
GAZİ ÜNİVERSİTESİ
Gazi
1926

CLASSIFICATION OF ABNORMAL RESPIRATORY SOUNDS USING DEEP LEARNING TECHNIQUES

Ahamadi Abdallah IDRISSE

M. Sc. THESIS

DEPARTMENT OF COMPUTER SCIENCE

GAZI UNIVERSITY

INSTITUTE OF INFORMATICS

MARCH 2023

Ahamadi Abdallah IDRISSE tarafından hazırlanan “SOLUNUM SESLERİNİN DERİN ÖĞRENME YÖNTEMLERİ İLE SINIFLANDIRILMASI” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgisayar Bilimleri Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Doç.Dr. Oktay YILDIZ
Bilgisayar Mühendisliği, Gazi Üniversitesi
Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum……

Başkan: Doç.Dr. Mustafa SERT
Bilgisayar Mühendisliği, Başkent Üniversitesi
Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum……..

Üye: Dr. Öğr. Üyesi Ceren GÜZEL TURHAN
Bilgisayar Mühendisliği, Gazi Üniversitesi
Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum

Tez Savunma Tarihi: 01/03/2023
Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

…….
Prof. Dr. Aslıhan TÜFEKCİ
Bilişim Enstitüsü Müdür

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Ahamadi Abdallah IDRISSE

01/03/2023

SOLUNUM SESLERİNİN DERİN ÖĞRENME YÖNTEMLERİ İLE SINIFLANDIRILMASI

(Yüksek Lisans Tezi)

Ahamadi Abdallah IDRISSE

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Mart 2023

ÖZET

Giriş

Akciğer sesleri, hava solunum sisteminden geçerken üretilmektedir. Bu sesler hastaların akciğer durumuna ilişkin önemli bilgiler vermektedir. Doktorlar akciğer rahatsızlığı olan hastaları teşhis etmek amacıyla akciğer seslerini dinlemek için genellikle stetoskop kullanmaktadır. Bir akciğer sesinin durumunun sağlıklı mı yoksa sağlıksız mı olduğunu belirlemek, hatalı bir teşhis konulması halinde hastanın sağlığının riske girebileceği gerçeğinden dolayı çok kritik bir görevdir. Akciğer seslerinin doğrusal ve durağan olmayan yapısal özellikler içerdiği göz önüne alındığında, bu görevden sorumlu olan doktor veya hekimin çok deneyimli olması gerekmektedir. Otomatik tanıma sistemleri (Automatic Recognition System - ARS), sağlık çalışanları için akciğer sesi sınıflandırma zorluklarına bir çözüm sağlayabilmektedir [1]. Mekanize solunum sesi muayenesi, geleneksel oskültasyon tekniklerinin dezavantajlarının üstesinden geldiği ve klinisyenlere daha doğru teşhis için güvenilir bir araç sağladığı için akciğer seslerinin teşhisinde geleneksel oskültasyon yöntemlerine daha güvenilir bir alternatif olarak son zamanlarda popülerlik kazanmıştır. Bunun nedeni, elektronik stetoskopların örüntü tanıma ve yapay zekâ ile birleştirilmesidir [3].

Normal ve anormal solunum sesleri iki ana akciğer sesi türüdür. Solunum zorlukları olmadığında normal sesleri ortaya çıkar, ancak akciğer hastalıkları mevcut olduğunda anormal solunum sesleri ortaya çıkmaktadır. Sağlıklı bir solunum sesi, nefes borusunda hem nefes alma hem de nefes verme aşamalarında duyulabilen daha yüksek frekanslı bileşenlerle tanımlanmaktadır. Anormal solunum sesleri, sağlıklı solunum seslerine ek sesler eşlik ettiğinde ortaya çıkar. Anormal solunum sesleri hırıltılı solunum gibi sürekli, çatırtı gibi süreksiz olabilir. Bu tür seslerin varlığı bir akciğer rahatsızlığına işaret eder [1, 4, 5].

Güler ve arkadaşları [6] aynı veri kümesi üzerinde farklı akciğer seslerini sınıflandırmak için makine öğrenmesi teknikleri önermiştir. Ancak, literatürde kabul gören son teknoloji yöntemlerin hiçbirinin akciğer sesi sınıflandırması için bu veri kümesi üzerinde derin öğrenme tekniklerini kullanmamıştır.

Bu tezde, ses alanında en çok tercih edilen MFCC özellik çıkarımı yapılmış, ardından bunlar derin öğrenme modeller için girdi olarak kullanılmıştır. Seçilen derin öğrenme modelleri arasında VGG, MobileNet, Xception ve ResNet bulunmaktadır. Yapılan tez çalışmasının ana katkıları şu şekilde özetlenebilir: (1) [6]'da kullanılan akciğer sesi veri kümesini

sınıflandırmak için önceden eğitilmiş derin öğrenme tabanlı modeller kullanmayı önermek (2) önceden eğitilmiş modellerin bu spesifik veri kümesi üzerindeki performansını dikkatlice incelemek (3) veri artırma tekniklerinin akciğer seslerini sınıflandırmada derin öğrenme modellerin performansı üzerindeki etkisini incelemek. Seçilen derin öğrenme modellerinin halihazırda büyük veri kümeleri üzerinde eğitilmiş olması nedeniyle, verimli ve düşük eğitim süresi için önceden eğitilmiş ağırlıklarını uygun bir ağ başlatma olarak kullanabiliriz. Ancak, veri kümesinin az örneklem içermesi göz önünde bulundurulduğunda, daha fazla eğitim örneği oluşturmak ve derin öğrenme modellerinin büyük miktarda veri üzerinde iyi eğitilmesinin yanı sıra veri artırma tekniklerini kullanmayı tercih ettik. Sonuç olarak, derin öğrenme modellerimizle birlikte perde kaydırma gibi veri artırma tekniklerinin kullanılması çok daha iyi sınıflandırma doğruluğuna katkıda bulunmuştur.

İlgili Çalışmalar

Bu bölümde, akciğer sesi sınıflandırması ile ilgili önceki araştırmalar hakkında literatür taraması sunulmaktadır. İncelenen makalelerde önerilen yöntemler ve performansları bu bölümde sunulmuştur.

Altan ve arkadaşları [9] astımlı ve sağlıklı akciğer seslerini sınıflandırmak için bir Derin İnanç Ağları (DBN) modeli önermiştir. Sınıflandırıcıyı beslemek için girdi olarak kullanılan istatistiksel özellikleri çıkarmak için Hilbert Huang Dönüşümü (HHT) yöntemini kullanmışlardır. Model, 5 çapraz doğrulama ölçütü kullanılarak değerlendirilmiş ve Derin İnanç Ağı modeli %84,61'lik bir doğruluk elde etmiştir. Islam ve arkadaşları [12], 30'u sağlıklı ve 30'u astımlı olmak üzere 60 hastadan akciğer seslerini toplamak için 4 kanallı bir veri toplama sistemi kullanmıştır. Özellik çıkarma işlemi için, her bir akciğer sesi solunum döngüsünden güç spektral yoğunluğunu hesaplamak ve ardından bunu tek tip alt bantlara ayırmak için Welch yöntemi kullanılmıştır. Sınıflandırma aşaması için Destek Vektör Makinesi ve Yapay Sinir Ağı sınıflandırıcıları alt bantlardan hesaplanan bazı istatistiksel değerlerle beslenmiştir. Destek Vektör Makinesi yöntemi, 4 kanaldan 3 kanalın (1,2,4) kombinasyonu ile %93,3 ile en yüksek sınıflandırma doğruluğuna ulaşmıştır. Messner ve arkadaşları [14] çok kanallı bir akciğer sesi veri kümesinin sınıflandırılması amacıyla bir konvolüsyonel sinir ağı modeli önermiştir. Bu veri kümesi, 16 kanallı bir ses kayıt cihazı kullanılarak farklı hastalardan kaydedilen sağlıklı ve patolojik akciğer seslerinden oluşmaktadır. Ses dosyalarından spektrogramlar çıkarmışlar ve bunları modelleri beslemek için girdi olarak kullanmışlardır. Değerlendirme sonrasında, önerdikleri CNN çerçevesi %92'lik bir F-skoruna ulaşarak diğer yöntemlerden daha iyi performans göstermiştir. Zulfiqar ve arkadaşları [3] solunum seslerinin sınıflandırılması için Yapay Gürültü Ekleme (ANA) tabanlı bir metodoloji önermişlerdir. Solunum sesinin spektrogramını oluşturmak için Fourier dönüşümünü uygulamışlar, ardından spektrogramı daha net ve daha sağlam hale getirmek için bazı yapay gürültüler eklemişlerdir. Daha sonra, önerdikleri sınıflandırıcıları girdi olarak beslemek için spektral veriler üzerinde bir özellik haritası aracılığıyla özellikler çıkardılar (AlexNet ve Inception gibi önceden eğitilmiş modeller). AlexNet modeli en yüksek sınıflandırma doğruluğunu elde eden model olmuştur. Bardou ve diğerleri [1] akciğer sesi veri kümesi üzerinde bazı makine öğrenimi tekniklerini bir CNN modeli ile karşılaştırmıştır. Mel-frekans cepstral katsayıları (MFCC'ler) ve Yerel İkili Patern (LBP) özellikleri makine öğrenmesi yaklaşımları için girdi olarak kullanılırken, akciğer sesleri ses kayıtlarından çıkarılan spektrogramlar CNN algoritmasının girdisi olmuştur. Bu çalışmada aşırı uyum sorununun üstesinden gelmek için sadece spektrogramlar üzerinde bazı veri artırma teknikleri kullanılmıştır. En yüksek sınıflandırma doğruluk değeri %95,56 ile CNN modeli için kaydedilmiştir. Elsetrønninga ve arkadaşları [2] normal ve anormal akciğer seslerinin sınıflandırılması için denetimli ve denetimsiz makine öğrenimi teknikleri geliştirmiştir. Bu çalışmada kullanılan veri kümesi 126 hastadan toplanan 920 ses kaydından

oluşmaktadır. Özellik çıkarma ve özellik seçme teknikleri kullanıldıktan sonra, MFCC'ler sınıflandırıcılar için en iyi girdiler olarak seçilmiştir. K-NN, %84,38 ile en yüksek sınıflandırma doğruluğuna ulaşan algoritma olmuştur. Bu çalışmada, araştırmacılar ayrıştırma özelliği çıkarma ve özellik seçimi yaklaşımlarına daha fazla önem vermişlerdir. Haider ve arkadaşları [13] normal ve maceralı solunum seslerinin sınıflandırılmasını analiz etmek için makine öğrenimi yöntemlerini kullanmışlardır. Çalışmalarında 30 anormal ve 25 sağlıklı kayıttan oluşan bir akciğer sesleri veri kümesi kullanmışlardır. Daha sonra bu akciğer seslerinden çeşitli özellikler çıkarılmış ve COPD ve normal solunum seslerini ayırt etmeyi amaçlayan k-en yakın komşular, lojistik regresyon gibi farklı makine öğrenimi teknikleri için girdi olarak kullanılmıştır. Deney sırasında modellere farklı parametreler atamışlardır. DVM modeli tarafından elde edilen en yüksek sınıflandırma doğruluğu bazı parametrelerle %83,6 olmuştur. Farklı parametreler kullanıldığında ise DVM ve LR sınıflandırıcıları %100 sınıflandırma doğruluğuna ulaşmıştır. Balli ve Kutlu [10], pencere boyutunun anormal solunum seslerinin sınıflandırılmasını nasıl etkileyebileceğini araştırmayı amaçlamıştır. Bu çalışmada kullanılan veri kümesi 92 anormal ve 35 normal sesten oluşmaktadır. Akciğer sesleri 44100Hz frekansında 20 saniye boyunca kaydedilmiştir. Akciğer seslerinden Mel-Frekans Cepstrum Katsayılarını çeşitli pencere boyutu değerlerinde çıkarmışlardır. Bu çalışmada sınıflandırıcı olarak makine öğrenimi algoritması KNN kullanılmıştır. Deneysel sonuçlar KNN'nin, sesler 5 saniyelik bir pencere boyutuna sahip olduğunda %93,21'lik en iyi sınıflandırma doğruluğuna ulaştığını göstermiştir. Bu çalışmada yazarlar, pencere boyutunun solunum seslerindeki anormalliklerin tespitinde sahip olduğu önemli rolü vurgulamışlardır. Pencere boyutu 2 ila 10 saniye arasında olduğunda veri kümesinin daha anlamlı hale geldiği sonucuna varmışlardır. Li ve Hong [7] çatırtıların tespiti için otomatik bir yöntem sunmuşlardır. Çalışmalarında kullandıkları akciğer sesi veri kümesi 70 çatırtı ve 70 normal sesten oluşmaktadır. Seslerin çoğunluğu 5 saniyelik bir periyoda ve 4000Hz veya 4800Hz'lik bir frekans değerine sahiptir. Önerdikleri yöntemin üç ana adımı vardır. İlk olarak, akciğer seslerinin frekanslarını 150Hz ile 1800Hz arasında olacak şekilde azaltmak için bir filtreleme yöntemi kullanılır. İkinci olarak, filtrelenen seslerden üç özellik çıkarılır. Bu özellikler frekans limbik sinyalinin fmin/fmax değeri, zaman limbik sinyalinin standart sapması ve yumuşatma zaman limbik sinyalidir. Üçüncü olarak, sınıflandırma görevi denetimli bir öğrenme yöntemi olan Destek Vektör Makinesi (SVM) kullanılarak gerçekleştirilir. Deneyler, DVM yönteminin akciğer seslerini %97,14 ile %100 arasında değişen bir sınıflandırma doğruluğu değeriyle sınıflandırabildiğini göstermiştir. Aras ve Gangal [11] tek kanallı akciğer seslerinin sınıflandırılması için otomatik bir tanıma yöntemi önermişlerdir. Tek kanallı çıkışı olan bir elektronik stetoskop kullanılarak kaydedilen bir veri kümesi kullanmışlardır. Akciğer sesleri, Karadeniz Teknik Üniversitesi hastanesinde muayene edilen çeşitli deneklerden toplanmıştır. Toplanan dosyalar 70 çatırtı, 70 ronküs ve 70 normal ses olmak üzere toplam 210 akciğer sesinden oluşmaktadır. Bu ses kayıt işlemi iki doktor tarafından gerçekleştirilmiştir. Özellik çıkarma aşamasında, akciğer seslerinden MFCC özelliklerini çıkarmışlardır. Daha sonra bu MFCC'lerden çeşitli temsiller türettiler ve sınıflandırıcılara girdi olarak kullanılabilmeleri için en iyi özelliği veya en iyi özellik kombinasyonunu bulmak amacıyla Sıralı İleri Seçim (SQS) adı verilen bir özellik seçim yöntemi kullandılar. Tek kanallı akciğer seslerini sınıflandırmak için farklı makine öğrenimi algoritmaları kullanılmıştır. KNN, algoritmaya belirli özelliklerin bir kombinasyonu sağlandığında en yüksek sınıflandırma doğruluğunu elde eden algoritma olmuştur. Shuvo ve arkadaşları [15] akciğer seslerinden solunum koşullarının tespiti için hafif bir konvolüsyonel sinir ağı modeli önermiştir. Bu makalede kullanılan veri kümesi, ICBHI bilimsel yarışması için sağlanan halka açık bir akciğer veri kümesidir. Bu veri kümesi, toplam 5,5 saatlik ses dosyasına sahip 920 ses örneği içermektedir. Sesler 10'lar ile 90'lar arasında değişen farklı

uzunluklara sahiptir. Veriler 126 hastadan toplanmıştır. Ön işleme aşamasında, yazarlar ses dosyalarını 22050Hz'lik benzersiz bir frekans değerine yeniden örneklemişlerdir. Ardından akciğer seslerini solunum döngülerine göre segmentlere ayırdılar. Bölütlenmiş verilerden CWT veya hibrit bir EMD-CWT yaklaşımı kullanılarak iki tür skalogram oluşturulmuştur. Önerilen hafif CNN modeli arasında, sınıflandırma görevini tamamlamak için iyi bilinen VGG16 da kullanılmıştır. Hibrit skalogramın daha önemli bir özellik olduğu kanıtlanmıştır çünkü en yüksek sınıflandırma doğruluğu olan %99,05 değeri, hibrit skalogram girdi olarak kullanıldığında yazarların önerdiği CNN modeli tarafından elde edilmiştir. Don [18] akciğer seslerinin sınıflandırılmasında özellik seçim yöntemlerinin önemini araştırmıştır. Çevrimiçi ve çevrimdışı olmak üzere farklı kaynaklardan veri toplamıştır. Son olarak, çalışma için 85 ses dosyası dikkate alınmıştır. Akciğer ses dosyalarından 15 farklı özellik çıkarmış ve sınıflandırıcılar için uygun özellikleri belirleyebilmeleri için iki özellik seçim yöntemi kullanmıştır. Seçilen özellikler daha sonra akciğer sesi sınıflandırma amacını gerçekleştirmek için üç makine öğrenimi algoritmasına beslenmiştir. Kesinlik, duyarlılık ve f-ölçümü yöntemleri, algoritmaların performansını değerlendirmek için kullanılan değerlendirme ölçütleridir. KNN, hassasiyet ve f-ölçümü açısından %94,1'lik bir değerle en iyi skorları elde etmiştir. Öte yandan hem Destek Vektör Makinesi hem de NB algoritmaları %100 geri çağırma değeri elde etmiştir. Yazar, özellik seçim yöntemlerinin etkisine daha fazla dikkat etmiş ve kullandığı 2 özellik seçim yönteminden RSFS'nin (Rastgele Alt Küme Özellik Seçimi) sınıflandırıcılar için SFS'den (Sıralı İleri Seçim) daha önemli özellikler seçtiği sonucuna varmıştır. Gairola ve arkadaşları [17], ICBHI genel akciğer sesi veri kümesinin sınıflandırılması için RespireNet adı verilen CNN tabanlı bir model geliştirmiştir. Akciğer ses dosyalarına veri artırma yaklaşımlarını uyguladıktan sonra, her bir akciğer sesinden Mel-spektrogram özelliklerini çıkarmışlardır. Ardından, her bir Mel-spektrogramdan boş alanı (bilgi aktarmayan) ve akciğer sesi hakkında bilgi içeren alanı ayıracak siyah bölge kırpma adı verilen bir yöntem kullandılar. Mel-spektrogramın ilgili bilgiye sahip bölümü CNN modeline girdi olarak kullanılacaktır. Önerilen modelin performansı, 4 sınıflı sınıflandırma problemleri için en son teknoloji sonuçlarından daha iyi performans göstermiştir. Rocha ve arkadaşları [16] hırıltı seslerini otomatik olarak sınıflandırmak için farklı makine öğrenme algoritmaları ve bir derin öğrenme yöntemi önermiştir. Temel amaçları, hırıltı sınıflandırmasında olay süresinin etkisini araştırmaktı. Respiratory Sound DataBase (RSD) adlı bir veri kümesi kullanmışlardır. Bu veri kümesi, toplam 5,5 saat süren 920 ses dosyası içermektedir. RSD'nin sesleri 126 hastadan toplanmıştır. Ses kayıtları farklı frekanslarda kaydedildiğinden, yazarlar ses dosyalarını veri tabanındaki en düşük frekans değerine (4000Hz) yeniden örneklemişlerdir. Her ses kaydından bir spektrogramı üretmek için STFT kullanmışlardır. Sınıflandırma görevi için doğrusal diskriminant analizi (LDA), güçlendirilmiş ağaçlar (Boost) gibi makine öğrenimi yöntemleri ve özellikle CNN modeli olmak üzere bir derin öğrenme modeli kullanılmıştır. Üretilen spektrogramlar doğrudan CNN modeli için girdi olarak kullanılır. Geleneksel ML yöntemleri girdileri için, her bir spektrogramdan kare başına 47 özellik türetmek için bir MIR Araç Kutusu kullanmışlardır. Sınıflandırıcıların performanslarını değerlendirmek için doğruluk, hassasiyet, duyarlılık ve özgüllük gibi çeşitli değerlendirme ölçütleri kullanılmıştır. Deneysel sonuçlar, parametreleri değiştirmeden elde edilen en iyi sonuçların %98 olduğunu ve bu sırayla %95 duyarlılık ve özgüllük değerlerinin CNN modeli tarafından elde edildiğini göstermiştir. Olay süresi parametresini değiştirdiklerinde, en iyi sınıflandırıcı (Boost) duyarlılık ve özgüllük için sırasıyla yalnızca %55 ve %76 değerlerini kaydetmiştir. Petmezas ve arkadaşları [28], ICBHI veri kümesindeki akciğer seslerinin sınıflandırılmasını otomatikleştirmek için bir Odak Kaybı (FL) işlevi kullanan hibrit bir derin öğrenme modeli tanıtmıştır. Bu veri kümesi, 2 farklı ülkeden uzmanlar tarafından manuel olarak notlandırırmış açık erişimli bir veri tabanıdır. Akciğer seslerinin farklı frekans değerleri

nedeniyle, yazarlar ses dosyalarını 4000Hz'lik benzersiz bir örnekleme hızı değerine yeniden örneklemeye karar verdiler. Ayrıca sesler için 2,7 saniyelik sabit bir uzunluk seçtiler. Özellik çıkarma sürecinde, spektrogram olarak bilinen bir sinyalin zaman-frekans temsilini sağlayan Kısa Zamanlı Fourier Dönüşümü (STFT) algoritması kullanıldı. Çıkarılan spektrogramlar, hibrit model için girdi olarak kullanılacak görüntüler olarak kaydedilmiştir. Önerilen hibrit ağ, bir 2D CNN ve ardından bir LSTM ve FL fonksiyonundan oluşmaktadır. Spektrogram görüntüleri CNN bloğuna beslenir, böylece sınıflandırma görevinin yapıldığı LSTM bloğuna itilecek derin özellikler çıkarabilir. ICHBI dengesiz bir veri tabanı olarak bilinmektedir. Bu nedenle yazarlar, bu dengesiz sorunu ele almak ve derin öğrenme modelinin etkinliğini değerlendirmek için bir FL işlevi uygulamıştır. Farklı deneyler gerçekleştirilmiş, elde edilen sonuçlar CNN, LSTM ve FL fonksiyonundan oluşan bu kombinasyonun kayda değer sonuçlar elde edebildiğini göstermiştir. Yazarlara göre bu, bu metodolojiyi (Hibrit CNN-LSTM Ağı ve Odak Kaybı Fonksiyonu) öneren ilk araştırmadır. Demir ve arkadaşları [19] akciğer seslerini otomatik olarak sınıflandırmak için konvolüsyonel sinir ağı tabanlı bir model önermişlerdir. ICBHI veri kümesinde bulunan akciğer ses dosyalarını kullanmışlardır. Frekans üzerinde zamanın 2 boyutlu bir temsili olan spektrogramlar, Kısa Zamanlı Fourier Dönüşümü (STFT) yöntemi kullanılarak ses kayıtlarından çıkarılmıştır. Spektrogramlar görüntü olarak kaydedilir. Ancak bunları derin öğrenme modellerine beslemeden önce, görüntüler derin öğrenme modellerine uygun olabilmeleri için 224×224 boyutuna yeniden boyutlandırılır. VGG16, AlexNet ve ResNet-50, derin özellikleri çıkarmak için yeniden boyutlandırılmış spektrogramlar üzerinde eğitilir. Daha sonra SVM ve SoftMax sırasıyla çıkarılan derin özelliklere ve transfer öğrenmeye dayalı akciğer sesleri sınıflandırmasını gerçekleştirmek için kullanılır. Hesaplama sonuçları, SVM'nin %65,50'lik bir sınıflandırma doğruluğu elde ederek daha iyi performans gösterdiğini ortaya koymuştur. Datta ve arkadaşları [20] akciğer rahatsızlıklarının tespiti için otomatik bir araç sunmuşlardır. Farklı kaynaklardan 52 akciğer sesi ses kaydı toplamışlardır. Dolayısıyla ses dosyaları farklı frekanslara sahiptir. Bu nedenle ön işleme aşamasında, yazarlar ilk olarak akciğer seslerini 4KHz'lik benzersiz bir frekans değerine indirgemişlerdir. Ardından akciğer sesleriyle birlikte gelen istenmeyen sesleri gidermek için Ampirik Mod Ayrıştırma (EMD) yöntemini kullanmışlardır. Özellik çıkarma aşamasında, önceden işlenmiş akciğer seslerinden spektral özellikler, spektrogram özellikleri, dalgacık özellikleri ve cepstral özellikler çıkarılmıştır. Alakasız özelliklerden kaçınmak için bir özellik seçim yöntemi olan Maksimal Bilgi Katsayısı (MIC) kullanılır. MIC yöntemi, özelliklerin önem sırasına göre düzenlenmesini sağlayan bir puan sağlar. Doğrusal bir Destek Vektör Makinesi (DVM), seçilen özelliklere dayalı olarak akciğer sesinin sınıflandırılması için kullanılan algoritmadır. Doğruluk, duyarlılık ve özgüllük gibi sınıflandırma ölçütleri modelin performansını değerlendirmek için kullanılmaktadır. Deneysel sonuçlar, önerilen yöntemin %80'lik bir sınıflandırma doğruluğu kaydettiğini göstermiştir. Bozkurt ve arkadaşları [24], girdi özelliklerini analiz etmek amacıyla kalp seslerini patolojik durumlarla otomatik olarak sınıflandıran bir derin öğrenme modeli geliştirmiştir. Yazarlara göre MFCC ve Mel-spektrogramı son teknolojide en çok kullanılan özelliklerdir. Bu çalışmada, yazarlar MFCC, Mel-spektrogram ve alt bant zarfları olmak üzere üç farklı özellik çıkarmışlardır. Alt bant zarfları da bir sinyalin zaman-frekans gösterimidir, ancak çoğunlukla alan bilgisinden etkilenirler. Kalp sinyallerinin sınıflandırılması için bir konvolüsyonel sinir ağı modeli geliştirilmiştir. Çıkarılan özellikler CNN modelini ayrı ayrı eğitmek için kullanılır, böylece yazarlar hangi özellik türünün sınıflandırıcıya daha yararlı bilgiler sağladığını bulabilir. Modelin performansını değerlendirmek için ROC eğrisi kullanılmıştır. Birkaç hesaplama deneyinden sonra yazarlar, alt bant zarf özelliklerinin CNN modeline daha önemli özellikler sağladığı sonucuna varabilmişlerdir. Naves ve arkadaşları [22] akciğer seslerinin sınıflandırılması için bir böl ve yönet tekniği önermişlerdir. Bu makalede ele alınan veri

