SIGNAL PROCESSING AND FEATURE ENGINEERING

OF RESPIRATORY DISEASE

Major project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

By

Priya Verma (191428)

Ajay Yadav (191422)

UNDER THE SUPERVISION OF

Dr. Sunil Datt Sharma & Dr. Diksha Hooda

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat, 173234, Himachal Pradesh, INDIA

DECLARATION

I hereby declare that the work presented in this report entitled “Signal Processing and Feature Engineering of Respiratory Disease” in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2022 to December 2022 under the supervision of **Dr. Sunil Datt Sharma**, Assistant Professor (SG) and Co-Supervision of **Dr. Diksha Hooda**, Assistant Professor(Grade-II).

I also authenticate that I have carried out the above mentioned project work under the proficiency stream Data Science.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Submitted by:

Ajay Yadav 191422

Priya Verma 191428

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Supervised by : Dr. Sunil Datt Sharma Assistant Professor (SG)

Department of ECE

Co-Supervised by : Dr. Diksha Hooda Assistant Professor (Grade-II)

Department of CSE

PLAGIARISM CERTIFICATE

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ....

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report | Paper

Name: ___________________________ **__Department:** _________________ **Enrolment No** _________

Contact No. ______________________________**E-mail.** ______________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): ________________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ....(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Generated Plagiarism Report Details (Title, Abstract & Chapters)	
	• All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String		Word Counts	
Report Generated on			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
Name & Signature **Librarian**

....

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

ACKNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully.

I am really grateful and wish my profound indebtedness to Supervisor **Dr. Sunil Datt Sharma, Assistant Professor (SG)** and Co-Supervisor **Dr. Diksha Hooda**, Jaypee University of Information Technology, Waknaghat deep Knowledge & keen interest of my supervisor and Co-Supervisor in the field of "**Audio Signal Processing**", "**Machine Learning" and "Deep Learning"** to carry out this project. Their endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Sunil Datt Sharma,** and **Dr. Diksha Hooda**,for their kind help to finish my project.

I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

Priya Verma (191428)

Ajay Yadav (191422)

Table of Content

TOPIC	Page No.
DECLARATION ………………………………………..	i
PLAGIARISM CERTIFICATE ……………………………	ii
ACKNOWLEDGEMENT …………………………………	iii
TABLE OF CONTENTS…………………………………...	iv
LIST OF ABBREVIATIONS……………………………...	vi
LIST OF FIGURES………………………………………..	vii
LIST OF TABLES……………………………………….	vii
ABSTRACT……………………………………………...	viii
CHAPTER-1 INTRODUCTION…………………………...	1
1.1 Introduction……………………………………………	1
1.2 Problem Statement……………………………………	2
1.3 Objective………………………………………………	2
1.4 Methodology…………………………………………..	3
1.5 Organisation…………………………………………...	5
CHAPTER-2 LITERATURE SURVEY……………………	6

CHAPTER-3 SYSTEM DEVELOPMENT………………... 13

CHAPTER-4 PERFORMANCE ANALYSIS…………… 32

CHAPTER-5 CONCLUSIONS……………………………. 56

REFERENCES……………………………………………... 57

List of Abbreviations

CT Scans - Computed Tomography

MFCC - Mel-Frequency Cepstral Coefficient

CNN - Convolutional Neural Network

COPD - Chronic Obstructive Pulmonary Disease

URTI - Upper RespiratoryTract Infection

LRTI - Lower Respiratory Tract Infection

STFT - Short Time Fourier Transform

FFT - Fast Fourier Transform

RSV - Respiratory Syncytial Virus

AUC - Area Under Curve

ROC - Receiver Operating Characteristic

SGD - Stochastic Gradient Descent

MB-SGD - Mini Batch Stochastic Gradient Descent

List of Figures

Fig. 1.1 Flow Diagram of Major Project
Fig. 3.1 Stationary Signal
Fig. 3.2 Non-Stationary Signal
Fig. 3.3 Mathematical model of Neutral Network
Fig. 3.4 Neural Network Layers
Fig. 3.5 Forward Propagation
Fig. 3.6 Backward Propagation
Fig. 3.7 Single Perceptron
Fig. 3.8 Multi-Layer Perceptron
Fig. 4.1 Web plot of Healthy person
Fig. 4.2 Web plot of COPD
Fig. 4.3 Web plot of URTI
Fig. 4.4 Web plot of Pneumonia
Fig. 4.5 Web plot of Bronchiectasis
Fig. 4.6 Web plot of Bronchiolitis
Fig. 4.7 Web plot of Audio Signal
Fig. 4.8 Web plot after Noise Injection
Fig. 4.9 Web plot after Shifting Time
Fig. 4.10 Web plot after Speed Adjustment
Fig. 4.11 Bar Graph before Augment Features
Fig. 4.12 Scatter plot before Augment Features
Fig. 4.13 Bar Graph after Augment Features
Fig. 4.14 Scatter Plot after Augment Features
Fig. 4.15 Confusion Matrix 1
Fig. 4.16 Accuracy And Loss 1
Fig. 4.17 Confusion Matrix 2
Fig. 4.18 Accuracy And Loss 2
Fig. 4.19 Confusion Matrix 3
Fig. 4.20 Accuracy And Loss 3

List of Table

Table 2.1 Literature Survey

Table 4.1 Performance Measure 1

Table 4.2 Performance Measure 2

Table 4.3 Performance Measure 3

ABSTRACT

An area of study that has recently attracted more attention is respiratory sound analysis. In fact, there is a chance that the irregularities in the early phases of a lung dysfunction may be automatically inferred in this area. In this study, we provide a technique for automatically analysing respiratory sounds. The objective is to demonstrate how well Deep learning methods work for analysing respiratory sounds. Although systems for evaluating audio signals already exist in this emerging era of new technologies, there is still a need to build a tool that can analyse audio signals, diagnose sickness at an early stage, and track the recovery of patients. By utilising Deep learning techniques, we are able to determine whether a feature vector is associated with a patient who is suffering from a lung disease. A feature vector is directly collected from breath audio. Asthma, bronchiectasis, bronchiolitis, chronic obstructive pulmonary disease, pneumonia, and lower or upper respiratory tract infections can all be diagnosed using the suggested method. We utilised Librosa's machine learning package in the tests we did, which includes features like MFCC, Mel-Spectrogram, and Chroma. The degree of the diagnosed ailment, such as mild, moderate, or acute, might also be determined by the system that was provided. The findings of the research support the effectiveness of the suggested deep learning strategy. A retrospective experimental analysis on 126 patients with 920 recording sessions showed the effectiveness of the proposed method.

Chapter 01: INTRODUCTION

1.1 Introduction

When compared to numerous other industries, the healthcare industry is one of the most important and autonomous sectors. It is one of the most important and crucial industries where consumers want the best standards of treatment, diagnosis and services in line with their expenditures. Due to their clarity, the pictures and audio produced by various medical devices may have a variety of restrictions. Because various interpreters/doctors analyze them differently, they may be prone to significant variances. A short time ago, we were solely dependent on human brains, talent, and capacity to evaluate and comprehend the vast amounts of medical data produced by different types of equipment and gear. COPD is a term used to describe a wide range of lung conditions that impair breathing by obstructing airflow into and out of the lungs owing to congested airways. Lack of oxygen causes the lungs to release more carbon dioxide. The two major diseases that contribute to COPD are chronic bronchitis and emphysema.These are two illnesses that frequently coexist and can have varying degrees of severity in people with COPD. Narrowed and inflamed airways are caused by bronchitis. The causes of COPD are frequently recognised and include tobacco use, genetic disorders, pollution in air, etc. COPD regular detection is under research and development to get 100% accuracy.

Practitioners of medicine presently employ the following traditional methods:

1: Lung tests: During examinations, determine if the oxygen is properly supplied to the lungs during. Spirometry is the most typical test for this. In this subject inhales the machine which then calculates how much air was exhaled.

2: Chest X-ray: It may detect a number of lung conditions, but Emphysema stands out. The primary cause of COPD is emphysema.

3: CT scan: Although not frequently employed, CT scans are performed when there is no chance of detection using more conventional techniques. It can determine whether the person needs COPD surgery of any kind.

4: The quantity of O2 that rises in the blood during breathing is measured by an arterial blood gas analysis.

Disadvantage :

1: People with cardiac issues or those who had their surgery in heart cannot undergo spirometry.

2: Unsteadiness, nausea, and dizziness are a few symptoms that are frequently experienced following the testing.

3: Both X-rays and scans put patients at danger for dying from radiation exposure.

1.2 Problem Statement

All industrial sectors are affected by the serious issue of occupational respiratory illnesses, which account for a sizable portion of occupational disease mortality. Asthma, pneumonia, TB, obliterative bronchiolitis, and chronic obstructive lung disease are among the illnesses and disorders brought on by work (COPD). These are a few of the ailments that might impair your ability to breathe. You can control your breathing issues with the proper diagnosis, treatment, and comprehension of your disease. So, we want to provide a technique for automatically analyzing these diseases by using respiratory sound.

1.3 Objective

Preprocessing of the dataset involves various techniques such as noise injection, time shifting, and speed adjustment to augment the data and account for the imbalance in the dataset. After preprocessing, we extract the features from each audio file using Mel Frequency Cepstral Coefficients (MFCC). The preprocessed data is split into training and testing sets. A convolutional neural network (CNN) model architecture is designed.The model is trained using the preprocessed training data as input. The model's performance is evaluated using the testing data. Performance measures are calculated for each label, which allows for an assessment of the model's effectiveness in accurately classifying the different classes in the dataset.

The following goals are part of this project:

1) For the respiratory condition, feature extraction based on time-frequency analysis.

2) Classification of respiratory disorders using machine learning tools.

1.4 Methodology

In this Major Project we have used CNN to classify different varieties of respiratory diseases using different features of audio signal. Creating augmented samples for distinct respiratory classes and classifying the samples using deep learning models are the two main components of our work. Feature extractions are part of data pre-processing after audio capture. We want to compare how well the classifiers perform before and after augmentation. Mel Frequency Cepstrum Coefficients are obtained in feature extraction steps and then supplied to the suggested classification models for the classification of respiratory disorders.

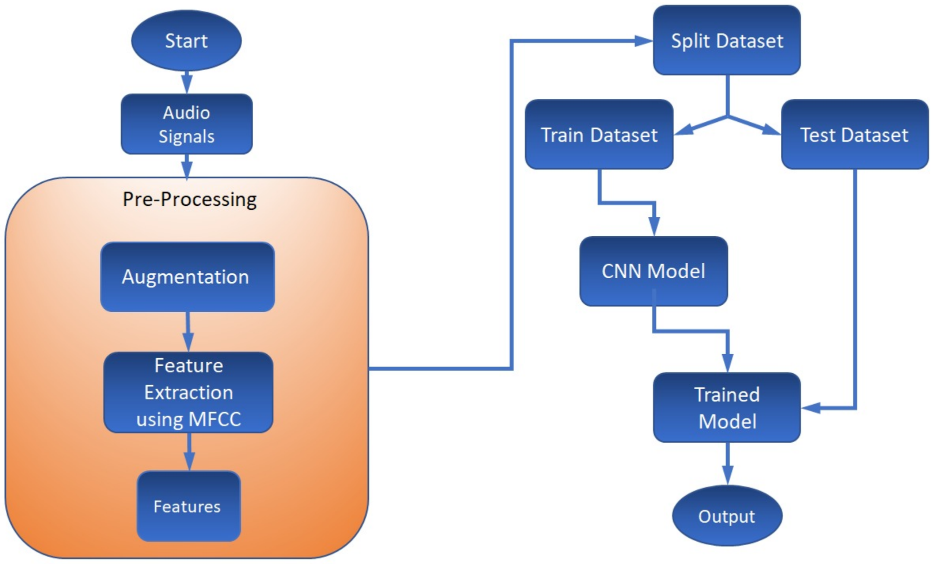

Fig. 1.1 Flow diagram of Major Project

Data: Collect respiratory sound dataset from Kaggle i.e. ICBHI dataset that contains Chronic Obstructive Pulmonary Disease (COPD), Asthma, Pneumonia etc. The data can be collected using a digital stethoscope or other appropriate devices.

Data pre-processing: Apply appropriate pre-processing techniques to the collected respiratory sound data to remove noise, artifacts, and other unwanted signals. This may include techniques such as augmentation using noise injection, time shifting, speed adjustment.

Feature extraction: Extract relevant features from the pre-processed respiratory sound data using appropriate signal processing techniques. The features can be based on time-frequency analysis, and may include features such as MFCC.

Model development: Develop a classification model using DL algorithms such as CNN , artificial neural networks (ANNs). The model should be trained and validated using appropriate datasets, such as cross-validation or leave-one-out validation.

Model evaluation: Evaluate the performance of the developed model using appropriate metrics such as accuracy, sensitivity, specificity, recall and precision based on confusion matrix. The performance of the model should be compared with existing diagnostic tools and techniques.

Interpretation: Interpret the results of the developed model and visualize the classification outcomes using appropriate techniques such as confusion matrices.

1.5 Organization

The structure of this paper is as follows:

Chapter 1: Introduction.

Chapter 2: Literature Survey: A comprehensive review of the existing literature on signal processing and feature engineering techniques used in respiratory disease research.

