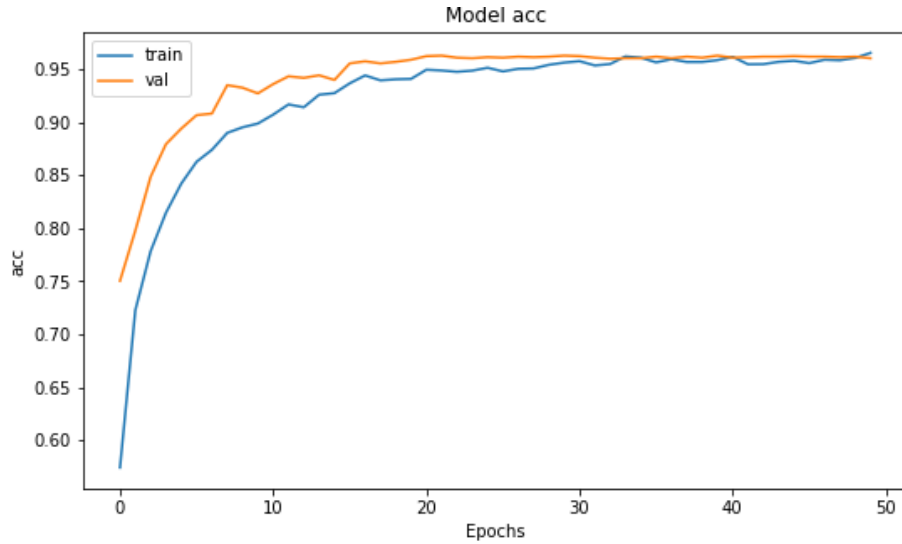
Tablo 4.2: VGG16 Modelinin SMOTE’li ve SMOTE’siz Varyasyonlarının Karşılaştırılması

Model	Testing Accuracy
VGG16 (with SMOTE)	92.89
VGG16	84.38

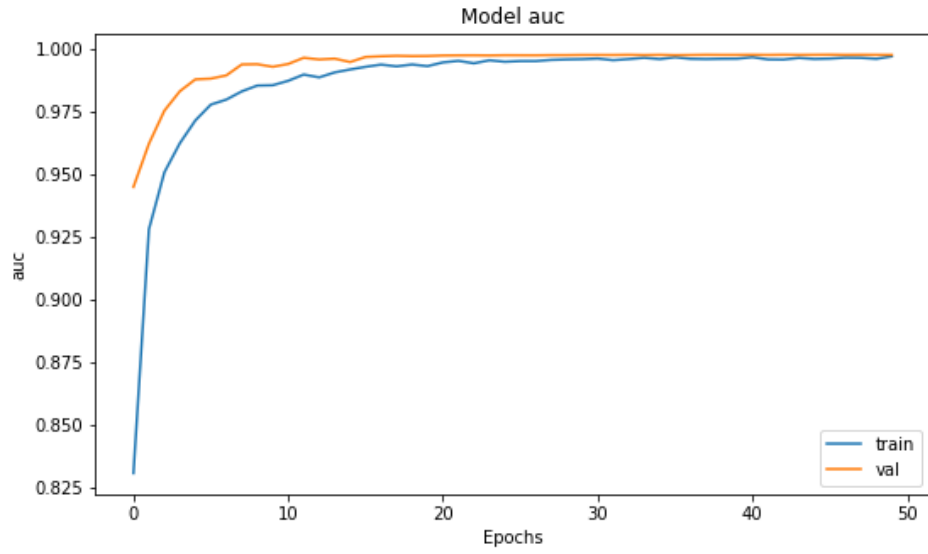
4.2.2. VGG19 Modeli

VGG19 modelinin SMOTE tekniği kullanılarak yapılan eğitim süreci ile alakalı Şekil 4.6, 4.7 ve 4.8’de sırasıyla accuracy, AUC ve Loss grafikleri verilmiştir. SMOTE tekniğine sahip model, tüm hastalık sınıflarında (ND, VMD, MD ve MOD) ortalama %99.95 eğitim veriseti doğruluğuna ve %96.04 doğrulama doğruluğuna ulaşır.

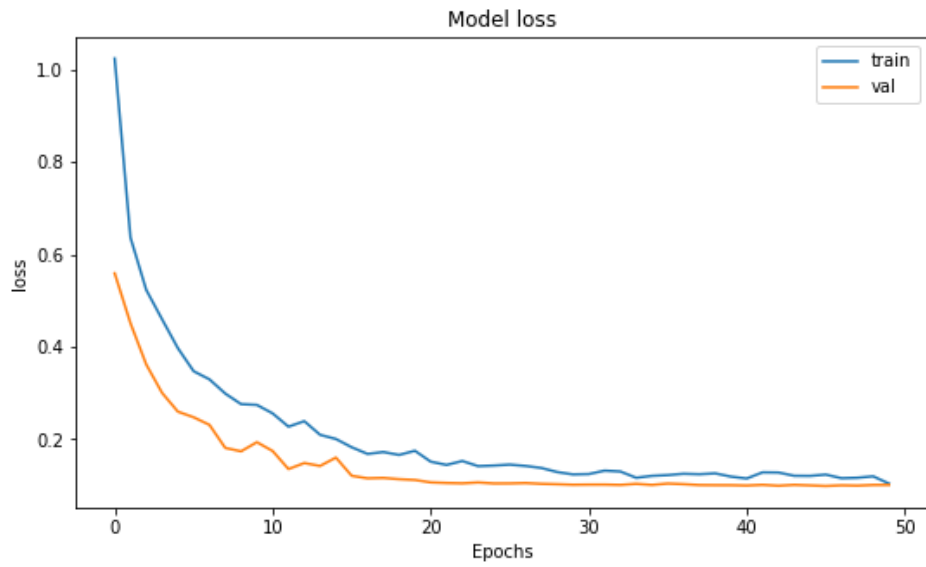
Şekil 4.6: SMOTE’li VGG19 Modelinin Accuracy Grafiği



Şekil 4.7: SMOTE’li VGG19 Modelinin AUC Grafiği

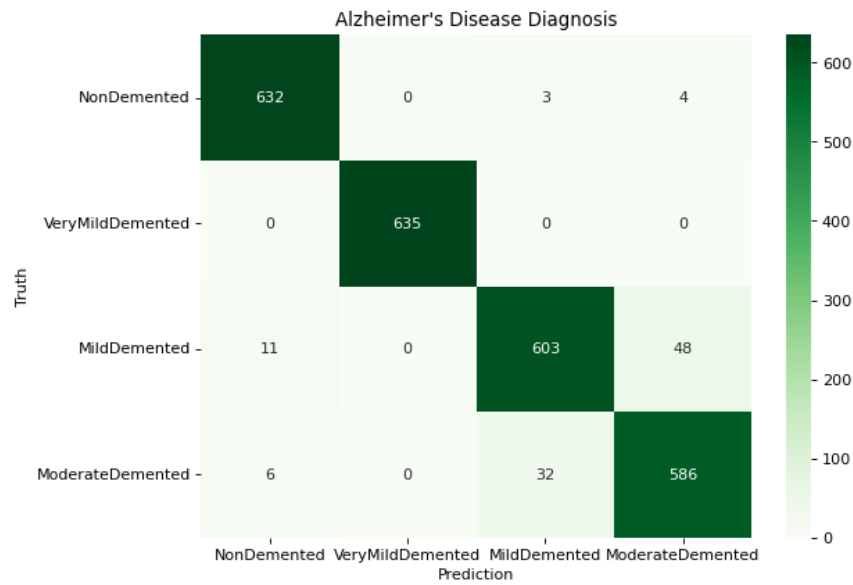


Şekil 4.8: SMOTE’li VGG19 Modelinin Loss Grafiği



Şekil 4.9’da SMOTE tekniği uygulanarak oluşturulan VGG19 modelinin hata matrisi verilmiştir.