kümesi, farklı sınıflardan toplam 35 ses dosyasıdır. Akciğer sesleri 8KHz'lik tek bir frekans değerinde kaydedilmiştir. Yazarlar, ses dosyalarını örtüşmeyen segmentlere ayırarak ön işleme tabi tutmuşlardır. Yüksek Dereceli İstatistik (HOS), ses kayıtlarından giriş özelliklerini çıkarmak için kullanılan yöntemdir. Bu özellik çıkarma yöntemi toplam 800 özellik ile sonuçlanmıştır. Fazlalıktan ve önemli olmayan özelliklerden kaçınmak için yazarlar Genetik Algoritma (GA) kullanmışlardır. GA tarafından seçilen özelliklere dayalı olarak akciğer sesi sınıflandırma görevini gerçekleştirmek için iki makine öğrenimi algoritması, KNN ve NB çağrılmıştır. Hesaplama sonuçlarından sonra, bir sınıflandırma ağacının daha iyi sınıflandırıcı olduğu kanıtlanmıştır. İlk olarak, ağaç sınıflandırıcı aşağıdaki sınıfları sınıflandırmak için KNN kullanarak başlar veziküller, hırıltılar ve çatırtılar. Daha sonra NB hem hırıltı hem de çatırtı alt sınıflarını sınıflandırmak için kullanılır. Kaydedilen sınıflandırma doğruluğu değerleri eğitim ve doğrulamada sırasıyla %98,1 ve 94,64'tür. Nguyen ve Pernkopf [4] akciğer seslerindeki çatırtıların tespiti için bir derin öğrenme modeli sunmuştur. Önerilen model transfer öğrenme tabanlı bir modeldir. Bu çalışmada iki farklı veri kümesi kullanılmıştır. İlk veri kümesi iyi bilinen ICBHI veri kümesidir ve sadece CNN modelini eğitmek için kaynak veri kümesi olarak kullanılmıştır. Bu eğitim sırasında CNN modeli tarafından öğrenilen bilgi, hedef veri kümesi olarak adlandırılan ikinci veri kümesinde akciğer sesleri sınıflandırma görevi gerçekleştirilirken kullanılır. İki veri kümesinin akciğer sesleri farklı örnekleme hızlarıyla toplanmıştır. Bu nedenle yazarlar tüm ses dosyalarını 16KHz'e yeniden örneklemiştir. Daha sonra yazarlar, seslerin solunum fazlarının ayrılması için sabit bir uzunluk da seçmişlerdir. Bu sabit uzunluk, CNN modellerinin girdilerinden benzersiz bir şekil gerektirmesi nedeniyle kolaylık sağlamaktadır. Spektrogramlar, akciğer seslerinden çıkarılan özellik gösterimleridir. Eğitim sürecinde aşırı uyumdan kaçınmak amacıyla farklı ses verisi artırma yöntemleri kullanılmıştır. Hesaplamalı deneylerde, yazarlar CNN modelini önce ICBHI verileri üzerinde eğitmişlerdir. Daha sonra CNN modelini, çok kanallı hedef veri kümesindeki akciğer seslerinin sınıflandırılması için önceden eğitilmiş bir model olarak kullandılar. Yazarlar, önerilen yöntemin oldukça iyi sonuçlar verdiği sonucuna ulaşmıştır. Fairwan ve arkadaşları [23] akciğer seslerinden akciğer rahatsızlıklarının tanınması için CNN+BDLSTM tabanlı bir derin öğrenme modeli sunmuştur. Veri hazırlama adımında iki farklı kaynak kullanılmıştır. İlk kaynak Ürdün'deki bir devlet üniversitesinde kaydedilen bir dizi akciğer sesidir. İkinci kaynak ise Uluslararası Sağlık ve Biyomedikal Bilişim Konferansı (ICBHI) adlı popüler halka açık akciğer sesi veri kümesidir. Bu ikinci veri kümesinin tamamı kullanılmamış, sadece 110 ses kaydı dikkate alınmıştır. Akciğer seslerinin farklı frekans değerleri nedeniyle, tüm akciğer ses dosyaları için 4KHz'lik sabit bir frekans seçilmiştir. Daha sonra her bir akciğer sesi 5 saniyelik segmentlere bölünerek toplam 1182 segment elde edilmiştir. Gürültüleri gidermek için akciğer sesleri filtrelenmiştir. Ön işleme adımlarında üç işlem gerçekleştirilmiştir. Bunlar dalgacık yumuşatma, yer değiştirme artefaktı giderme ve z-skorlarının normalleştirilmesidir. Bir konvolüsyonel sinir ağı (CNN) ve çift yönlü uzun-kısa süreli bellek (BDLSTM) kombinasyonu, segmente edilmiş-ön işlemden geçirilmiş akciğer seslerinden önemli uzamsal boyutluluk ve zaman alanı özelliklerini çıkarmak için kullanılmıştır. Ağ, karışıklık matrisi ve F1-skoru gibi çeşitli sınıflandırma ölçütleriyle değerlendirilmiştir. Model %99,62'lik bir genel sınıflandırma doğruluğu değeri elde edebilmiştir.

Materyaller ve Yöntemler

Bu bölümde, tezde kullanılan veri kümesi hakkında genel bilgi verilmiş ve önerilen metodolojinin farklı aşamaları ayrıntılı olarak açıklanmıştır.

a) Solunum Sesi Veri Kümesi

Bu tez çalışmasında kullanılan veri kümesi, Güler ve arkadaşları tarafından [6]'da kullanılan veri kümesinin güncellenmiş bir versiyonudur. Onların 2020 yılında yaptıkları araştırmada,

akciğer sesleri Türkiye'de bir devlet hastanesinde 60 hastadan toplanmıştır. Her ses kaydı, bir hastanın göğsünün belirli bölgelerinden elektronik stetoskop kullanılarak 16KHz frekansında 15 saniye boyunca kaydedilmiştir. Aynı veri hazırlama prosedürleri yeni 117 ek akciğer ses dosyasına da uygulanmıştır. Böylece, elimizdeki son veri kümesi 177 ses dosyasından oluşmakta ve 3 sınıfa (Normal 59, Patolojik Rhal 59 ve Patolojik Ronküs 59) ayrılmaktadır.

b) Yöntem

Bu bölümde, önerdiğimiz teşhis metodoloji açıklanmıştır. Akciğer sesi sınıflandırma problemimizi çözmek için izlenen farklı aşamalar Şekil 5.1'de gösterilmiştir.

İlk olarak, veri toplama aşamasında, akciğer sesleri Hacettepe Üniversitesi Tıp Fakültesi Fizyoloji Anabilim Dalı gözetiminde Türkiye'de bir devlet hastanesinde farklı hastalardan elektronik stetoskop kullanılarak kaydedilmiştir.

İkinci olarak veri artırma aşamasında, sınıflandırıcı olarak derin öğrenme tekniklerini kullandığımız için modellere girmeyi planladığımız eğitim verisi miktarına dikkat etmemiz gerekmektedir. Bunun nedeni, derin öğrenme modellerinin yüksek eğitim verisi tüketicileri olması ve yüksek başarılar elde etmek için çok sayıda eğitim örneği gerektirmesidir. Elimizdeki akciğer sesi veri kümesinin küçük boyutunu dikkate alarak, daha fazla eğitim örneği oluşturmak için veri artırma yöntemlerini kullandık. Bu aşamada iki ses verisi artırma yöntemi, zaman esnetme ve perde kaydırma, kullanılmıştır. Bir sese zaman esnetme yöntemi uygulandığında, bunu yavaşlatmak veya hızlandırmak anlamına gelmektedir. Perde kaydırma ise bir ses girdisinin perdesini değiştirmektir [46].

Üçüncü aşamada, Mel Frekans Cepstral Katsayıları (MFCC'ler) özellikleri çıkarılmış ve modeller için girdi olarak kullanılmıştır. Ses işleme alanında, Mel Frekansı Cepstrumu (MFC) bir ses sinyalinin kısa vadeli güç spektrumunu temsil eder. Logaritmik bir güç spektrumunun doğrusal olmayan bir Mel frekans ölçeğinde lineer bir kosinüs dönüşümüne dayanmaktadır, dolayısıyla Mel-frekans cepstral katsayıları bir MFC'yi oluşturan katsayılardır [53]. Bizim durumumuzda, modellerin girdilerini oluşturmak için her sesten 32 katsayı çıkarılmıştır. MFCC özellikleri, konuşma tanıma [53, 54] ve çevresel ses sınıflandırması [55, 56] dahil olmak üzere ses uygulamalarında en yaygın kullanılan özelliklerden biridir. Tiwari araştırmasında MFCC'ler için 32 katsayının verimli olabileceğini kanıtlamıştır [57]. Bir akciğer sesinin özelliklerini çıkarmadan önce, her ses dosyası 5 bölüme ayrılmış ve her bölümün MFCC'leri çıkarılmıştır. Her segmentteki çıkarılan MFCC'ler ve bunlara karşılık gelen etiketler, modelleri eğitmek için kullanılacak son veri kümesini oluşturan bir JSON dosyasında saklanır. MFCC özellikleri daha sonra akciğer seslerinden türetilir ve ardından sınıflandırma görevini yerine getirmek ve hastanın akciğerlerinin sağlıklı (Normal) veya sağlıksız (PatolojikRhal veya PatolojikRonküs) olup olmadığını belirlemek için derin öğrenme modelleri için girdi olarak kullanılır.

Dördüncü aşamada, sağlanan girdiden (MFCC'ler) öğrenmek için popüler derin öğrenme mimarilerini kullandık. Bu CNN yapıları arasında VGG16, MobileNet, ResNet ve Xception yer almaktadır. Bu mimarilerin halihazırda sahip oldukları bilgiden yararlanmak için bu seçeneği tercih ettik çünkü bu mimariler uzun süre boyunca büyük ve yeterli miktarda veriyle eğitilmişlerdir. Bu yöntem transfer öğrenme olarak bilinir, modeli sıfırdan yeniden eğitmek yerine, önceki eğitimi sırasında edindiği bilgilerle kolayca kullanılabilir [58]. Bu transfer öğrenme seçeneğini tercih etmemizin ikinci nedeni de elimizdeki veri kümesinin miktarından kaynaklanmaktadır. Önceden eğitilmiş derin öğrenme modellerini MFCC'lerden ayrı olarak eğittikten sonra, sınıflandırma görevini tamamlamak için bir SoftMax aktivasyon fonksiyonu ile birlikte tam bağlı bir katman kullanılır. SoftMax kullandık çünkü bizim problemimiz, yani çok sınıflı bir sınıflandırma problemi için uygun

aktivasyon fonksiyonudur. Daha sonra modellerimizin performansını değerlendirmek için doğruluk, kesinlik ve karışıklık matrisi gibi değerlendirme metrikleri kullanılmıştır.

Deneyler Çalışma ve Sonuçlar
Önerilen yöntemin deneysel incelemesi ve deney sonuçları bu bölümde sunulmuştur. Dört farklı modelin sonuçları sunulmuş ve karşılaştırılmıştır. Deneysel çalışma sırasında veri kümesi rastgele bir şekilde %80 eğitim kümesi ve %20 test kümesi olmak üzere ikiye ayrılmış, ardından eğitim kümesinin %20'si doğrulama kümesi olarak kullanılmıştır.

a) Veri artırma teknikleri kullanılmadan elde edilen sonuçlar
Bu bölümde, veri artırma yöntemleri kullanılmadığında elde edilen sonuçları verdik. Akciğer seslerinden MFCC özelliklerini çıkardık ve bunları sınıflandırma aşaması için modellere ilettik. Tablo 5.1, veri artırma teknikleri kullanılmadan modellerin performans sonuçlarını göstermektedir. Eğitim örneklerinin sayısının ne kadar az olduğu göz önüne alındığında, modellerin tatmin edici sonuçlar elde ettiği sonucuna varabiliriz. Bu koşullar altında, ResNet modeli %81'lik bir sınıflandırma doğruluğu değeri kaydederek, MobileNet yakın sonuçlar elde etmesine rağmen diğer yöntemleri geride bırakmıştır. VGG ve Xception modellerinin her ikisi de tüm farklı metriklerde %77 değerine ulaşmıştır. Kesinlik açısından MobileNet ve ResNet %81 ile aynı sonucu paylaşmaktadır.

b) Veri artırma teknikleri kullanıldığında elde edilen sonuçlar
Veri artırma yöntemlerinin, örnek sayısını artırarak modellerin yüksek doğruluklara ulaşmasına yardımcı olmada etkili olduğu görülmektedir. Özellikle de daha iyi sonuçlar elde etmek için tonlarca veriye ihtiyaç duyan derin öğrenme modelleri söz konusu olduğunda. Salamon ve Bello [26] tarafından yürütülen ve veri artırma tekniklerinin kullanılmasının derin öğrenme modellerinin performansını büyük ölçüde artırdığını öne süren bir çalışmadan esinlenerek, deneylerimizde farklı veri artırma araçları kullandık. Veri artırma yöntemlerini uygulamak için, zaman genişletme ve perde kaydırma olmak üzere iki tür ses deformasyon tekniği kullandık. Zaman genişletme yönteminde, akciğer seslerini yavaşlatmak için 0.81 değeri kullanılırken, hızlandırmak için 1.07 değeri kullanılmıştır. Perde üzerinde değişiklik yapıldığında ise bu değerler {-3.5, -2.5, -2, -1.5. -1, 1, 1.5, 2, 2.5, 3.5}, perde kaydırma deformasyonunu tamamlamak için kullanılır. Salamon ve Bello çalışmalarında, ses dosyalarının perde kaydırmasını gerçekleştirmek için {-2, -1, 1, 2} olmak üzere 4 değerle başladıkları perde kaydırma yönteminin etkinliğinin altını çizmişlerdir. Daha sonra, bu yöntemin ne kadar faydalı olduğunu fark ettikten sonra, 4 değer daha eklediler {-3.5, -2.5, 2.5, 3.5}. Bizim durumumuzda, [26]'da kullanılan katsayıları {-1.5, 1.5} olmak üzere iki değer daha ekleyerek benimsedik. Veri artırma teknikleri kullanıldıktan sonra elde edilen sonuçlar ilerleyen alt bölümde sunulmuştur.

<u>VGG modelinin sonuçları</u>
Tablo 5.2, kesinlik metriği aracılığıyla Ral ve Ronküs sınıfları için 0,98 puan kaydedildiğini, Normal sınıfı için ise 0,97 puan rapor edildiğini göstermektedir. Geri çağırma metriğinde ise skorlar Normal sınıfı için 0,98 ve Ronküs sınıfı için 0,97'den başlayıp Ral sınıfı için 0,96'da sona ermektedir. 0.97, F1-skor metriği ile hem Normal hem de Ral sınıfları için kaydedilen değer iken, Ronküs sınıfı için 0.98 rapor edilmiştir. Tüm sınıflarda VGG'nin genel doğruluk oranı %97'dir.
VGG modelinin eğitim sürecindeki performansı Şekil 5.8'de gösterilmektedir. Diğer grafiklerle karşılaştırıldığında, VGG'nin eğitim doğruluğu grafiği en istikrarlı grafiklerden biridir. Eğitim epoklarının sonuna kadar düz bir çizgi görülürken, 100. epoktan önce bazı küçük değişimler meydana gelmektedir. Aynı yörünge, doğrulama doğruluğu eğrisinde ve eğitim kaybı eğrisinde de göze çarpmaktadır. Doğrulama kaybı eğrisinde de 100. epoktan

önce düz bir çizgi görülebilir. Ancak eğitim epoklarının sonunda yükseldiği görülmektedir. Modelin eğitim süreci boyunca sağlanan verilerden etkili bir şekilde öğrendiği görülmektedir.

MobileNet modelinin sonuçları

Tablo 5.3'te gösterildiği gibi, Ral sesleri için MobileNet 0,96'lık bir duyarlılık değerine ve 0,97'lik bir F1-Skoruna sahiptir. Normal sesler için ise 0,97 hassasiyet değerine sahiptir. Kesinlik, duyarlılık ve F1-skorunun diğer sonuçları tüm sınıflar için ortalama 0,98 puana ulaşmıştır. Doğruluk söz konusu olduğunda, MobileNet her üç sınıf için de ortalama %98 doğruluk elde etmiştir.

Şekil 5.9, MobileNet'in 1000 epok boyunca eğitildikten sonra deney sırasında nasıl performans gösterdiğini göstermektedir. Eğitim süresi boyunca eğitim doğruluğu grafiğinde bazı küçük değişiklikler vardır. Ancak genel olarak aynı seviyededir. Aynı eğilimler, bazı küçük değişikliklerle eğitim kaybı grafiğinde de görülür. Doğrulama hatası grafiğiyle karşılaştırıldığında, doğrulama doğruluğu eğrisi eğitim epoklarının başından sonuna kadar tutarsız değişikliklere sahiptir. Eğitim aşamasının neredeyse her aşamasında büyük değer değişiklikleri göze çarpmaktadır.

Xception modelinin sonuçları

Tablo 5.4, Xception modelinin en yüksek kesinlik skorunu 0,99 ile Ral sınıfı için elde ettiğini göstermektedir. Buna karşılık, en düşük kesinlik puanı 0,96 ile Normal sınıfı için elde edilmiştir. Duyarlılık metriğinin sonuçları, Normal ve Ral sınıflarında sırasıyla 0,99 ve 0,96 ile kesinlik sonuçlarına kısmen simetriktir. Ronküs sınıfı için hassasiyet, geri çağırma ve F1-skoru metriklerinin her birinde kaydedilen puan 0,98 iken Normal ve Ral sınıfları sırasıyla 0,97 ve 0,98 F1-Skoruna ulaşmıştır. Doğruluk açısından, Xcetpion'un tüm sınıflar için genel doğruluğu %98 olarak raporlanmıştır.

Şekil 5. 10'da gösterildiği gibi, Xception modelinin eğitim süreci 500 epok boyunca sürmüştür. Eğitim doğruluğu grafiği tüm iterasyonlar boyunca neredeyse sabittir. Ancak epok sayısı arttıkça bazı küçük değişiklikler gözlemlenebilir. Modelin kendisine beslenen girdi verileriyle oldukça etkili bir şekilde öğrendiği sonucuna varabiliriz. Xception modelinin doğrulama eğrisi özellikle başlangıçta ve bazı durumlarda ortada bazı tutarsızlıklar göstermektedir. Ancak bu düzensizliğin iterasyonların sonunda azaldığı görülmektedir. Eğitim hatası grafiği yüksek değerlerle başlarken, daha sonra bazı minimal değişimler gözlemlense de sıfıra yakın bir değere düşmüştür. Doğrulama kaybı grafiği, eğitim aşamasının ortasında aralıklı değer değişiklikleri görüldüğü için net bir düzensizlik göstermektedir.

ResNet modelinin sonuçları

Bu modelin sonuçları Tablo 5.5'te açıklanmıştır. Üç metrik (hassasiyet, duyarlılık ve F1-skoru) için kaydedilen en yüksek değerlerin Ronküs ve Normal sınıflarına ait olduğunu göstermektedir. Hem kesinlik hem de duyarlılık için 0.98 ortalama puan kaydedilirken, F1-skoru için 0.97 rapor edilmiştir. En düşük hassasiyet puanı 0,95 ile Normal sınıfı için elde edilmiştir. ResNet modelinin genel doğruluğu %97'dir.

Şekil 5.11, ResNet modelinin deney sırasında nasıl performans gösterdiğini yansıtmaktadır. Eğitim doğruluğu grafiği bazı küçük düzensizlikler göstermektedir. Ancak eğitim aşaması boyunca neredeyse aynı seviyede kalmaktadır. Bu özellik eğitim hatası grafiğiyle aynı kalmaktadır, ancak loss grafiğinde, düzensizlikler eğitim doğruluğunda büyük ölçüde kaybolma eğilimindedir. Diğer taraftan, doğrulama doğruluğu ve hata grafikleri tüm eğitim süreci boyunca büyük değişimler yaşamıştır.

Tartışmalar

Doktorların akciğerleri incelemek için kullandıkları en yaygın tekniklerden biri, klinik muayenelerin rutin bir parçası haline gelen oskültasyondur. Bu yöntemin öznel doğası ve akciğer seslerinin durağan olmayan yapısı nedeniyle analiz, tanıma ve ayırt etme görevleri karmaşıktır. Sonuç olarak, bu süreci otomatikleştirmek ve bu sınırlamaların üstesinden gelen otomatik tanıma sistemleri (ARS) geliştirmek için makine öğrenimi ve yapay zekadan yararlanmaya yönelik bir ilgi vardır [1]. Bu tezde, önceden eğitilmiş farklı modeller, Türkiye'de bir devlet hastanesinde farklı hastalardan toplanan 177 ses dosyasından oluşan bir akciğer sesi veri kümesi üzerinde değerlendirilmiştir. Bu veri kümesi normal, ronküs ve rhal olmak üzere üç farklı sınıftan oluşmaktadır.

Veri kümesinin küçük boyutlu olması nedeniyle, orijinal veri kümesinden daha fazla örnek üretmek amacıyla bazı veri artırma teknikleri uygulanmıştır. Bunlar ya akciğer sesinin hızını değiştirerek ya da farklı parametreler kullanarak perdesini değiştirerek yapıldı. Bu veri artırma yöntemleri, model performansını optimize etmede etkili olabileceklerini kanıtlamıştır. Bundan sonra, Librosa kütüphanesi kullanılarak MFCC özelliklerinin çıkarıldığı özellik çıkarma aşaması gelmektedir. Çıkarılan özellikler daha sonra önceden eğitilmiş modeller için girdi olarak kullanılmıştır. Önceden eğitilmiş modeller, eğitim süresinden tasarruf etmek ve büyük veri kümeleri üzerinde önceden eğitilmiş ağırlıkları sayesinde sınıflandırma görevlerinde verimli oldukları bilindiği için sınıflandırıcı olarak seçilmiştir. Önceden eğitilmiş modeller, farklı epoklar için bu akciğer sesi veri kümesi üzerinde ayrıca eğitilmiştir. Xception ve MobileNet modelleri, diğer algoritmalar arasında %98 ile en yüksek sınıflandırma doğruluğuna ulaşmıştır. Bu iki algoritma sırasıyla 500 ve 1000 epok için eğitilmiştir. Eğitim doğruluğu grafikleri küçük farklılıklarla birbirine oldukça benzemektedir. Ardından %97 sınıflandırma doğruluğuna sahip VGG ve ResNet modelleri gelmektedir. Bu son algoritmalardan ResNet 400 epok için eğitilirken diğeri 500 epok için eğitilmiştir. ResNet grafiği, VGG modeline kıyasla çok fazla varyasyona sahiptir, özellikle de Şekil 5.11'de gösterildiği gibi doğrulama grafikleri (doğruluk ve kayıp). Derin öğrenme modelleri, tahminlerde bulunurken daha doğru olabilmek için büyük miktarda veriye ihtiyaç duyar. Bu nedenle ses verisi deformasyon yöntemleri uygulanmadan önce daha düşük performansları gözlemlenmiştir. Modeller veri artırma teknikleri kullanılmadan eğitildiğinde, VGG ve Xception modelleri en düşük sınıflandırma doğruluğuna sahip modeller olmuş ve her ikisi de ortalama %77 doğruluk elde etmiştir. Oysa bu ses büyütme yöntemleri kullanıldıktan sonra %97 ve %98 gibi daha yüksek sınıflandırma doğruluklarına ulaşabilmişlerdir. Tablo 6.1, derin öğrenme modellerinin farklı değerlendirme metriklerindeki performans sonuçlarını özetlemektedir. Bu sonuçlar, veri artırma tekniklerinin kullanılarak elde edilmiştir. Akciğer sesi veri kümesinin sınıflandırılmasında en başarılı sınıflandırıcılarımızın (MobileNet ve Xception) karışıklık matrisleri sırasıyla şekil 6.1 ve 6.2'de gösterilmiştir.

Sonuç ve Değerlendirme

Literatüre göre MFCC ses özellikleri, akciğer seslerinin sınıflandırılması için makine öğrenmesi yöntemleri ile birlikte girdi olarak kullanılmak üzere tercih edilmektedir. Bu durum [1, 2, 6]'da görülebilmektedir. Öte yandan, spektrum türleri, akciğer sesi veri kümelerini sınıflandırmak için [1, 14, 17]'de olduğu gibi derin öğrenme modellerini beslemek için en çok kullanılan özelliklerdir. Önceden eğitilmiş derin öğrenme modelleriyle birlikte girdi olarak MFCC'ler, bilgimize göre çok fazla araştırılmamıştır. Bu boşluğu araştırmak istememizin nedenlerinden biridir. Denemelerden sonra, bu yaklaşımın etkili ve önceki tablolarda gösterildiği gibi doğru olduğu gösterilmiştir. [1, 6, 17] veri artırma yöntemlerini kullanmıştır. [6] ve [17]'de veri artırma yöntemleri özellik çıkarma aşamasından önce uygulanırken, [1]'de spektrogram üzerinde, yani özellik çıkarma

işleminden sonra kullanılmıştır. Bu çalışmada, çevrimdışı olarak (MFCC'leri çıkarmadan önce) iki farklı ses verisi artırma türü kullandık ve modellerin öğrenme performansına olumlu katkıda bulunmuştur.

Önerilen yaklaşımımız %97 ve %98 doğruluk değerlerine ulaşarak daha iyi sonuçlar elde edilmiştir. Bu çalışmada kullanılan veri kümesi daha önce Güler ve arkadaşları tarafından [6]'da kullanılmış ve sınıflandırıcı olarak makine öğrenmesi yöntemleri kullanılmıştır. Naïve Bayes algoritması, 10 çapraz doğrulama doğruluğu altında en yüksek sınıflandırma doğruluğunu (%95) kaydetmiştir. Bu değer, VGG gibi önerilen önceden eğitilmiş modellerimiz tarafından elde edilen en düşük doğruluktan (%97) daha düşüktür.

Abnormal solunum sesleri, normal solunum sesleriyle birlikte gelen istenmeyen seslerdir. Yedi farklı anormal ses sınıfı ayırt edilebilir: hırıltı, ronküs, stridor, squawk, kaba çıtırtı ve plevral sürtünme [3]. Ne yazık ki, bu araştırmada kullanılan veri kümesi sınırlı örneklere sahiptir ve yalnızca iki anormal sınıf (ronküs ve rhal) mevcuttur. Veri kümesi tüm anormal akciğer sesi türlerini içeriyorsa, sınıflandırıcılar belirli bir akciğer sesinin sınıfını belirlerken daha spesifik olabilir.

Gelecekteki çalışmalar için, akciğer ses dosyalarından MFCC'ler veya Mel-spektrogramlar gibi ses özelliklerini çıkarmayı, bu özellikleri önceden eğitilmiş derin öğrenme modellerine beslemeyi ve ardından çıktılarını birleştirmeyi planlıyoruz. Bu kombinasyonun sonucu, sınıflandırma görevini tamamlamak, KNN veya SVM gibi geleneksel bir sınıflandırıcı için girdi olarak kullanılabilir. Özellikle, MobileNet gibi hafif ağırlıklı önceden eğitilmiş bir modelin kullanılması, mobil cihazlarla entegrasyonunu kolaylaştıracak ve böylece hastaların akciğer seslerinin sınıflandırılması için sağlık uzmanları tarafından kolayca kullanılabilecektir [59]. Bir başka alternatif araştırma da Librosa kullanarak Mel-spektrogramlarını çıkarmak, OpenL3 [60] kullanarak akciğer seslerinden gömülmeleri çıkarmak ve çıktılarını birleştirmek olabilir. Daha sonra bu kombinasyonun sonucunu, akciğer sesleri sınıflandırma problemine devam etmek için bir derin öğrenme modeline girdi olarak kullanılabilir.