Chapter 3: System Development: Detailed description of the research methodology.

Chapter 4: Experiment & Result Analysis: Presentation and analysis of the experimental results obtained from the evaluation of various signal processing and feature engineering techniques.

Chapter 5: Conclusion: Summary of the research, key findings, and future research directions.

References: List of all the references cited in the research document.

Chapter 02: LITERATURE SURVEY

2.1 Journal Review

Table 2.1 : Literature Survey

Author(s)	Year	Methodology	Limitations
D. Bardou, K. Zh., S. M. Ahmad	June, 2018	CNN in Lung sounds classification	The lack of information on the model's decision-making process may limit its interpretability.
Jiuxiang Gu, Zhenhua Wang, JasonKuen, LianyangMa	May, 2018	Recent advances in CNN	The study focused on CNN architecture advancements but lacked clear explanations of their applications in various fields.
Ch. Wac. , Martin Reuter, Tassilo Klein	April, 2018	Deep CNN for segmenting neuroanatomy	The paper's lack of in-depth comparison of CNN architectures may restrict its real-world applications.

Nandini Sengupta, Md Sahidullah, Goutam Saha	Aug, 2016	Cepstral-based statistical features in Lung sound classification	The limited lung sound dataset used in the study may not cover all possible variations of lung sounds
J. Chen, Y. Gao, L. Li and Y. Zhang	2020	Deep Learning approach for diseases using X-ray images	The proposed approach only considers three diseases and may not detect other diseases
R. Kumar, V. Rathi, G. Kaur and A. Bala	2018	An ensemble approach for classification of diseases using machine learning techniques	Does not provide a detailed analysis of the feature selection process and the choice of hyperparameters for each algorithm

CNN in Lung sounds classification

D. Bardou, K. Zh., S. M. Ahmad[7], in 2018 they contrasted three machine learning methods for classifying lung sounds. The 3rd strategy is dependent on the creation of CNN, the starting 2 approaches are dependent on a collection of manually created trained features using 3 distinct algorithms(SVM, KNN, and Gaussian mixture models). To improve the final accuracy of the CNN, they also have evaluated augmentation in dataset methods over spectrograms. Accordingly CNN performed better than manually created feature-based classifiers.

The authors[7] provide details on the methodology used for creating and training the CNN, including the architecture of the CNN model, such as the number and type of layers, activation functions, and pooling strategies used. Authors also describe the process of collecting and preprocessing the dataset used for training and evaluation.

The authors[7] go through the usage of spectrogram data augmentation approaches to increase CNN accuracy. Authors also describe the specific data augmentation methods used, such as flipping, rotating, or scaling the spectrograms, and how these techniques were incorporated into the training process of the CNN.

Recent advances in CNN

Jiuxiang Gu, Zhenhua Wang, JasonKuen, LianyangMa, Amir Sh., B. Shuai, T. Liu, X. Wang,G. Wang, in 2018 they demonstrated CNN's most current development. The authors discuss how CNNs have improved in terms of layer architecture, loss function, activation function, regularization, optimization, and computational efficiency. Authors provide examples of CNNs applied to various domains, including voice, natural language processing (NLP), and computer vision.

In terms of layer architecture, the authors discuss the use of residual connections, dense connections, and multi-scale feature fusion to improve the performance of CNNs. Authors also discuss the use of different loss functions, such as focal loss and dice loss, which can be used to handle class imbalance and improve segmentation accuracy.

In terms of activation functions, the authors presented the use of rectified linear units (ReLU), leaky ReLU, and parametric ReLU, as well as more recent activation functions such as swish and mish. They also discuss the use of different regularization techniques, such as dropout, batch normalization, and weight decay, which can prevent overfitting and improve the generalization performance of CNNs.

The authors also presented recent developments in optimization algorithms for CNNs, including the use of adaptive learning rate methods such as Adam and RMSprop. They also discuss the use of more efficient convolution operations, such as depthwise separable convolution and dilated convolution, which can reduce the computational cost of CNNs.

Deep CNN for segmenting neuroanatomy

Christian Wachinger , M. Reuter, Tassilo Klein[8], in 2018 for segmenting neuroanatomy, they offer a Deep Convolutional Neural network. 3 convolutional layers with normalization, non-regularities, and pooling are used, for fully connected layers.

The authors[8] provide details on the methodology used for creating and training the CNN, including the architecture of the CNN model, such as the number and type of layers, activation functions, and pooling strategies used. The authors[8] explore how to increase the accuracy of the CNN by using data augmentation techniques on spectrograms.

Authors[8] also describe the specific data augmentation methods used, such as flipping, rotating, or scaling the spectrograms, and how these techniques were incorporated into the training process of the CNN.

Image recognition with Deep residual learning

X. Zhang, K. He, and Shaoqing Ren introduce a new architecture called deep residual learning for improving the performance of convolutional neural networks (CNNs) in image recognition tasks. The authors propose a recurrent learning approach that builds on top of the previously used deep networks.

They demonstrate the effectiveness of their approach on the CIFAR-10 dataset with a network consisting of 1000 layers.

One of the challenges in visual recognition tasks is the depth of representations in deep networks, which can lead to degradation in performance due to the vanishing gradient problem. The authors address this issue by introducing skip connections, which allow information to bypass certain layers and be directly fed to deeper layers. This helps to mitigate the degradation problem and enables the training of very deep neural networks.

The authors evaluate their deep residual learning architecture on the COCO (Common Objects in Context) dataset and report a significant improvement of 28% over the performance of networks without skip connections. This demonstrates the effectiveness of their approach in achieving better representations with deep neural networks.

Cepstral-based statistical features in Audio of Lungs

Md Sahidullah, Nandini Sengupta[10], in 2018 they suggest a new set generated using the characteristics in cepstral for quick and effective categorization. The authors conducted experiments using a dataset of 30 subjects and an artificial neural network (ANN) for classification.

The authors[10] found that their proposed statistical features derived from cepstral analysis outperformed conventional cepstral coefficients and wavelet-based features, including the commonly used mel-frequency cepstral coefficients (MFCCs), in terms of both robustness and recognition accuracy. This is particularly important for lung sound recordings, which are often of low quality due to background noise and interference.

The authors[10] also noted that their proposed features showed better performance even in low signal-to-noise ratios, which is important for the accurate classification of lung sounds in noisy environments. The results of this study suggest that cepstral-based statistical features may be a promising approach for the analysis and classification of lung sounds.

Deep Learning approach for diseases using X-ray images

J. Chen, Y. Gao, L. Li and Y. Zhang[17], in 2020 authors introduce a new hybrid deep learning technique that uses both CNNs and RNNs for classifying respiratory diseases based on X-ray images. The CNNs extract features from the images, while the RNNs model the temporal dependencies among the features. The proposed approach is evaluated on a dataset containing X-ray images of patients with pneumonia, tuberculosis, and lung cancer. The results indicate that the proposed approach outperforms several existing deep learning techniques, achieving an accuracy of 89.75%. The paper also presents an in-depth analysis of the proposed approach's performance, including an ablation study that examines the contribution of each component.

Based on the findings, the authors[17] conclude that the proposed approach is an efficient and effective method for classifying respiratory diseases using X-ray images. The approach has the potential to be utilized in clinical settings to help doctors in diagnosing and treating respiratory diseases.

Classification of respiratory diseases using machine learning techniques

R. Kumar, V. Rathi, G. Kaur and A. Bala[18], in 2018 authors[18] introduce an ensemble approach combining decision tree, random forest, support vector machine (SVM), and k-nearest neighbors (KNN) algorithms to enhance the classification accuracy. The study uses a dataset of respiratory sounds recorded from patients with asthma, bronchiectasis, pneumonia, and chronic obstructive pulmonary disease (COPD).

The experimental results demonstrate that the proposed ensemble approach outperforms the individual machine learning algorithms with an overall accuracy of 94.7%. The authors provide a detailed analysis of the performance of each algorithm in the ensemble and compare the results with previous studies. They suggest that the ensemble approach can effectively and accurately classify respiratory diseases using machine learning techniques, which can be beneficial for medical professionals in the diagnosis and treatment of respiratory diseases.

Chapter 03: SYSTEM DEVELOPMENT

3.1 Dataset used in Respiratory Diseases classification

ICBHI[13] 2017 hosted a sci. challenge, and here is where the Respiratory Sound database was first created. The current version of this database, which includes private and public dataset of ICBHI[13], is made openly for study.

The database includes audio recordings that were collected in many years by distinct nations. The majority of the database is made up of audio samples that were captured by the University of Aveiro in Portugal. The 2nd study by the Aristotle University of Thessaloniki recorded respiratory sounds at the Govt. Hospital of Imathia , Greece.

It includes 920 audio samples of 126 patients, recording of 5.5 hours with 6898 breathing, of which 886 have wheezes, 1864 have crackles, and 506 have both.

Experts of respiratory biology annotated the cycles as having wheezes and crackles, or no accidental noises. These were made with various tools, and they ranged in length from 10 to 90 seconds. It also includes the location in the chest where these were made. Some respiration cycles have high noise levels that mimic real-world settings.[13]

Each file name is composed of five components.:

1. Recording index

2. Patient number (101,102,...,226)

3. Chest location

   a. Lateral right (Lr)
   b. Lateral left (Ll)
   c. Posterior right (Pr)
   d. Trachea (Tc)

e. Anterior left (Al)
f. Anterior right(Ar)
g. Posterior left (Pl)

4. Recording equipment

a. WelchAllyn Meditron Master Elite Electronic Stethoscope (Meditron)
b. 3M Littmann 3200 Electronic Stethoscope (Litt3200)
c. 3M Littmann Classic II SE Stethoscope (LittC2SE)
d. AKG C417L Microphone (AKG C417L)

5. Acquisition

a. Simultaneous/multichannel (mc)
b. Sequential/single channel (sc)

These having 4 columns :

1. Presence/absence of wheezes (presence=1, absence=0)

2. Presence/absence of crackles (presence=1, absence=0)

3. Beginning of respiratory cycle(s)

4. End of respiratory cycle(s)

The abbreviations are used :

1. LRTI

2. COPD

3. URTI

3.2 STFT

A window's Fourier transform is discovered using the STFT. We discover that the common Fourier transform gives us frequency information for the whole signal. However, with STFT, we continue to move the window while locating the Fourier transform across it. For signals with time-varying frequency components, STFT gives us the time-frequency information. STFT is defined as:

$$F(W,t) = \int_{-\infty}^{\infty} f(\tau)h(\tau - t)e^{-j\omega\tau}d\tau$$

Where

h (t) = hamming window

w = frequency

t = time

f (t) = input signal having STFT to be calculated

3.3 Compute STFT

1. Choose a window with a finite length.

2. Window at time t = 0.

3. Determine that segment's Fourier Transform.

4. Now, slowly move the window.

5. Return to step 3 till we reach the signal's conclusion.

3.4 Size of Window

- The window has to be narrow so that the part of the audio that falls within it remains stationary.
- An extremely small window does not always enable adequate frequency domain localisation.
- A broad window aids in offering poor temporal resolution but strong frequency resolution.
- The size of the window is normally a power of two. A window size of 32 samples is approximately 0.0007 seconds if your sampling rate is

44,100 samples per second.

3.5 Why are we using STFT?

The FFT of a Stationary Signal clearly identifies the 3 components that exist but the FFT of a Non-Stationary audio signal just identifies the frequencies that are present; the location of these frequencies in time is not disclosed. Therefore, as a benefit to Fourier, we have employed the Short Time Fourier Transform in this study. By splitting the larger time signals into shorter, equal-length parts, the STFT may be determined. Then, for each brief piece, the Fourier Transform is calculated. Each Fourier Transform gives time and frequency information together with the spectrum information for that segment.

3.6 Limitation of FFT

- The primary constraint of the FFT is that it cannot represent both time and frequency at the same time.
- The FFT cannot be used for nonstationary signals.
- FFT is ineffective for representing discontinuities.

3.7 Stationary and Non-Stationary Signals :

A great illustration of a stationary signal is a sine wave, its frequency component is not affected by the passage of time. Stationary signals are ones whose components of frequency don’t change w.r.t time.

Signals that shift in frequency over time, such as a voice signal, are considered non-stationary signals.