Şekil 4.9: SMOTE’li VGG19 Modelinin Hata Matrisi



Tablo 4.3’te VGG19 modelinin test edildikten sonra elde edilen değerlendirme metrikleri verilmiştir.

Tablo 4.3: SMOTE’li VGG19 Modeli ile Elde Edilen Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.97	0.99	0.98	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.95	0.91	0.93	662
Orta	0.92	0.94	0.93	624

VGG19 modeli, test veri setini test eder ve Tablo 4.4’te gösterildiği gibi SMOTE ile %95,94 ve SMOTE olmadan %82,19 test doğruluğuna ulaşır.

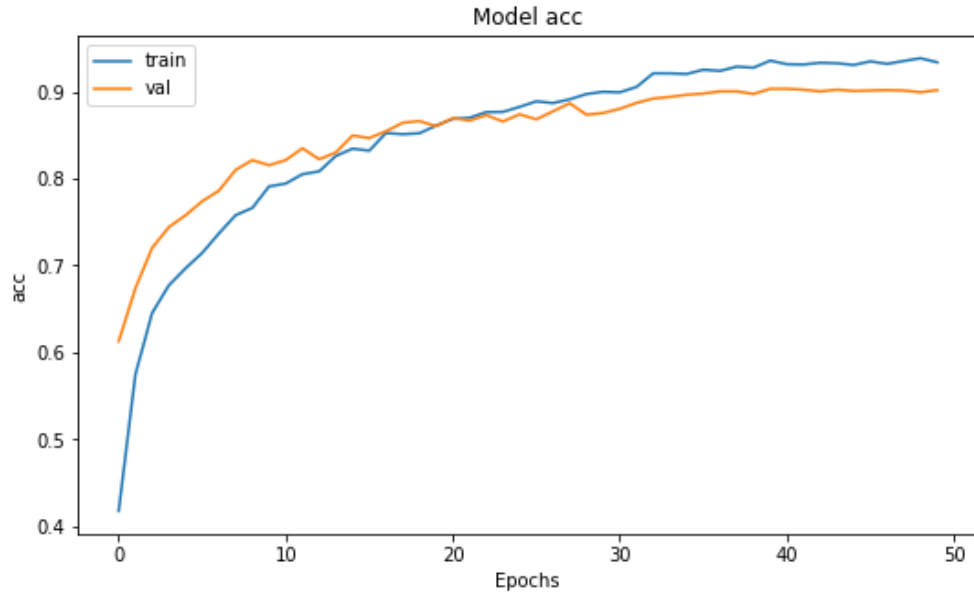
Tablo 4.4: VGG19 Modelinin SMOTE’li ve SMOTE’siz Varyasyonlarının Karşılaştırılması

Model	Testing Accuracy
VGG19 (with SMOTE)	95.94
VGG19	82.19

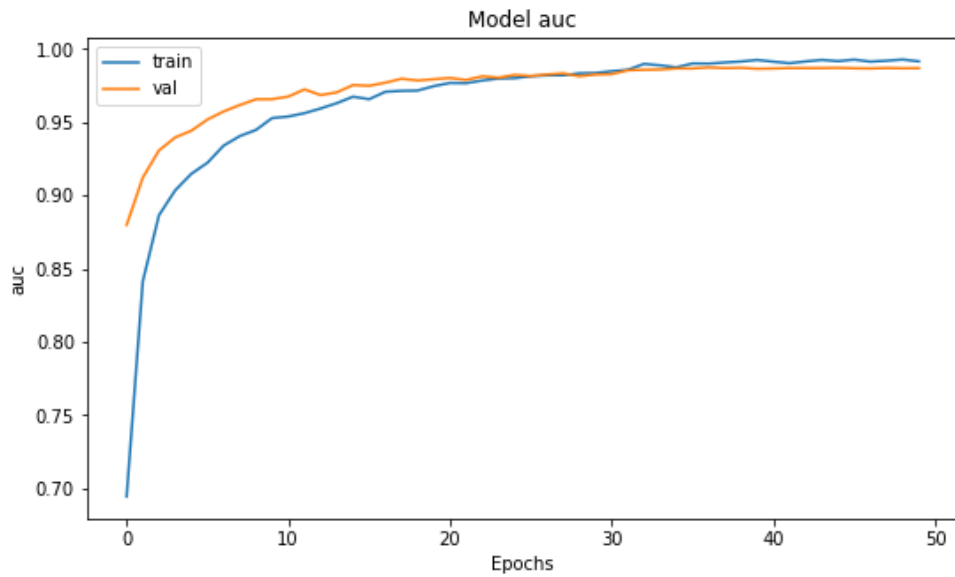
4.2.3. InceptionV3 Modeli

InceptionV3 modelinin SMOTE tekniği kullanılarak yapılan eğitim süreci ile alakalı Şekil 4.10, 4.11 ve 4.12’de sırasıyla accuracy, AUC ve Loss grafikleri verilmiştir. SMOTE tekniğine sahip model, tüm hastalık sınıflarında (ND, VMD, MD ve MOD) ortalama %99.91 eğitim veriseti doğruluğuna ve %91.02 doğrulama doğruluğuna ulaşır.

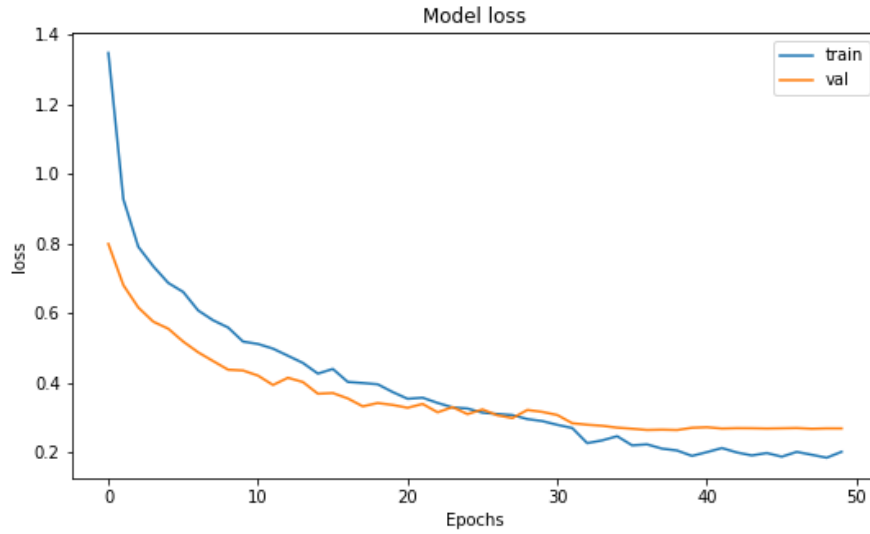
Şekil 4.10: SMOTE’li InceptionV3 Modelinin Accuracy Grafiği