Bilim Kodu : 92432
Anahtar Kelimeler : Akciğer sesleri sınıflandırması, derin öğrenme, veri artırma, MFCC
Sayfa Adedi : 73
Danışman : Doç.Dr. Oktay YILDIZ

CLASSIFICATION OF ABNORMAL RESPIRATORY SOUNDS USING DEEP LEARNING TECHNIQUES

(M. Sc. Thesis)

Ahamadi Abdallah IDRISSE

GAZI UNIVERSITY

INSTITUTE OF INFORMATICS

March 2023

ABSTRACT

Lung disorders encompass some of the deadliest diseases worldwide. The health status of a lung could be discovered by examining the sounds that are generated when the air flows in the respiratory system. Classification of lung sounds has to be done accurately to avoid misdiagnosis and thus minimize lung-related deaths. A doctor traditionally uses a stethoscope to listen to a patient's lung sounds, then provides a diagnosis, in which the trustworthiness of the outcomes depends massively on the skills of the doctor who is in charge of this task. An Automatic Recognition System (ARS) can help health care professionals achieve an accurate and efficient lung sound classification. The objective of this research is to suggest using pre-trained deep learning-based models that automatically classify lung sounds found in a limited dataset collected from different patients at a public hospital in Turkey. The well-known MFCCs (Mel-Frequency Cepstrum Coefficients) audio characteristics are derived from the lung sounds and used as inputs to four different pre-trained models for lung sound classification. From the experiment, we noticed that the utilization of data augmentation techniques on lung sounds facilitated a better sound classification performance, and as a result, we reported an increase in the average accuracy from 81% to 98%.

Science Code : 92432
Key Words : Lung sounds classification, deep learning, data augmentation, MFCC
Page Number : 73
Supervisor : Assoc. Prof. Oktay YILDIZ

ACKNOWLEDGMENTS

İlk olaraka, bana yardım eden ve yüksek lisans araştırmamı tamamlamamı sağlayan Yüce Allah'a hamd ediyorum. Tezim boyunca beni yönlendiren Sayın Doç. Dr. Oktay YILDIZ'a özel ve sonsuz teşekkürlerimi sunarım. Onun değerli tavsiyeleri, sürekli desteği, engin bilgisi ve tecrübesiyle araştırmam sırasında bana oldukça faydalı oldu. Bu tez ona borçludur. Tüm bunlar için ve sabrınız için teşekkür ederim hocam. Bana destek olan, tavsiyeleri ve dualarıyla her zaman yanımda olan anne ve babama en içten teşekkürlerimi sunarım. Son olarak, bana öğreten, bana tavsiyede bulunan, beni cesaretlendiren ve bu aşamaya gelmeme, yüksek lisans çalışmalarımı tamamlamama yardım eden herkese teşekkür etmek istiyorum.

Ayrıca; Tez çalışmamızda kullanılan veri kümesinin sağlanmasında katkıları bulunan Prof.Dr. Uğur BAYSAL'a teşekkür ederim.

Klinik çalışmalarda kullanılan veri kümesi, Hacettepe Üniversitesi Girişimsel Olmayan Araştırmalar Etik Kurulu 29.06.2021 tarih 2021/13 nolu kararı ile onaylanmıştır. İlgi ve yardımları için teşekkür ederim.

TABLE OF CONTENTS

Page

ÖZET iv

ABSTRACT xv

ACKNOWLEDGMENTS xvi

TABLE OF CONTENTS xvii

LIST OF TABLES xx

LIST OF FIGURES xxi

LIST OF ABBREVIATIONS xxiii

1. INTRODUCTION 1

2. LITERATURE REVIEW 3

3. BACKGROUND 13

3.1. Lung Sounds 13

3.1.1. Normal lung sounds 14

3.1.2. Abnormal lung sounds 15

3.2. Audio Data Features 17

3.3. Time domain features 17

3.3.1. Frequency domain features 19

3.3.2. Cepstral domain features 21

3.3.3. Discrete wavelet features 24

3.3.4. Image-based features 24

3.3.5. Deep features 24

3.3.6. Sparsity features 25

4. MATERIALS AND METHOD 27

4.1. Dataset 27

4.2. Machine Learning 27

Page

4.2.1. Classification 28
4.2.2. Artificial neural network 28
4.2.3. Multilayer perception 29
4.2.4. Activation functions 30
4.2.5. Training neural network 34
4.3. Convolutional Neural Networks 34
4.3.1. VGG 36
4.3.2. MobileNet 37
4.3.3. Xception 39
4.3.4. ResNet 40
4.4. Data Augmentation 40
4.5. Evaluation Metrics 42
4.5.1. Accuracy 43
4.5.2. Confusion matrix 43
4.5.3. Precision 43
4.5.4. Recall 44
4.5.5. Specificity 44
4.5.6. F1-Score 44
5. EXPERIMENTAL STUDIES AND RESULTS 45
5.1. The Proposed Methodology 45
5.1.1. Data acquisition phase 47
5.1.2. Audio data augmentation phase 48
5.1.3. Feature extraction phase 49
5.1.4. Classification phase 50
5.2. Results Without Using Data Augmentation Techniques 51
5.3. Results When Using Data Augmentation Techniques 53

Page

5.3.1. VGG 54
5.3.2. MobileNet 55
5.3.3. Xception 56
5.3.4. ResNet 58
6. DISCUSSION 61
7. CONCLUSION 65
REFERENCES 67
CURRICULUM VITAE 73

LIST OF TABLES

Table **Page**

Table 4.1. Confusion matrix..... 43

Table 5.1. Results before the utilization of data augmentation techniques..... 51

Table 5.2. The results achieved by the VGG model 55

Table 5.3. The results achieved by the MobileNet model..... 56

Table 5.4. The results achieved by the Xception model. 57

Table 5.5. The results achieved by the ResNet model. 58

Table 6.1. Results after the utilization of data augmentation techniques..... 63

LIST OF FIGURES

Figure **Page**

Figure 3.1. A stethoscope 13

Figure 3.2. An MFCCs representation of a lung sound 23

Figure 4.1. A basic artificial neuron [38] 29

Figure 4.2. Multilayer perception 29

Figure 4.3. Graphical representation of the Sigmoid activation function 31

Figure 4.4. Graphical representation of the ReLU activation function 32

Figure 4.5. Graphical representation of the Tanh activation function 33

Figure 4.6. Graphical representation of the SoftMax activation function 34

Figure 4.7. A basic CNN architecture showing its principal phases [41] 35

Figure 4. 8. The architecture of the VGG16 model [61] 37

Figure 4.9. The architecture of the MobileNet [62]. 38

Figure 4.10. Overall Architecture of Xception. 39

Figure 4.11. The ResNet-50 architecture [63]. 40

Figure 5.1 Our proposed diagnostic methodology 47

Figure 5.2 Applying data augmentation on some samples. 48

Figure 5.3 MFCCs when applying data augmentation techniques. 49

Figure 5.4 VGG curves before data augmentation. 52

Figure 5.5 MobileNet curves before data augmentation. 52

Figure 5.6 Xception curves before data augmentation. 53

Figure 5.7 ResNet curves before data augmentation. 53

Figure 5.8. VGG curves after data augmentation 55

Figure 5.9. MobileNet curves after data augmentation 56

Figure 5.10. Xception curves after data augmentation 58

Figure 5.11. ResNet curves network after data augmentation 59

Figure **Page**

Figure 6.1. Confusion matrix of MobileNet network 61

Figure 6.2. Confusion matrix of the Xception network 62

LIST OF ABBREVIATIONS

The definitions of the acronyms used in our thesis are listed below.

Acronym	Meaning
1D	One Dimensional (Tek Boyutlu)
AD	Amplitude Descriptor (Genlik Açıklayıcısı)
ADSR	Attach, Delay, Sustain, Release (İliştirme, Geciktirme, Sürdürme, Bırakma)
ANA	Artificial Noise Addition (Yapay Gürültü Ekleme)
ANN	Artificial Neural Network (Yapay Sinir Ağı)
ARS	Automatic Recognition Systems (Otomatik Tanıma Sistemleri)
BLSTM	Bidirectional Long Short-Term Memory (Çift Yönlü Uzun Kısa Süreli Bellek)
BN	Background Of Noise (Gürültü Arka Planı)
CELP	Code Excited Linear Prediction (Kod Uyarımlı Doğrusal Tahmin)
CNN	Convolutional Neural Networks (Evrişimli Sinir Ağları)
COPD	Chronic Obstructive Pulmonary Disease (Kronik Obstrüktif Akciğer Hastalıkları)
COVID-19	Coronavirus Disease (Koronavirüs Hastalıkları)
CRD	Compression Of Range Dynamically (Aralığın Dinamik Olarak Sıkıştırılması)
CWT	Continuous Wavelet Transform (Sürekli Dalgacık Dönüşümü)
DBN	Deep Belief Networks (Derin İnanç Ağları)
DCT	Discrete Cosine Transform (Ayrık Kosinüs Dönüşümü)
DDAE	Data De-Noising Auto Encoder (Veri Gürültü Arındırma Otomatik Kodlayıcı)
DNN	Deep Neural Networks (Derin Sinir Ağları)
EMD	Empirical Mode Decomposition (Ampirik Mod Ayrıştırması)

Acronym	Meaning
EMS	Envelope Modulation Spectrum (Envelope Modülasyon Spektrumu)
FF	Fundamental Frequency (Temel Frekans)
FFT	Fast Fourier Transform (Hızlı Fourier Dönüşümü)
FL	Focal Loss (Odak Kaybı)
FT	Fourier Transform (Fourier Dönüşümü)
GA	Genetic Algorithm (Genetik Algoritma)
GDF	Group Delay Function (Grup Gecikme Fonksiyonu)
GFCCs	Greenwood Function Cepstral Coefficients (Greenwood Fonksiyonu Cepstral Katsayıları)
GPU	Graphical Processing Unit (Grafik İşlemci Birimi)
GTCCs	Gammatone Cepstral Coefficients (Gammatone Cepstral Katsayıları)
HOG	Histogram Of Gradients (Gradyanların Histogramı)
HOS	High Order Statistics (Gradyanların Histogramı)
ICBHI	International Conference On Biomedical Health Informatics (Uluslararası Biyomedikal Sağlık Bilişimi Konferansı)
ILSA	International Lung Sounds Association (Uluslararası Akciğer Sesleri Derneği)
ILSVRC	ImageNet Large Scale Visual Recognition Challenge (ImageNet Büyük Ölçekli Görsel Tanıma Yarışması)
KNN	K-Nearest Neighbours (K-En Yakın Komşular)
LAT	Long Attack Time (Uzun Saldırı Süresi)
LBP	Local Binary Pattern (Yerel İkili Patern)
LDA	Linear Discriminant Analysis (Doğrusal Diskriminant Analizi)
LPC	Linear Predictive Coding (Doğrusal Tahmine Dayalı Kodlama)
LPCCs	Linear Prediction Cepstral Coefficients (Doğrusal Tahmine Dayalı Kodlama Katsayıları)

Acronym	Meaning
LR	Logistic Regression (Lojistik Regresyon)
LSF	Linear Spectral Frequency (Doğrusal Spektral Frekans)
LSTM	Long Short-Term Memory (Uzun Kısa Süreli Bellek)
LTAS	Long-Term Average Spectrum (Uzun Süreli Ortalama Spektrum)
LTPs	Local Ternary Patterns (Yerel Üçlü Paternler)
MFCCs	Mel-Frequency Cepstrum Coefficients (Mel-Frekans Cepstrum Katsayıları)
MIC	Maximal Information Coefficient (Maksimal Bilgi Katsayısı)
ML	Machine Learning (Makine Öğrenimi)
MSAF MULTIEXPANDED	Method Of Selection Of Amplitudes Of Frequency Multi-Expanded Filter Based Features (Frekans Genişletilmiş Filtre Tabanlı Özelliklerin Genliklerinin Seçimi Yöntemi)
MUDA	Music Data Augmentation (Müzik Verilerinin Artırılması)
NB	Naïve Bayes
OBSC	Octave Based Spectral Contrast (Oktav Tabanlı Spektral Kontrast)
PLP	Perceptual Linear Prediction (Algısal Doğrusal Tahmin)
RADWT	Rational-Dilation Wavelet Transform (Rasyonel-Dilasyon Dalgacık Dönüşümü)
ResNet	Residual Network (Artık Ağ)
RNN	Recurrent Neural Networks (Tekrarlayan Sinir Ağları)
ROC	Receiver Operating Characteristic (Alıcı İşletim Karakteristiği)
RSD	Respiratory Sound Database (Solunum Sesi Veritabanı)
RSFS	Random Subset Feature Selection (Rastgele Alt Küme Özellik Seçimi)
SAE	Deep Stacked Auto-Encoder (Derin Yığılmış Otomatik Kodlayıcı)
SIFT	Scale Invariant Feature Transform

Acronym	Meaning
SMOFS-MULTIEXPANDED	Shortened Method Of Frequencies Selection MULTIEXPANDED (Kısaltılmış Frekans Seçimi Yöntemi MULTIEXPANDED)
SP1	Shift Pitch1 (Kaydırma Perdesi 1)
SP2	Shift Pitch2 (Kaydırma Perdesi 2)
SPSF	Stereo Panning Spectrum Feature (Stereo Kaydırma Spektrumu Özelliği)
SQS	Sequential Forward Selection (Sıralı İleri Seçim)
ST	Stretching Time (Süre Gerilmek)
STE	Short Time Energy (Kısa Zamanlı Enerjisi)
STFT	Short-Time Fourier Transform (Kısa Zamanlı Fourier Dönüşümü)
SVM	Support Vector Machine (Destek Vektör Makinesi)
TC	Temporal Centroid (Zamansal Centroid)
TPU	Tensor Processing Unit (Tensor İşleme Birimi)
ULSTM	Unidirectional Long Short-Term Memory (Tek Yönlü Uzun Kısa Süreli Bellek)
VGG	Visual Geometry Group (Görsel Geometri Grubu)
ZCR	Zero Crossing Rate (Sıfır Geçiş Oranı)

1. INTRODUCTION

Lung sounds are made when air flows into the respiratory system. These sounds provide vital information about a patient's pulmonary condition. In order to examine patients with pulmonary disorders, medical professionals traditionally utilize a stethoscope to listen to lung noises. Identifying whether lung sounds are normal or abnormal is a critical process because a wrong diagnosis could endanger a patient's health. Because lung sounds are difficult to distinguish due to their nonlinearity and non-stationarity structures, the doctor or physician in charge of this task must be well-trained. Automatic recognition systems (ARS) may offer a solution to the issue of lung sound classification for healthcare professionals [1, 2]. The mechanized respiratory sound examination has recently gained popularity as a more reliable alternative to traditional auscultation methods for diagnosing lung sounds by using electronic stethoscopes along with AI and pattern recognition [3]. It does this by overcoming the drawbacks of traditional auscultation techniques and giving clinicians a solid tool for a quicker and more accurate diagnosis.

Lung sounds are categorized into two main types: normal and adventitious breathing. When there are no respiratory difficulties, normal breathing sounds occur; however, adventitious breathing sounds arise when various pulmonary disorders present. Normal breathing sounds are a combination of tracheal, bronchial, and bronchovesicular noises. A healthy breathing sound is recognized on the thoracic wall as a low noise in intake and a scarcely audible noise in exhalation. When regular breathing sounds are accompanied by extra noises, adventitious breathing sounds are detected. This type of noise could be continuous for long periods, such as wheezing, or it could be discontinuous, like crackles. There are some lung sounds, such as squawks, that possess both aforementioned features. The existence of this kind of sound means a lung disorder [4, 5].

In this work, we dealt with three categories of lung sounds, which are normal, rhonchus, and crackle. Rhonchus is a low-pitched, continuous sound that is detectable during breathing. Rhonchus sounds can be found in people struggling with bronchial diseases like asthma and bronchitis. Crackles are discontinuous abnormal sounds that occur frequently in breathing. Crackle sounds can be characterized according to their duration or according to their location in the cycle of respiration. Crackle sounds have two types based on duration: fine crackle and coarse crackle. Fine crackle has a short duration, whereas coarse crackle has a long

duration. Crackle's appearance in lung sounds causes difficulties in the airways [4]. Fine crackles and coarse crackles were not separated in the dataset used for this work.

Guler et al. [6] proposed machine learning techniques on this dataset for classifying different lung sounds. However, we discovered that none of the state-of-the-art methods that are accepted in the literature have utilized deep learning mechanisms on this dataset for lung sound classification. In this research, we are planning to extract the most used feature in the audio domain, MFCCs (Mel-Frequency Cepstrum Coefficients), and use them as inputs for the deep learning models which are the VGG, MobileNet, Xception, and ResNet. Our main contributions could be summarised in this way: (a) to propose using pre-trained deep learning-based models for classifying the lung sound dataset used in [6] (b) to carefully investigate the performance of the models on our dataset (c) to inspect the influence of data augmentation techniques on the results of the pre-trained models in classifying lung sounds. Since these deep learning models were already trained with huge datasets, we can use their pre-trained weights as a proper network initialization for efficient and reduced training time. Since deep learning models train well on large amounts of data and our dataset is very small, we opted to deploy audio data augmentation techniques to create additional training samples. As a result, the use of data deformation techniques, such as pitch-shifting, along with our pre-trained models has contributed to much better classification accuracy.

The structure of the thesis is as follows: Within Chapter 2 (Related Work), to familiarise ourselves with the feature extraction methods, and classification techniques like machine learning and deep learning used for categorizing lung sounds as well as the various performance metrics used to assess the models, a literature review on prior research related to lung sound classification was conducted. We explained the fundamental concepts behind respiratory sounds in Chapter 3 (Background), along with their classes and many feature types that may be retrieved from an audio signal. In Chapter 4 (Materials and Methods), the dataset, feature extraction techniques, data augmentation techniques, as well as evaluation metrics that were employed in the thesis, are all presented. In Chapter 5 (Experimental Studies and Results), we presented our method, discussed how the experiments were carried out using our proposed approach, and presented the various outcomes obtained from the experiments. Finally, in Chapters 6 and 7 (Discussion and Conclusion), we discuss the experimental findings of the proposed methodology and give feature research ideas.

2. LITERATURE REVIEW

In this thesis, we went through research related to lung sound classification in order to get knowledge on the different datasets, feature extraction methods, machine learning, and deep learning that are used in lung sound classification problems. The proposed methods in the reviewed articles and their performances are presented in this section.

Li and Hong [7] presented an automatic model for detecting crackles. The lung sound dataset used in their work is a composition of 140 lung audio files. The majority of the sounds have a period of 5 seconds and a frequency value of either 4000Hz or 4800Hz. Their proposed method has three main steps. First, a filtering method is used to reduce the frequencies of the lung sounds to be in between 150Hz and 1800Hz. Second, three features are derived from the filtered sounds. Third, the classification task is accomplished by using SVM, a supervised learning method, that was able to obtain a classification accuracy value ranging from 97.14% to 100%.

Ulukaya et al. [8] implemented a wavelet-based system to classify a lung sound dataset that was collected from 14 male and 6 female subjects by using a 14-channel data acquisition system. Q-factor features, which can be obtained by using Rational-Dilation Wavelet Transform (RADWT) method, were extracted and used as inputs for two machine learning methods (KNN and SVM). The results showed that the SVM method achieved a total classification accuracy value of 95.17%.

Altan et al. [9] proposed a Deep Belief Networks (DBN) model to classify asthmatic and healthy lung sounds. They employed the Hilbert Huang Transformation (HHT) method for extracting statistical features which will be used as input to feed the classifier. After assessing the model by using a 5-cross validation metric, the DBN model achieved an accuracy of 84.61%.

Balli and Kutlu [10] conducted aimed to investigate how window size can affect the classification of abnormal respiratory sounds. Their dataset is composed of 127 lung sounds recorded for 20 seconds at a frequency of 44100Hz. MFCC are derived from the lung sounds in several window size values. KNN is the machine learning algorithm used as a classifier in this work, and it reached a classification accuracy of 93.21% when the sounds have a

window size of 5 seconds. In this work, the authors emphasized the important role that window size has in the detection of abnormalities in breathing sounds. They concluded that the dataset becomes more significant when the window size is between 2 and 10 seconds.

Guler et al. [6] deployed machine learning techniques to classify a dataset of lung sounds that was recorded form 60 patients in a public hospital in Turkey. They used Librosa (a python library) and Jaudio (a Java-based software), for extracting features several audio features from the audio files. Specific features were chosen as input for the models by using feature selection methods. They assessed the models by utilizing the f-cross validation metric, and the Naïve Bayes algorithm achieved better results than the other algorithms, 95% with 10-cross validation.

Aras and Gangal [11] proposed an automatic recognition method for classifying single-channel lung sounds dataset that was recorded by using an electronic stethoscope that has a single-channel output. The lung sounds were collected from several subjects who were examined at the hospital of Karadeniz Technical University. The collected files are 70 crackles, 70 rhonchi, and 70 normal files which totalize 210 lung sounds. This audio recording process was conducted by two doctors. They extracted MFCCs features, derived several representations from these MFCCs, and used a feature selection method called Sequential Forward Selection (SQS) for the purpose of finding the best feature or best feature combination so they can be used as inputs to the different machine learning classifiers. KNN obtained the highest classification accuracy when a combination of specific features is provided to the algorithm.

Islam et al. [12] used a 4-channel data acquisition system for the collection of lung sounds from 60 patients, of which 30 are healthy and 30 are asthmatic. For the feature extraction process, Welch's method is used to compute the power spectral density from each lung sound respiratory cycle and then decompose it into uniform subbands. For the classification phase, SVM and ANN algorithms were fed by some statistical values calculated from the subbands. The SVM method achieved the highest classification accuracy, 93.3%, through a combination of 3 channels (1,2,4) from the 4-channels.

Bardou et al. [1] compared some machine learning techniques (SVM, KNN, and GMM) with a CNN model on a lung sound dataset. MFCCs and LBP features were used as input for the

machine learning approaches while spectrograms extracted from the lung sounds audio recordings were the input of the CNN algorithm. Some data augmentation techniques were employed in this work only on the spectrograms to overcome the problem of overfitting. The CNN model comes out as the best classifier with an accuracy value of 95.56%.

Haider et al. [13] employed machine learning algorithms to distinguish normal and adventitious respiratory sounds. They used a dataset of lung sounds composed of 30 COPD and 25 healthy recordings. Then some various features were extracted from these lung sounds and fed to different machine learning techniques like SVM, and KNN. During the experiment, they assigned different parameters to the models. SVM was the best classifier by reaching an accuracy value of 83.6% with some parameters. While they used different parameters, SVM and LR classifiers achieved accuracy values of 100%.

Elsetrønninga et al. [2] developed supervised and unsupervised machine learning techniques for the classification of crackle and no-crackle lung sounds. ICBHI is the dataset used in this work. This dataset is composed of 920 audio files. After employing feature extraction and feature selection techniques, MFCCs were chosen as the best inputs for the classifiers. K-NN was the algorithm that reached the highest classification accuracy 84.38%. In this work, the researchers paid more attention to decomposition feature extraction and feature selection approaches.

Messner et al. [14] deployed a CNN model to classify a multi-channel lung sound dataset that is composed of consists of healthy and pathological lung sounds recorded from different patients by using a 16-channel sound recording device. They extracted spectrograms from the lung sounds and utilized them as input to feed the models. After evaluation, their proposed CNN framework outperformed the other methods by reaching an F-score of 92%.

Shuvo et al. [15] implemented a lightweight CNN model for the categorization of respiratory conditions from lung sounds. The dataset used in this article contains 920 audio samples which have a total of 5.5 hours of audio files. The sounds have different lengths which range between 10s to 90s. In the pre-processing phase, they resampled the audio files to a unique frequency value of 22050Hz. Then they segmented the lung sounds according to their respiratory cycle. Two types of scalograms were created from the segmented data, by using CWT or by using a hybrid EMD-CWT approach. Among their proposed lightweight CNN

model, the well-known VGG16 was also used to complete the classification task. The hybrid scalogram proved to be the more significant feature because the best classification accuracy, a value of 99.05%, was registered by the authors' proposed CNN algorithm when using the hybrid scalogram as input.

Rocha et al. [16] proposed different machine learning and deep learning techniques for automatically classifying wheeze sounds. They aimed at investigating how the event duration affects the classification of wheeze sounds. They used a data set called Respiratory Sound DataBase (RSD) that was collected from 126 patients. Since the audio recordings were recorded with different frequencies, the authors resampled the audio files to the lowest frequency value (4000Hz) that was in the database. They employed the STFT to produce a spectrogram from each audio file. The generated spectrograms are directly fed to the CNN model. While for the traditional ML algorithms, they used a MIR Toolbox to derive 47 features per frame from each spectrogram. The experimental findings explain that without changing parameters, the best-obtained results are 98%, and 95% values of sensitivity and specificity in that order were gained by the CNN model. When they changed the event duration parameter, the best classifier (Boost) registered only 55% and 76% for sensitivity and specificity respectively.

Gairola et al. [17] proposed a CNN-based method, called RespireNet, for classifying a lung sound dataset. They deployed data augmentation approaches to the lung audio files, they extracted Mel-spectrogram features from each lung sound. Then they employed a method called black region clipping which will separate from each Mel-spectrogram the empty area (that does not convey information) and the area that holds information about the lung sound. The section of the Mel-spectrogram that has relevant information is selected as input to the CNN model. RespireNet outperformed the state-of-art result for 4-class classification problems.

Don [18] investigated the role of feature selection methods in classifying lung sounds. He collected data from different sources, online and offline. Finally, 85 audio files were considered for the study. He extracted 15 different features from the lung audio files and used two feature selection methods so they can determine the suitable features for the classifiers. The selected features are then fed to three machine learning algorithms to accomplish the lung sound classification purpose. KNN achieved the best scores with a value

of 94.1% in terms of precision and f-measure. Whereas, both SVM and NB algorithms obtained a 100% of recall value. The author paid more attention to the influence of feature selection methods, and he concluded that among the 2 feature selection methods that he employed RSFS (Random Subset Feature Selection) selected more significant features for the classifiers than the SFS (Sequential Forward Selection).