3.7.1 Stationary signal audio signal

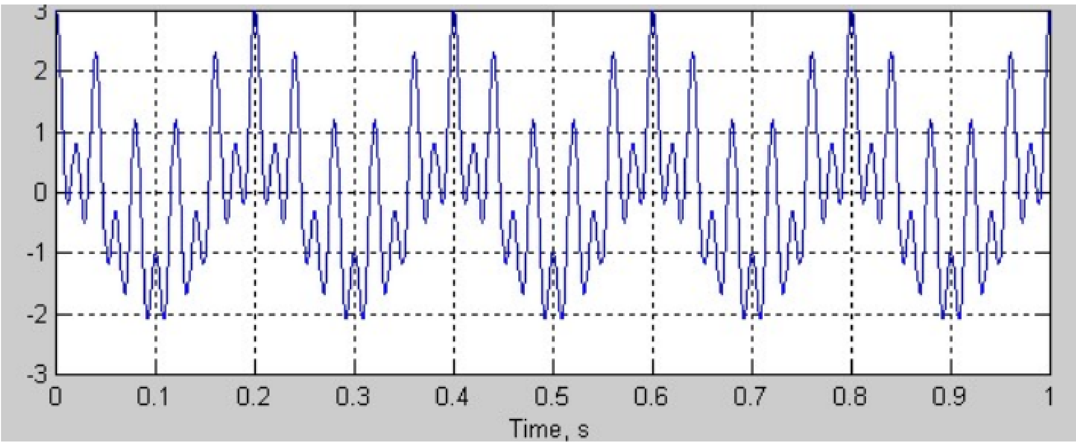

Fig. 3.1 Stationary Signal

3.7.2 Non-stationary audio signal

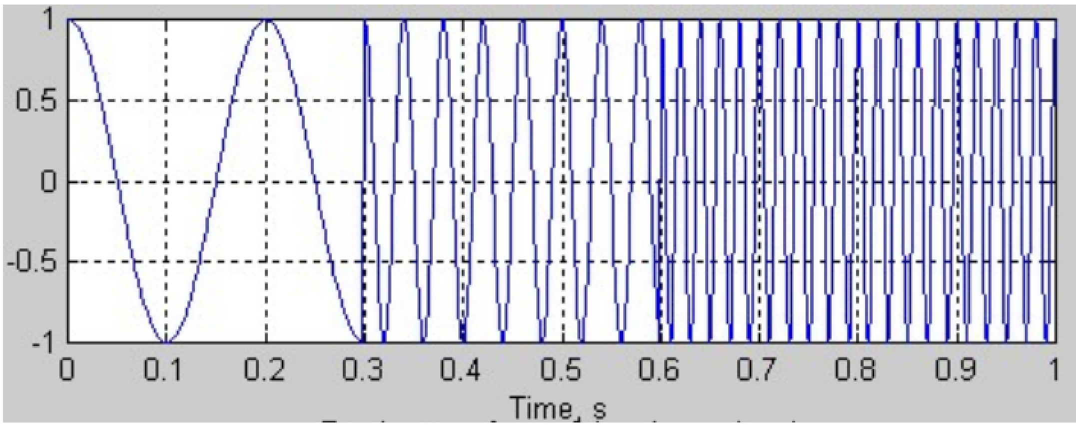

Fig. 3.2 Non-stationary signal

3.8 Librosa

Librosa is a Python library that is commonly used for audio and music signal processing tasks. It provides a wide range of tools and functions for various audio analysis tasks, including feature extraction, time-series analysis, and visualization. Librosa is widely used in the field of audio processing and music information retrieval (MIR) due to its user-friendly interface and extensive functionality.

Librosa provides an easy-to-use API for loading, manipulating, and analyzing audio signals in different formats, such as WAV, MP3, and FLAC. It includes various tools for spectral analysis, such as computing spectrograms, chromatograms, and mel-frequency cepstral coefficients (MFCCs), which are commonly used features for audio classification and recognition tasks. Librosa also offers functions for beat tracking, pitch estimation, and tempo analysis, making it a versatile tool for a wide range of audio analysis tasks.

One of the key features of Librosa is its integration with popular machine learning libraries such as NumPy and SciPy, making it easy to combine audio analysis tasks with machine learning algorithms for tasks such as audio classification, genre recognition, and music recommendation. Overall, Librosa is a powerful and widely used tool for audio signal processing and feature extraction in Python, making it a popular choice among researchers and practitioners in the field of audio processing and music information retrieval.

3.9 Keras

Keras is an open-source deep learning library written in Python that provides a high-level interface for building and training neural networks. It is widely used in the field of machine learning and deep learning due to its user-friendly and intuitive API, which allows researchers and developers to easily create complex neural network models with minimal coding efforts.

Keras provides a wide range of pre-processing functions for data preparation, such as data normalization, image augmentation, and sequence padding. It also offers a diverse collection of pre-defined neural network layers, including convolutional layers, recurrent layers, and dense layers, which can be easily stacked to build complex neural network architectures.One of the key features of Keras is its ability to run on top of various deep learning backends, including TensorFlow. This allows users to seamlessly switch between different backend engines without changing their code, making it a flexible choice for deep learning tasks across different platforms.

Keras also provides a rich set of tools for model evaluation, including built-in support for model training and validation, model checkpointing, and model visualization. It also supports various advanced features, such as transfer learning, model ensembling, and custom loss functions, which make it a powerful tool for deep learning research and applications.

3.10 Neural Network

Neural networks, also known as artificial neural networks or simply "neural nets", are a type of machine learning model that is inspired by the structure and function of the human brain. Neural networks are composed of interconnected nodes or neurons organized in layers, and they are designed to learn complex patterns and representations from input data through a process called training.

In a neural network, input data is passed through an input layer, then processed through one or more hidden layers, and finally output through an output layer. Each neuron in the network receives input from the neurons in the previous layer, processes it using an activation function, and produces an output. The outputs of the neurons in the output layer represent the predictions or classifications made by the model.

During training, the neural network learns to adjust the weights and biases associated with each neuron in order to minimize the difference between its predictions and the actual target values. This is typically done using optimization algorithms and loss functions that quantify the error between predicted and actual values. The training process iteratively updates the weights and biases to improve the model's performance until a certain stopping criterion is met.

Neural networks have been widely used in various domains, including computer vision, natural language processing, speech recognition, and many other applications. They are capable of learning complex patterns and representations from large amounts of data, and can achieve high levels of accuracy in tasks such as image recognition, speech recognition, and language translation. However, neural networks can also be computationally expensive and require large amounts of training data to achieve optimal performance. Proper hyperparameter tuning, regularization techniques, and model evaluation are important considerations when working with neural networks.

There are many types of neural networks, including feedforward neural networks (also known as multi-layer perceptrons or MLPs), convolutional neural networks (CNNs) for image processing, recurrent neural networks (RNNs) for sequential data, long short-term memory (LSTM) networks for handling temporal dependencies, and many others. The choice of neural network architecture depends on the specific problem and data characteristics, and experimenting with different architectures can be an important part of developing an effective neural network model.

A neuron is the most fundamental computing unit in the human brain. The human nervous system has around 86 billion neurons, which are linked by approximately 10^{14} - 10^{15} . Figure depicts real left neuron and right neuron.

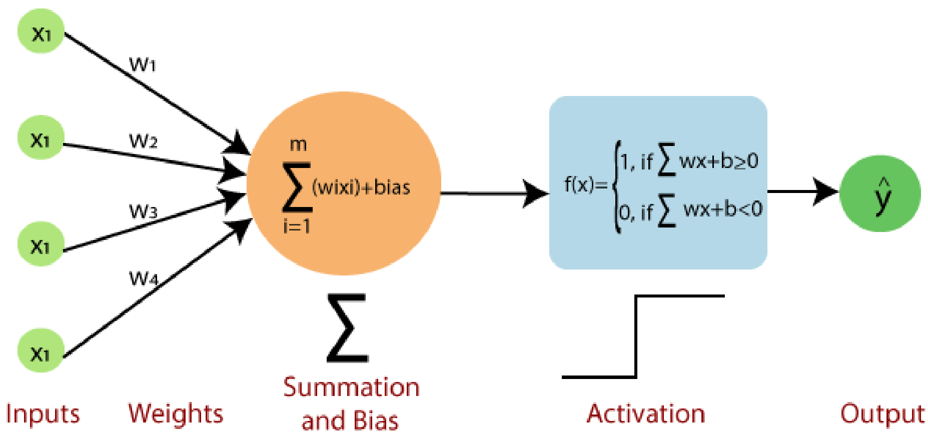

Fig. 3.3 Mathematical model of Neural Network

3.11 How does the Neural Network work ?

1. Input Nodes : The first layer of a Neural Network contains input nodes. There is no computation in this layer. This layer simply forwards the information provided by the user to the next layer.

2. Hidden Nodes : the 2, i/p and o/p layers are hidden nodes. This layer handles intermediate processing. It passes the weights from the i/p layer to next after calculating data.

3. Output Node : The Output layer contains Output Node. In this layer, we employ an activation function with bounds. Data then transferred to the o/p.

4. Weights and Connections: It is made up of different connection among synapses and Neurons, with every simply transferring the o/p of neuron *i* to the i/p of neuron *j*. *i* is the predecessor of *j*. All of these links have a weight *Wij* assigned to it.

5. Activation function : It appears only before the o/p . It simply determines the o/p of a node based on the i/p. The activation functions of a conventional chip circuit of a network are "on" and "off", based on the i/p. This behavior is remarkably same to the perceptron.

6. Learning Rule : It updates its arguments so that it can create a desired o/p from a given set of i/p. Weights and thresholding are modified during this learning process.

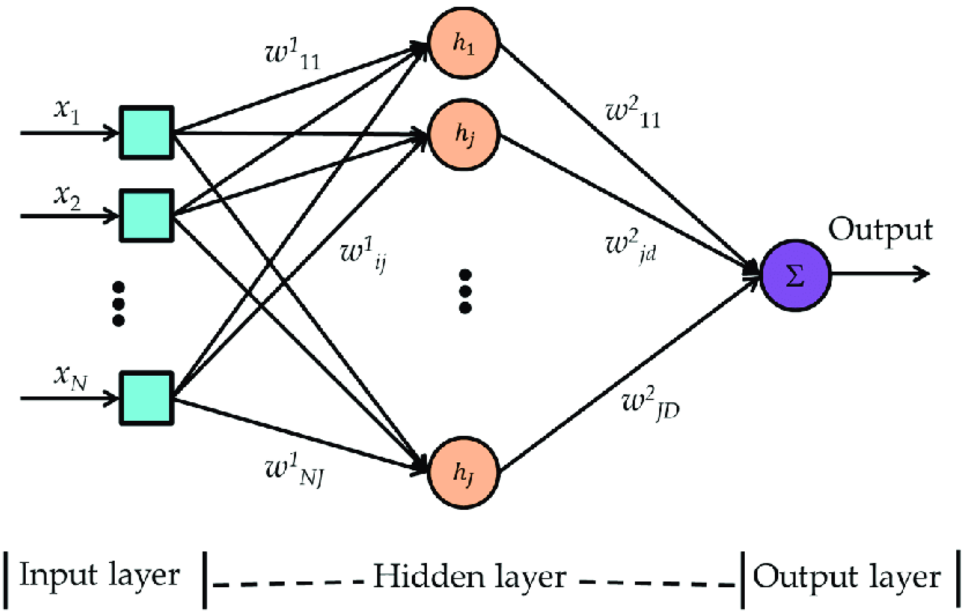

Fig. 3.4 Neural Network Layers

3.11.1 Forward Propagation

Forward propagation is the process in which an input signal is passed through a neural network to produce an output signal. It is the first step in the training and prediction process of a neural network.During forward propagation, the input data is first passed through the first layer of the neural network, where each neuron applies a weighted sum of its inputs and an activation function to produce an output signal. This output signal is then passed as input to the next layer, and the process is repeated until the final layer of the neural network is reached.

The output signal produced by the final layer of the neural network represents the network's prediction or classification for the given input. The process of forward propagation can be represented mathematically as a series of matrix multiplications and non-linear transformations.

The weights and biases of the neural network are initially randomly initialized and are updated during training using backpropagation, which uses the difference between the predicted output and the actual output to adjust the weights and biases of the network in order to minimize the error. Data at the input layer is multiplied with the corresponding weights of the input layer .

- a1 = (b1 * w1) + (b2 * w1)
- a2 = (b1 * w2) + (b2 * w2)
- a3 = (b1 * w3) + (b2 * w3)

Neurons can fire a pulse if the o/p's final total greater than a predetermined threshold value; otherwise, it can suppress the pulse.

- x1 = fn(a1 , a2, a3)

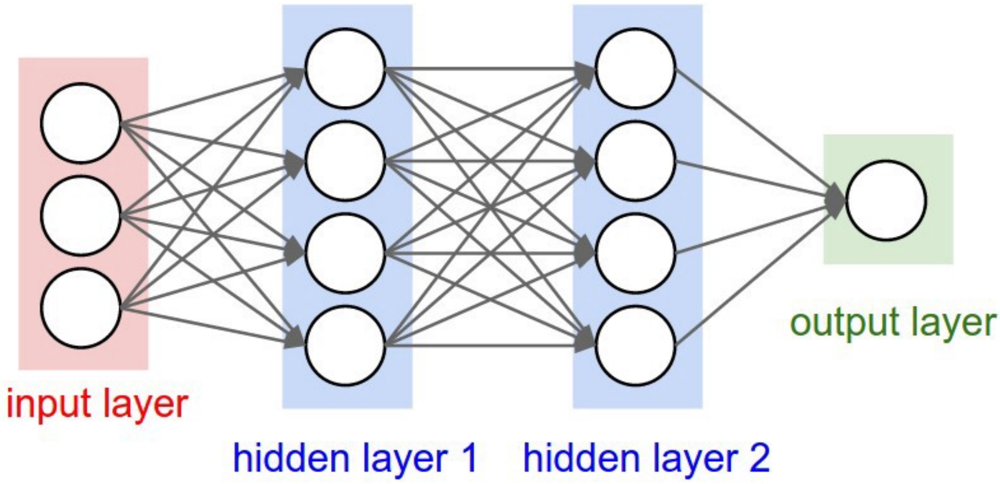

Fig. 3.5 Forward Propagation

3.11.2 Backward Propagation

Backpropagation is a technique used to train neural networks by adjusting the weights and biases of the network in order to minimize the difference between the predicted output and the actual output. It is based on the chain rule of calculus and works by propagating the error backwards through the network from the output layer to the input layer.