Şekil 4.11: SMOTE’li InceptionV3 Modelinin AUC Grafiği

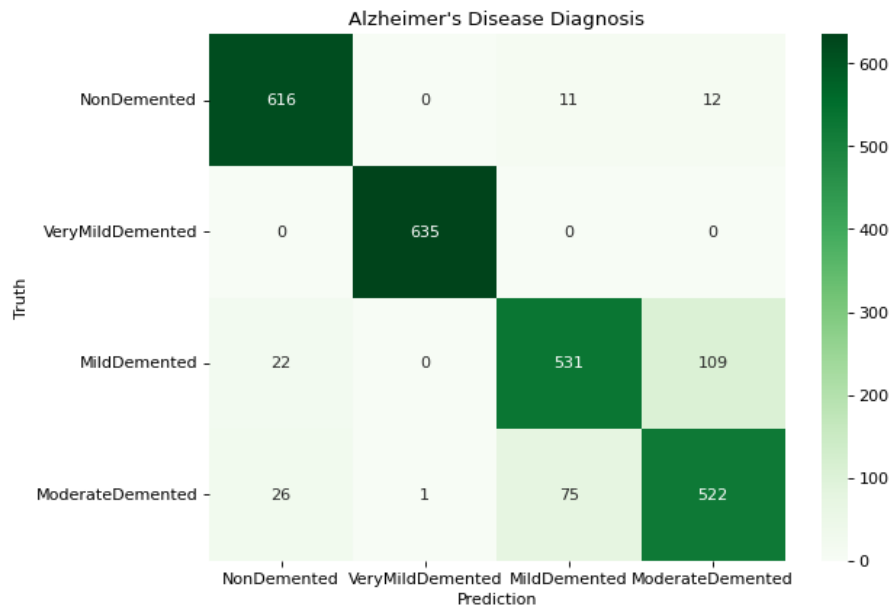


Şekil 4.12: SMOTE’li InceptionV3 Modelinin Loss Grafiği



Şekil 4.13’te InceptionV3 modeline ait hata matrisi verilmiştir. Test edilen model, MD etiketli olan görüntülerin 22’sini ND, 109’unu MID olarak 531’ini de doğru sınıf etiketi ile tahmin etmiştir. MOD etiketli görüntülerin ise 522’sini doğru, 75’ini MID, 1 tanesini VMD ve 26’sını ND olarak tahmin etmiştir. InceptionV3 modeli kullanılarak önerilen modelde ND ve VMD sınıflarının tahmin başarısı diğer sınıflara göre yüksektir.

Şekil 4.13: SMOTE’li InceptionV3 Modelinin Hata Matrisi



InceptionV3 hata matrisinden elde edilen sayısal değerlere göre oluşan sınıflandırma metrikleri Tablo 4.14’te sunulmuştur.

Tablo 4.14: SMOTE’li InceptionV3 Modeli ile Elde Edilen Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.93	0.96	0.95	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.86	0.80	0.83	662
Orta	0.81	0.84	0.82	624

InceptionV3 modeli, test veri setini test eder ve Tablo 4.6’da gösterildiği gibi SMOTE ile %90 ve SMOTE olmadan %75,47 test doğruluğuna ulaşır.

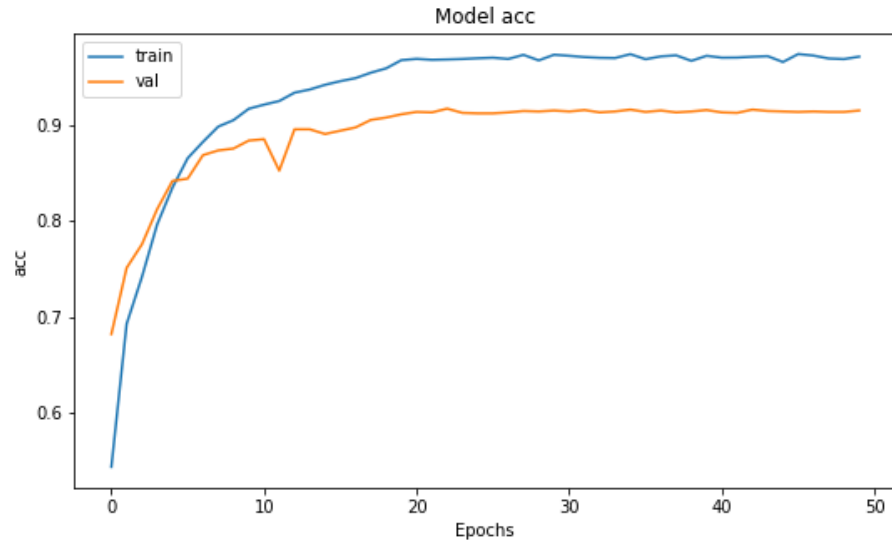
Tablo 4.6. SMOTE’li ve SMOTE’siz Çalışılan InceptionV3 Modelinin Test Doğruluklarının Karşılaştırılması

Model	Testing Accuracy
InceptionV3 (SMOTE)	90
InceptionV3	75.47

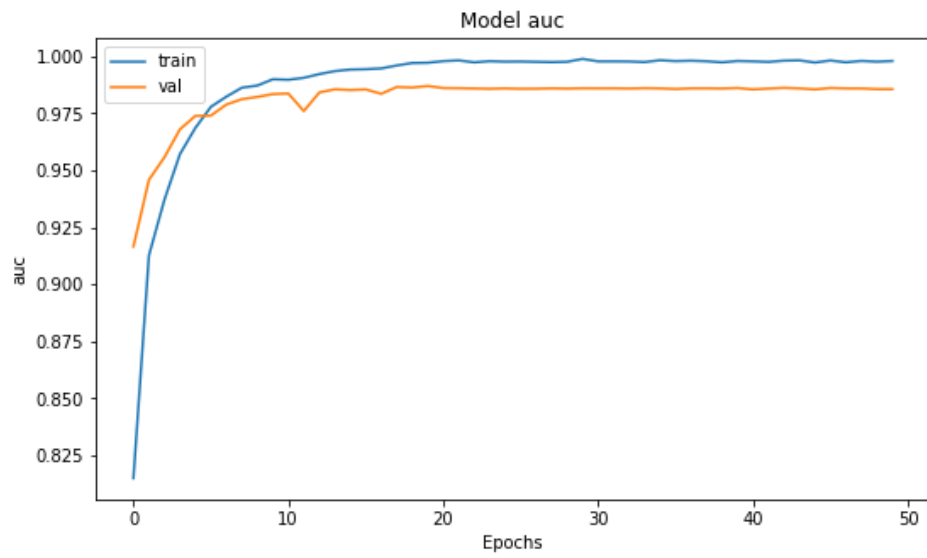
4.2.4. Xception Modeli

SMOTE’li önerilen Xception modelinin Şekil 4.14’te accuracy grafiği verilmiştir. Bu grafiğe göre model %97.17 eğitim doğruluğuna, %91.46 doğrulama doğruluğuna ulaşmıştır. Şekil 4.15 ve 4.16’da modelin AUC ve Loss grafikleri verilmiştir.