Demir et al. [19] proposed a CNN-based model to automatically classify lung sounds. They used the lung audio files that are available on the dataset ICBHI. Spectrograms that are a 2D representation of time over frequency, are extracted from the audio recordings by using the STFT method. The spectrograms are saved as images. But before feeding them to the deep learning models, the images are resized to 224×224 so they can be suitable for the deep learning models. VGG16, AlexNet, and ResNet-50 are trained on the resized spectrograms to extract deep features. Then SVM and SoftMax are utilized to make the lung sounds classification from the deep features and transfer learning respectively. The computational results demonstrated that SVM outperformed by obtaining a classification accuracy of 65.50%.

Zulfiqar et al. [3] proposed an ANA based methodology for classifying respiratory sounds. They applied the Fourier transformation to create a spectrogram of respiratory sound, then added some artificial noise to make the spectrogram clearer and more robust. Next, they extracted features through a feature map on spectral data to feed their proposed classifiers as inputs (pre-trained models such AlexNet, and Inception). AlexNet was the model that gained the highest classification accuracy.

Datta et al. [20] presented an automatic tool for the detection of pulmonary conditions. They collected 52 lung sounds audio files from diverse sources. Hence the audio files have different frequencies. That is why in the pre-processing stage, first the authors down-sampled the lung sounds to a unique frequency value of 4KHz. Then they deployed Empirical Mode Decomposition (EMD) to remove unwanted sounds that come with the lung sounds. In the feature extraction stage, spectral features, spectrogram features, wavelet features, and cepstral features are extracted from the pre-processed lung sounds. MIC (Maximal Information Coefficient) which is a feature selection method is used to avoid irrelevant features. MIC method provides a score that allows arranging the features in order of importance. A linear SVM algorithm is utilized for the classification of lung sounds based

on the selected features. Finally, the proposed tool registered a classification accuracy of 80%.

Aras et al. [21] proposed a machine learning technique to classify healthy and pathological lung sounds. The dataset used in this research is composed of 21 normal and pathological sounds. LPC coefficients, MFC Coefficients mean, and standard deviation are the features extracted from the audio files. The features are used first separately and second combined when being fed to the KNN algorithm. A classification accuracy value of 100% was obtained.

Naves et al. [22] proposed a divide-and-conquer technique to classify a lung sounds dataset that has a total of 35 audio files of different classes. The authors pre-processed the audio files by splitting them into non-overlapping segments. High Order Statistics (HOS) is the method used to extract characteristics from the audio files. This feature extraction method resulted in a total of 800 features. In order to avoid redundancy and non-significant features, the authors deployed a Genetic Algorithm (GA). Two machine learning algorithms, KNN and NB are deployed to perform the lung sound categorization task based on the GA-selected features. After the computational results, a classification tree has proved to be the better classifier. First, the tree-classifier starts by using KNN to classify the following classes vesicular, wheezes, and crackles. Then NB is used to classify the subclasses of both wheezes and crackles. The classification accuracy values registered are 98.1% and 94.64 in training and validation respectively.

Fairwan et al. [23] presented a CNN+BDLSTM-based model for the recognition of pulmonary conditions from lung sounds. In the data preparation step, two different sources were used. The first source is a set of lung sounds recorded at a public university in Jordan. The second source is the popular public lung sound dataset ICBHI. This second dataset was not used completely, only 110 audio files were considered. Due to the diverse frequencies of the lung sounds, a fixed frequency of 4KHz is chosen for the entire lung audio files. Then each lung sound was divided into segments of 5 seconds, hence resulting in a total of 1182 segments. The lung sounds were filtered in order to remove noises. In the pre-processing steps, three processes were conducted. These are wavelet smoothing, displacement artifact removal, and normalization of z-scores. The combination of a CNN and BDLSTM is used to extract significant spatial dimensionality and time-domain features from the segmented-

pre-processed lung sounds. The network was assessed by several classification metrics such as confusion matrix, and F1-score. The model gains a total classification accuracy value of 99.62%.

Nguyen and Pernkopf [4] introduced a deep learning model for detecting crackles in lung sounds. The proposed model is a transfer learning-based model. Two different datasets were utilized in this study. The first dataset is the well-known ICBHI dataset, and it was employed as a source dataset to train the CNN network only. The learned knowledge by the CNN model during this training is used when performing the lung sounds classification task in the second dataset called the target dataset. The lung sounds of the two datasets were collected with different sample rates. Hence the authors resampled all the audio files to 16KHz. Then, the authors opted also for a fixed length for the separation of respiratory phases of the sounds. This fixed length facilitates because CNN models require a unique shape from their inputs. Spectrograms are the feature representations extracted from the lung sounds. For the aim of avoiding overfitting during the training process, different audio data augmentation methods were deployed. In the computational experiments, they trained the CNN network on the ICBHI dataset first. Then they used the CNN model as a pre-trained model for classifying lung sounds in the multichannel target dataset. The authors conclude that the proposed method showed remarkable results.

Bozkurt et al. [24] developed a deep learning model to automatically classify heart sounds with pathological conditions with the aim of analyzing the input features. Two large datasets are used in this study, and they are the UoC murmur database and the physioNet-2016 database. In this study, the authors extracted three different features which are MFCC, Mel-spectrogram, and sub-band envelopes. The extracted features are used to train the CNN model separately, hence the authors can find out which feature type is providing more useful information to the classifier. ROC curve is employed to evaluate the performance of the model. After several computational experiments, the authors were able to conclude that the sub-band envelopes features are providing more significant features to the CNN model.

Lella and Pja [25] proposed a 1D CNN to automatically discover COVID-19 disease according to the cough, breath, and voice signals of different patients. The dataset used in this study was recorded by Cambridge University in the United Kingdom. The authors applied five data augmentation methods to the sounds before the feature extraction phase.

The data deformation methods are Stretching Time (ST), Shift Pitch1 (SP1), Shift Pitch2 (SP2), Compression of Range Dynamically (CRD), and Background of Noise (BN). They were implemented by using a python library called MUDA which stands for Music Data Augmentation. More information about this library can be found available in [26]. DDAE algorithm is deployed to extract deep features for the 1D CNN model. In this study, the authors preferred the DDAE deep features over MFCCs. Several computational experiments were conducted, and as result, the model could classify healthy sounds, asthmatic sounds, and COVID-19 sounds with an accuracy of 90%.

Srivastava et al. [27] presented a deep learning-based model in order to support healthcare professionals in the diagnosis of COPD according to respiratory sounds. The dataset used is the public database ICBHI. This dataset was created with the collaboration of 2 laboratories from Greece and Portugal. The dataset was collected from healthy and non-healthy male and female subjects who had different types of respiratory issues. The authors utilized Librosa, which is a Python library, for the pre-processing and feature extraction phases. They extracted several audio features from the respiratory sounds. These features are used separately as inputs to the classifier. Srivastava et al. also applied data augmentation methods to balance the dataset because non-healthy sounds were approximately quadruple of healthy sounds. A CNN model was employed to accomplish the classification purposes. The experiment results showed that MFCC inputs fed the classifier more significant features for the classification of COPD. Hence the proposed CNN model with MFCCs as inputs was able to reach an ICBHI score of 93%.

Petmezas et al. [28] introduced a hybrid deep learning model that utilizes a Focal Loss (FL) function to automatize the classification of lung sounds in the ICBHI dataset. This dataset is an open-access database that was manually annotated by experts from 2 different countries. Because of the diverse frequency values of the lung audio files, the authors decided to resample the audio files to a unique sample rate value of 4000Hz. They also chose a fixed length of 2.7 seconds for the sounds. In the feature extraction process, the STFT algorithm, which provides a time-frequency representation of a signal known as a spectrogram, was deployed. The extracted spectrograms were saved as images that will be used as input for the hybrid model. The proposed hybrid network is composed of a 2D CNN followed by an LSTM and FL function. The spectrogram images are fed to the CNN block, so it can extract deep features that will be propelled to the LSTM block in which the classification task is

held. The ICHBI is known as an imbalanced database. For this reason, the authors implemented an FL function to handle this non-balanced issue and to evaluate the performance of the deep learning model. Several experiments were carried out, the obtained results demonstrated that this combination of CNN, LSTM with FL function was able to achieve remarkable results. According to the authors, this is the first research that proposed this methodology.

In this section, we studied the literature regarding respiratory sound classification and then proposed a deep learning-based model for classifying an upgraded version of the lung sound dataset used in [6]. The details of our proposed methodology are given in Chapter 5.

3. BACKGROUND

This chapter provides the foundational knowledge and concepts needed to successfully achieve our goal of categorizing lung sounds using deep learning methods. Thus, there are details provided on respiratory sounds, different categories of features that could be extracted from an audio signal, along with machine learning and deep learning methods that perform audio classification, amongst others.

3.1. Lung Sounds

Auscultation is a technique through which healthcare professionals listen to the sounds over the chest wall by using a stethoscope. A stethoscope is a medical instrument used by professionals in the healthcare domain for diverse purposes. Figure 3.1. below illustrates what a stethoscope is.



Figure 3.1. A stethoscope

The sounds coming from the chest region can come from different organs for example the heart, and lungs. The noises produced when the air flows into the respiratory system are called lung sounds. These lung sounds convey important information about a patient's pulmonary health. Traditionally doctors prefer using a stethoscope to listen to lung sounds, so they will be able to provide a diagnosis in a short time. This technique is opted for mostly because of its simplicity, cheapness, and rapidity to provide a diagnosis. However, when using the auscultation technique, the skills of the doctor are crucial in providing a correct

diagnosis. Because lung sounds are sounds with nonlinearity and non-stationarity structures, that why they are difficult to be distinguished [1, 3].

The classification criteria for lung sounds were presented at the 10th ILSA Conference held in Tokyo, Japan in 1980. Since then, there has been an increasing interest in studying and categorizing respiratory sounds. Breathing sounds can be divided into two categories: normal and adventitious breathing sounds. Normal lung sounds are the noises that are heard without additional respiratory sounds, while adventitious lung sounds come with additional sounds, in the existence of pulmonary conditions [3]. These two categories of respiratory sounds are discussed in more depth in the section that follows.

3.1.1. Normal lung sounds

Breathing sounds are produced by the respiratory system and they can be heard by performing auscultation which is commonly utilized to determine a patient's physical health, as well as his or her respiratory and cardiac sounds. In the absence of respiratory problems, a patient's breathing sounds are considered normal. The characteristics (sound quality, duration, and pitch) of a normal breathing sound differ according to the region at which the stethoscope is placed during the auscultation [3, 29].

Normal vesicular sounds are heard if the auscultation is done over most lung fields. These sounds are smooth, non-musical sounds and are generally heard during the entire inhalation process. They can also be audible in the exhalation phase, but only in the earliest stages. Vesicular normal sounds have a long duration during the inhalation phase than they have in the exhalation phase. Therefore, they have their highest frequencies during the inspiration. The range of frequency of the vesicular normal sounds is between 100-200Hz, which is a low pitch. This is because the chest wall behaves as a low-pass filter for the sound produced [29, 30].

Over the major airways on the chest, normal bronchial noises can be heard, particularly in the region between the second and third intercostal spaces. Bronchial normal noises are detectable during both the inhalation and exhalation phases [1]. Unlike vesicular normal noises, since the bronchial noises came from broader airways, the bronchial noises during the expiratory phase often last longer than those during the inspiration phase. In general,

bronchial normal sounds are loud, high-pitched, and tubelike with more energy at higher frequencies than vesicular normal sounds do have [30].

When the auscultation is done at the posterior chest between the scapulae or at the middle section of the anterior chest, normal broncho-vesicular sounds should be audible. Broncho-vesicular sounds last the same amount of time in both the inspiration and expiratory phases. The sound quality is somewhere between the quality of bronchial and vesicular sounds. In contrast to bronchial sounds, they are softer, but yet imitate tubular sounds [30].

The central airways, which are influenced by turbulent airflow below the glottis, are the source of the breathing sounds that are audible from the mouth. The frequency range of a mouth breathing sound is between 200 to 2,000 Hz. When a patient does not have respiratory issues, his mouth's breathing sounds are silent [30].

Tracheal normal sounds are breathing sounds detectable when the doctor does auscultation over the trachea, especially on the jugular notch of a patient. Due to turbulent airflow in the pharynx glottis, normal tracheal sounds are generally harsh, loud, and cylindrical. This type of normal sound contains more power in higher-frequency components than the other types. The scope of their frequency values varies between 800 to 5,000 Hz. Because tracheal normal sounds are heard through the trachea and are not filtered by the chest wall, they have a higher intensity and can provide more information related to the actual health condition of a patient’s lung [30].

3.1.2. Abnormal lung sounds

Abnormal sounds are undesirable breathing sounds that are overly imposed on normal noises. Adventitious respiratory sounds have no force during both of inspiration and expiration phases. These unwanted noises are categorized according to various criteria that aid in the identification of each class independently. Abnormal breathing sounds are widely categorized into two types: continuous and discontinuous. This distinction is based on the variation in their period of recurrence when breathing. Continuous abnormal sounds have a duration of 250ms generally, e.g., wheezes. According to the pitch, continuous abnormal sounds are either high-pitched (wheeze and stridor) or low-pitched (squawk and rhonchi) sounds. Discontinuous abnormal sounds have a duration not superior to 25ms and are often

subdivided into fine crackle, coarse crackle, and pleural rub [1, 3]. Below further details on the different categories of adventitious lung sounds are provided.

Wheeze

Wheeze is a continuous adventitious breathing sound generated in the respiratory airway during respiration. Wheezing occurs when a portion of the respiratory tree is constricted or blocked, or when the rate of airflow through the respiratory tree increases. The duration of a wheezing sound is generally no more than 250ms and its pitch has a minimum value of 400Hz. Asthma and COPD are the most prevalent causes of wheezing. When a Wheeze sound is detected, it might be because of a foreign body limiting the airflow, e.g., a tumor [3, 29].

Rhonchi

Rhonchi is a type of continuous adventitious breathing sound that is melodious with a low pitch. Rhonchi noises are hearable during both breathing phases, but they are mainly detected in the expiration phase. They are distinguished by a prominent frequency of around 200Hz or less and a duration ranging from 80 to 100ms. Airway thickness in large sections is among the main causes of rhonchi [1, 3].

Stridor

A stridor noise is a continuous abnormal sound that is classified as a musical inhalation sound. Stridor sounds are made from turbulent airflow that passes through the bronchial tree or larynx [31]. Because of their coming source, stridor noises are generally audible during the inspiration phase, but they can also be detectable in the exhalation phase. In contrast to wheeze sounds, stridor sounds are louder and harsher. The duration of a stridor sound is greater than 250 milliseconds and the pitch is more than 500 Hz. Epiglottitis, croup, and laryngeal edema are illnesses connected to stridor since they all include airway blockage [3].

Squawk

Squawk noises are discontinuous abnormal sounds that are detected during the inhalation phase. They are known to be like wheezing sounds, with short durations of less than 200ms

and a frequency range of 200 to 300Hz [3]. Hypersensitive pneumonia or common pneumonia are examples of lung diseases that may be caused by squawks [32].

Coarse crackle

A coarse crackle sound is a discontinuous abnormal sound that is generated in large bronchi by air bubbles. Coarse sounds are detectable in the earliest stages of inspiration and during the entire expiration phase. Coarse crackle sounds have a short pitch of 350 Hz, but their duration is long, with a value of around 15 milliseconds. Coarse crackling noises can cause lung disorders such as chronic bronchitis, or COPD [3].

Pleural rub

A pleural rub sound, a discontinuous adventitious sound, is a rhythmless and raspy breathing sound. Pleural rub noises have a short pitch, usually less than 350 Hz, and last for 15 milliseconds. These types of lung sounds are detectable during the exhalation and intake phases. Pleural rub sounds are generally caused by inflammation in the pleural tissue and may lead to pleural tumors as well [3, 33].

3.2. Audio Data Features

In general, feature extraction is a technique used for identifying the most dominant and distinctive properties of a given input. In audio signal processing, a feature extraction method has to find a significant feature that contains the characteristics of a signal in a much more compact manner [34]. Different audio signal features are depicted in the next subsections.

3.3. Time domain features

A sound is an audio signal that is composed of time and amplitude axes. We can analyze significant characteristics of a signal by visualizing it in the time domain, and this information could be utilized to analyze and predict related signals. Windowing is a concept of arranging an audio signal with samples in the data flow direction. Several features can be obtained from the audio signal by applying the windowing method at certain time intervals of the signal [34]. Time domain audio characteristics that can be derived from a signal in the time domain are as follows.

Zero crossing rate (ZCR)

The ZCR is the quantity that indicates how many times a signal crosses the zero level in a second. ZCR is known to be effective in voice detecting activities that determine if a signal frame is spoken, unvoiced, or quiet [34].

Amplitude-based features

Amplitude Descriptor (AD) is a feature that considers the energy, and the variation of the duration of signal segments from high and low amplitudes via an adaptative threshold. This feature is mostly utilized when classifying environmental sounds. The ADSR (Attach, Delay, Sustain, Release) envelope is a feature calculated by the relationship of variations in time with amplitude. It is employed in the analysis of music and music genre classification. LAT is a logarithmic (base 10) representation of the time duration; from the time it begins to the time it goes to its constant state. It is utilized to detect musical onsets or recognize environmental sounds. Shimmer is a feature that calculates variations in amplitude from cycle to cycle. it can be utilized in areas such as speaker recognition, voice activity detection, and music sound classification [34].

Energy-based features

Short Time Energy (STE) is the average energy per frame, and it is utilized in voiced-unvoiced segment detection, environmental sound detection, and audio-based surveillance systems. Volume is defined as the root mean square value of a signal frame's magnitude. Applications such as speech discrimination, and speech segmentation employed this feature [34].

Temporal centroid (TC)

This characteristic is time-averaged throughout the energy envelope, it can be utilized in recognizing environmental sounds and in classifying acoustical scenes [34].

Auto-correlation based features

This kind of feature assesses the degree to which the signal and its delayed form are equivalent. A negative association is represented by a value of -1, a neutral association by a value of 0, and a strong positive association by a value of 1. Auto-correlation-based features are used to determine the pitch of a signal and analyze musical beats [34].

Rhythm based

Generally, these features have patterns that recur over time. They are deployed in audio applications such as music discrimination, or music instrument classification [34].

3.3.1. Frequency domain features

The time domain graphical representation depicts the signal fluctuation according to time. When studying a frequency-domain signal, the time-domain signal must be translated into the frequency domain. This signal transformation is accomplished through the use of techniques such as the Fourier Transform (FT). Different frequency domain features can then be extracted. The majority of these types of features are discussed below.

Auto-regression-based features

LPC Coefficients start by eliminating redundancies in a signal, then attempt to predict upcoming values by linearly merging the previous coefficients. LPC coefficients are utilized in audio retrieval and audio segmentation. CELP is built on three techniques. LSF is a feature that is utilized to describe LPC coefficients for the transmission across a channel. The value of this feature varies depending on whether the glottis is open or closed. Audio applications like speaker segmentation, music discrimination, and voiced-unvoiced segment recognition make use of the LSF feature [34].

Peak frequency

Peak frequency may be defined as the frequency at which the most power is present. It helps determine the basic frequency of the signal and gives an approximation of the most

prominent frequency of a given audio signal. When classifying music or speech, or classifying gender, peak frequency is a useful feature [34].

Method of Selection of Amplitudes of Frequency Multi-expanded Filter based features (MSAF MULTIEXPANDED)

It is an acoustical feature that computes variances between the FFT spectrum. This feature is utilized to discover errors in electric drilling or griding motors and also to identify faulty commutator motors [34].

Shortened method of frequencies selection MULTIEXPANDED (SMOFS-MULTIEXPANDED)

It is a hand-crafted feature similar to MSAF-MULTIEXPANDED utilized in industrial applications to examine faults in motors [34].

STFT-based features/Time-frequency features

For example, time-frequency matrix, sub-band energy ratio, spectrum envelope, and SPSF (Stereo Panning Spectrum Feature) are types of features that aim to compute stereo shift information for several frequencies based on the STFT of the left and right channels of a given audio signal. For example, a group delay function (GDF) is a feature that provides information on a signal's temporal events, such as detecting spectral peaks. Speaker identification and speech segmentation are examples of audio applications that utilize GDF [34].

Envelope modulation spectrum (EMS) based

An envelope modulation spectrum depicts a gradual amplitude modulation in the signal as well as the energy distribution in amplitude variations across particular frequencies. When it comes to classifying pathological speech and controlling speech, the EMS features are the suitable features [34].

Long-term average spectrum (LTAS)

The signal's unusual spectral information is captured using LTAS-based characteristics. They are utilized for the classification of pathological speech from controlled normal speech [34].

Chroma related features

Chroma features are powerful music audio resolutions that the whole spectra is mapped into 12 bins that symbolize the 12 semitones (or chroma) of the musical octave. Chroma features are calculated by using the logarithmic STFT of the audio signal. These characteristics are used to discover similarities across interpretations of a given piece of music [34].

Tonality-based features

The fundamental frequency in a stationary harmonized audio stream makes up the tonal tones. Tonality in music refers to the arrangement of the notes of a musical scale. Following are the main tonality-based audio characteristics: Fundamental Frequency (FF), pitch histogram, pitch profile, harmonicity, harmonic-to-noise ratio, and jitter [34].

Spectrum shape-based features

Spectral centroid, spectral center, spectral roll-off, spectral spread, spectral skewness, spectral kurtosis, spectral slope, spectral decrease, bandwidth, spectral flatness, spectral crest factor, entropy, spectral flux, Octave based spectral contrast (OBSC) [34].

3.3.2. Cepstral domain features

A Cepstrum of a given signal is computed by applying the inverse Fourier transform of the logarithm of the spectrum of that signal [34]. The cepstral audio characteristics are explained in the coming following subsections.

Linear prediction cepstral coefficients (LPCCs)

They are the cepstral domain transformed coefficients of the linear prediction coefficients (LPC). They are used in several audio applications such as speech analysis, speech recognition, noise removal, etc [34].

Perceptual linear prediction (PLP) cepstral coefficients

These coefficients can be extracted from the linear prediction coefficients (LPC), and they are built from the concepts of critical band spectral resolution, equal-loudness curve, and intensity loudness power law [34].

RelAtive-SpecTrAl PLP (RASTA-PLP) feature

It is a hybrid feature that attempts to utilize the human hearing system's ability for cancelling noise by applying a method called RASTA. RASTA is a method that employs a band-pass purifier to the power in each frequency sub-band with the aiming of smoothing out short-term noise changes and eliminating any stable offset produced by constant spectral coloring in the audio signal. This mixed property is often employed in voice recognition, gender categorization, speaker verification, and other applications [34].

Greenwood function cepstral coefficients (GFCCs)

They are a generalized version of MFCCs. They are utilized in audio applications working on animals and environmental sound classification [34].

Gammatone cepstral coefficients (GTCCs)

This feature uses gammatone filter banks that output a frequency-time resolution of the audio clip. They are deployed in speech and environmental sound recognition [34].

Mel frequency cepstral coefficients (MFCCs)

MFCC features are generated based on the cepstral representation of an audio signal. MFCCs are used to represent the short-time power spectrum of audio clips. They are according to

the discrete cosine transform of the log power spectrum on a non-linear Mel scale. The frequency bands of MFCCs are evenly spaced on the Mel scale, which closely replicates the human hearing system [34]. MFCCs are key components in many audio signal processing applications.

Algorithm 1: Algorithm to Extract MFCC [34]
Input: Audio signal (for our case, a lung sound)
Output: Mel Frequency Cepstrum Coefficients MFCCs

1. Divide the audio signal into short window frames
2. For each frame do
   a. Compute the periodogram estimate of the power spectrum;
   b. Apply the Mel filter bank to the power spectrum and sum up the power of the filters;
   c. Take the logarithm of all filter-bank energies;
   d. Apply the DCT of the log filter-bank energies;
3. endFor
4. Take the desired number of DCT coefficients called MFCCs (for our case 32 coefficients)

The algorithm above can be used to produce MFCCs from a given audio file. An example of an MFCC extracted from a lung sound is displayed in Figure 3.2. below.

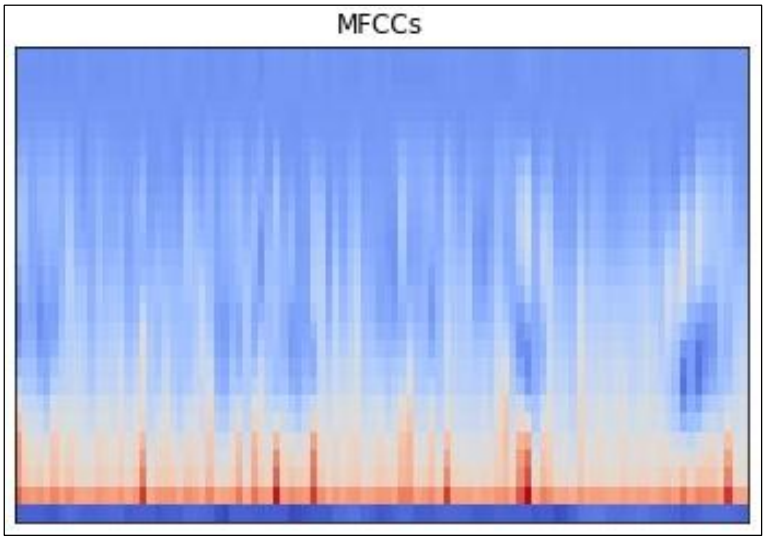


Figure 3.2. An MFCCs representation of a lung sound

3.3.3. Discrete wavelet features

The wavelet transform is a method for transforming a time-domain waveform into a frequency representation. It calculates the inner product of the audio by using one member of the wavelet family, which are the CWT and the DWT [34].

3.3.4. Image-based features

Image-based features are obtained after analyzing a spectrogram converted into an image. Local Binary Pattern (LBP), Local Ternary Patterns (LTPs), Histogram of gradients (HOG) features, and Scale Invariant Feature Transform (SIFT) are examples of image-based that can be derived from a spectrogram image [34].

3.3.5. Deep features

Features that are extracted from an audio signal by using any deep neural network are called deep features [34]. Any one of the mentioned audio features can be fed to a deep neural network as inputs and then the model will be able to provide deep features. Techniques that are used to extract deep features are as follows:

Convolutional Neural Network	(CNN)
Deep Neural Networks	(DNN)
Recurrent Neural Networks	(RNN)
Deep stacked auto-encoder	(SAE)
Unidirectional long short-term memory	(LSTM)
Bidirectional long short-term memory	(BLSTM)

3.3.6. Sparsity features

Mathematically, a sparse matrix has a large percentage of zeros and a small percentage of non-zero values. In audio signal processing, this idea may result in relatively few non-zero values when analyzing it in a certain domain. Sparsity may be noticed in any of the above-mentioned domains. When sparsity is established in one domain, domain-specific characteristics are retrievable and can be examined. For example, if sparsity is in the cepstral domain, cepstral characteristics could be recovered from the sparse signal [34]. There are several techniques to obtain sparsity.