During backpropagation, the difference between the predicted output and the actual output is computed and propagated backwards through the network. The weights and biases of each neuron are then adjusted based on their contribution to the overall error, with the goal of reducing the error in subsequent iterations.

The specific algorithm for computing the weight and bias updates during backpropagation involves computing the gradient of the error with respect to each weight and bias in the network, and using this gradient to update the weight and bias values. This process is repeated for each training example in the dataset until the error is minimized.

Backpropagation is a crucial component of training neural networks, as it allows the network to learn from its mistakes and make more accurate predictions over time. It is an iterative process that requires careful tuning of hyperparameters such as learning rate and regularization strength to ensure the network converges to a good solution.

Final result :

- w 1 = w1 - (a * d(err) / d(w1))
- w2 = w2 - (a * d(err) / d(w2))
- w3 = w3- (a * d(err) / d(w3))

Model uses an algorithm known as the gradient descent approach to identify the minimal value of the error. Weights which minimizes error are therefore regarded learning issue solution.Gradient descent may be defined as an optimization technique for iteratively minimizing the advancing in the local minima direction based on the computation of negative Gradient.

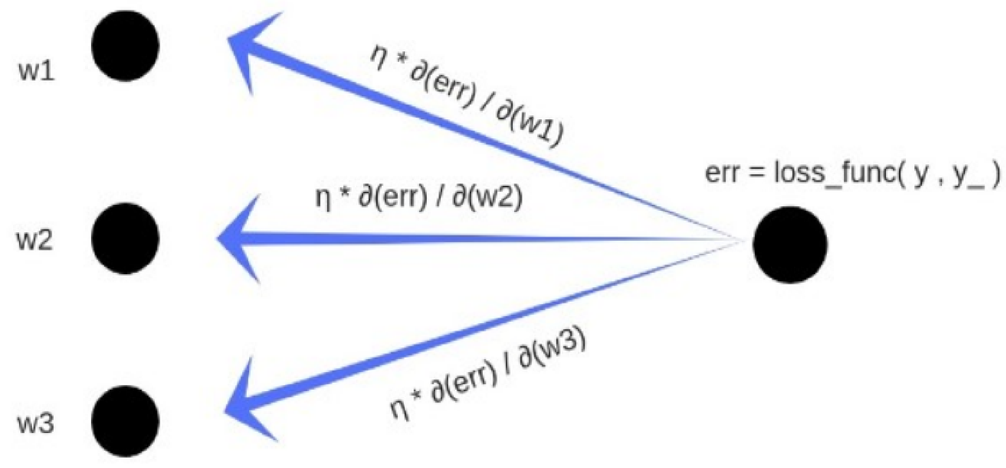

Fig. 3.6 Backward Propagation

3.12 Types of Neural Network

Two types ::

1. Multi layer Perceptron

2. Single layer Perceptron

3.12.1 Single Layer Perceptron

The Single Layer Perceptron (SLP) is a type of artificial neural network that consists of a single layer of interconnected neurons, also known as nodes or perceptrons. It is one of the simplest forms of neural networks, often used for binary classification tasks.

In an SLP, each input feature is connected to a corresponding node in the input layer, and each node in the input layer is connected to a single output node in the output layer. The output node computes a weighted sum of the input features and applies an activation function to produce the final output, which is typically a binary value (e.g., 0 or 1) representing the predicted class label.

During training, the SLP uses a learning algorithm to alter the weights of the connections between the input features and the output node, such as the perceptron learning rule or the delta rule. These weight modifications are based on the difference between the projected and actual target outputs, with the objective of minimizing the error and increasing the network's prediction accuracy.

The SLP has limitations in its ability to model complex nonlinear relationships in data, as it can only learn linear decision boundaries. However, it can be used for simple binary classification tasks where the data is linearly separable. The SLP is often used as a basic building block in more complex neural network architectures, such as multi-layer perceptrons (MLPs), which can model more complex patterns in data.

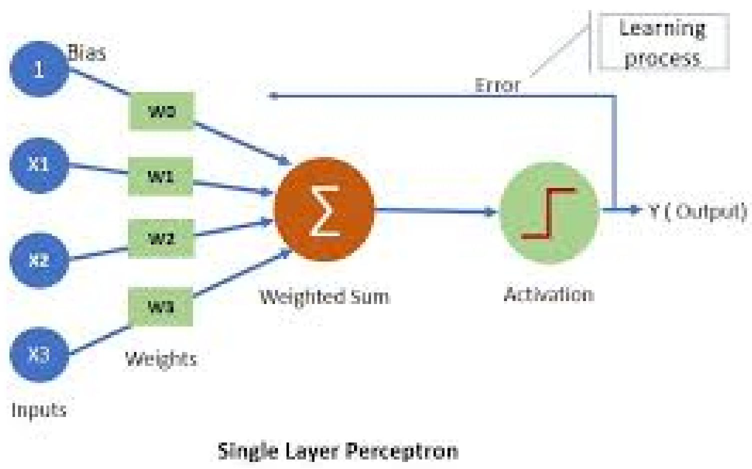

Fig. 3.7 Single Perceptron

3.12.2 Multi Layer Perceptron

The Multi-Layer Perceptron (MLP) is a type of artificial neural network that consists of multiple layers of interconnected neurons, also known as hidden layers, in addition to an input layer and an output layer. The MLP is a feedforward neural network, where the signals flow in one direction, from the input layer through the hidden layers to the output layer, without any loops or feedback connections. Each neuron in the MLP applies a weighted sum of its inputs, passes the result through an activation function, and produces an output that serves as input to the next layer.

The MLP can learn complicated patterns and provide predictions for a variety of applications, including classification, regression, and time-series forecasting. In an MLP, the learning process involves modifying the weights and biases of the neurons depending on the difference between the expected and actual output, a process known as supervised learning. Sigmoid, tanh, and ReLU are examples of common activation functions used in MLPs. Additionally, the MLP can have varying numbers of neurons and hidden layers, allowing for flexibility in designing the network architecture.

The MLP is a widely used type of neural network due to its ability to learn non-linear relationships in data and its versatility in handling a wide range of tasks. However, it also has some limitations, such as the potential for overfitting, the need for careful tuning of hyperparameters, and the possibility of getting stuck in local optima during training. Nonetheless, the MLP serves as a fundamental building block in many advanced neural network architectures and has been successfully applied in numerous applications, including image and speech recognition, natural language processing, recommendation systems, and financial prediction, among others.

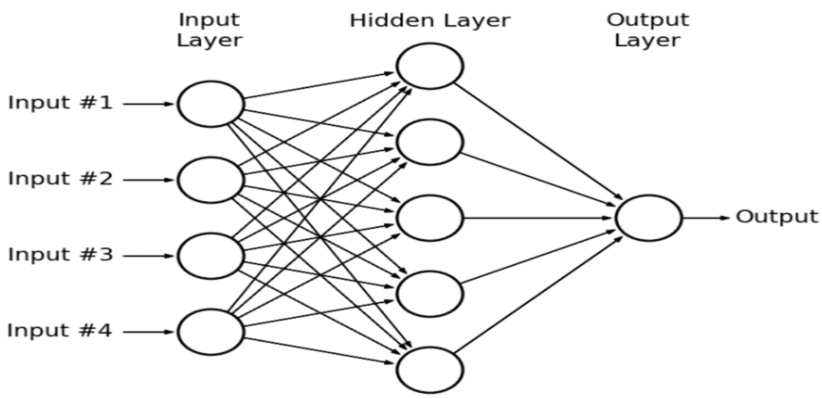

Fig. 3.8 Multi Layer Perceptron

3.13 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a type of artificial neural network that are designed for processing data with a grid-like structure, such as images, videos, and audio signals. CNNs are specifically designed to automatically learn and extract relevant features from the input data through a series of convolutional, pooling, and fully connected layers.

The key feature of CNNs is the convolutional layer, which applies filters (also known as kernels or feature detectors) to the input data in a localized manner, capturing local patterns or features. The output of the convolutional layer is then passed through a pooling layer, which reduces the spatial dimensions of the feature maps while preserving the most important information. This pooling process helps to reduce the computational complexity of the network and make it more robust to small translations or distortions in the input data.

After the convolutional and pooling layers, the extracted features are flattened and passed through one or more fully connected layers, which are similar to those in traditional feedforward neural networks. The fully connected layers learn the high-level representations and make the final predictions or classifications based on the learned features.

CNNs have been highly successful in a wide range of tasks, such as image classification, object detection, facial recognition, speech recognition, and natural language processing, among others. They have surpassed human-level performance in many image recognition tasks and are widely used in state-of-the-art deep learning models. CNNs have revolutionized fields such as computer vision, and they continue to be an active area of research and development in the field of artificial intelligence.

To understand CNN we go with Convolution and Pooling.

3.13.1 Convolution

Convolution is the process of applying filters (also known as kernels or feature detectors) to the input data in a localized manner. The filter is typically a small matrix that is convolved (element-wise multiplied and summed) with a corresponding patch of the input data at a time. The filter acts as a feature detector and captures local patterns or features, such as edges, corners, and textures, from the input data. The convolution operation allows the network to learn meaningful representations of the input data, and multiple filters can be applied in parallel to learn different features.

It works on 2 signals: one as the "i/p " signal, and the other as a "filter" in the i/p image, generating the o/p image (so convolution takes 2 images as i/p and produces a 3rd as o/p). It takes an i/p audio and filters it, basically multiplying by i/p audio by kernel to produce the modified audio. In this, it is simpler to imagine a sliding kernel through a whole img. , altering the pixels in process.

$$x_{ij}^{l} = \sum_{a=0}^{m-1}\sum_{b=0}^{m-1} \omega_{ab} y_{(i+a)(j+b)}^{l-1}$$

3.13.2 Pooling

Pooling is the technique of reducing the spatial dimensionality of feature maps generated by convolutional layers. Pooling is often accomplished through the use of procedures such as max pooling and average pooling. The largest value from a local region in the feature map is selected by max pooling, whereas the average value is calculated by average pooling. Pooling reduces the spatial dimensions of the feature maps, making the network more computationally efficient and resilient to minor translations or distortions in the input data. Pooling also helps to capture the most important information from the local regions, preserving the relevant features while reducing the spatial resolution.

It is a procedure that uses samples. Pooling is classified into two types: maximum pooling and minimum pooling. As the name implies, max pooling selects the largest value from the specified region, whereas min pooling selects the least value.

A CNN is essentially a DNN with layers that are hidden that have Pooling and Convolution that supplement the activation function and increase non-linearity.

3.14 Optimizers

Optimizers are algorithms used in machine learning and deep learning to optimize the parameters of a model during the training process. They play a crucial role in improving the efficiency and effectiveness of the model training process, by finding the optimal values of the model's parameters that minimize the loss function.

Loss here, which indicates how weak our model is. Using this loss to train our model. Essentially, for minimizing we use loss. It is the process to reduce any statement.

These used to modify the neural network's properties, such as its Learning Rate and Weights to reduce losses. In order to solve these problems, optimizers use function minimization. These have been researched in recent years, having their own pros and cons.

The various types of optimizers :

1. Mini Batch Stochastic Gradient Descent(MB-SGD)
2. Nesterov Accelerated Gradient(NAG)
3. Stochastic Gradient Descent(SGD)
4. Adaptive Gradient(AdaGrad)
5. SGD with momentum
6. Gradient Descent
7. AdaDelta
8. RMSprop
9. Adam

Deep learning frequently employs gradient descent-based optimizers such as stochastic gradient descent (SGD), mini-batch gradient descent, and batch gradient descent. The parameters of the model are updated by these optimizers based on the gradients of the loss function with respect to the parameters. They try to minimize the loss function by adjusting the parameters in the direction of the negative gradient. Gradient descent-based optimizers are relatively simple and computationally efficient, but they may converge slowly and get stuck in local optima.

Adaptive optimizers, such as Adagrad, RMSprop, and Adam, modify the learning rate for each parameter depending on past gradients or other data. These optimizers can dynamically adjust the learning rate during training, which can help accelerate the convergence and improve the model's performance. Adaptive optimizers are commonly used in deep learning as they are more robust to variations in the data and can handle different scales of gradients.

Stochastic optimizers, such as stochastic gradient descent with momentum, Nesterov accelerated gradient (NAG), and RMSprop with momentum, incorporate momentum or velocity-based updates to the gradient descent algorithm. These optimizers accumulate past gradients to update the parameters, allowing for faster convergence and better handling of noisy gradients.

Chapter 04: PERFORMANCE ANALYSIS

4.1 ICBHI Dataset

ICBHI 2017 hosted a sci. challenge, and here is where the Respiratory Sound database was first created. The current version of this database, which includes private and public dataset of ICBHI, is made openly for study.

The collection contains audio recordings gathered over many years by several nations. The vast part of the collection consists of audio recordings collected by the University of Aveiro in Portugal. The second research, conducted by Aristotle University of Thessaloniki, captured respiratory sounds at the Imathia Government Hospital in Greece.