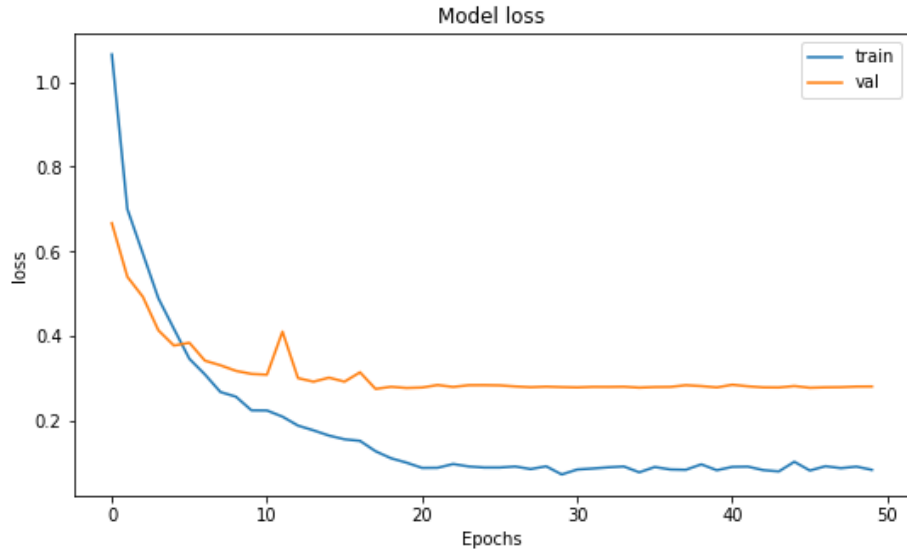
Şekil 4.14: SMOTE’li Önerilen Xception Modelinin Accuracy Grafiği



Şekil 4.15: SMOTE’li Önerilen Xception Modelinin AUC Grafiği

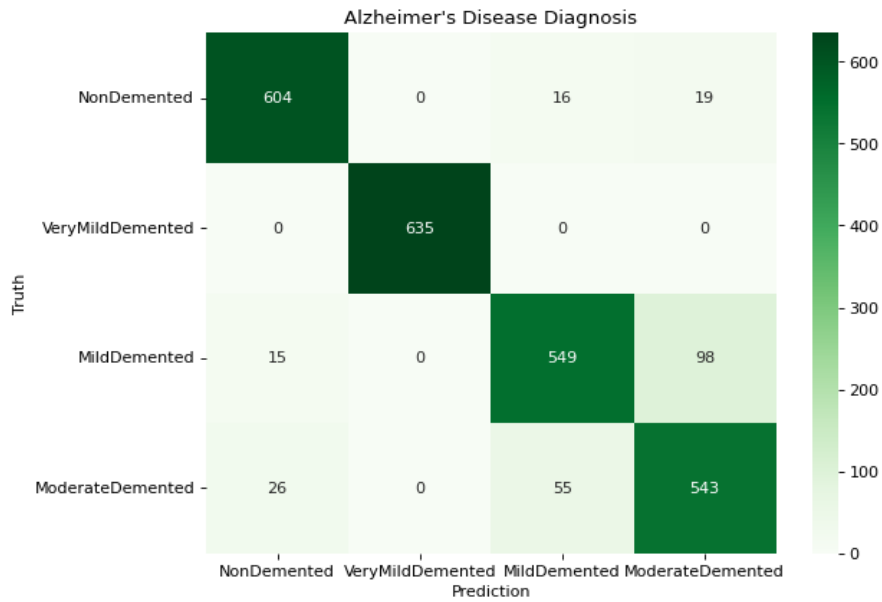


Şekil 4.16: SMOTE’li Önerilen Xception Modelinin Loss Grafiği



Şekil 4.17’de SMOTE uygulanarak önerilen Xception modelinin hata matrisi verilmiştir. Bu matrise göre MD ve MOD etiketli görüntülerin tahmini diğer sınıflara göre daha başarısız olmuştur.

Şekil 4.17: SMOTE’li Xception Modelinin Hata Matrisi



Tablo 4.7’de SMOTE uygulanarak Xception derin transfer öğrenme yönteminin kullanıldığı modelin test sonuçlarına yönelik sınıflandırma metrikleri verilmiştir.

Tablo 4.7: SMOTE’li Xception Modeli Test Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.94	0.95	0.94	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.89	0.83	0.86	662
Orta	0.82	0.87	0.85	624

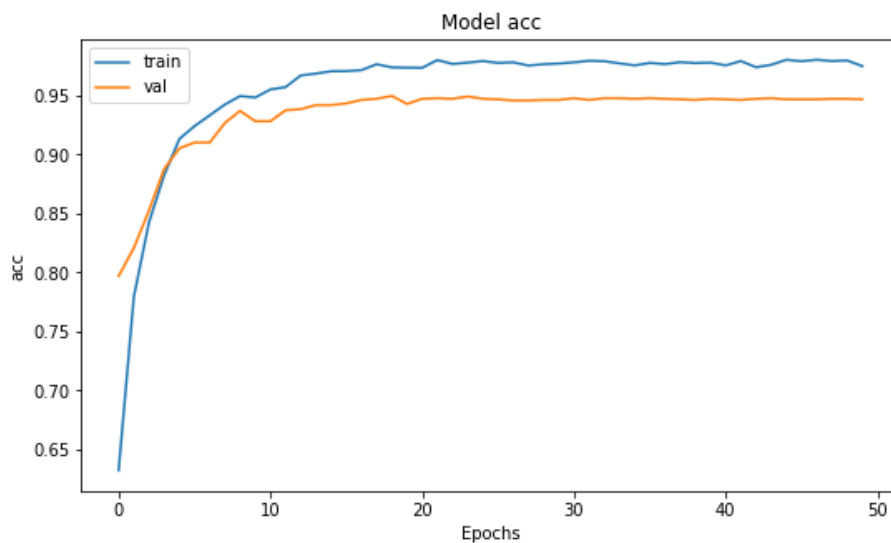
Xception modeli, test veri setini test eder ve Tablo 4.8’de gösterildiği gibi SMOTE ile %91,05 ve SMOTE olmadan %77,03 test doğruluğuna ulaşır.

Tablo 4.8: SMOTE’li ve SMOTE’siz Çalışılan Xception Modelinin Test Doğruluklarının Karşılaştırılması

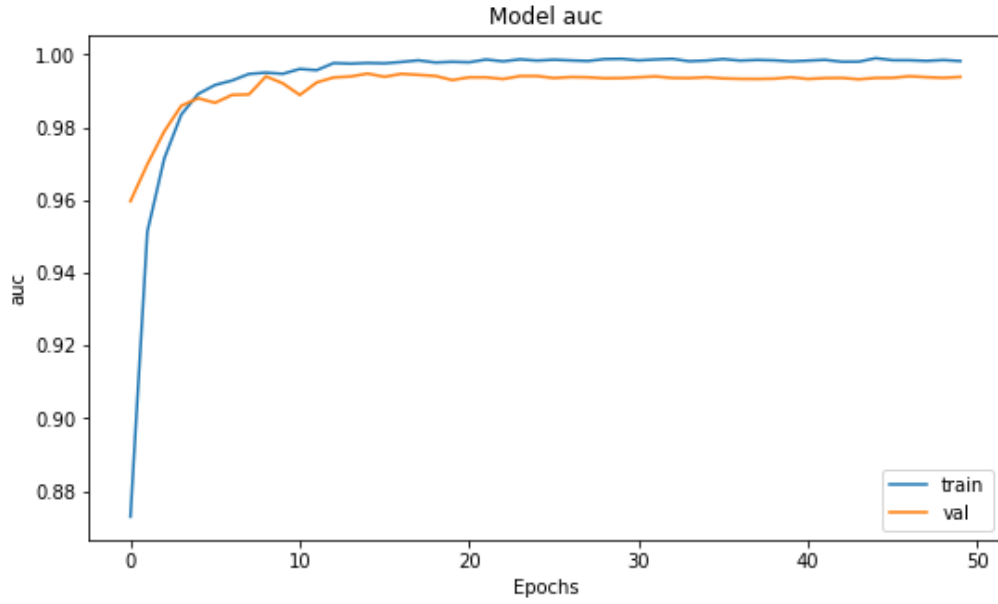
Model	Testing Accuracy
Xception (SMOTE)	91.05
Xception	77.03