Matching Pursuit (MP)

Orthogonal matching pursuit

Stage-wise greedy method

Coordinate descent

4. MATERIALS AND METHOD

In this chapter, we provided more information about our dataset and different concepts that can help us achieve our goal.

4.1. Dataset

The dataset utilized in this research is a newer version of the dataset used by Güler et al. in [6]. During their research in 2020, lung sounds were collected from 60 patients at a public hospital in Turkey. Each audio file was recorded from specific regions of the chest of a patient by using an electronic stethoscope for 15 seconds at a frequency of 16KHz. These same data preparation procedures have been applied to the new 117 additional lung audio files. So, the final dataset in our possession is composed of 177 audio files and categorized into 3 classes (Normal 59, PathologicalRhal 59, and PathologicalRhonchus 59).

4.2. Machine Learning

Machine learning is when a computer program uses computational methods to perform a task effectively without being explicitly instructed on how to perform it. Implying that, the program will learn from experience, available past information, and how to achieve a specific goal [35]. An example of a learning problem: Based on previous meteorological data, forecast whether it will be sunny, cloudy, or rainy on a specific future coming day [36].

There are two main types of learning scenarios: supervised learning and unsupervised learning. The difference between them comes from the type of provided data to train the learner program. When a set of labeled examples is used as training data for the program to make it able to predict unseen data, this is called supervised learning. Spam detection is a common problem of this kind. Here, the learning problem is to automatically classify email messages as either spam or not spam. While in unsupervised learning, unlabelled examples are provided to train the learner program and based on this data, the program will make predictions for new unseen data. In this case, the program will need to find similarities, characteristics, and differences on its own from the provided dataset to be able to divide the dataset into two or three different groups based on the patterns discovered. An example of an unsupervised learning problem is clustering [35].

4.2.1. Classification

Classification is a supervised machine learning task that aims to train a model that is capable of identifying which category a new data sample belongs to. Because classification is a supervised learning type, classification algorithms learn from available past information provided for the training. In classification, the output will be a class or a category, such as sunny or rainy, 0 or 1. The spam or no-spam problem is an example of a classification problem. A classifier is an algorithm that classifies a set of labelled examples [37].

Once a classifier is completed, evaluation metrics are used to evaluate how the model performs at its task. Common metrics used to evaluate a classifier include Accuracy, Precision, and Recall [37]. More details about the different metrics used to assess classifiers are discussed with details in subsection 4.8.

4.2.2. Artificial neural network

Artificial Neural Networks (ANN) are a branch of machine learning methods. Their concepts are driven by the neuronal architecture of the nervous system. The brain usually learns from available information and experience. The main elements of the human brain are known as neurons. This cell, the neuron, is the one that provides humans with the ability to remember, think, and act based on previous experience. There are billions of these kinds of cells in the brain, and they are all connected to each other. One neuron can be connected to 200,000 other neurons. This huge number of neurons and the multiple connections between them give birth to the power of the human mind. Artificial Neural Networks work as if trying to duplicate this powerful organism; with new ways to solve problems. Since the last decade, computing methods based on this biological inspiration have become the preferred choice for many machine learning industries [37, 38].

In the human brain, a neuron is the basic element of a neural network. Generally, a neuron comes up with four essential components: dendrites, soma, axon, and synapses. Dendrites accept inputs coming from various sources (might be from other neurons), then the inputs are processed in the soma section. To convert inputs into outputs, the axon performs a nonlinear transformation [38]. Figure 4.1. below shows in more detail the operation of this process.

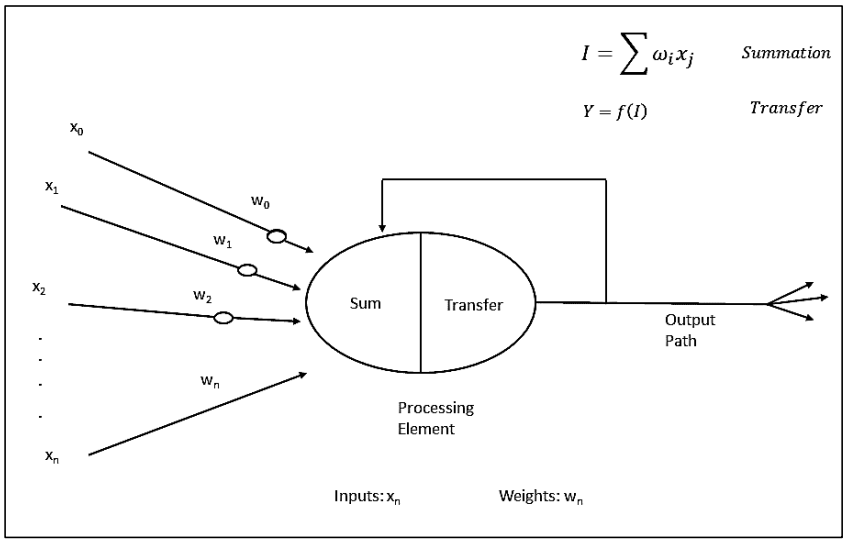


Figure 4.1. A basic artificial neuron [38]

4.2.3. Multilayer perception

Multilayer Perception (MLP) is a feedforward artificial neural network that consists of three types of layers: input, hidden, and output layer as shown in Figure 4.2. below.

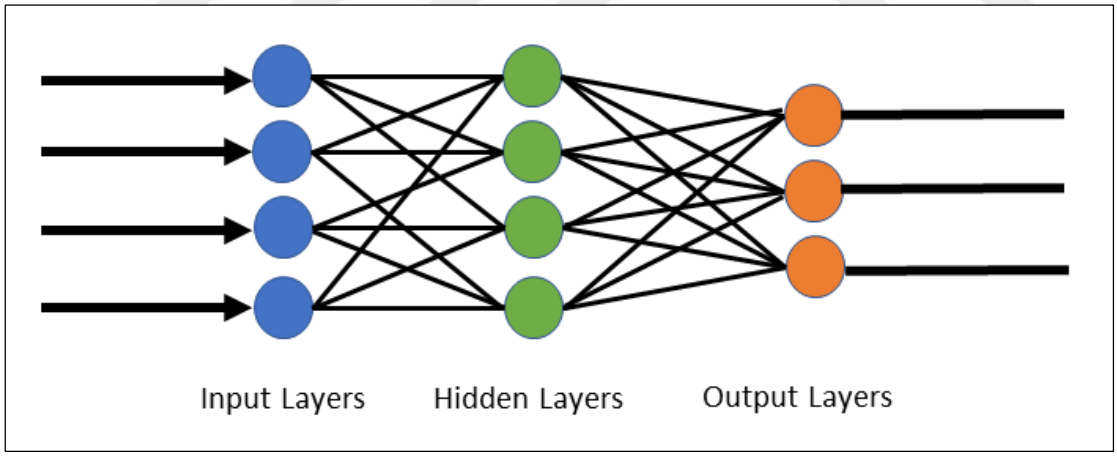


Figure 4.2. Multilayer perception

Input signals are received by the input layer. Hidden layers, which are the main computational engine of an MLP, compute the input signals and the output layer does the required task (classification or prediction). The neurons in the MLP are trained following the backpropagation algorithm, meaning that the data flows in the forward direction, from the input layers to the output layers. MLP is an example of a neural network [37].

4.2.4. Activation functions

An activation function of a node is the decision-making element defining the output of that node according to a given input. Activation functions are utilized to represent non-linear interactions between input and output. Because without these activation functions, the output is a linear function which is a single-grade polynomial that lets the neurons behave as a linear regression model [37].

There are many non-linear activation functions; among them are Tanh and ReLU activation functions which are commonly used for general purposes. Then Sigmoid and SoftMax activation functions are generally used at the classification stage, the output phase, of a neural network model. When dealing with binary classification tasks, for example, cat and dog problems, the sigmoid activation function is suitable. Whereas in multi-class categorization cases, such as our problem in this work, three classes which are normal, rhal, and rhonchus; SoftMax is the suitable activation function to be used in this situation [37]. The mathematical functions and the graphical representations of the activation functions are provided below.

Sigmoid function

The sigmoid activation function ranges from 0 to 1. It is used for predicting a probability as a model output [38]. The equation eq. 3.1 below is utilized to compute the Sigmoid.

$$\sigma = \frac{1}{1 + e^{-x}} \tag{4.1}$$

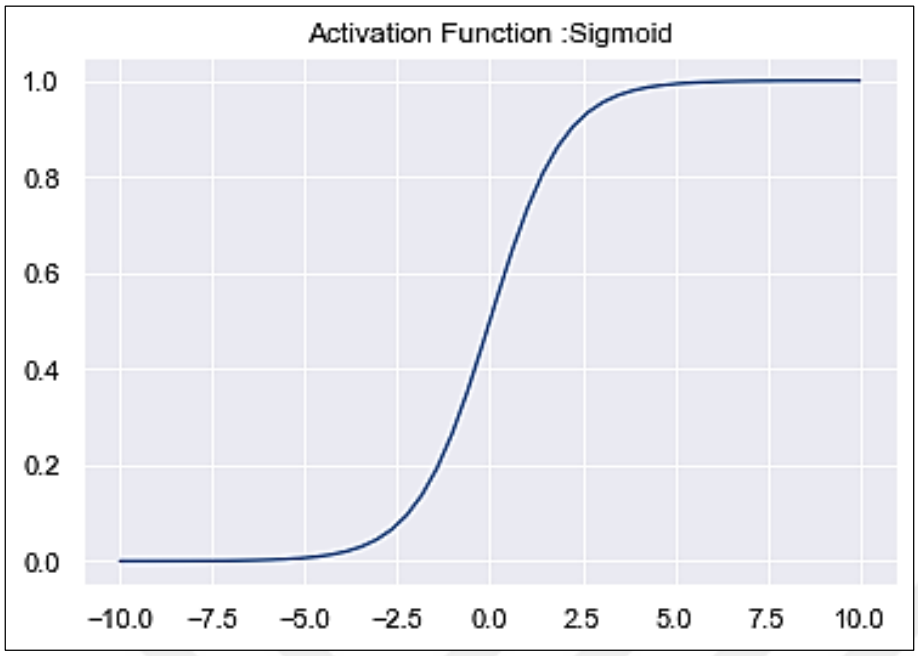


Figure 4.3. Graphical representation of the Sigmoid activation function

ReLU

Rectified Linear Unit is a linear function that outputs its entry itself when it is greater than zero and a zero if the input is negative [39]. Eq. 4.2 below is the ReLU equation.

$$f(x) = \begin{cases} 0, & \text{if } x \leq 0 \\ x, & \text{if } x > 0 \end{cases} \tag{4.2}$$

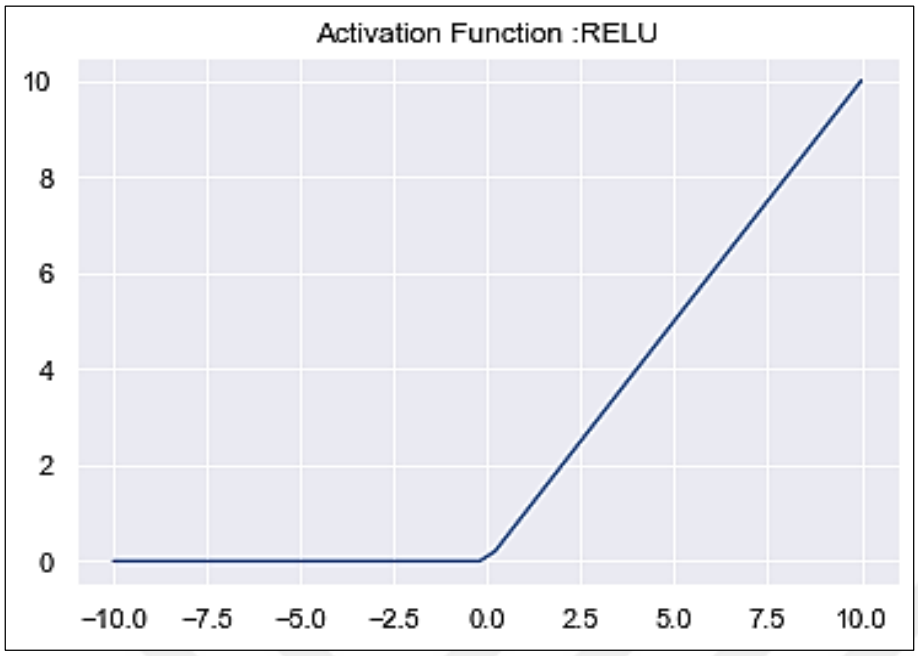


Figure 4.4. Graphical representation of the ReLU activation function

Tanh

The tangent hyperbolic function, or simply tanh, behaves quite similarly to the sigmoid function. The only difference is that sigmoid outputs values that are in the range of 0 to 1, whereas tanh outputs are around -1 to 1 [37]. Eq. 4.3 below is used to calculate the Tanh function.

$$\tanh(x) = \frac{e^{x} + e^{-x}}{e^{x} - e^{-x}} \tag{4.3}$$

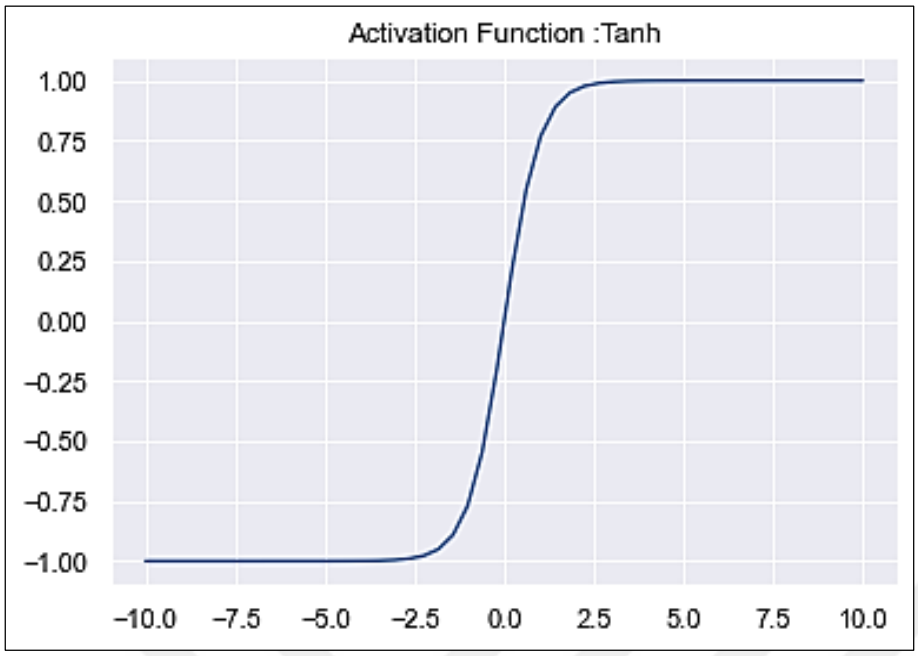


Figure 4.5. Graphical representation of the Tanh activation function

SoftMax

The SoftMax function is utilized for scaling model output into probabilities and is utilized as the last activation on the output layer of a neural network. It is mostly used for multiclass classification problems [37]. The eq. 4.4 below computes the SoftMax function.

$$s(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{4.4}$$

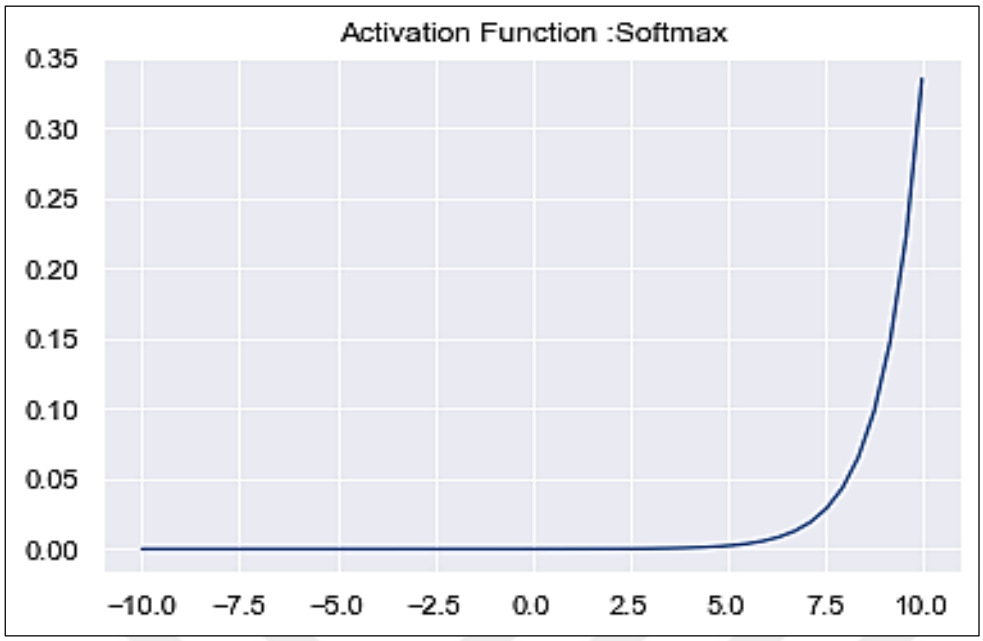


Figure 4.6. Graphical representation of the SoftMax activation function

4.2.5. Training neural network

The numerical optimization of an objective function is used to train neural networks. The objective function is the metric employed to assess how closely the model's outputs match the proper values. There are two sorts of objective functions: loss functions and reward functions. In the case of loss functions, the lower the loss function, the more accurate the model is. On the other side, reward functions are the inverse of loss functions. The higher the level of accuracy, the higher the reward function. Typically, reward functions are used in reinforcement learning to optimize a given result. Loss functions are commonly encountered while dealing with supervised learning problems [37].

4.3. Convolutional Neural Networks

A convolutional neural network (CNN) is a deep learning subclass that contains at least one convolutional layer. CNN uses backpropagation to automatically and adaptively learn the spatial hierarchy of features. CNN architectures are widely employed in computer vision, an artificial intelligence discipline that investigates how software applications, such as image categorization or self-driving vehicles, may acquire a high-level understanding of digital pictures and videos. They are excellent at inferring information from data with spatial structure and correlation. As a result, images are the most appropriate input for this kind of

deep learning model. The image is transformed using a kernel (filter) without losing the underlining structure. In practice, convolution is the process of sliding the kernel over each pixel in the picture and determining its new value. Convolution may be expressed mathematically by the equation eq. 4.5, where *I* is the original image, *K* is the filter matrix, and *S* is the result of the transformed image [40].

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_m I(m,n)K(i - m, j - n) \quad (4.5)$$

A CNN architecture has two major parts: feature extraction and classification. In the feature extraction phase, a convolution operation is performed on a given input to separate and identify the different features of the input data. Several layers such as pooling layers can participate in this process. In the classification part, a fully connected layer uses the results of the convolution process to predict the class of the given input [40]. Figure 4.7. below describes a CNN architecture.

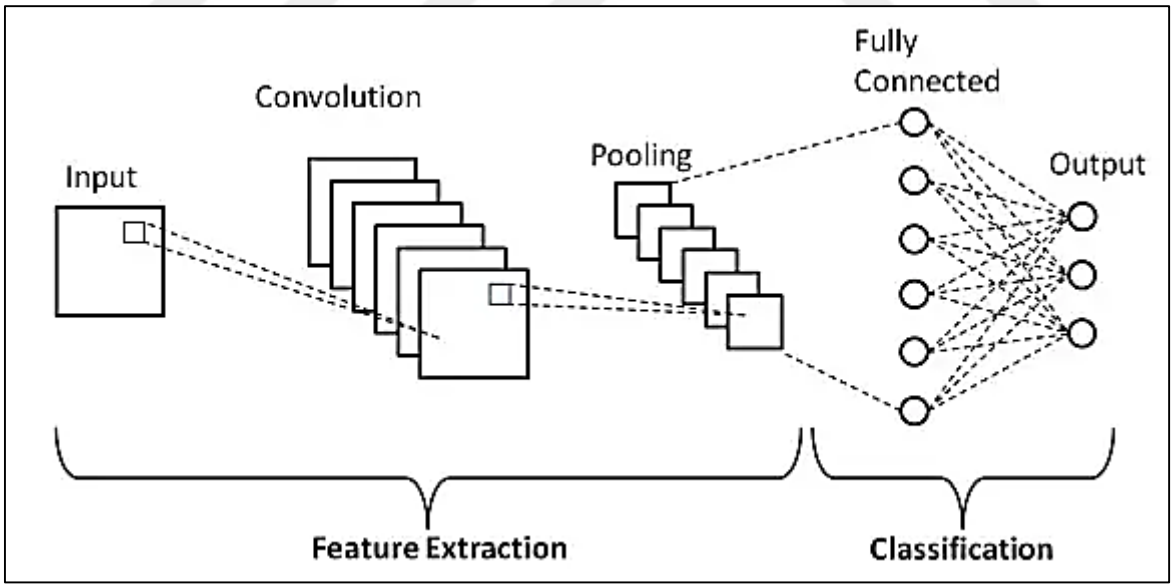


Figure 4.7. A basic CNN architecture showing its principal phases [41]

When convolutional, pooling, and fully-connected layers are stacked together, they form a CNN architecture. Additional layers such as dropout and different parameters such as activation functions can be added to the architecture because they have a remarkable effect on the overall result [41].

Convolutional layer: It is a significant, even the most significant layer in the CNN architecture as it is the first layer utilized to extract the different characteristics of a given input data. It applies the convolution operation (dot), which is provided in equation (3.5). The output of this operation is called a feature map which contains the most relevant features of the provided input data [41].

Pooling layer: Generally, after a convolutional layer comes a pooling layer whose main task is reducing the size of the convolved feature map to minimize the computational complexity. The pooling layer act as a bridge. The feature map (the output of the convolution operation) passes through a pooling layer to reach the fully connected layer. Max pooling and average pooling are different methods that perform pooling operations [41].

Fully connected layer: This layer is the last layer of a CNN architecture. It receives as input a flattened vector from the previous CNN layers. Then by using fully connected layers' weights and biases, mathematical operations are performed so the layer will be able to do the classification task. It is noticeable that this layer is like a neural network; meaning that it consists of input, hidden, and output layers [41].

Dropout: The dropout is utilized to overcome the overfitting issue that comes if all the neurons are connected to each other at the fully connected layer. The dropout concept is randomly dropping and nullifying a portion of the neurons. This dropout process is done only in training while during testing all neurons are present. The dropout rate has to be specified which is generally between 25% and 30% [41].

Activation function: An activation function is a key parameter for a CNN model. To put it simply the main aim of an activation function is to decide if a specific neuron should be activated or not. This decision is taken based on a mathematical function. Commonly utilized activation functions include ReLu, sigmoid, and softMax [41].

4.3.1. VGG

VGG, which stands for Visual Geometry Group from Oxford University, is a deep convolutional neural network proposed by Simonyan and Zisserman [43]. In the ILSVRC-2014 competition, the VGG model reached 92.7% classification accuracy. With the VGG

model, the depth of layers is investigated using a very small convolutional filter (3 * 3) stacked together to consider very large images. The authors have published a family of VGG models that range from 11 to 19 layers deep[43]. In this thesis, the 16th version is used for classifying lung sounds. The architecture of the VGG16 model can be visualized in the graphical representation shown in Figure 4.8., which illustrates the 16 layers of the network, including 13 convolutional layers and 3 fully connected layers.

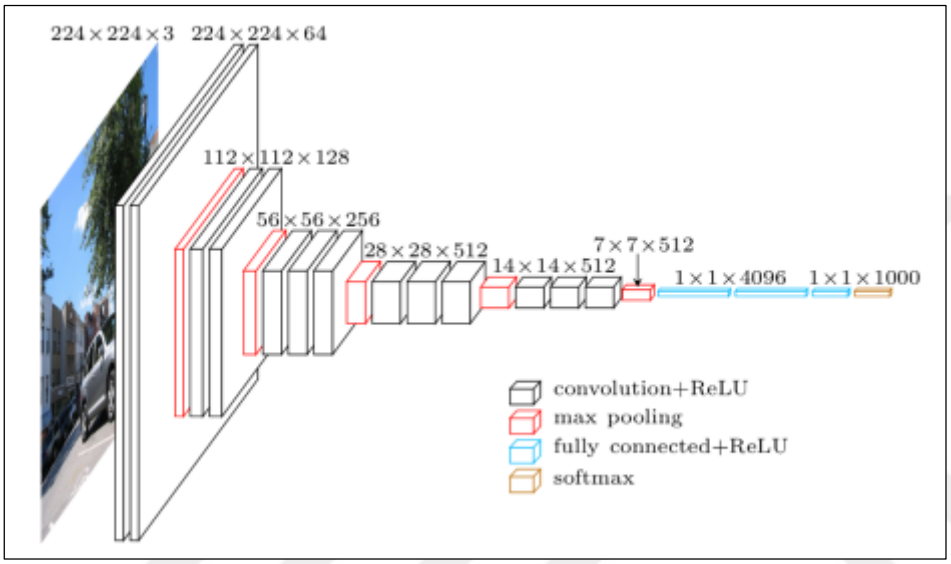


Figure 4. 8. The architecture of the VGG16 model [61]

4.3.2. MobileNet

A MobileNet is a lightweight deep neural network based on a simplified architecture that utilizes depth separable convolutions. MobileNet algorithm transforms a standard convolution into 2 convolutions that are depthwise and pointwise. The depthwise convolution uses a single kernel for each input channel and then the pointwise applies a 1x1 convolution in order to fuse the outputs of the depthwise convolution together. This separation leads to the formation of a lighter network by reducing the computation and the size of the MobileNet model. With depthwise and pointwise convolutions counted as distinct layers, MobileNet consists of 28 layers. MobileNet was introduced in 2017 by Howard et al. to work on mobile and embedded vision platforms [43]. The MobileNet architecture's detailed components are presented in Figure 4.9., which includes convolutional depthwise separable convolutions (DS), average pooling, fully connected (FC), and softmax layers. In addition, a more comprehensive breakdown of the DS layer comprises depthwise

convolution, batch normalization (BN), and rectified linear unit (ReLU) modules, respectively. Figure 4.9. below gives an illustration of the overall MobileNet architecture (A) and an explanation of the depthwise separable convolutions layer (B)

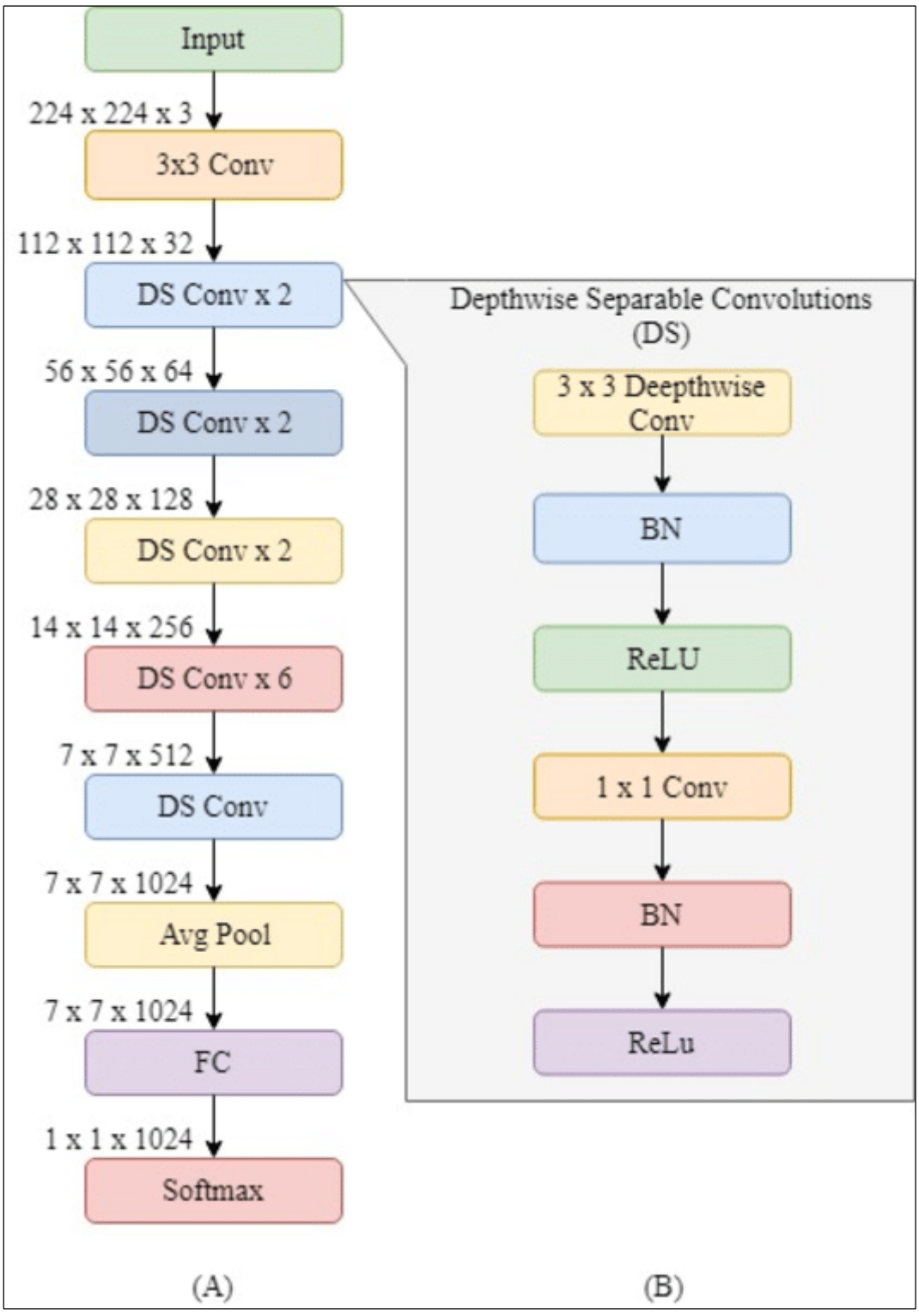


Figure 4.9. The architecture of the MobileNet [62].