It includes 920 audio samples of 126 patients, recording of 5.5 hours with 6898 breathing, of which 886 have wheezes, 1864 have crackles, and 506 have both.

Experts of respiratory biology annotated the cycles as having wheezes and crackles, or no accidental noises. These were made with various tools, and they ranged in length from 10 to 90 seconds. It also includes the location in the chest where these were made. Some respiration cycles have high noise levels that mimic real-world settings.

The abbreviations are :

1. LRTI

2. COPD

3. URTI

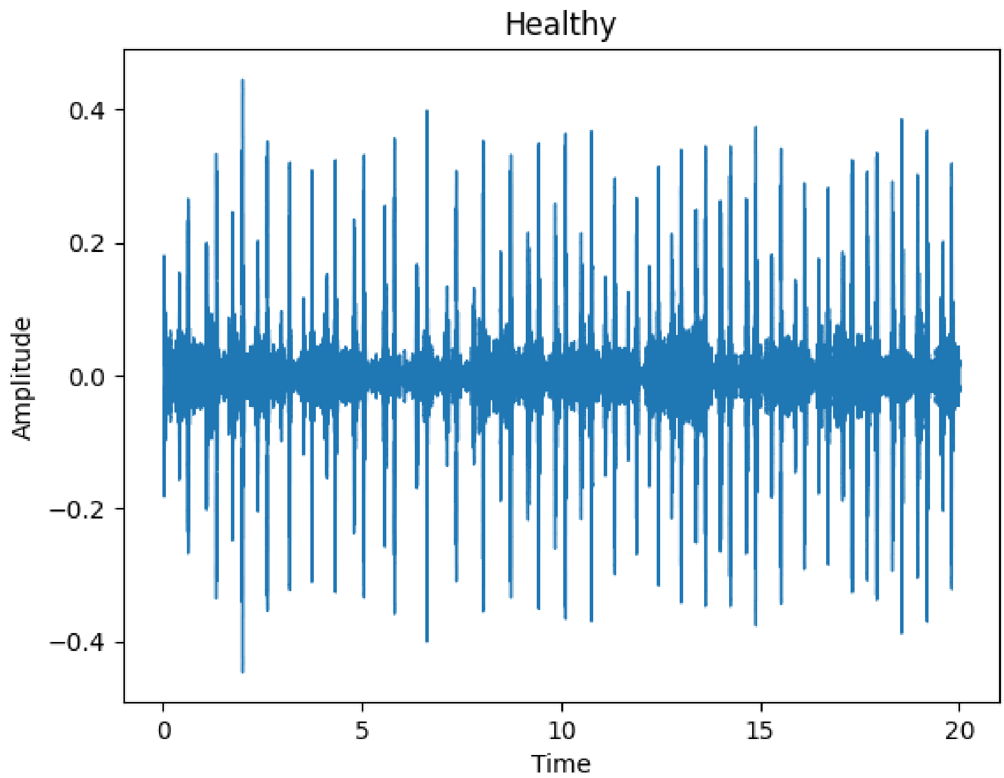

Fig. 4.1 Web plot of Healthy Person

4.2 Web Plot of different Disease

1. COPD
2. URTI
3. Pneumonia
4. Bronchiectasis
5. Bronchiolitis

4.2.1 COPD

COPD is a chronic respiratory disease that affects the lungs, causing breathing difficulties due to the narrowing of the airways and damage to the lung tissue. Cigarette smoking is the major cause of COPD, although other factors such as air pollution, occupational exposure to dust and chemicals, and genetic susceptibility can also contribute to the disease's development.

The main symptoms of COPD include shortness of breath, coughing, wheezing, and chest tightness, which typically worsen over time. COPD is a progressive disease, meaning that it gradually worsens and can lead to severe disability or death if left untreated.

COPD is typically diagnosed through a combination of medical history, physical examination, lung function tests, and imaging tests. Treatment for COPD aims to relieve symptoms, slow the progression of the disease, and improve quality of life. This may involve lifestyle changes, medications, oxygen therapy, pulmonary rehabilitation, and, in severe cases, surgery. Quitting smoking is essential in the management of COPD, as it can help slow the progression of the disease and reduce the risk of complications.

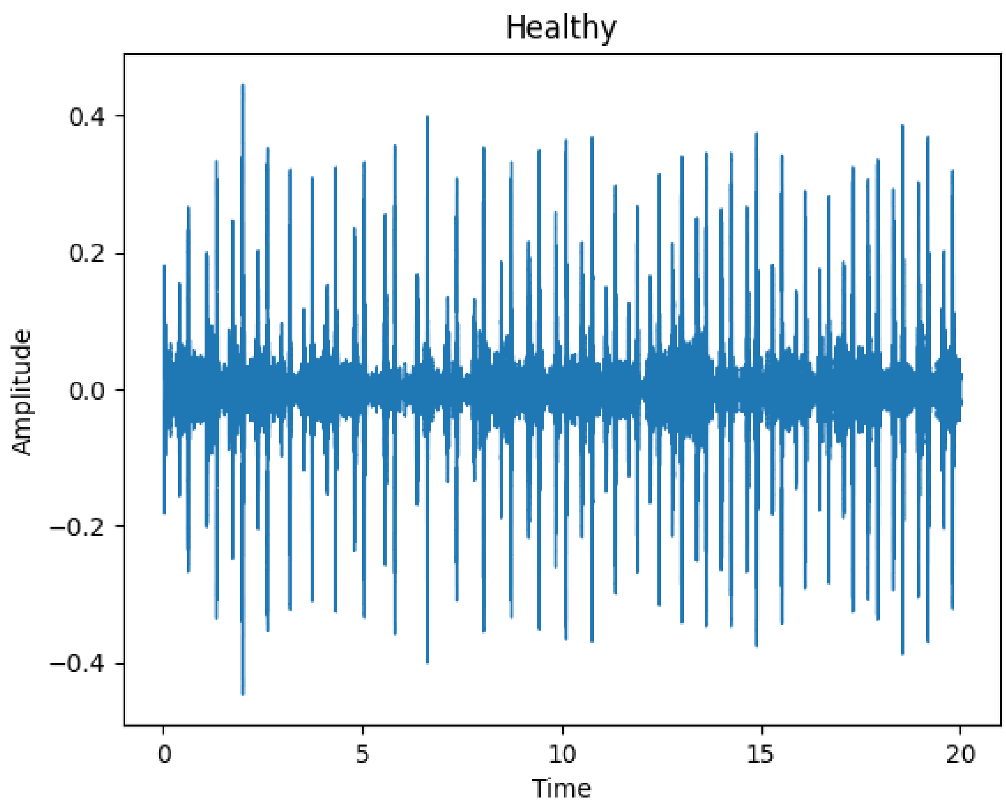

Fig. 4.2 Web Plot of COPD

4.2.2 URTI

URTI stands for Upper Respiratory Tract Infection, which is a common medical condition affecting the upper respiratory tract, including the nasal passages, sinuses, pharynx, and larynx. URTIs are typically caused by viral infections, such as the common cold, flu, or respiratory syncytial virus (RSV).

The symptoms of URTI can vary depending on the specific virus causing the infection, but commonly include a runny or stuffy nose, sore throat, cough, fever, and body aches. URTIs are highly contagious and can spread easily through contact with infected respiratory secretions, such as from coughing or sneezing.

Treatment for URTI typically focuses on relieving symptoms and may include over-the-counter pain relievers, decongestants, and cough suppressants. Antibiotics are generally not effective against URTIs, as they are caused by viruses rather than bacteria. Most cases of URTI will resolve on their own within a week or two, with rest and adequate hydration being the most important aspects of management. However, complications can occur, especially in individuals with weakened immune systems, such as the elderly or those with chronic medical conditions, and prompt medical attention may be necessary in such cases.

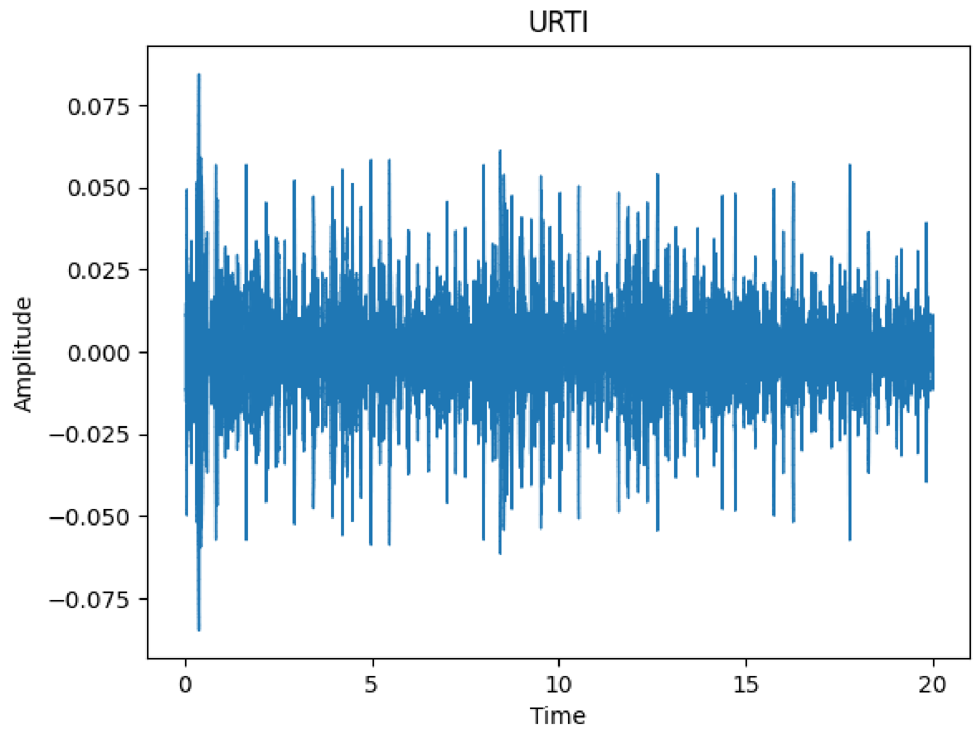

Fig. 4.3 Web Plot of URTI

4.2.3 Pneumonia

Pneumonia is a type of disease in which the lungs can be infected by various microorganisms, including bacteria, viruses, fungi, and parasites. It is a serious condition that can affect people of all ages but is most common in young children, older adults, and individuals with weakened immune systems.

The symptoms of pneumonia can vary, but commonly include cough, fever, chest pain, shortness of breath, fatigue, and muscle aches. In severe cases, individuals may develop difficulty breathing, confusion, or a bluish tint to their lips or nails due to lack of oxygen.

Diagnosis of pneumonia involves a combination of medical history, physical examination, chest x-rays, and laboratory tests. Treatment for pneumonia typically involves antibiotics for bacterial infections, antiviral medications for viral infections, and antifungal medications for fungal infections. Supportive care, including oxygen therapy and hydration, may also be necessary in severe cases.

Prevention of pneumonia can be achieved through vaccination against bacterial and viral pathogens, practicing good hygiene, and avoiding exposure to environmental irritants that can damage the lungs, such as cigarette smoke or air pollution.Prompt medical attention is crucial in the management of pneumonia, as complications can arise, such as lung abscesses or sepsis, which can be life-threatening.

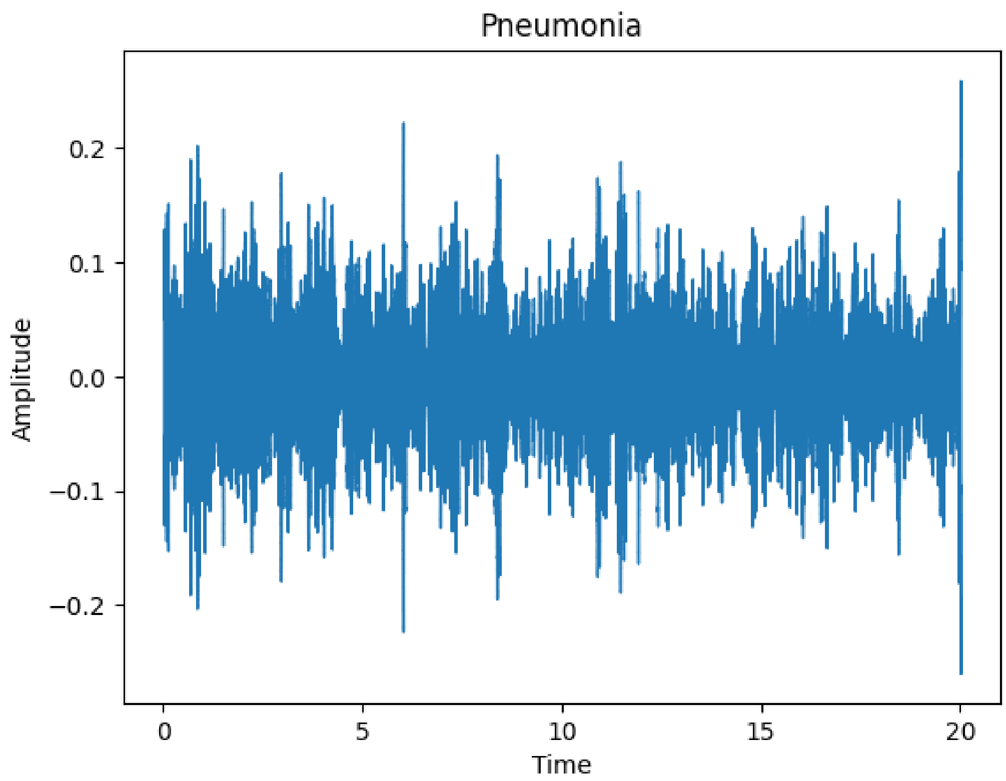

Fig. 4.4 Web Pot of Pneumonia

4.2.4 Bronchiectasis

Bronchiectasis is a chronic respiratory condition that involves permanent and abnormal dilation (widening) of the bronchi, the tubes that carry air to the lungs. This dilation can cause mucus to accumulate in the bronchi, leading to recurrent infections and damage to the lung tissue over time.