4.2.5. DenseNet201 Modeli

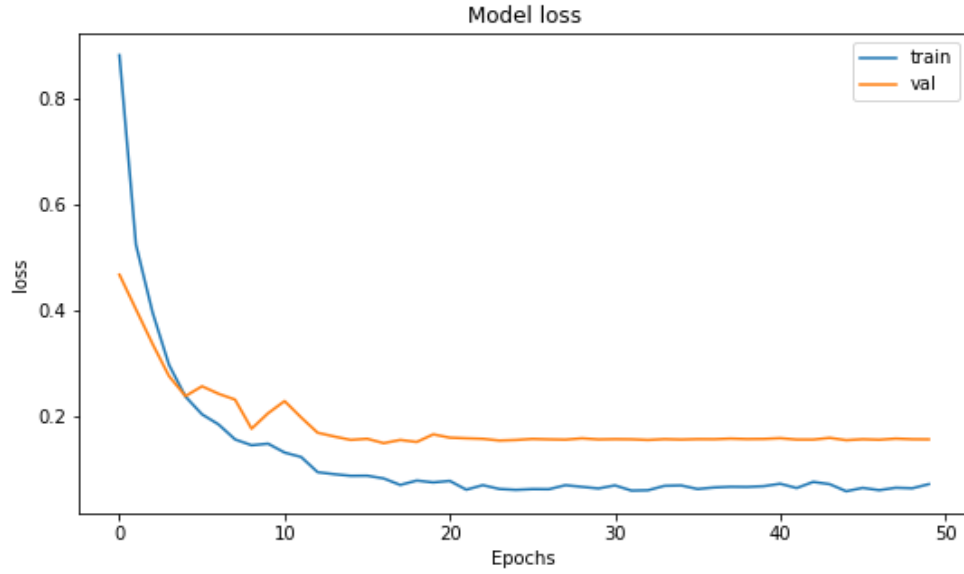
Derin Transfer Öğrenme modellerinden olan DenseNet201 fine-tuning yapılmış ve modelin SMOTE’li, SMOTE’siz varyasyonlarının performansları karşılaştırılmıştır. SMOTE’li önerilen DenseNet201 modelinin Şekil 4.18’de accuracy grafiği verilmiştir. Bu grafiğe göre model %100 eğitim doğruluğuna, %94,68 doğrulama doğruluğuna ulaşmıştır. Şekil 4.19 ve 4.20’de modelin sırasıyla AUC ve Loss grafikleri verilmiştir.

Şekil 4.18: SMOTE’li Önerilen DenseNet201 Modelinin Accuracy Grafiği

Şekil 4.19: SMOTE’li Önerilen DenseNet201 Modelinin AUC Grafiği

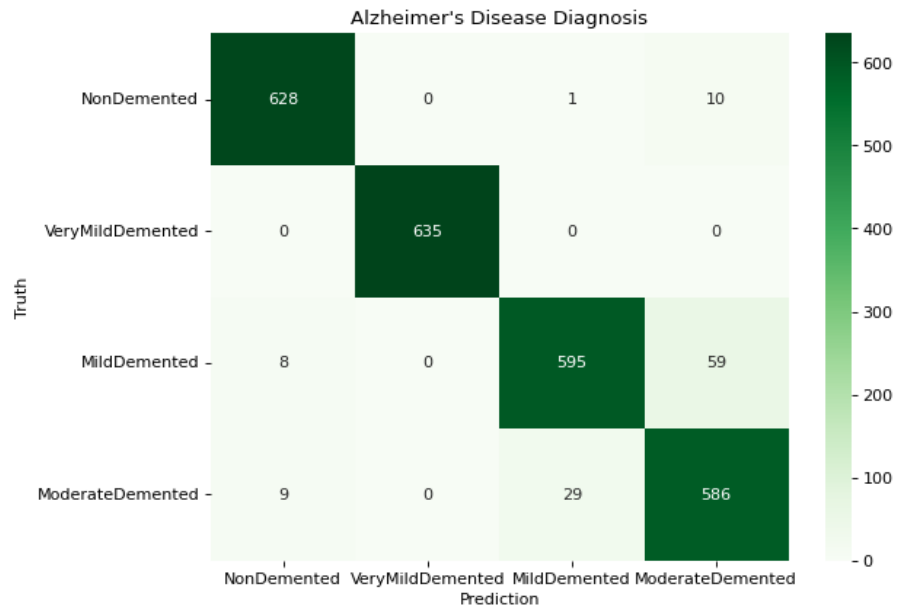


Şekil 4.20: SMOTE’li Önerilen DenseNet201 Modelinin Loss Grafiği



Şekil 4.21’de SMOTE uygulanılarak önerilen DenseNet201 modelinin hata matrisi verilmiştir. Bu matrise göre MD ve MOD etiketli görüntülerin tahmini diğer sınıflara göre daha başarısız olmuştur.

Şekil 4.21: SMOTE’li Önerilen DenseNet201 Modelinin Hata Matrisi



Tablo 4.9’da SMOTE uygulanarak DenseNet201 derin transfer öğrenme yönteminin kullanıldığı modelin test sonuçlarına yönelik sınıflandırma metrikleri verilmiştir.

Tablo 4.9: SMOTE’li DenseNet201 Modeli Test Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.97	0.98	0.98	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.95	0.90	0.92	662
Orta	0.89	0.94	0.92	624

DenseNet201 modeli, test veri setini test eder ve Tablo 4.10’da gösterildiği gibi SMOTE ile %95,47 ve SMOTE olmadan %88,98 test doğruluğuna ulaşır.

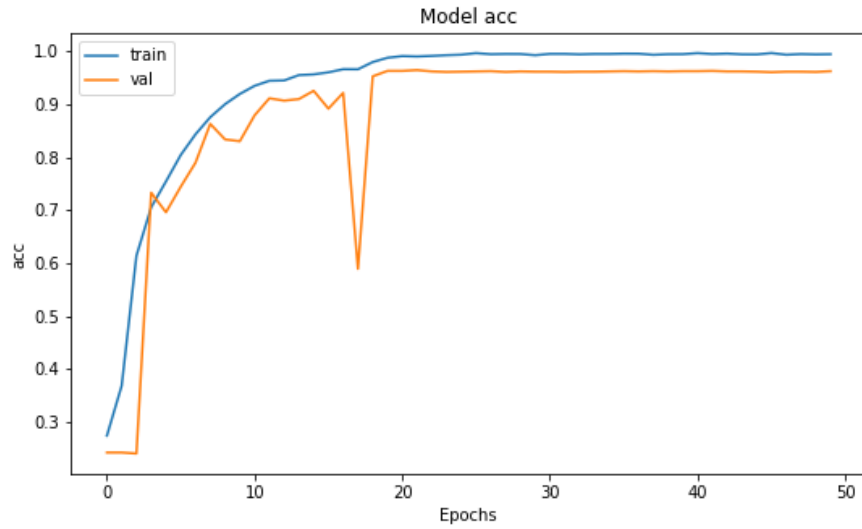
Tablo 4.10: SMOTE’li ve SMOTE’siz Çalışılan DenseNet201 Modelinin Test Doğruluklarının Karşılaştırılması