4.3.3. Xception

Xception means Extreme Inception. It is a depth-separable convolutional neural network architecture based on the Inception algorithm. In the Xception model, depthwise separable convolutions are used instead of inception modules. The Xception architecture includes 36 convolutional layers which serve as the network's feature extraction foundation, and except for the first and last convolutions, the 36 layers are structured into 14 modules connected by linear residual connections. A linear stack of depth-separable convolutional layers with residual connections forms the Xception architecture [44]. Figure 4.10. gives a complete description of the specifications of the Xception network. The architecture functions in a sequence of steps, starting with the data passing through the entry flow, followed by the middle flow, which is repeated eight times, and ultimately concluding with the exit flow [44].

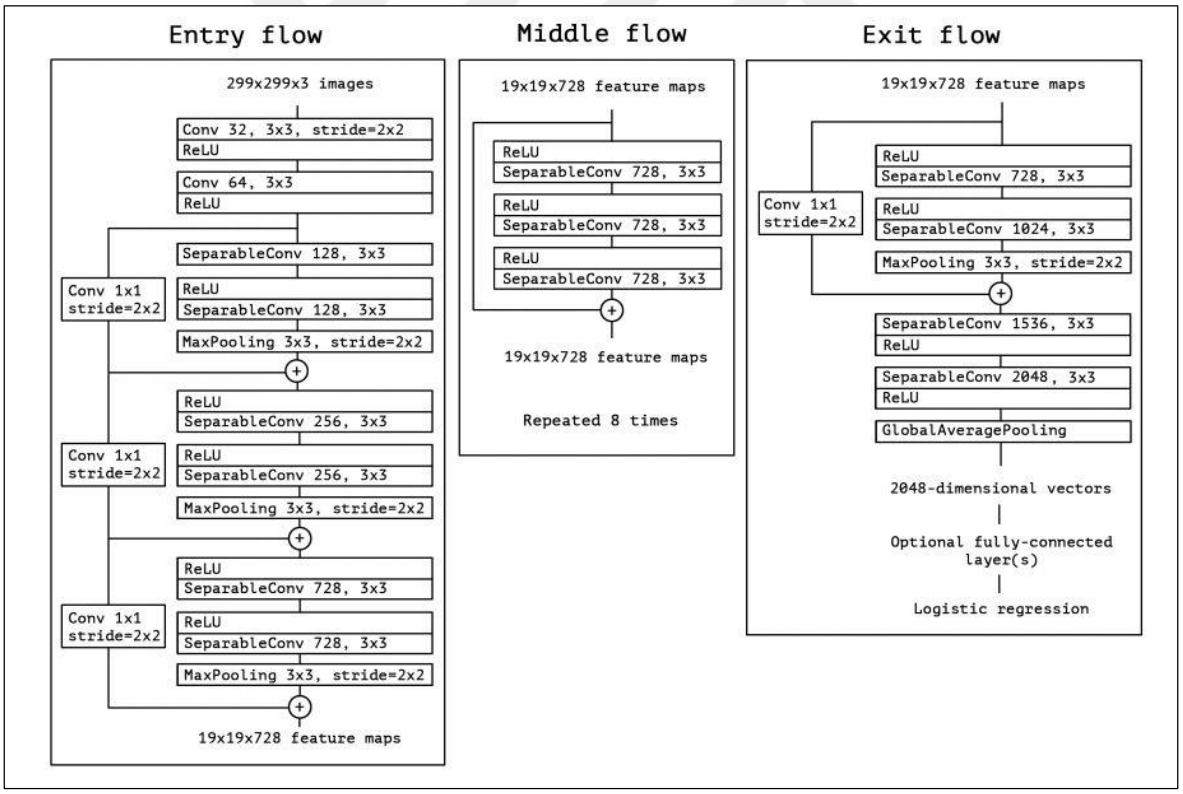


Figure 4.10. Overall Architecture of Xception.

4.3.4. ResNet

In the study “Deep Residual Learning for Image Recognition”, conducted by He et al. in 2015, they introduced the ResNet, which is a particular kind of neural network. The trend in previous articles was that the deeper the network the better it performs because the network has more parameters, which means more capacity. But as the network gets more powerful, the rate of errors goes up, meaning that the network fails in making accurate predictions. The idea behind their research was to fix this problem. They added a shortcut connection from one layer to the other in the architecture, so when it gets too complex for the weights to go through the layers, the network follows the shortcut path, and the layers in between are ignored. This innovation allows this deep and residual network to perform better than networks with low parameters. There are various versions of ResNet models, and they differ according to the number of layers (how deep they are). Different variants of ResNet include ResNet18, ResNet34, ResNet50, and Resnet101 [45]. The ResNet version used in this work is the ResNet50v2. The ResNet architecture can be visualized through the graphical representation depicted in Figure 4.11.

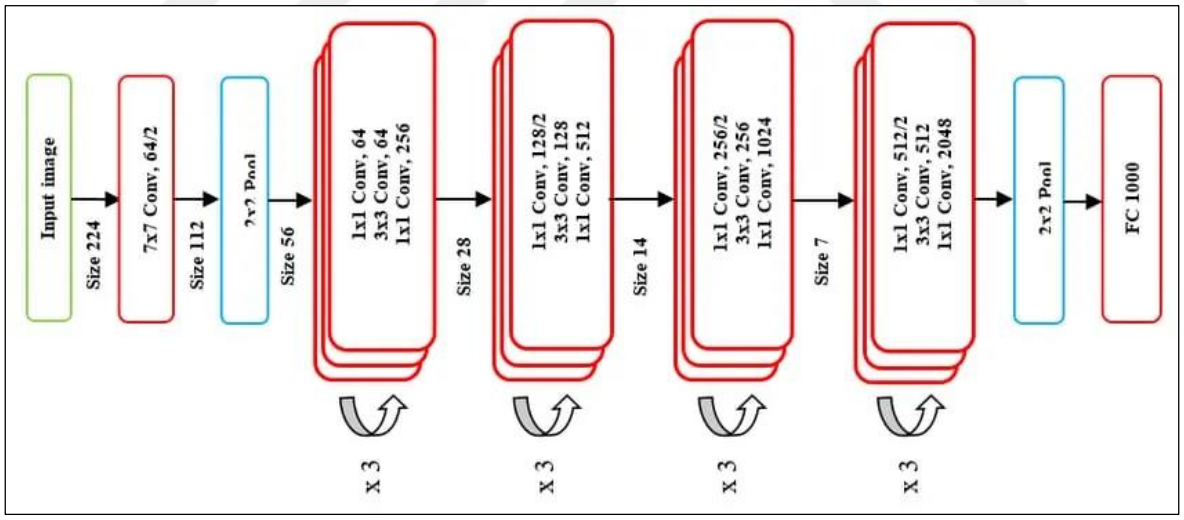


Figure 4.11. The ResNet-50 architecture [63].

4.4. Data Augmentation

When creating a dataset, most of the time you will face some difficulties to acquire a sufficient number of examples to train your algorithms because labeled samples are generally expensive to collect. Whereas the size of the dataset plays a huge role in the good

performance of a deep learning model. Here comes the data augmentation technique. The data augmentation technique, also known as the data deformation technique, is a process used to synthetically generate new labeled samples from the available ones to increase the effective training set. When dealing with images, an example of a data augmentation technique can be done by performing slightly some modifications like rotating, blurring, or resizing a given image [37].

In audio processing applications, time-stretching and pitch-shifting are the common audio data augmentation techniques used to generate new audio samples. When a time-stretching method is applied to a sound, this refers to slowing it down or speeding it up. Whereas the pitch-shifting method is utilized for modifying the pitch of an audio input [46].

One common method of time stretching is the phase vocoder algorithm [64], which can be represented mathematically as follows:

$$X'(t,\omega) = X(t,\omega) * e^{(j\Delta\varphi(t,\omega))} \tag{4.6}$$

where X(t,ω) is the short-time Fourier transform of the audio signal at time t and frequency ω, X'(t,ω) is the magnitude of the transform, and Δφ(t,ω) is the phase of the transform. To stretch the signal, we adjust the time scale of the phase φ(t,ω) while preserving the magnitude X(t,ω). This can be accomplished by dividing the phase by a time-varying function:

$$\Delta\varphi'(t,\omega) = \Delta\varphi(t,\omega) * a(t) \tag{4.7}$$

where a(t) is a time-varying scaling factor that adjusts the rate of change of the phase. The time-stretched signal can then be obtained by taking the inverse short-time Fourier transform of the modified phase and the original magnitude:

$$x'(t) = ISTFT(X'(t,\omega)) \tag{4.8}$$

where x'(t) is the time-stretched signal and ISTFT represents the inverse short-time Fourier transform.

Pitch shifting involves changing the pitch of an audio signal by a certain number of semitones, which can be represented mathematically as follows:

$$f' = f \times 2^{(\frac{n}{12})} \tag{4.9}$$

where *f'* is the new frequency of the audio signal, *f* is the original frequency, and n is the number of semitones to shift by [65]. For example, if we want to shift the pitch of an audio signal up by one semitone, we would set n = 1.

Different approaches could be used to create additional labelled examples. An offline approach, meaning that the new examples are obtained as audio files, before the feature extraction process. While there is also another way, which is the online approach, to generate new examples after the feature extraction phase. This method is mostly used when extracting spectrograms from the audio files, so the new samples will be generated from an extracted spectrogram while being forwarded to the model for the training process [37].

Mixup [47] is another offline data augmentation method; its concept is to create a new sample by mixing two examples from different categories. More details about this approach can be found in [47]. In this thesis, time stretching, and pitch shifting methods are the data augmentation techniques utilized for generating additional training samples.

4.5. Evaluation Metrics

Evaluation metrics, also called performance metrics, are utilized to evaluate the efficiency of a model on a dataset. In the following section, we are explaining some of these performance metrics in detail.

4.5.1. Accuracy

Accuracy is known as the fundamental metric to assess the performance of classification models, it is the proportion of the total number of correct predictions [48]. The equation eq. 4.7 below can be used to compute the accuracy of a model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.7}$$

4.5.2. Confusion matrix

In the domain of machine learning, an algorithm's performance can be visualized via a confusion matrix. Confusion matrix rows represent instances belonging to the predicted class while columns represent instances that belong to the actual class [48]. Table 4.1. below shows what a confusion matrix can look like.

Table 4.1. Confusion matrix

	True 0	True 1
Predicted 0	TN	FN
Predicted 1	FP	TP

Here, True positives (TP) are outcomes predicted by the model correctly. Similar to a true positive, a true negative (TN) occurs when the model accurately predicts the negative outcome. A false positive (FP) outcome results if the model incorrectly predicts the positive class, while a false negative (FN) occurs when the model incorrectly predicts a negative outcome.

4.5.3. Precision

Precision can be deduced from the confusion matrix; the precision metric is the ratio of positive data samples classified correctly to the total number of samples classified as positive [48]. The precision is used to assess the model's accuracy in recognizing positive training samples. Precision can be calculated also by using the equation eq. 4.7 below.

$$Precision = \frac{TP}{TP + FP} \tag{4.7}$$

4.5.4. Recall

The recall (or sensitivity) is measured as the ratio between the number of positive data samples correctly classified as positive to the total number of positive examples [48]. It can be computed through the equation eq. 4.8 below.

$$Recall = \frac{TP}{TP + FN} \tag{4.8}$$

4.5.5. Specificity

It is the percentage of true negatives that the model accurately predicted. The equation eq. 4.9 can be used to compute its value.

$$Specificity = \frac{TN}{TN + FP} \tag{4.9}$$

4.5.6. F1-Score

It may be regarded as a harmonic mean of accuracy and recall, the closer the value is to 1, the better it is [48]. Precision and recall contribute equally to the F1 score which can be computed by utilizing the equation eq. 4.10 below.

$$F1 - Score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{4.10}$$

5. EXPERIMENTAL STUDIES AND RESULTS

In this section, we started by explaining the proposed method, then following by experimentation of the proposed method, and finally, the outcomes of this experimentation were presented, discussed, and compared.

5.1. Experimental Environment and Tools

Some of the experiments of this research are conducted on a Windows 11 system with Intel(R) Core (TM) i7-1065G7 CPU @ 1.30GHz, 1501 MHz, 4 Core(s), 8 Logical Processor(s), 1 TB HDD, 8 GB RAM. The rest of the experiments were completed on the Google Colab platform. We preferred to run some of our experimentations on Google Colab because we require special computing and hardware resources such as GPU (Graphical Processing Unit) and TPU (Tensor Processing Unit) during model training and testing phases. Especially appealing is the availability of Google Colab's dedicated, free, and easy-to-use development platform and its powerful GPUs and TPUs. By utilizing Google Colab, we could run most of our experiments in a very short period of time.

Keras

Keras is a deep learning API written in Python that works on top of the TensorFlow machine learning framework. It was designed with the intention of enabling fast experimentation. It is critical to be able to move quickly from idea to outcome when conducting research. By lessening the cognitive strain on developers. Keras enables you to concentrate on the crucial aspects of the issue. It adheres to the concept of complexity is gradually revealed and provides industry-strength performance and scalability [49].

Scikit-learn (Sklearn)

Sklearn is Python's most usable and reliable machine learning package. It offers a collection of effective tools for statistical modeling and machine learning, including classification, regression, clustering, and dimensionality reduction. In this thesis, we used the sklearn library when splitting the dataset into training, testing, and validation sets, in addition also, when evaluating the performance of the models [50].

Numpy

NumPy is a Python library that contains multidimensional array objects and a set of array processing functions. NumPy can execute Fourier transformations and routines for shape manipulation, as well as other operations related to linear algebra. It is an abbreviation for Numerical Python [51].

Librosa

Librosa is a python library for music and audio analysis. It offers the fundamental elements required to develop music information retrieval systems [52].

5.2. The Proposed Methodology

In this subsection, we are explaining the workflow of our proposed diagnostic methodology. Figure 5.1. illustrates a layout of the different stages to address the lung sound classification problem that we aimed to tackle in this thesis.

Starting with the data collection part, followed by the data augmentation phase, then the feature extraction stage, and finalizing with the classification part which will determine the real class that a given lung audio class belongs to. More details on each phase of our methodology are detailed bein the following subsections.

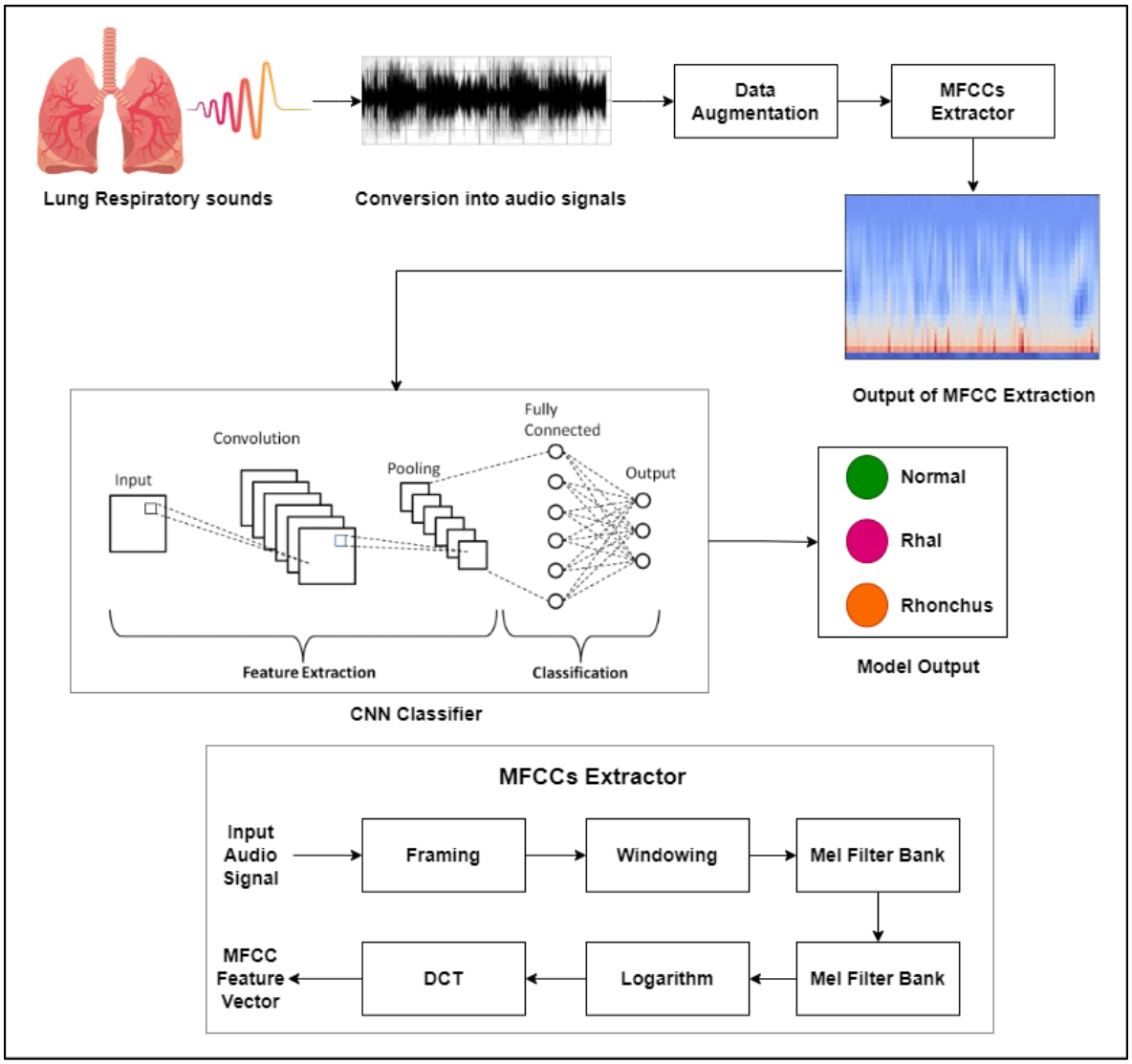


Figure 5.1. Our proposed diagnostic methodology

5.2.1. Data acquisition phase

First, in the data acquisition phase, the lung sounds are recorded using an electronic stethoscope from variant patients at a public hospital in Turkey under the supervision of the Department of Physiology Faculty of Medicine at Hacettepe University. Thanks to them, we were able to have this real dataset, to conduct our research. Specific configurations and settings were followed when collecting the audio signals and following the ethical directives. The result of this phase is the obtention of digital audio signals that were saved in wave (.wav) file format, and a total of 177 lung audio signals were registered and given to us.

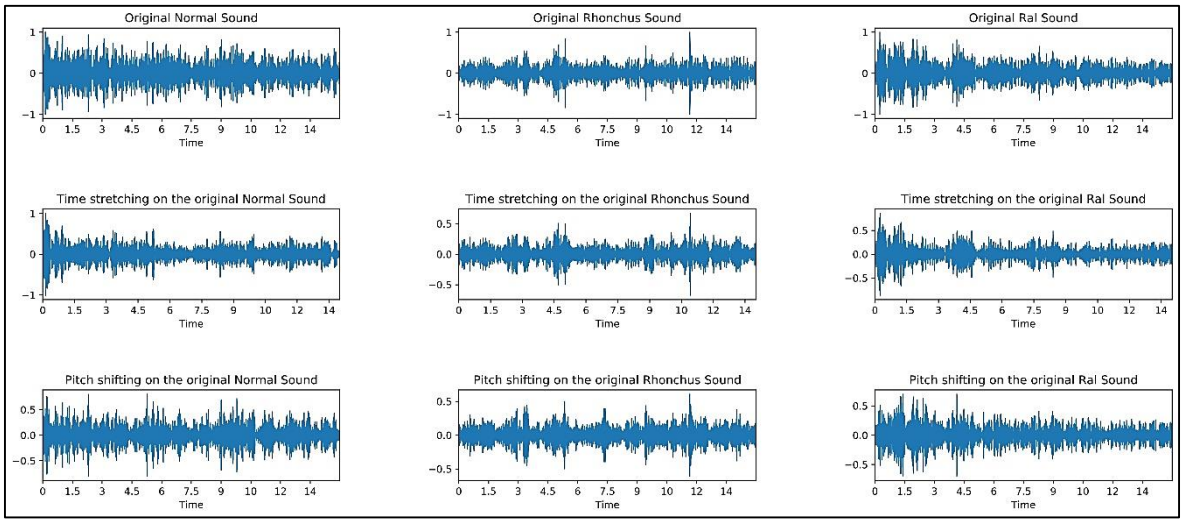


Figure 5.2. Applying data augmentation on some samples.

5.2.2. Data augmentation phase

Secondly comes the data augmentation phase, because we are using deep learning techniques as classifiers in this thesis, we had to pay attention to the training amount of data that we intend to input into the models. The reason is that deep learning models are high training data consumers and require a lot of training examples to achieve great results. Noticing the small size of the lung sound dataset in our possession, we deployed data deformation methods to create additional training samples. Two audio data augmentation methods were used for this phase: time stretching, and pitch shifting. When a time stretching method is applied to a sound, this refers to slowing it down or speeding it up. Whereas pitch shifting is modifying the pitch of an audio input [46]. This data augmentation phase was achieved by using a python library known as Librosa. Figure 5.2. shows some examples of data augmentation techniques applied to some samples from the original dataset. A graphical representation of the MFCCs of the newly generated examples is provided in Figure 5.3. to visualize the effects of the audio data deformation methods on some random samples.

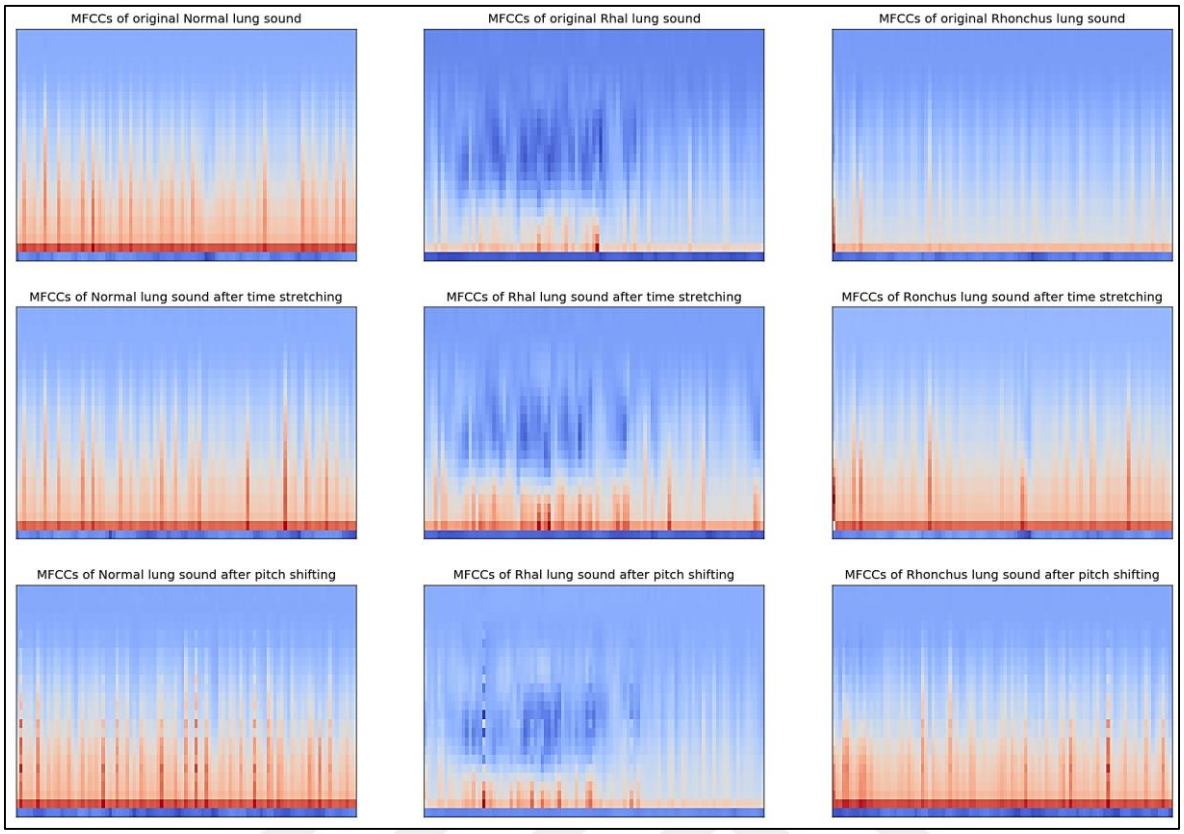


Figure 5.3. MFCCs when applying data augmentation techniques.