The most common cause of bronchiectasis is recurrent respiratory infections, although it can also be caused by genetic conditions, immune system disorders, or other underlying medical conditions. The primary symptoms of bronchiectasis include chronic cough, excessive mucus production, wheezing, shortness of breath, and chest pain.

Diagnosis of bronchiectasis typically involves a combination of medical history, physical examination, chest imaging tests, and lung function tests. Treatment for bronchiectasis aims to manage symptoms, prevent complications, and slow the progression of the disease. This may involve antibiotics to treat infections, bronchodilators to open up the airways, and expectorants to help loosen and clear mucus.

Other management strategies may include chest physiotherapy, pulmonary rehabilitation, and oxygen therapy. In severe cases, surgery may be necessary to remove damaged lung tissue or to address other complications. Bronchiectasis is a chronic condition that can significantly impact quality of life, so it is important for individuals with this condition to work closely with their healthcare providers to develop an individualized treatment plan.

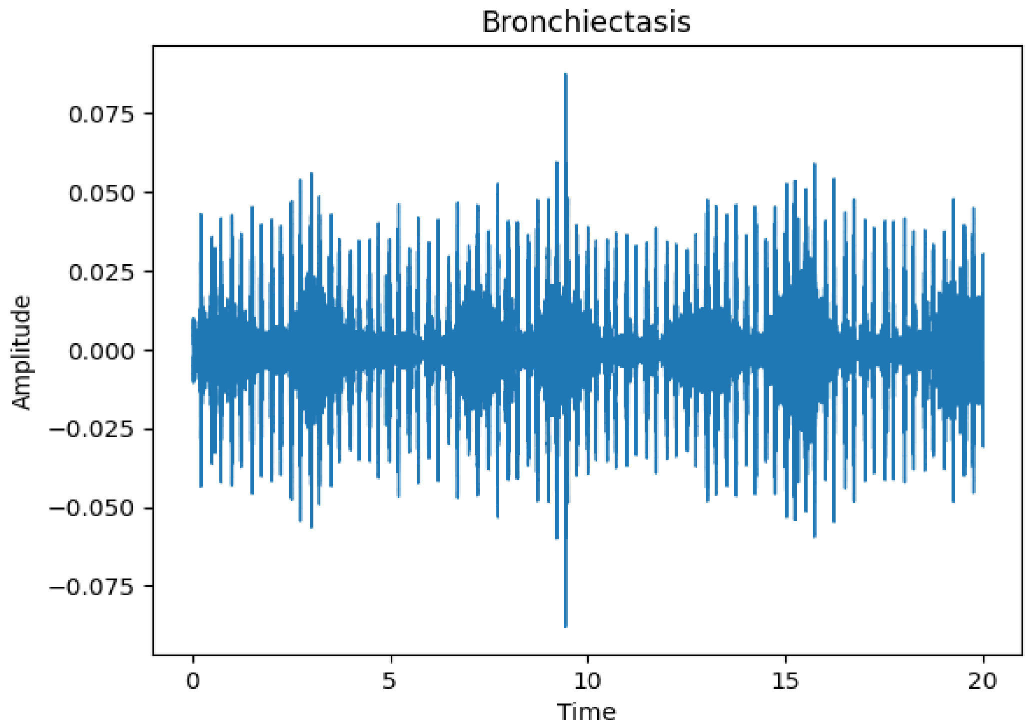

Fig. 4.5 Web Plot of Bronchiectasis

4.2.5 Bronchiolitis

Bronchiolitis is a respiratory condition that affects the small airways (bronchioles) in the lungs, primarily in infants and young children.It is often caused by a viral infection, most frequently respiratory syncytial virus (RSV), which can induce inflammation and blockage of the bronchioles, making air movement in and out of the lungs problematic.

The symptoms of bronchiolitis usually begin with mild cold-like symptoms, such as a runny nose, cough, and fever, which may progress to wheezing, difficulty breathing, rapid breathing, and poor feeding or lethargy in infants. In severe cases, bronchiolitis can lead to respiratory failure and the need for hospitalization.

Diagnosis of bronchiolitis can be treated by examining the medical history, physical examination, and sometimes laboratory or imaging tests. Treatment for bronchiolitis is typically supportive, aimed at relieving symptoms and preventing complications. This may include providing supplemental oxygen, administering medications to reduce inflammation and relieve bronchospasm, and ensuring adequate hydration and nutrition.

Most cases of bronchiolitis will resolve on their own within a few weeks, but some infants may be at increased risk of complications, such as those with underlying medical conditions or prematurity. Prevention of bronchiolitis can be achieved through good hygiene practices, such as frequent hand washing, avoiding contact with sick individuals, and vaccination against RSV in high-risk populations.

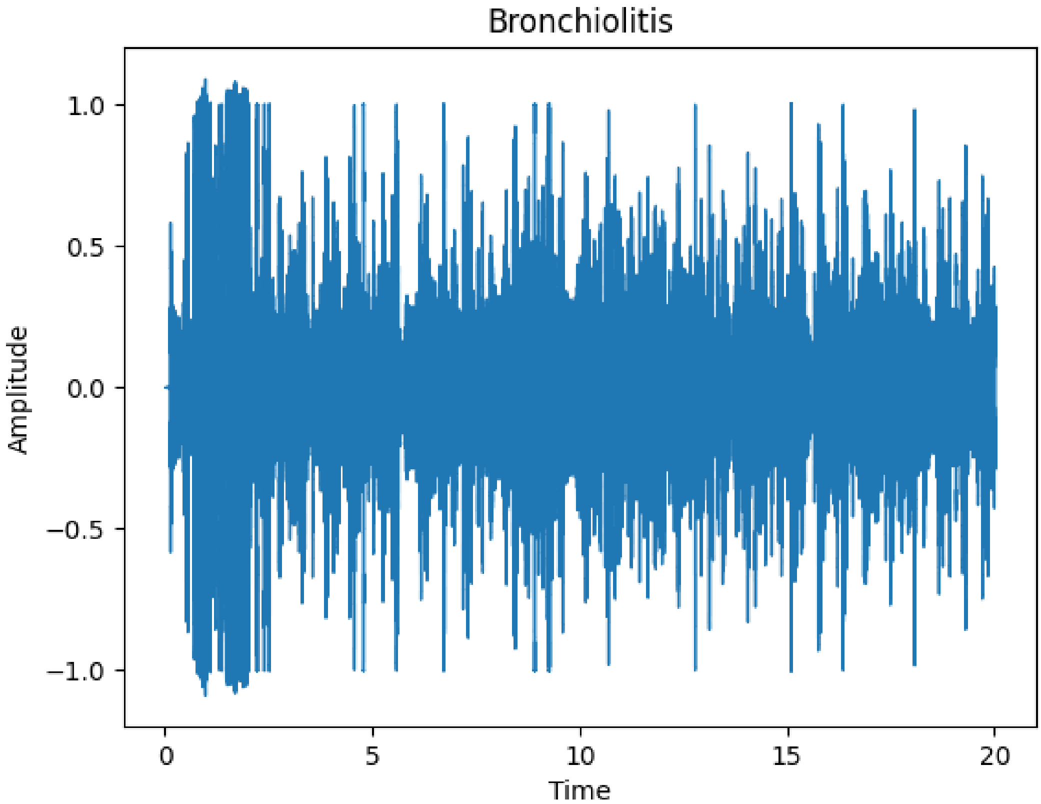

Fig. 4.6 Web Plot of Bronchiolitis

4.3 Feature Extraction :

Feature extraction is a process of extracting relevant information or features from an audio signal that can be used for various purposes, such as classification, clustering, and machine learning. Some commonly used features for audio analysis include:

1. Time-domain features: These features capture information about the amplitude and duration of the audio signal over time, such as the maximum and minimum amplitude, energy, zero-crossing rate, and root-mean-square (RMS) amplitude.
2. Frequency-domain features: These features capture information about the spectral content of the audio signal, such as the frequency range, spectral centroid, spectral flatness, spectral rolloff, and spectral density.
3. Mel-frequency cepstral coefficients (MFCCs): MFCCs are commonly used features in speech and audio processing, which capture information about the shape of the spectrum and the perceptual properties of the human ear.
4. Chroma features: Chroma features are derived from the harmonic content of the audio signal and represent the distribution of musical notes and their relative strengths.
5. Pitch and tempo features: These features capture information about the fundamental frequency (pitch) and beat rate (tempo) of the audio signal, which can be used for tasks such as music transcription and beat detection.

Here , MFCCs are used in this model.

4.3.1 MFCCs

MFCC which is also called Mel-frequency cepstral coefficients. These are the commonly used feature extraction techniques in audio signal processing, particularly in speech recognition and music analysis. They are based on the human auditory system's response to sounds of different frequencies, and they can effectively capture the spectral characteristics of audio signals.

The MFCC extraction process involves several steps. First, the audio signal is divided into overlapping frames of a fixed duration, typically 20-30 milliseconds. Each frame is then windowed using a windowing function, such as the Hamming window, to reduce spectral leakage. Next, the discrete Fourier transform (DFT) is applied to each frame to obtain its frequency spectrum.

The frequency spectrum is then converted into the mel frequency scale, which is a nonlinear scale that approximates the frequency response of the human ear. This is done by applying a set of triangular filters to the frequency spectrum, with the filters spaced closer together at lower frequencies and further apart at higher frequencies. The output of each filter is then log-transformed to obtain the log-mel spectrum.

Finally, the MFCCs are obtained by applying the discrete cosine transform (DCT) to the log-mel spectrum. The resulting coefficients represent the spectral envelope of the audio signal, which captures the shape of the frequency spectrum and the perceptual properties of the human ear.

MFCCs are widely used in speech recognition and music analysis because they are relatively robust to variations in recording conditions, such as background noise and microphone placement. They are also compact and efficient to compute, making them suitable for real-time applications. However, they may not capture all of the relevant information in some audio signals, such as those with non-stationary or time-varying spectra.

4.3.2 Data Augmentation for Audio Signals

To generate synthetic data for audio, various techniques can be applied, including noise injection, shifting time, and speed adjustment. These techniques can help to increase the diversity of the audio data and make the neural network more robust to variations in the input data.

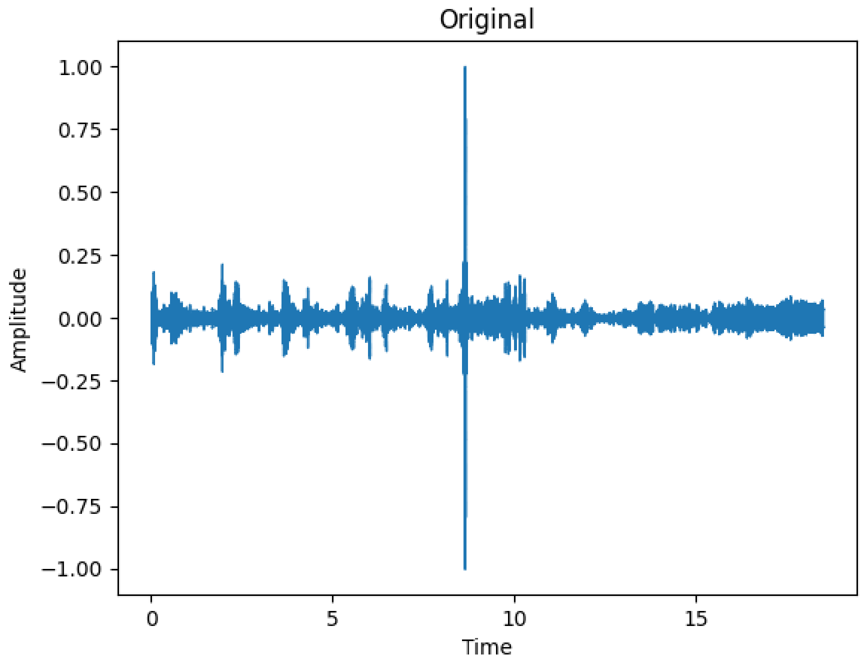

Fig. 4.7 Web plot of Audio Signal

1. Noise Injection: It is a technique where random noise is added to the audio signal.The noise can be either additive or multiplicative and can be applied at different levels of intensity. This technique can help to simulate real-world scenarios where the audio signal may be contaminated by noise or interference. For example, in speech recognition tasks, background noise can significantly degrade the performance of the system. By adding different levels of noise to the audio data, the neural network can learn to recognize and filter out background noise, leading to improved performance.