Model	Testing Accuracy
DenseNet201 (SMOTE)	95.47
DenseNet201	88.98

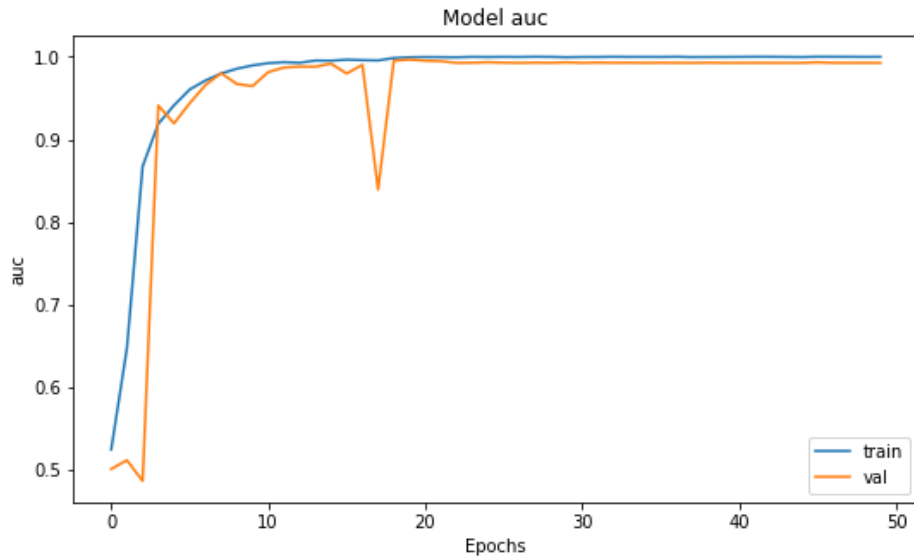
4.2.6. DEMXNET Modeli

SMOTE'li önerilen DEMXNET modelinin Şekil 4.22'de accuracy grafiği verilmiştir. Bu grafiğe göre model %99.66 eğitim doğruluğuna, %96.24 doğrulama doğruluğuna ulaşmıştır. Şekil 4.23 ve 4.24'te modelin sırasıyla AUC ve Loss grafikleri verilmiştir.

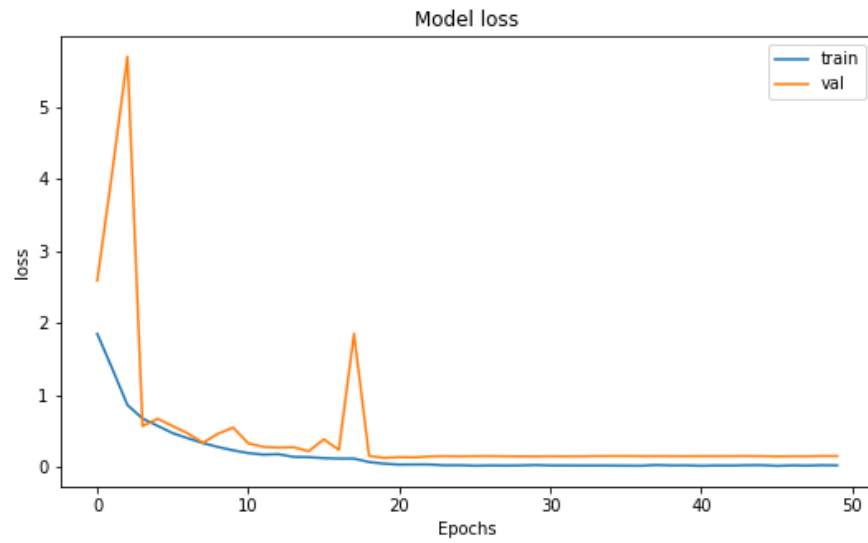
Şekil 4.22: SMOTE'li Önerilen DEMXNET Modelinin Accuracy Grafiği



Şekil 4.23: SMOTE'li Önerilen DEMXNET Modelinin AUC Grafiği

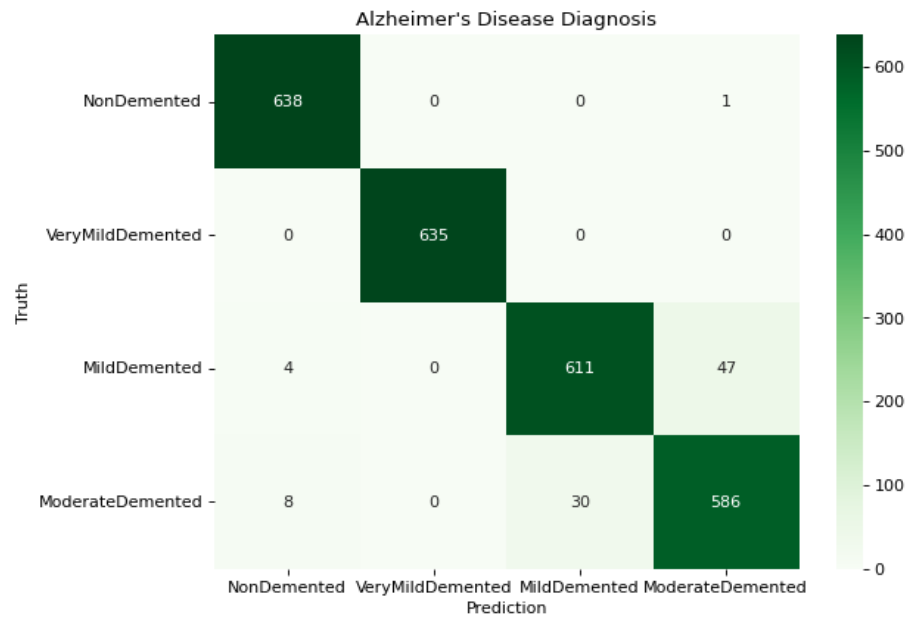


Şekil 4.24: SMOTE’li Önerilen DEMXNET Modelinin Loss Grafiği



Şekil 4.25’te SMOTE uygulanılarak önerilen DEMXNET modelinin hata matrisi verilmiştir.

Şekil 4.25: SMOTE’li Önerilen DEMXNET Modelinin Hata Matrisi



Tablo 4.11’de SMOTE uygulanarak DEMXNET derin transfer öğrenme yönteminin kullanıldığı modelin test sonuçlarına yönelik sınıflandırma metrikleri verilmiştir.

Tablo 4.11: SMOTE’li DEMXNET Modeli Test Sınıflandırma Raporu

Class	Precision	Recall	F1-Score	Support
Demanssız	0.98	1.00	0.99	639
Çok Hafif	1.00	1.00	1.00	635
Hafif	0.95	0.92	0.94	662
Orta	0.92	0.94	0.93	624

DEMXNET modeli, test veri setini test eder ve Tablo 4.12’de gösterildiği gibi SMOTE ile %96,48 ve SMOTE olmadan %64,06 test doğruluğuna ulaşır.