5.2.3. Feature extraction phase

Phase three, the well-known MFCCs are the features extracted and utilized to train the models. A python library called Librosa was used to extract the MFCCs. When extracting these features, the number of coefficients must be specified. By default, the Librosa function used for extracting the MFCCs returns 20 coefficients. In our case, 32 coefficients were extracted from each audio file to constitute the inputs of the models. MFCC features are one of the most widely utilized features in audio applications, including speech recognition [53, 54] and environmental sound classification [55, 56]. Tiwari proved in his research that 32 coefficients for the MFCCs can be efficient [57]. Before extracting the features of a lung sound, each audio file was split into 5 segments and the MFCCs of each segment were extracted. The extracted MFCCs in each segment and their corresponding ground-truth labels are stored in a JSON file which forms the final dataset that will be used for training the models. Some of the parameters used in Librosa's MFCC feature extraction function include hope_length (sliding window for FFT) and n_fft (considered interval to apply FFT). The value of hope_length was set to 512 while the n_fft was given a value of 2048. As

result, each MFCC segment has a shape of (130, 32). MFCC features are then derived from the audio lung sounds and then used as inputs for the deep learning models to accomplish the classification task and identify whether the patient's lungs are healthy (Normal) or non-healthy (PathologicalRhal or PathologicalRhonchus). During the experiment, the dataset was split randomly into 80% for the training set and 20% for the testing set, then 20% of the training set was used as a validation set. Then, in the evaluation phase, different evaluation metrics including accuracy and F1-score were used to evaluate the performance of the models on the classification problem.

5.2.4. Classification phase

In the fourth phase, we utilized popular deep learning architectures to learn from the input (MFCCs) that is provided. These well-known CNN architectures include VGG16, MobileNet, ResNet, and Xception. We opted for this option to take advantage of the knowledge that these architectures already have because they were trained for a long time in huge and enough amounts of data. This method is known as transfer learning, instead of retraining the model from scratch, we can easily use it with the knowledge it has acquired during its previous training [58]. The second reason for opting for this transfer learning option is also due to the amount of the dataset in our possession. After training the pre-trained deep learning models separately from the MFCCs, then a fully connected layer is used along with a SoftMax activation function to complete the classification task. We used SoftMax because it is the suitable activation function for our case, meaning a multiclass classification problem. Then to evaluate the efficiency of our models, we deployed machine learning evaluation metrics to do so.

During the experiment, the dataset was split randomly into 80% for the training set and 20% for the testing set, then 20% of the training set was used as a validation set. Then, in the evaluation phase, different evaluation metrics including accuracy and F1-score were used to evaluate the performance of the models on the classification problem. In the next sections, the results of each model at different evaluation metrics, when using data augmentation methods or not, and graphs of the models to show how the models were performing during the training and also validation phases are presented.

5.3. Results Without Using Data Augmentation Techniques

In this section, we provided the results achieved when not using data augmentation methods. We extracted the MFCCs features from the lung sounds and forwarded them directly to the models for the classification phase. The results are illustrated in Table 5.1 below.

Table 5.1. Results before the utilization of data augmentation techniques

Models	Accuracy	Precision	Recall	F1-Score
VGG	0.77	0.78	0.77	0.77
MobileNet	0.80	**0.81**	0.80	0.80
Xception	0.77	0.77	0.77	0.77
ResNet	**0.81**	**0.81**	**0.81**	**0.81**

Table 5.1 shows the performance results of the models without using data augmentation techniques on the lung sound dataset. Considering how small the number of training examples was, we are able to draw that the models achieved satisfying results. Under these circumstances, the ResNet model has registered a classification accuracy value of 81%, surpassing the other methods even though, the MobileNet achieved close results. VGG and Xception models have both reached 77% value in all the different metrics. In terms of precision, MobileNet and ResNet are sharing the same result of 81%.

The accuracy and error curves of the models during training and validation phases under these circumstances (without applying data augmentation) are shown in Figures 5.4., 5.5., 5.6., and 5.7. Deep learning models require a large amount of training data to learn the underlying patterns in the data. If the training dataset is small, as it is in our case, the model may overfit the training data and fail to generalize to new data. Here we facing the same problem and its results can be seen from the four figures. To prevent overfitting in deep learning models, it is important to apply appropriate regularization techniques, use an appropriate validation set, preprocess the data properly, and train the model on a sufficient amount of data. We utilized some of the previously mentioned solutions and data augmentation techniques were one of the effective ones.

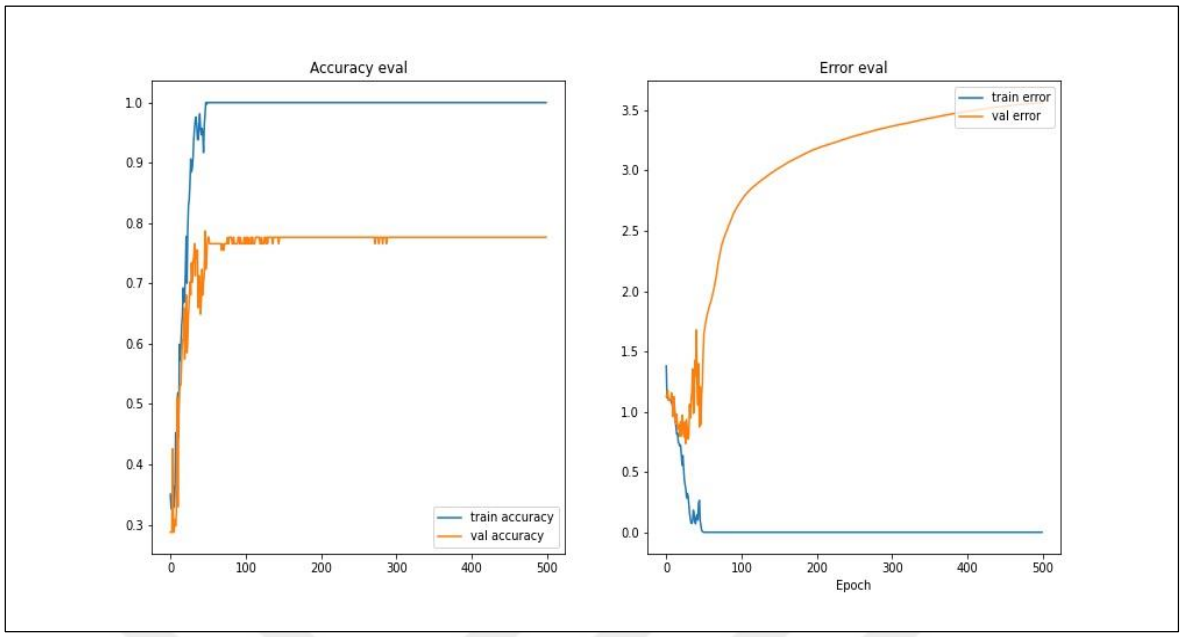


Figure 5.4. VGG curves before data augmentation.

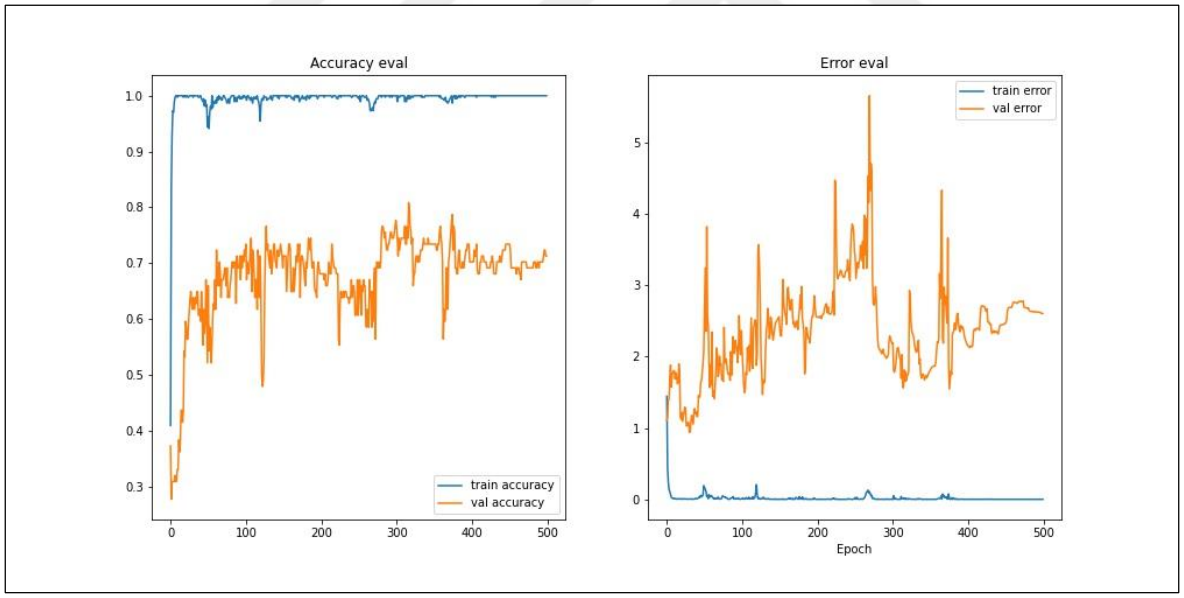


Figure 5.5. MobileNet curves before data augmentation.

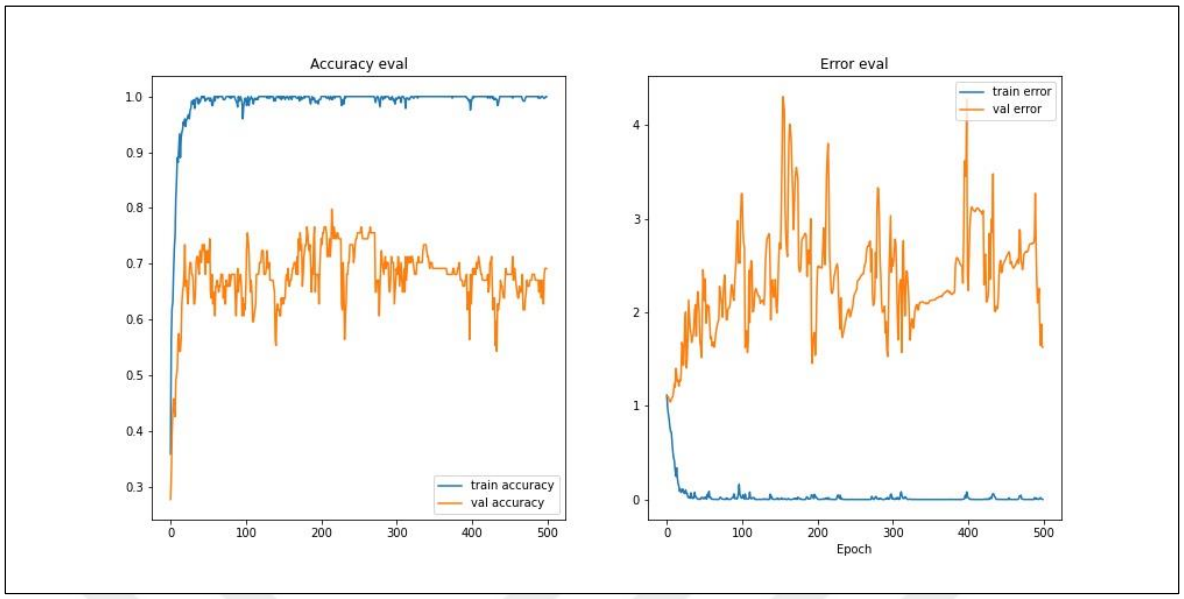


Figure 5.6. Xception curves before data augmentation.

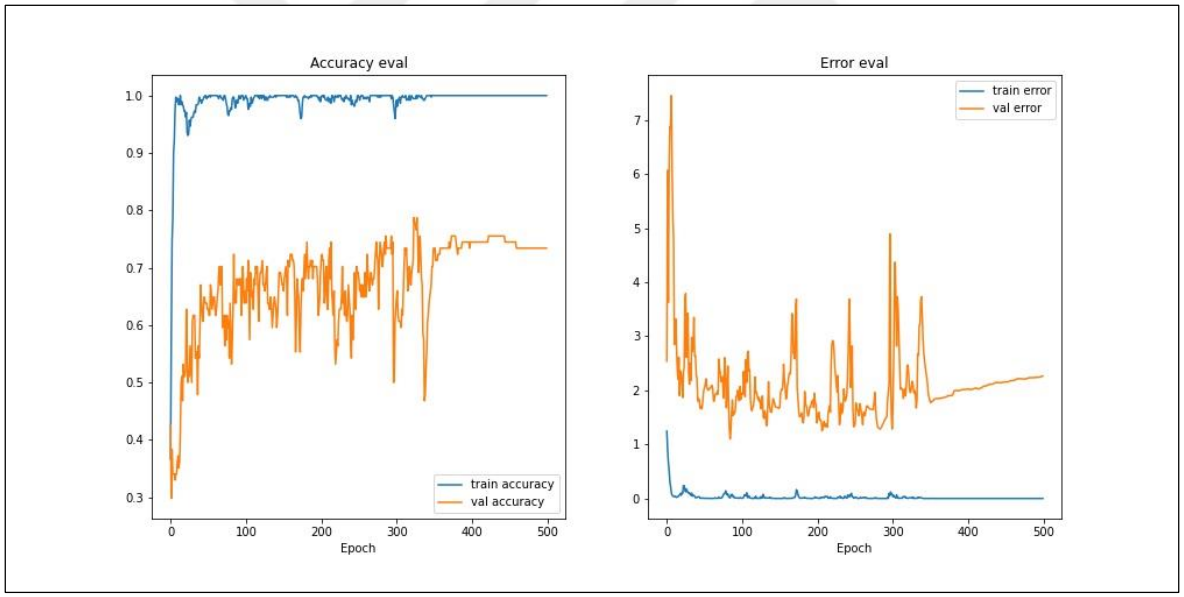


Figure 5.7. ResNet curves before data augmentation.

5.4. Results When Using Data Augmentation Techniques

Data augmentation methods appear to be effective in helping the models reach high accuracies by increasing the number of samples, specifically when dealing with deep learning models that require tons of data to perform better. Inspired by a study conducted by Salamon and Bello [26] which suggested that the utilization of data augmentation techniques immensely increases the performance of deep learning models, we adopted different types of data augmentation tools in our experiments. To implement the data augmentation

methods, we utilized two types of audio deformation techniques which are time stretching and pitch shifting. For the time stretching method, the value of 0.81 was used for slowing down the lung sounds, while 1.07 was deployed for speeding it up. Whereas when the changes are made on the pitch, these values {-3.5, -2.5, -2, -1.5. -1, 1, 1.5, 2, 2.5, 3.5}, are used to complete the pitch-shifting deformation. In their study, Salamon and Bello underlined the effectiveness of the pitch-shifting method in which they started with 4 values, {-2, -1, 1, 2} to accomplish the pitch shifting of the audio files. Then, after noticing how beneficial this method was, they added 4 more values {-3.5, -2.5, 2.5, 3.5}. In our case, we adopted the coefficients used in [26] with the addition of two more values which are {-1.5, 1.5}. After applying data augmentation techniques, the number of audio files increased to 2031, providing a larger and more diverse dataset for analysis and model training. The results obtained after using the data augmentation techniques are presented in the coming subsections.

5.4.1. VGG

Table 5.2 shows that through the precision metric, a 0.98 score was registered for Ral and Rhonchus classes while 0.97 was reported for the Normal class. In the recall metric, the scores started from 0.98 (for the Normal class), and 0.97 (for the Rhonchus), and ended at 0.96 (for the Ral class). 0.97 is the value recorded for both Normal and Ral classes with the F1-score metric, whereas 0.98 was reported for the Rhonchus class. Across all classes, VGG's overall accuracy stands at 0.97%.

The VGG model’s performance during the training process is depicted in Figure 5.1. Compared to the other graphs, VGG's training accuracy graph is one of the most stable ones as a straight line can be seen until the end of training epochs with some little variations taking place before the 100th epoch. The same trajectory is noticeable on the validation accuracy curve and also on the training loss curve. A straight line can also be seen in the validation loss curve before the 100th epoch, but it seems to rise at the end of the training epochs. The model appears to learn effectively from the provided data during the training process.

Table 5.2. The results achieved by the VGG model

	Precision	Recall	F1-Score
Normal	0.97	0.98	0.97
Rhal	0.98	0.96	0.97
Rhonchus	0.98	0.97	0.98
Accuracy			0.97
Weighted avg	0.97	0.97	0.97

5.4.2. MobileNet

As shown in Table 5.3, MobileNet for Ral sounds has a recall of 0.96 and an F1-Score of 0.97, whereas, for Normal sounds, it has a precision score of 0.97. Other results of precision, recall, and F1-score have reached an average score of 0.98 for all classes. When it comes to accuracy, MobileNet has achieved an average accuracy of 0.98 for all three classes.

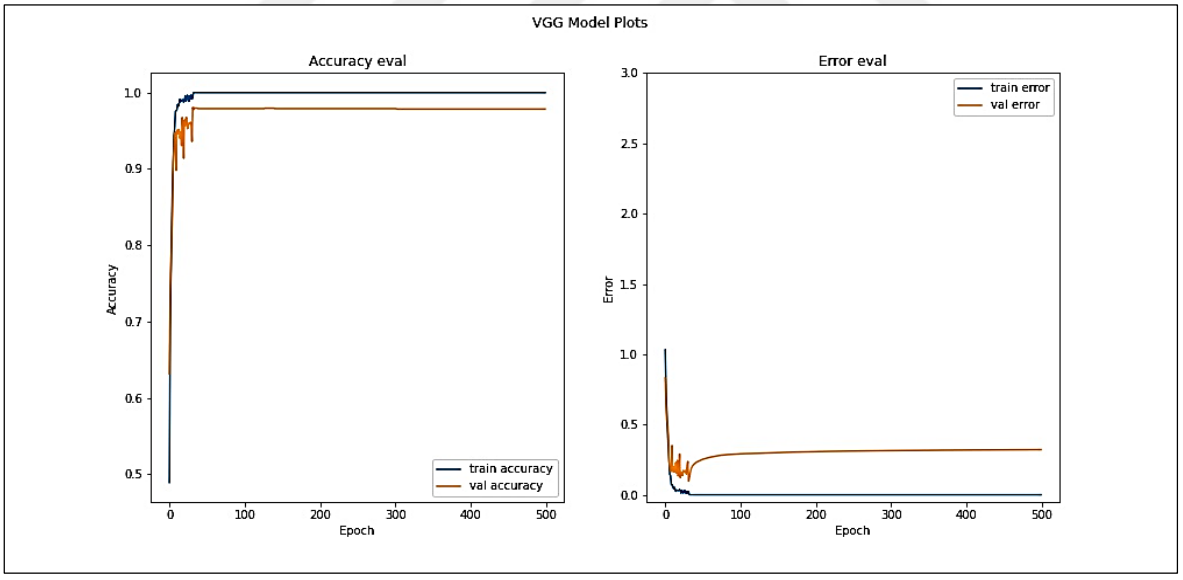


Figure 5.8. VGG curves after data augmentation

Figure 5.2. shows how MobileNet performed during the experiment after training it for 1000 epochs. There are some slight variations in the training accuracy graph throughout the training period, but generally, it is on the same level. These same trends also appear in the training loss graph with some tiny changes. Compared to the validation error graph, the validation accuracy curve has inconsistent changes from the beginning until the end of the

training epochs as huge value changes are noticeable in almost every stage of the training phase.

Table 5.3. The results achieved by the MobileNet model

	Precision	Recall	F1-Score
Normal	0.97	0.98	0.98
Rhal	0.98	0.96	0.97
Rhonchus	0.98	0.98	0.98
Accuracy			0.98
Weighted avg	0.98	0.98	0.98

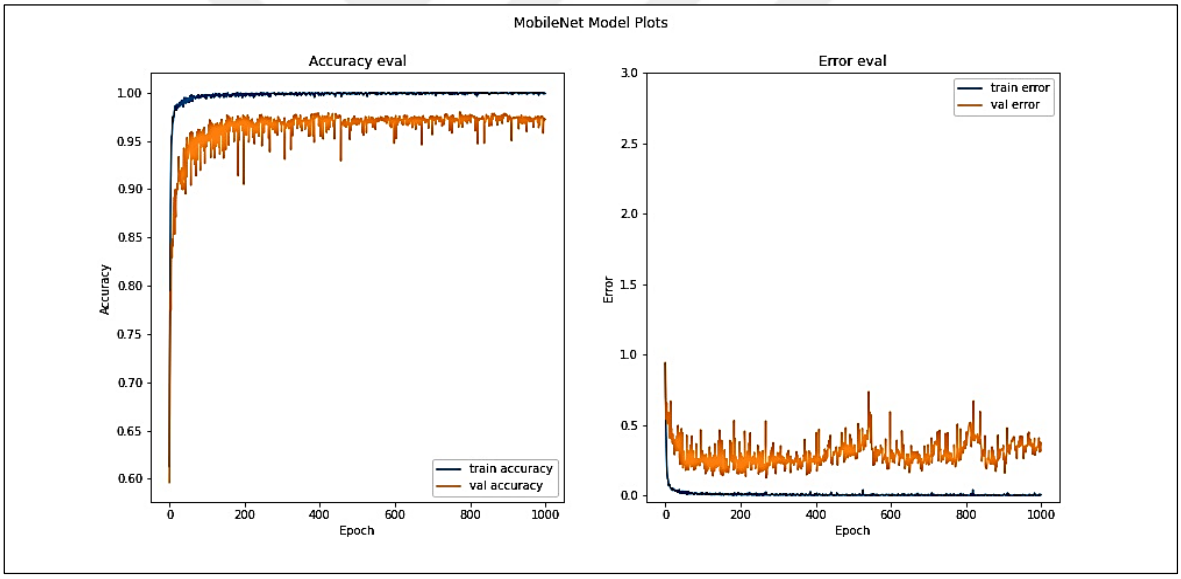


Figure 5.9. MobileNet curves after data augmentation

5.4.3. Xception

Table 5.4 shows that the Xception model has achieved its highest precision score for the Ral class with a score of 0.99. In contrast, its lowest precision score stood at 0.96 for the Normal class. Results of the recall metric are Partially symmetric to the results in precision with 0.99 and 0.96 in the Normal and Ral classes respectively. For the Rhonchus class, 0.98 is the score recorded in each of the precision, recall, and F1-score metrics, while Normal and Ral

classes have reached an F1-Score of 0.97 and 0.98 respectively. In terms of accuracy, Xcetpion's overall accuracy for all classes was reported at 0.98%.

Table 5.4. The results achieved by the Xception model.

	Precision	Recall	F1-Score
Normal	0.97	0.98	0.98
Rhal	0.98	0.96	0.97
Rhonchus	0.98	0.98	0.98
Accuracy			0.98
Weighted avg	0.98	0.98	0.98

As shown in Figure 5.3., the training process of the Xception model lasted for 500 epochs. The training accuracy graph is almost stable during the whole iterations, except that some little changes (up and down) can be observed as the number of epochs is increased. But we can conclude that the model is learning quite effectively with the input data fed to it. While the validation curve of the Xception model shows some inconsistency, and obvious ups and downs, specifically in the beginning and in some cases in the middle. But it appears that this irregularity is decreasing at the end. Whereas the training error graph started with high values, it then decreased close to zero, even though some minimal variations were observed. The validation loss graph shows a clear irregularity as intermittent value changes are seen in the middle of the training phase.

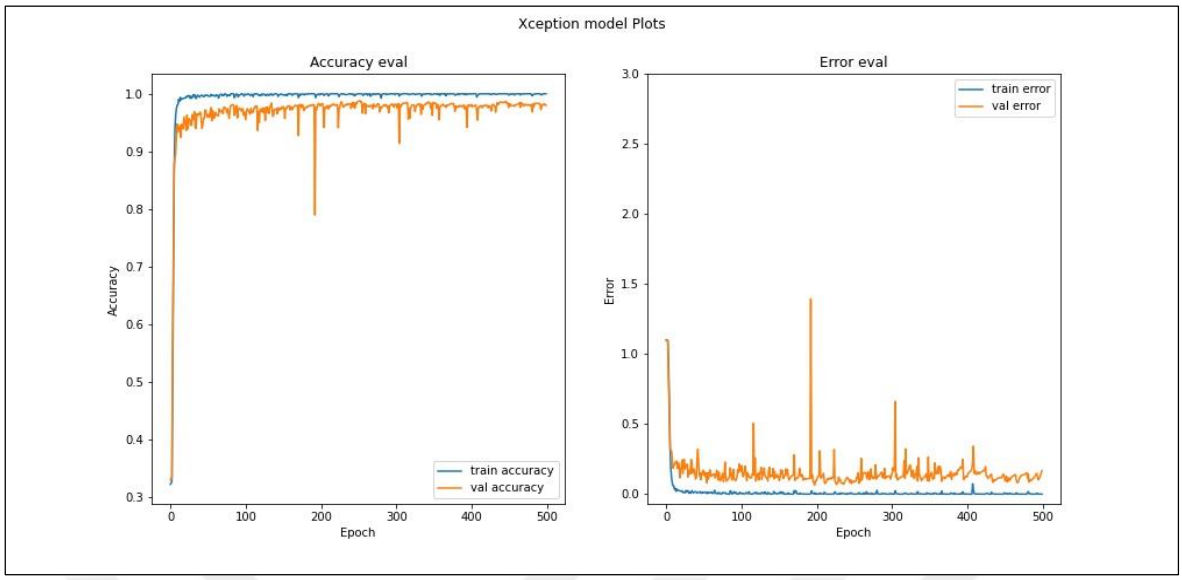


Figure 5.10. Xception curves after data augmentation

5.4.4. ResNet

The results of this model as described in Table 5.5 show that the highest values recorded for the three metrics (precision, recall, and F1-score) belong to the Rhonchus and Normal classes. A 0.98 average score was recorded for both precision and recall while a 0.97 was reported for the F1-score. The lowest score of precision stood at 0.95 for the Normal class. The overall accuracy of the ResNet model is 0.97.