Tablo 4.12: SMOTE’li ve SMOTE’siz Çalışılan DEMXNET Modelinin Test Doğruluklarının Karşılaştırılması

Model	Test Doğruluğu
DEMXNET (SMOTE)	96.48
DEMXNET	64.06

4.3. MODELLERİN PERFORMANS ANALİZİ

Bütün modellerin eğitimi SMOTE’li ve SMOTE’siz olarak yapılmış olup testleri gerçekleştirilmiştir. Tablo 4.13’te bütün modellerin eğitim (train) doğruluğu, doğrulama (validation) doğruluğu ve test doğruluğu değerleri verilmiştir. Buna göre en yüksek doğruluk oranlarına sahip model (test %96,48, validation %96,24 ve train %99,66) önerilen DEMXNET modeli olmuştur. İkinci en yüksek doğruluk oranlarına sahip olan model ise fine tuning yapılmış VGG19 (test %95,94, validation %96,04 ve train accuracy %99,95) modelidir. En başarısız model ise SMOTE tekniği uygulanmadan eğitilen %64,06 test doğruluğuna sahip DEMXNET modelidir. Bu tabloya göre veri setindeki sınıf dengesizliği problemini gidermek için kullanılan SMOTE tekniği ile eğitilen modellerin daha yüksek performanslara sahip olduğu görülmektedir.

Tablo 4.13: Modellerin Performans Analizi

Model	Train Acc	Validation Acc	Test Acc
VGG16	83.28	84.86	84.38
VGG16 (with SMOTE)	88.90	93.31	92.89
VGG19	79.59	82.03	82.19
VGG19 (with SMOTE)	99.95	96.04	95.94
InceptionV3	87.48	75.39	75.47
InceptionV3 (with SMOTE)	99.91	91.02	89.65
Xception	94.56	76.95	77.03
Xception (with SMOTE)	97.17	91.46	91.05
DenseNet201	100	88.09	88.98
DenseNet201 (with SMOTE)	100	94.68	95.47
DEMXNET	69.24	63.67	64.06
DEMXNET (with SMOTE)	99.66	96.24	96.48



BEŞİNCİ BÖLÜM
SONUÇLAR VE ÖNERİLER

5.1. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında derin transfer öğrenme yöntemlerinden olan VGG16, VGG19, InceptionV3, Xception ve DenseNet201 modelleri ve önerilen DEMXNET modelinin performansları karşılaştırılmıştır. Veri seti Kaggle'dan elde edilmiş olup sınıf dengesizliği problemini gidermek amacıyla önerilen her modele SMOTE tekniği uygulanmıştır. Önerilen modellerin performanslarının karşılaştırılması için aynı hiper parametre ayarları ve Fine-tuning adımları uygulanmıştır.

Önerilen DEMXNET modeli karşılaştırılan transfer öğrenme yöntemlerinin başarısını geçmiştir. Elde edilen sonuçlar umut verici olsa da veri setinin boyutu artırılıp gelecek çalışmalarda önerilen model test edilecektir. Farklı modeller üzerinde eğitim işlemi ve performans değerlendirmeleri yapılabilir. Oluşturulan modellerin performanslarını artırmak amacıyla hiper parametre ayarları ve Fine-tuning adımlarında değişiklikler yapılabilir.

KAYNAKÇA

- Akgözlüoğlu, K., & Özçift, A. (2021). *Transfer öğrenmesi tekniği tabanlı derin öğrenme yöntemiyle ürün tanıma* (Yayımlanmamış Yüksek Lisans Tezi). T.C. Manisa Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü, Manisa.
- Alom, M. Z., vd. (2019). A state-of-the-art survey on deep learning theory and architectures. *Electronics*, 8(3), 292.
- Alzubaidi, L., vd. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 1-74.
- Andresen, S. L. (2002). John McCarthy: Father of AI. *IEEE Intelligent Systems*, 17(5), 84-85.
- Auer, P., Burgsteiner, H. ve Maass, W. (2008). A learning rule for very simple universal approximators consisting of a single layer of perceptrons. *Neural Networks*, 21(5), 786-795.
- Baumgart, M., vd. (2015). Summary of the evidence on modifiable risk factors for cognitive decline and dementia: A population-based perspective. *Alzheimer's and Dementia*, 11(6), 718-726.
- Bilgic B., vd. (2020). *Alzheimer ve diğer demans hastalıkları klinik protokolü*. Ankara: T.C. Sağlık Bakanlığı.
- Boureau, Y. L., Ponce, J. ve LeCun, Y. (2010, 21-24 Haziran). *A theoretical analysis of feature pooling in visual recognition* [Sözlü bildiri]. In Proceedings of the 27th international conference on machine learning, ICML, Haifa.
- Brownlee, J. (2018). What is the difference between a batch and an epoch in a neural Network. Retrieved from: <https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/>
- Caruana, R., & Niculescu-Mizil, A. (2006, Haziran). *An empirical comparison of supervised learning algorithms* [Sözlü bildiri]. In proceedings of the 23rd international conference on machine learning, ICML, Pittsburgh.
- Chawla, N. vd. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

- Chollet, F. (2017, 21-27 Temmuz). *Xception: Deep learning with depthwise separable convolutions* [Sözlü bildiri]. 30th IEEE conference on computer vision and pattern recognition, IEEE, Honolulu.
- Cipriani, G., vd. (2011). Alzheimer and his disease: A brief history. *Neurological Sciences*, 32(2), 275-279.
- Dike, H. U., vd. (2018, 25-27 Ekim). *Unsupervised Learning Based On Artificial Neural Network: A Review* [Sözlü bildiri]. 2018 IEEE International Conference on Cyborg and Bionic Systems, CBS 2018, IEEE, Shenzhen.
- Dubey Sarvesh. (2019). Alzheimer's dataset. Retrieved from: <https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images>
- Elfallah, S., vd. (2021, 25-27 Mayıs). Investigating the impact of different data representation with several classification models on magnetic resonance imaging (MRI) [Sözlü bildiri]. 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering, IEEE, Tripoli.
- Esteva, A., vd. (2019). A guide to deep learning in healthcare. *Nature Medicine*, 25(1), 24-29.
- Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1(2), 119-130.
- Gharbi-Meliani, A., vd. (2021). The association of APOE ϵ 4 with cognitive function over the adult life course and incidence of dementia: 20 years follow-up of the Whitehall II study. *Alzheimer's Research and Therapy*, 13(1), 1-11.
- Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. *Metabolism: Clinical and Experimental*, 69, S36-S40.
- He, G., vd. (2019, 23-25 Ağustos). *Alzheimer's disease diagnosis model based on three-dimensional full convolutional densenet* [Sözlü bildiri]. Proceedings - 10th International Conference on Information Technology in Medicine and Education, IEEE, Quingdao.
- Hebb, D. O. (2002). *The organization of behaviour : A neuropsychological theory* (1st ed.). New York: Psychology Press.

- Huang, G., vd. (2017, 21-26 Temmuz). *Densely connected convolutional networks* [Sözlü bildiri]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Honolulu.
- İnik Özkan, & Ülker Erkan. (2017). Derin öğrenme ve görüntü analizinde kullanılan derin öğrenme modelleri. *Gaziosmanpasa Journal of Scientific Research*, 6(3), 85-104.
- Ioffe, S., & Szegedy, C. (2015). *Batch normalization: Accelerating deep network training by reducing internal covariate shift* [Sözlü bildiri]. In International conference on machine learning, PMLR, Lille.
- Isaac Abiodun, O., vd. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), 938.
- Islam, M. S., vd. (2018). IncePTB: A CNN based classification approach for recognizing traditional Bengali games. *Procedia Computer Science*, 143, 595-602.
- Kandel, I., & Castelli, M. (2020). The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT Express*, 6(4), 312-315.
- Knopman, D. S., vd. (2021). Alzheimer disease. *Nature Reviews Disease Primers*, 7(1), 1-21.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90.
- Krogh, A. (2008). What are artificial neural networks? *Nature biotechnology*, 26(2), 195-197.
- Lazarczyk, M. J., vd. (2012). Preclinical Alzheimer disease: Identification of cases at risk among cognitively intact older individuals. *BMC Medicine*, 10(1), 1-13.
- LeCun, Y., vd. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2323.
- Li, Y. (2018). Deep reinforcement learning: An overview. Retrieved from: <https://arxiv.org/abs/1810.06339>

- Bkassiny, M., Li, Y., & Jayaweera, S. K. (2012). A survey on machine-learning techniques in cognitive radios. *IEEE Communications Surveys & Tutorials*, 15(3), 1136-1159.
- Fang, X., Luo, H., & Tang, J. (2005). Structural damage detection using neural network with learning rate improvement. *Computers & structures*, 83(25-26), 2150-2161.
- Mathew, N. A., Vivek, R. S. ve Anurenjan, P. R. (2018, 5-7 Temmuz). *Early diagnosis of Alzheimer's disease from MRI images using PNN* [Sözlü bildiri]. 2018 International CET Conference on Control, Communication, and Computing (IC4), IEEE, Thiruvananthapuram.
- Mcculloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5(4), 115-133.
- Minsky, M., Papert, S. ve Bottou, L. (1969). *Perceptrons: an introduction to computational geometry*. Cambridge: MIT Press.
- Muneeb ul Hassan. (2018, Kasım 20). VGG16 model. Retrieved from: <https://neurohive.io/en/popular-networks/vgg16/>
- Murugan, S., vd. (2021). DEMNET: A deep learning model for early diagnosis of Alzheimer diseases and dementia from MR images. *IEEE Access*, 9, 90319-90329.
- Sibi, P., Jones, S. A., & Siddarth, P. (2013). Analysis of different activation functions using back propagation neural networks. *Journal of theoretical and applied information technology*, 47(3), 1264-1268.
- Plaut, D. C., & Hinton, G. E. (1987). Learning sets of filters using back-propagation. *Computer Speech & Language*, 2(1), 35-61.
- Rabeh, A. ben, Benzarti, F. ve Amiri, H. (2016, 29 Mart-1 Nisan). Diagnosis of Alzheimer diseases in early step using SVM (Support Vector Machine) [Sözlü bildiri]. Proceedings - Computer Graphics, Imaging and Visualization: New Techniques and Trends, IEEE, Beni Mellal.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386-408.

- Sadiq, R., & Rodriguez, M. J. (2004). Disinfection by-products (DBPs) in drinking water and predictive models for their occurrence: a review. *Science of the total Environment*, 321(1-3), 21-46.
- Saratxaga, C. L., vd. (2021). Mri deep learning-based solution for Alzheimer's disease prediction. *Journal of Personalized Medicine*, 11(9), 902.
- Seyyarer, E., vd. (2020). Derin öğrenmede kullanılan optimizasyon algoritmalarının kıyaslanması. *Computer Science*, 5(2), 90-98
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1-48.
- Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architectures. *IEEE Access*, 7, 53040-53065.
- Özdemir, D., & Arslan, N. N. (2021). Analysis of deep transfer learning methods for early diagnosis of the Covid-19 disease with chest x-ray images. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 10(2), 628-640.
- Srivastava, N., vd. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1), 1929-1958.
- Sun, R. Y. (2020). Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 8(2), 249-294.
- Suzuki, K. (2011). *Artificial neural networks : methodological advances and biomedical applications / monograph*. Croatia: InTech.
- Szegedy, C., vd. (2015, 7-12 Haziran). *Going deeper with convolutions* [Sözlü bildiri]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, Boston.
- Tamma, S. (2019). Transfer learning using VGG-16 with deep convolutional neural network for classifying images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), s9420.
- Chlap, P., vd. (2021). A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology*, 65(5), 545-563.

- Yaman, A. (2015). Aile hekimliğine başvuran yaşlı bireylerde bilişsel değerlendirme. *Konuralp Medical Journal*, 7(2), 121-124.
- Turing, A. M. (1950). Mind a quarterly review of psychology and philosophy. *Computing Machinery and Intelligence*.
- Wang, X., Zhao, Y., & Pourpanah, F. (2020). Recent advances in deep learning. *International Journal of Machine Learning and Cybernetics*, 11(4), 747-750.
- Weiss, K., Khoshgoftaar, T. M., & Wang, D. D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1), 1-40.
- Werbos, P. J. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78(10), 1550-1560.
- Williams, M. M., Roe, C. M., & Morris, J. C. (2009). Stability of the clinical dementia rating, 1979-2007. *Arch Neurol*, 66(6), 773-777.
- Yamashita, R., vd. (2018, Ağustos 1). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(4), 611-629.
- Yashodhar, A., & Kini, S. (2021, 19-20 Kasım). *Performance measurements of different classification techniques for the Alzheimer's disease neuroimaging initiative* [Sözlü bildiri]. 2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics, DISCOVER 2021 – Proceedings, IEEE, Nitte.
- Yigit, A., & Isik, Z. (2018, 2-5 Mayıs). *Application of artificial neural networks in dementia and alzheimer's diagnosis* [Sözlü bildiri]. 26th IEEE Signal Processing and Communications Applications Conference, IEEE, İzmir.
- Ying, X. (2019). An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, 1168(2), 022022.
- Zaheer, R., & Shaziya, H. (2019, 10-11 Ocak). *A study of the optimization algorithms in deep learning* [Sözlü bildiri]. 2019 Third International Conference on Inventive Systems and Control (ICISC), Coimbatore, Hindistan.
- Zhu A., vd. (2017, 5-8 Eylül). *Wind power prediction based on a convolutional neural network* [Sözlü bildiri]. 2017 International Conference on Circuits, Devices and Systems, IEEE, Chengdu.

Zhu Xiaojin. (2008, 19 Temmuz). Semi-supervised learning literature survey. Retrieved from: https://pages.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf



DİZİN

-A-

Alzheimer, iii, v, vi, x, 15, 16, 17, 18,
20, 21, 46, 50, 51, 56

-D-

DEMXNET, i, iii, v, vi, 18, 21, 49, 51,
52, 70, 71, 72, 73, 74
DenseNet201, v, vi, 18, 21, 43, 67, 68,
69, 73, 74
Derin öğrenme, 32, 33, 40, 47

-I-

InceptionV3, v, vi, 18, 21, 42, 62, 63,
64, 73, 74

-S-

SMOTE, v, vi, x, 19, 21, 22, 47, 48, 49,
50, 56, 57, 58, 59, 60, 61, 62, 63, 64,
65, 66, 67, 68, 69, 70, 71, 72, 73, 74

-V-

VGG16, v, vi, 18, 21, 41, 42, 56, 57, 58,
59, 73, 74
VGG19, v, vi, 18, 21, 41, 42, 59, 60, 61,
73, 74

-X-

Xception, v, vi, 18, 21, 43, 65, 66, 67,
73, 74

-Y-

Yapay Sinir Ağları, x, 20, 25, 30