Table 5.5. The results achieved by the ResNet model.

	Precision	Recall	F1-Score
Normal	0.95	0.98	0.97
Rhal	0.97	0.96	0.96
Rhonchus	0.98	0.96	0.97
Accuracy			0.97
Weighted avg	0.97	0.97	0.97

Figure 5.4. shows how the ResNet model performs during the experiment. The training accuracy graph depicts some tiny irregularities, but it almost stays at the same level during the training stage. This property remains identical to the training error graph, but in the loss graph, the irregularities tend to disappear greatly in the training accuracy. On the other side,

the validation accuracy and error graphs have experienced huge variations throughout the entire training process.

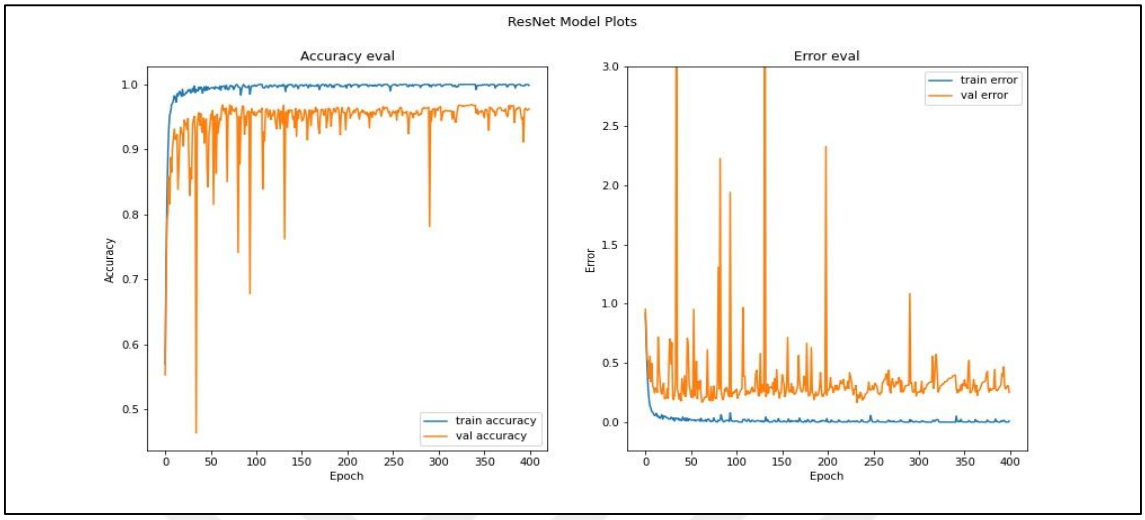


Figure 5.11. ResNet curves network after data augmentation

6. DISCUSSION

Among the most common techniques that physicians use to examine lungs is auscultation, which has become a routine part of clinical exams. Due to the subjective nature of this method, as well as the non-stationary nature of lung sounds, analysis, recognition, and differentiation tasks are complicated. Consequently, there is an interest in employing machine learning and artificial intelligence to automate this process and to develop automatic recognition systems (ARS) that tackle these limitations [1]. In this thesis, different pre-trained models were assessed on a lung audio dataset that has 177 audio files recorded at a public hospital in Turkey.

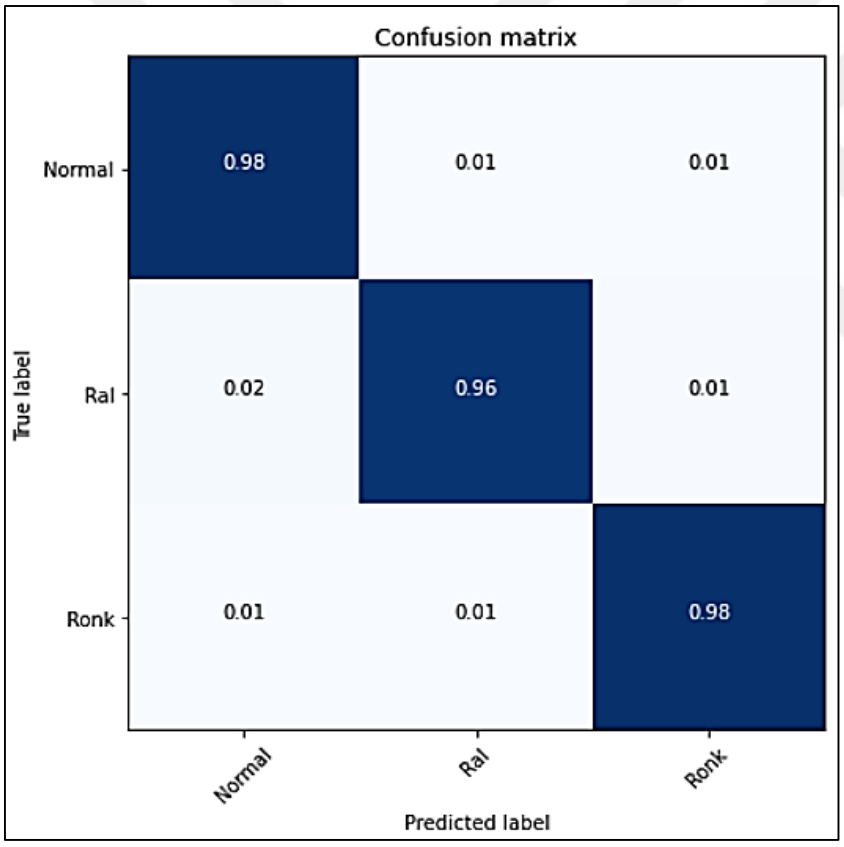


Figure 6.1. Confusion matrix of MobileNet network after data augmentation

This dataset is composed of three different classes: normal, rhonchus, and rhal. Due to the small size nature of the dataset, some data augmentation techniques were applied for the sake of generating more samples from the original dataset. These were done either by changing the speed of the lung sound or by changing its pitch with the use of different parameters. These data augmentation methods proved that they can be effective in

optimizing model performance. After that comes the feature extraction stage in which the MFCC features are extracted by using the Librosa library, they are then forwarded to the models. The pre-trained models were chosen as classifiers to save training time as well as for the fact that they are recognized to be efficient in classification problems thanks to their pre-trained weights on huge datasets. The pre-trained models were further trained on this lung sound dataset for different epochs. The Xception and MobileNet models have attained the highest classification accuracy of 98% among the other algorithms. These two algorithms were trained for 500 and 1000 epochs respectively. Their training accuracy graphs look quite similar to each other with little variations. Then comes the VGG and ResNet models with a classification accuracy of 97%. Among these latter algorithms, ResNet was trained for 400 epochs while the other was trained for 500 epochs. The ResNet graph has too many variations compared to that of the VGG model, specifically the validation graphs (accuracy and loss) as shown in Figure 5.4. Deep learning models demand huge amounts of data to learn from, so they can achieve greater performances in making correct predictions.

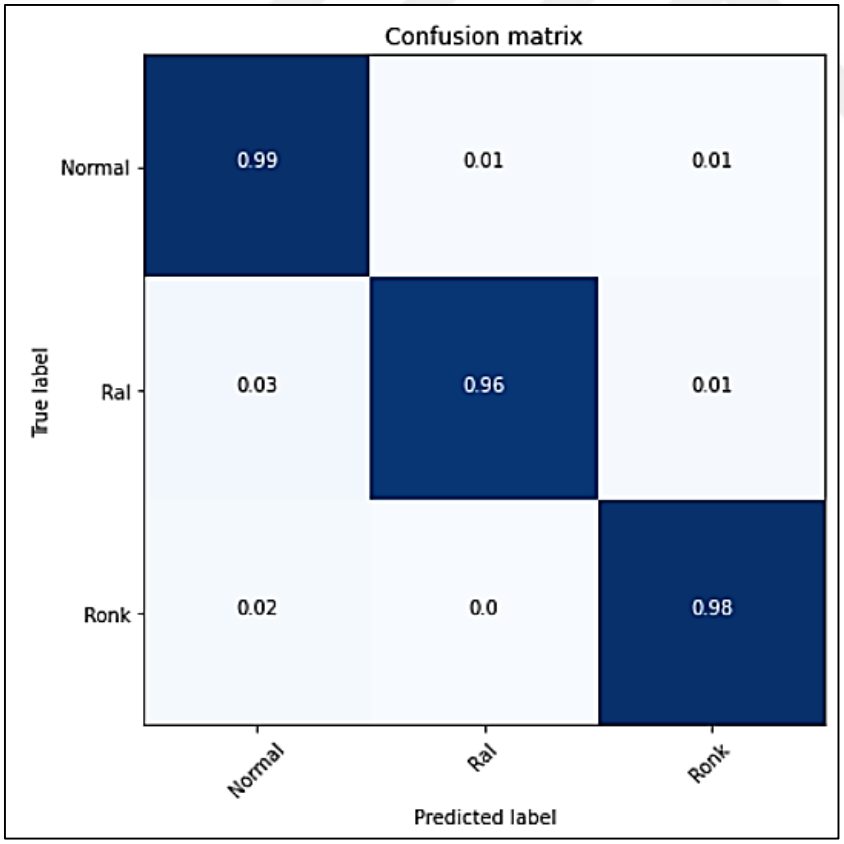


Figure 6.2. Confusion matrix of the Xception network after data augmentation

This is why lower model performances were observed before applying audio data deformation methods. When the models were trained without using data augmentation techniques, VGG and Xception models were the ones with the lowest classification accuracies, and both achieved an average accuracy of 77%. Whereas they were able to reach higher classification accuracies of 97% and 98% after the deployment of these audio deformation methods. Table 6.1 sum up the performance results of the deep learning models in different evaluation metrics. Note that these results were obtained with the utilization of data augmentation techniques. The confusion matrixes of our best classifiers (MobileNet and Xception) on the classification of the lung sound dataset are illustrated respectively in Figure 6.1. and Figure 6.2.

Table 6.1. Results after the utilization of data augmentation techniques

Models	Accuracy	Precision	Recall	F1-Score
VGG	0.97	0.97	0.97	0.97
MobileNet	**0.98**	**0.98**	**0.98**	**0.98**
Xception	**0.98**	**0.98**	**0.98**	**0.98**
ResNet	0.97	0.97	0.97	0.97

The ImageNet dataset is a large-scale dataset of labeled images used for training and evaluating computer vision algorithms. It was created by researchers at Stanford University and is one of the most widely used image datasets in computer vision. The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is an annual competition that evaluates the performance of computer vision algorithms on the ImageNet dataset. The challenge consists of two tasks: object localization and image classification. In the image classification task, the goal is to classify an image into one of 1,000 categories. The challenge has been running since 2010 and has played a significant role in advancing the field of computer vision. The ImageNet dataset has been used to train many deep learning models, including AlexNet, VGG, ResNet, Inception, and others. These models have achieved state-of-the-art performance on various computer vision tasks and have significantly advanced the field of computer vision. The four pre-trained models (VGG, MobileNet, Xception, and ResNet) that we utilized in our thesis were trained on the ImageNet dataset, this explains in somehow why their results are close to each other, in both scenarios when applying data augmentation techniques and when not applying them.

When comparing the performance of different models, it can be difficult to choose between them when their accuracy scores are similar. In such cases, looking at the number of trainable parameters in each model can be a useful factor to consider. In this case, it appears that MobileNet has the lowest number of trainable parameters, with approximately 8 million. This makes it an attractive choice for scenarios where computational resources are limited, such as on mobile or embedded devices. Next in line is VGG16 with around 17 million trainable parameters, followed by Xception with 29 million parameters. ResNet50V2 has the highest number of trainable parameters, with approximately 35 million. It's worth noting that the number of trainable parameters is not always a reliable predictor of performance, as other factors such as the architecture, pre-training, and fine-tuning techniques can also impact a model's accuracy. However, it can still be a useful factor to consider when choosing a model for a specific use case.

7. CONCLUSION

According to the literature, we can deduce that MFCC audio features are preferred as inputs along with machine learning methods for classifying lung sounds. This can be seen in [1, 2, 6]. On the other hand, spectrum types are the most used features to feed deep learning models to classify lung sound datasets [1, 14, 17]. MFCCs as inputs along with pre-trained deep learning models are not explored much, according to our knowledge, hence this is what our research is about. After experimenting, this approach showed to be effective, and accurate as it is illustrated in the previous tables of the different methods. [1, 6, 17] deployed data augmentation methods. In [6] and [17] the data augmentation methods were applied before the feature extraction phase, while in [1] they were utilized on the spectrogram meaning that after the feature extraction process. Here in this study, we employed two different types of audio data deformations offline (before extracting the MFCCs) and they contributed positively to the learning performance of the pre-trained models.

Our proposed approach achieved better results by reaching accuracy values of 97% and 98%. The dataset used in this study was used previously in [6] by Guler et al., in which they used machine learning methods as classifiers. The Naïve Bayes algorithm registered the highest classification accuracy (95%) under 10-cross validation accuracy. This value is less than the lowest accuracy (97%) achieved by our proposed pre-trained models such as VGG.

Adventitious breathing sounds are unwanted sounds that come along with normal breathing sounds. Seven different classes of abnormal sounds can be distinguished: wheeze, rhonchi, stridor, squawk, coarse crackle, and pleural rub [3]. Unfortunately, the dataset used in this research has limited samples and only two abnormal classes (rhonchi and crackle) are available. Classifiers could be more specific when determining the class of a given lung sound if a dataset includes all types of abnormal lung sounds mentioned above.

For future studies, researchers plan to extract audio features such as MFCCs or Mel-spectrograms from lung audio files, feed these features to pre-trained deep learning models, then combine their outputs. The result of this combination can be used as inputs for a traditional classifier such as KNN, or SVM to complete the classification task. Specifically, the use of a light-weighted pre-trained model such as MobileNet would facilitate its integration with mobile devices, hence so they can be used easily by healthcare professionals

for the classification of patients' lung sounds [59]. An alternative research idea also might be to extract Mel-spectrograms using Librosa, extract embeddings using OpenL3 [60] from the lung sounds and combine their outputs. Then use the result of this combination as input for a deep learning model to proceed to the lung sounds classification problem.

REFERENCES

1. Bardou D., Zhang K., and Ahmad S. M. (2018). Lung sounds classification using convolutional neural networks. *Artificial intelligence in medicine,* vol. 88, 58–69.

2. Elsetrønning A., Rasheed A., Bekker J., and San O. (2020). On the effectiveness of signal decomposition, feature extraction and selection on lung sound classification. *ArXiv, abs/2012.11759*.

3. Zulfiqar R., Majeed F., Irfan R., Rauf H. T., Benkhelifa E., and Belkacem A. N. (2021) Abnormal respiratory sounds classification using deep cnn through artificial noise addition. *Frontiers in Medicine,* 8.

4. Nguyen T. and Pernkopf F. (2021). Crackle detection in lung sounds using transfer learning and multi-input convolutional neural networks. *43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE*, Guadalajara, Mexico, 80–83.

5. Reichert S., Gass R., Brandt C., and Andrès E. (2008). Analysis of respiratory sounds: state of the art. *Clinical medicine. Circulatory, respiratory, and pulmonary medicine*, 2, CCRPM–S530.

6. Guler H. C., Yildiz O., Baysal U., Cinyol F. B., Koksal D., Babaoglu E., and Ulasli S. S. (2020). Classification of abnormal respiratory sounds using machine learning techniques. *Medical Technologies Congress (TIPTEKNO),* Antalya, Turkey, 1–4.

7. Li, J., and Hong, Y. (2016). Crackles detection method based on time-frequency features analysis and SVM. *IEEE 13th International Conference on Signal Processing (ICSP)*, 1412-1416.

8. Ulukaya, S., Serbes, G., Sen, I., and Kahya, Y. P. (2016). A lung sound classification system based on the rational dilation wavelet transform. *38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 3745-3748.

9. Altan, G., Kutlu, Y., Pekmezci, A. Ö., and Nural, S. (2021). The diagnosis of asthma using Hilbert-Huang transform and deep learning on lung sounds. *ArXiv, abs/2101.08288*.

10. Balli, O., and Kutlu, Y. (2021). Effect of window size for detection of abnormalities in respiratory sounds. *ArXiv, abs/2101.08495*.

11. Aras, S., and Gangal, A. (2017). Comparison of different features derived from mel frequency cepstrum coefficients for classification of single channel lung sounds. *40th International Conference on Telecommunications and Signal Processing (TSP)*. 346-349.

12. Islam, M. A., Bandyopadhyaya, I., Bhattacharyya, P., and Saha, G. (2018). Multichannel lung sound analysis for asthma detection. *Computer methods and programs in biomedicine*, 159, 111-123.

13. Haider, N. S., Singh, B. K., Periyasamy, R., and Behera, A. K. (2019). Respiratory sound based classification of chronic obstructive pulmonary disease: a risk stratification approach in machine learning paradigm. *Journal of medical systems*, 43(8), 1-13.

14. Messner, E., Fediuk, M., Swatek, P., Scheidl, S., Smolle-Jüttner, F. M., Olschewski, H., and Pernkopf, F. (2020). Multi-channel lung sound classification with convolutional recurrent neural networks. *Computers in Biology and Medicine*, 122, 103831.

15. Shuvo, S. B., Ali, S. N., Swapnil, S. I., Hasan, T., and Bhuiyan, M. I. H. (2020). A lightweight cnn model for detecting respiratory diseases from lung auscultation sounds using emd-cwt-based hybrid scalogram. *IEEE Journal of Biomedical and Health Informatics*, 25(7), 2595-2603.

16. Rocha, B. M., Pessoa, D., Marques, A., Carvalho, P., and Paiva, R. P. (2020). Influence of Event Duration on Automatic Wheeze Classification. 2020 *25th International Conference on Pattern Recognition (ICPR)*, 7462-7469.

17. Gairola, S., Tom, F., Kwatra, N., and Jain, M. (2020). Respirenet: A deep neural network for accurately detecting abnormal lung sounds in limited data setting. *43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 527-530.

18. Don, S. (2020). Random subset feature selection and classification of lung sound. *Procedia Computer Science*, 167, 313-322.

19. Demir, F., Sengur, A., and Bajaj, V. (2020). Convolutional neural networks based efficient approach for classification of lung diseases. *Health information science and systems*, 8(1), 1-8.

20. Datta, S., Choudhury, A. D., Deshpande, P., Bhattacharya, S., and Pal, A. (2017). Automated lung sound analysis for detecting pulmonary abnormalities. *39th Annual International Conference of the Ieee Engineering in Medicine and Biology Society (Embc)*, 4594-4598.

21. Aras, S., Gangal, A., and Bülbül, Y. (2015). Classification of healthy and pathologic lung sounds recorded with electronic auscultation. *23nd Signal Processing and Communications Applications Conference (SIU)*, 252-255.

22. Naves, R., Barbosa, B. H. and Ferreira, D. D. (2016). Classification of lung sounds using higher-order statistics: A divide-and-conquer approach. *Computer methods and programs in biomedicine*, 129, 12-20.

23. Fraiwan, M., Fraiwan, L., Alkhodari, M., and Hassanin, O. (2021). Recognition of pulmonary diseases from lung sounds using convolutional neural networks and long short-term memory. *Journal of Ambient Intelligence and Humanized Computing*, 1-13.

24. Bozkurt, B., Germanakis, I., and Stylianou, Y. (2018). A study of time-frequency features for CNN-based automatic heart sound classification for pathology detection. *Computers in biology and medicine*, 100, 132-143.

25. Lella, K. K., and Pja, A. (2021). Automatic COVID-19 disease diagnosis using 1D convolutional neural network and augmentation with human respiratory sound based on parameters: cough, breath, and voice. *AIMS Public Health*, 8(2), 240.

26. McFee, B., Humphrey, E. J., and Bello, J. P. (2015). A software framework for musical data augmentation. *ISMIR*, 248-254.

27. Srivastava, A., Jain, S., Miranda, R., Patil, S., Pandya, S., and Kotecha, K. (2021). Deep learning based respiratory sound analysis for detection of chronic obstructive pulmonary disease. *PeerJ Computer Science*, 7.

28. Petmezas, G., Cheimariotis, G. A., Stefanopoulos, L., Rocha, B., Paiva, R. P., Katsaggelos, A. K., and Maglaveras, N. (2022). Automated Lung Sound Classification Using a Hybrid CNN-LSTM Network and Focal Loss Function. *Sensors*, 22(3), 1232.

29. Dokur, Z. (2009). Respiratory sound classification by using an incremental supervised neural network. *Pattern Analysis and Applications*, 12(4), 309-319.

30. Pramono, R. X. A., Bowyer, S., and Rodriguez-Villegas, E. (2017). Automatic adventitious respiratory sound analysis: A systematic review. *PloS one*, 12(5), e0177926.

31. Baughman, R. P., and Loudon, R. G. (1989). Stridor: Differentiation from asthma or upper airway Noise1-3. *Am Rev Respir Dis*, 139, 1407-1409.

32. Rizal, A., Hidayat, R., and Nugroho, H. A. (2016). Lung sounds classification using spectrogram's first order statistics features. *6th International Annual Engineering Seminar (InAES)*, 96-100.

33. Gurung, A., Scrafford, C. G., Tielsch, J. M., Levine, O. S., and Checkley, W. (2011). Computerized lung sound analysis as diagnostic aid for the detection of abnormal lung sounds: a systematic review and meta-analysis. *Respiratory medicine*, 105(9), 1396-1403.

34. Sharma, G., Umapathy, K., and Krishnan, S. (2020). Trends in audio signal feature extraction methods. *Applied Acoustics*, 158, 107020.

35. Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT press.

36. Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (Vol. 2, pp. 1-758). New York: springer.

37. Nordby, J. O. (2019). *Environmental sound classification on microcontrollers using Convolutional Neural Networks* (Master's thesis, Norwegian University of Life Sciences, Ås).

38. Anderson, D., and McNeill, G. (1992). Artificial neural networks technology, DACS State-ofthe-Art Report. *Utica (NY): Data and Analysis Center for Software*.

39. Nair V. and Hinton G. E. (2010). Rectified linear units improve restricted boltzmann machines. *27th international conference on machine learning (icml- 10)*, 807–814.

40. Albawi S., Mohammed T. A., and Al-Zawi S. (2017). Understanding of a convolutional neural network. *International conference on engineering and technology (ICET). Ieee*, Antalya, Turkey, 1–6.

41. Internet: Basic CNN architecture: Explaining 5 layers of Convolutional Neural Network. upGrad blog. (2022, October 3). Retrieved December 22, 2022, from https://www.upgrad.com/blog/basic-cnn-architecture/

42. Simonyan K. and Zisserman A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR, vol. abs/1409.1556*.

43. Howard A. G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., and Adam H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv, vol. abs/1704.04861*.

44. Chollet F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, Hawaii, United States, 1251–1258.

45. He K., Zhang X., Ren S., and Sun J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, United States, 770–778.

46. Salamon J. and Bello J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal processing letters*, 24 (3), 279–283.

47. Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. *ArXiv, abs/1710.09412*.

48. Grandini M., Bagli E., and Visani G. (2020). Metrics for multi-class classification: an overview. *ArXiv, vol. abs/2008.05756*.

49. Internet: Keras documentation: About keras. URL: https://keras.io/about/, last accessed date: 22.12.2022

50. Internet: Scikit learn - introduction. URL:https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.htm#:~:text=Scikit%2Dlearn%20(Sklearn)%20is,a%20consistence%20interface%20in%20Python, last accessed date: 22.12.2022

51. Internet: Numpy introduction. URL:https://www.tutorialspoint.com/numpy/numpy_introduction.htm, last accessed date: 22.12.2022

52. Internet: Librosa. URL: https://librosa.org/doc/latest/index.html, last accessed date: 22.12.2022

53. Rahaman M. E., Alam S. S., Mondal H. S., Muntaseer A. S., Mandal R., and Raihan M. (2019). Performance analysis of isolated speech recognition technique using mfcc and cross-correlation. *10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE*, Kanpur, India, pp. 1–4.

54. Wahyuni E. S. (2017). Arabic speech recognition using mfcc feature extraction and ann classification," *2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). IEEE*, Yogyakarta, Indonesia, 22–25.

55. Mushtaq Z. and Su S. F. (2020). Environmental sound classification using a regularized deep convolutional neural network with data augmentation. *Applied Acoustics*, 167, 107389.

56. Toffa O. K. and Mignotte M. (2020). Environmental sound classification using local binary pattern and audio features collaboration. *IEEE Transactions on Multimedia*, 23, 3978–3985.

57. Tiwari V. (2010). Mfcc and its applications in speaker recognition. *International journal on emerging technologies*, 1(1),19–22.

58. Internet: Transfer learning and fine-tuning: Tensorflow Core. URL: https://www.tensorflow.org/tutorials/images/transfer_learning, last accessed date: 23.12.2022

59. Sutaji D. and Yıldız O. (2022). Lemoxinet: Lite ensemble mobilenetv2 and xception models to predict plant disease. *Ecological Informatics*, 70, 101698.

60. Arandjelovic R. and Zisserman A. (2017). Look, listen and learn. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, Italy, 609– 617.

61. Internet: VGG in tensorflow. URL: https://www.cs.toronto.edu/~frossard/post/vgg16/, last accessed date: 03.03.2023

62. Phiphiphatphaisit S. and Surinta O. (2020). Food image classification with improved MobileNet architecture and data augmentation. *Proceedings of the 3rd International Conference on Information Science and Systems*, 51-56.

63. Internet: The annotated resnet-50. URL: https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758, last accessed date: 03.03.2023

64. Flanagan J. and Golden R. (1966). Phase vocoder. *Bell System Technical Journal*, 45(9), 1493-1509.

65. Amara M. and Kumar P. (2017). Implementation of Pitch Shifter using Phase Vocoder Algorithm on Artix-7 FPGA. *International Journal of Control Theory and Applications*, 10, 6.

CURRICULUM VITAE

Personal Information

Surname, Name	: IDRISSE, Ahamadi Abdallah
Nationality	: Komorlar
Date and Place of Birth	:
Marital status	:
Phone number	:
E-mail	:

Education

Degree	School / Program	Graduation Date
M. Sc.	Gazi University / Computer Science	Ongoing
B. Sc.	University of Kassala / Computer Science	2017
High School	École Nationale Technique et Professionnelle de Ouani (ENTP)	2011

Professional Experience

Year	Place of Work	Position

Foreign Language

Türkçe, İngilizce, Arapça, Fransızca

Publications

A. A. Idrisse, O. Yıldız, and U. Baysal. (2022). A Lightweight Deep Learning Model for The Classification Of A Limited Lung Sound Dataset. *International Conference on Engineering Sciences (IENSCI)*, Baku Azerbaijan.

Hobbies

Gerçek hikayelere dayalı belgeseller



GAZİLİ OLMAK AYRICALIKTIR…