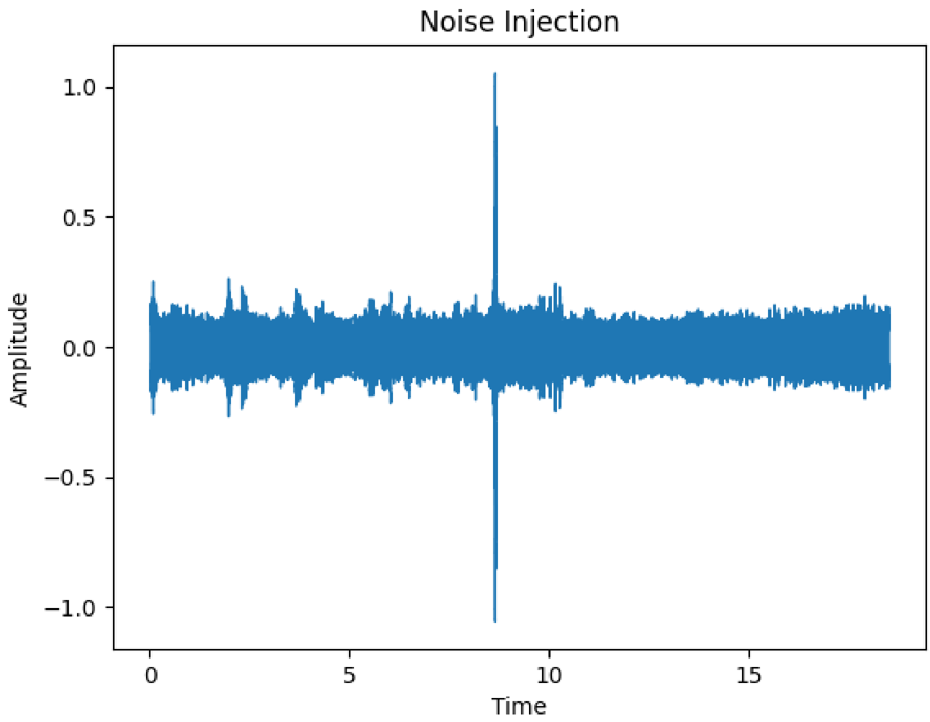

Fig. 4.8 Web plot after Noise Injection

2. Shifting time: It is a technique where the audio signal is shifted in time by a certain amount. This can be done either forward or backward in time and can help to simulate variations in the timing of the audio input. For example, a speaker may speak faster or slower than usual, and shifting the audio signal can help to capture these variations. This technique can help the neural network to better generalize to variations in the timing of the input signal, leading to improved performance.

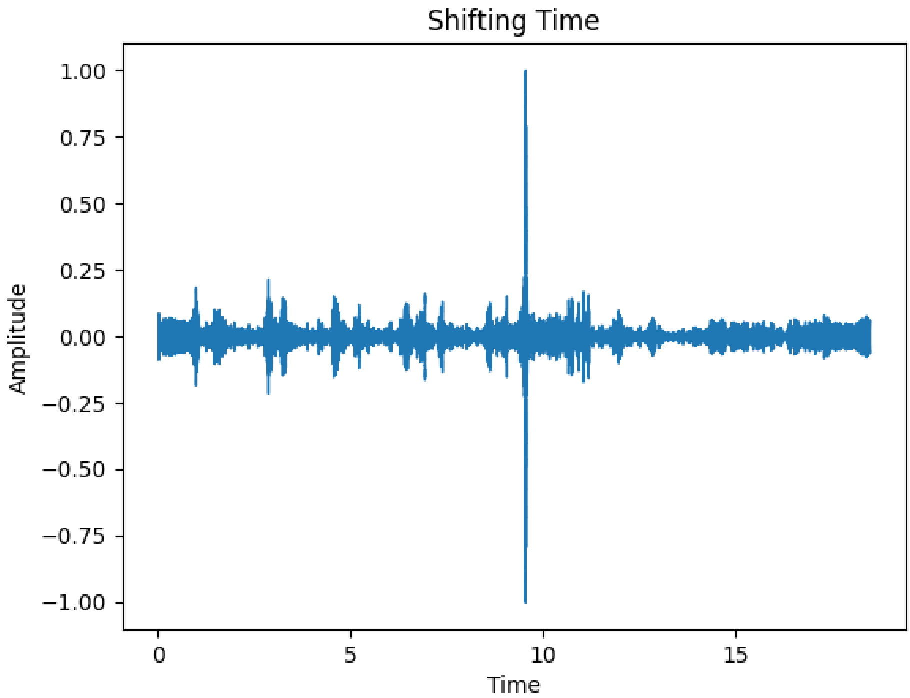

Fig. 4.9 Web plot after Shifting Time

3. Speed adjustment: It is a technique where the speed of the audio signal is adjusted by a certain factor. The speed can be increased or decreased, and this can help to simulate variations in the tempo of the audio input. For example, music may be played at a faster or slower tempo, and adjusting the speed of the audio signal can help to capture these variations. This technique can help the neural network to better generalize to variations in the tempo of the input signal, leading to improved performance.

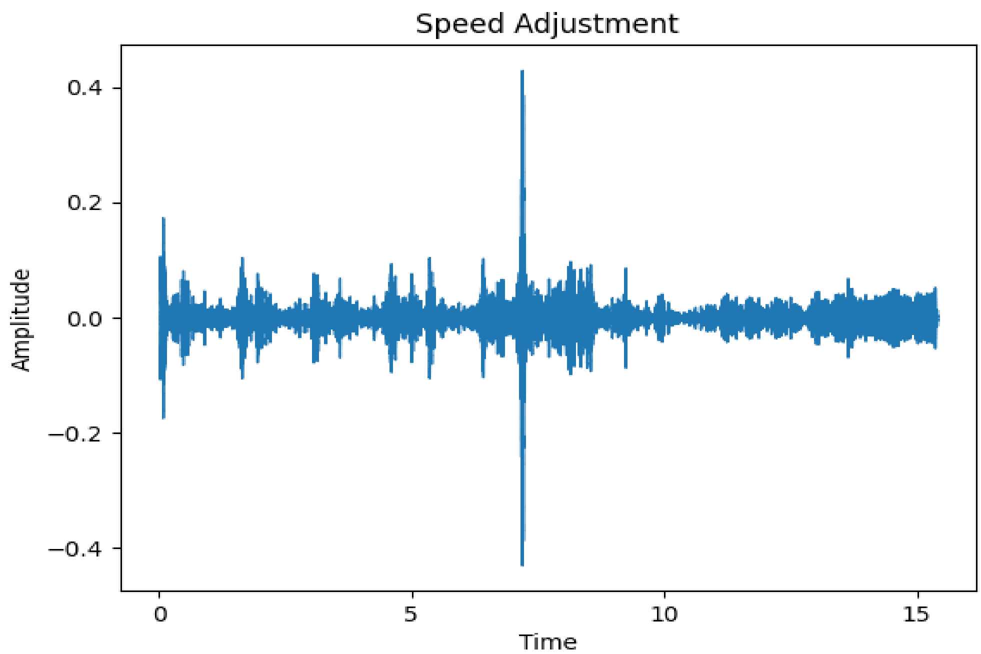

Fig. 4.10 Web plot after Speed Adjustment

The disease samples before oversampling are represented as follows:

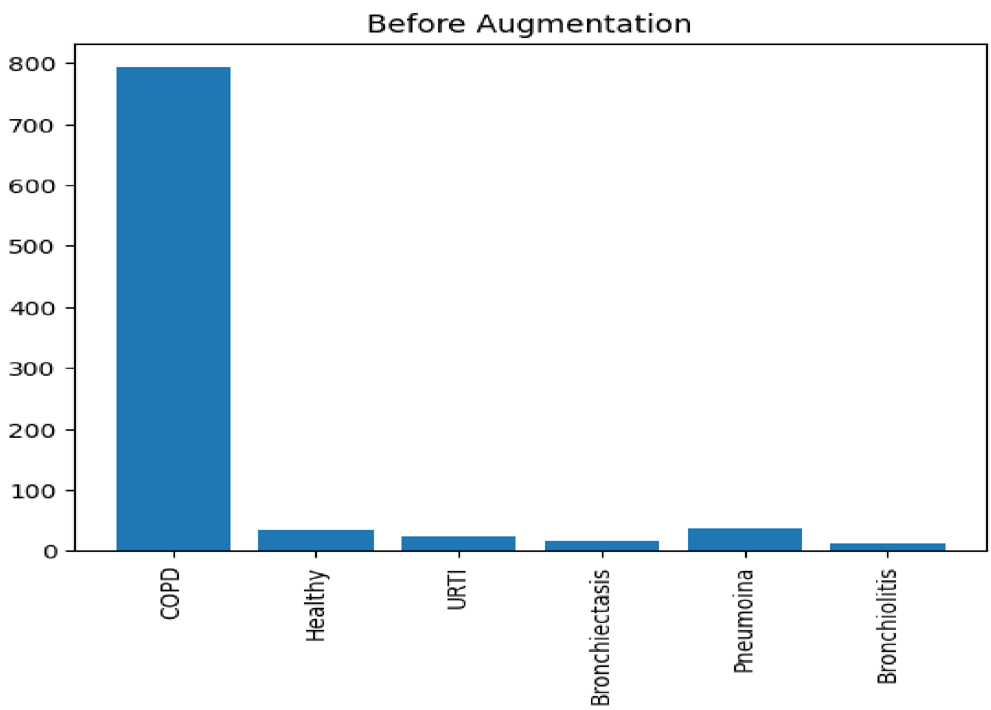

Fig. 4.11 Bar Graph before Augment Features

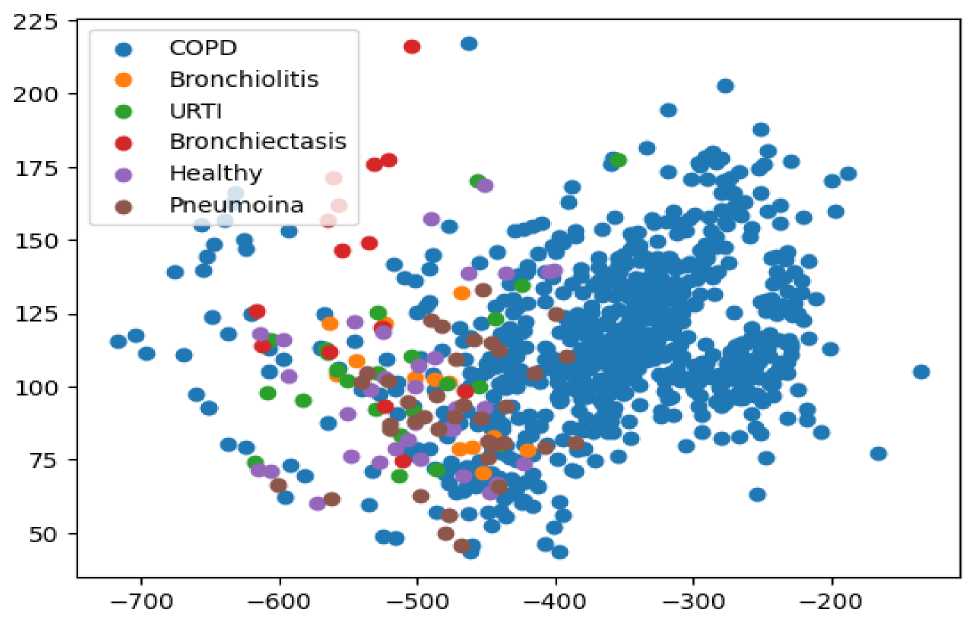

Fig. 4.12 Scatter plot before Augment Features

The disease samples after augmentation are represented as follows:

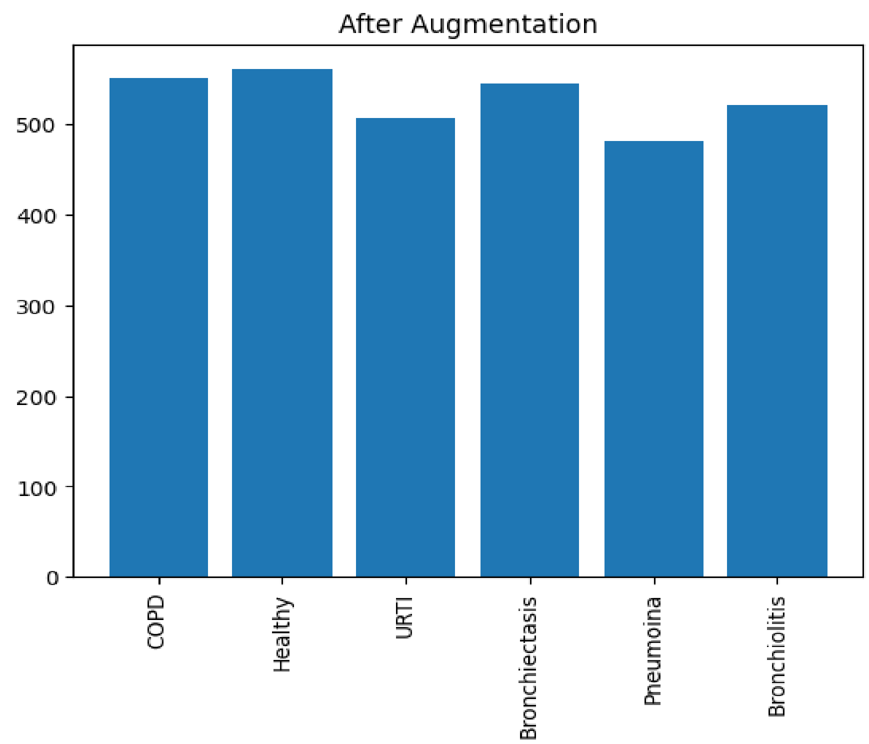

Fig. 4.13 Bar Graph after Augment Features

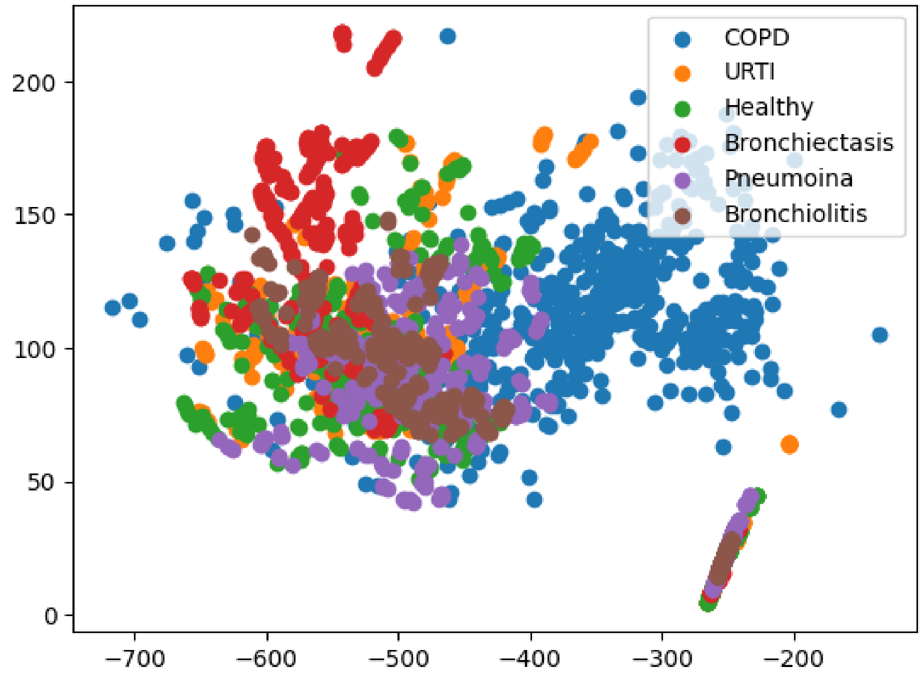

Fig. 4.14 Scatter Plot after Augment Features

4.4 Performance Measures

Performance measures are used to calculate the effectiveness and accuracy of a predicted model in machine learning and data mining. They are essential for comparing the performance of different models and selecting the best one for a given task. Some commonly used performance measures include:

1. Accuracy: Accuracy is the fraction of properly categorized examples in the dataset divided by the total number of occurrences. It is a simple and intuitive measure that works well for balanced datasets, but it may not be informative for imbalanced datasets where the minority class is of interest.
2. Precision and Recall: It is the proportion of properly categorized positive instances out of all positive instances predicted, whereas recall is the proportion of correctly classified positive instances out of all positive instances in the dataset. They are commonly used in binary classification problems.

3. F1-score: It is the HM(Harmonic Mean) of precision and recall and provides a single measure of a classifier's performance in binary classification problems. It is a useful measure for imbalanced datasets where both precision and recall are important.
4. Area Under the ROC Curve (AUC-ROC): AUC-ROC measures the trade-off between true positive rate (TPR) and false positive rate (FPR) at different classification thresholds. It is a popular measure for evaluating classifiers in multi-class and binary classification problems and is particularly useful for imbalanced datasets.
5. MSE (Mean Squared Error): In regression problems, MSE calculates the average squared difference between predicted and actual values. It is frequently employed in linear regression and other regression models..
6. Root Mean Squared Error (RMSE): RMSE is the square root of MSE and provides a measure of the average absolute difference between the predicted and actual values in regression problems.

Accuracy :

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

Precision :

$$\textbf{Precision} = \frac{TP}{TP+FP}$$

Recall / Sensitivity :

$$\textbf{Recall} = \frac{TP}{FN+TP}$$

Specificity:

$$\textbf{Specificity} = \frac{TP}{FN+TN}$$

F-measure :

$$\textbf{F-measure} = \frac{2 * Recall * Precision}{Recall + Precision}$$

4.5 Results and Discussion

Here, all the results are mentioned based on different observations such as batch-size, training-testing ratio, dropout layers drop , optimizers, number of convolutional layers and number of neurons .

1. Firstly batch-size : 32, training-testing ratio : 80/20, dropout layer : 50%, epochs : 50, optimizer : Adam, Convolutional layer : 2 [Neurons(64,64)].

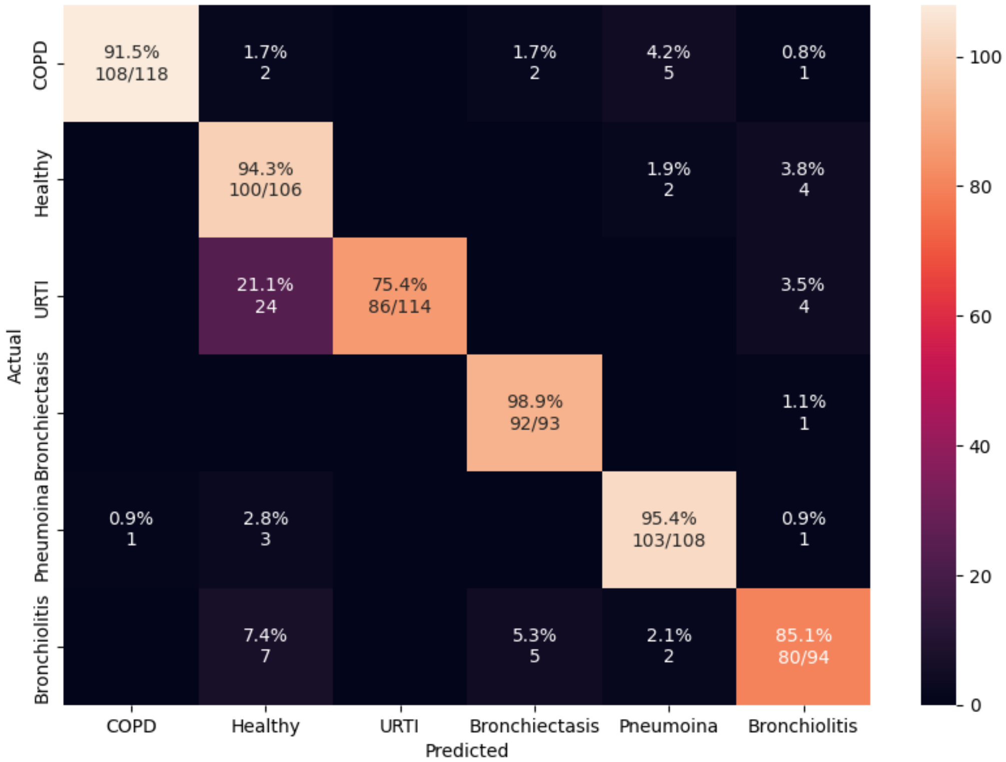

Fig. 4.15 Confusion Matrix 1

This confusion matrix shows , the model has 91.5% prediction on COPD, 94.3% prediction on Healthy, 75.4% prediction on URTI, 98.9% prediction on Bronchiectasis, 95.4% prediction on Pneumonia, and 85.1% prediction on Bronchiolitis.

Table 4.1 : Performance Measure 1

Labels	Precision	Recall	Sensitivity	Specificity	F1-Score
Healthy	0.735	0.943	0.943	0.931	0.826
COPD	0.990	0.915	0.915	0.998	0.951
URTI	1.000	0.954	0.954	1.000	0.860
Bronchiectasis	0.929	0.989	0.989	0.987	0.958
Pneumonia	0.919	0.953	0.953	0.982	0.936
Bronchiolitis	0.879	0.851	0.851	0.979	0.864

Accuracy and Loss :

Training Accuracy: 0.894

Testing Accuracy: 0.898

Training loss: 0.295

Testing loss: 0.294

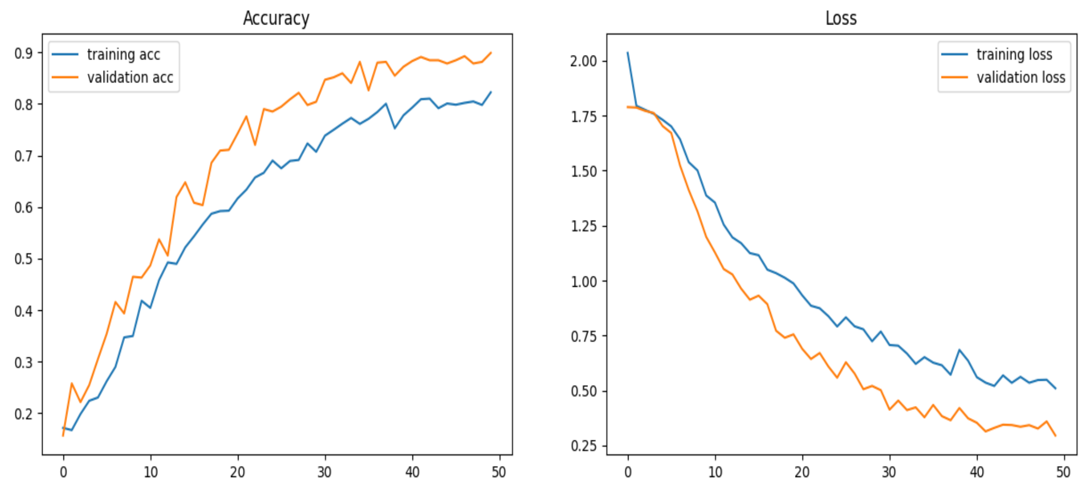

Fig. 4.16 Accuracy And Loss 1

This graph shows the accuracy and loss of Training data and Validation data. Here Validation accuracy is higher than Training accuracy at higher epochs and in case of loss , Validation loss is lower than Training loss at higher epochs.

2. Batch-size: 32, training-testing ratio: 80/20, dropout layer: 50%, epochs:50, optimizer: Adam, Convolutional layer: 3 [Neurons(64,64,64)].

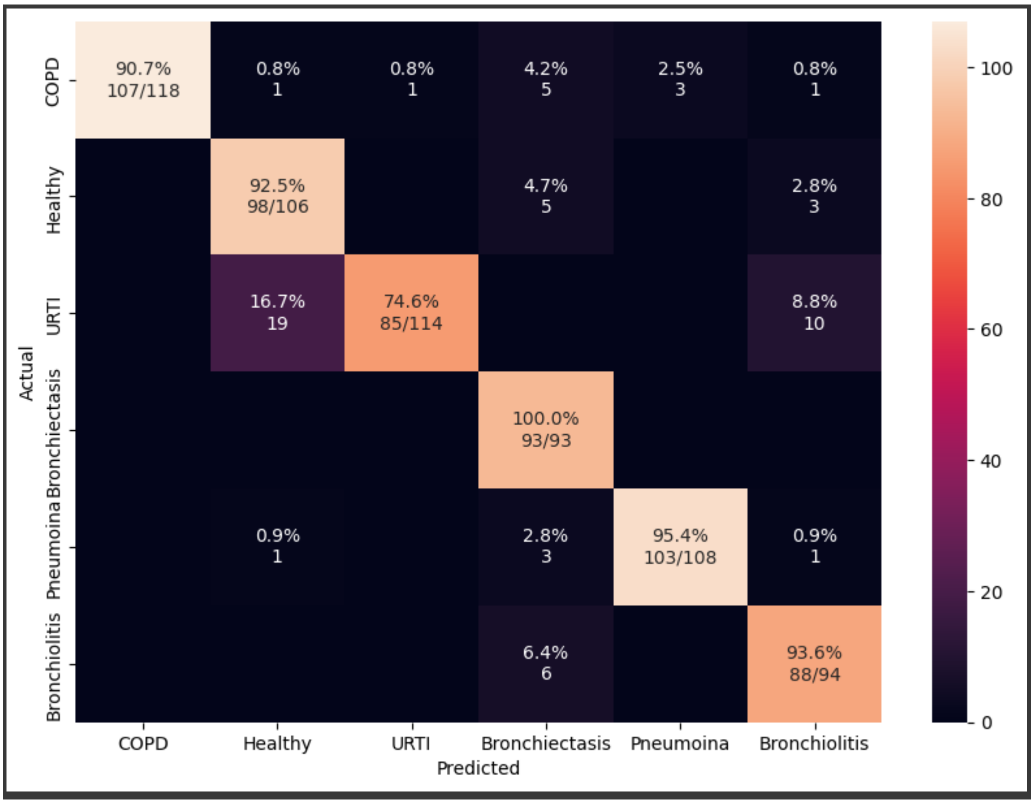

Fig. 4.17 Confusion Matrix 2

This confusion matrix shows , the model has 90.7% prediction on COPD, 92.5% prediction on Healthy, 74.6% prediction on URTI, 100.0% prediction on Bronchiectasis, 95.4% prediction on Pneumonia, and 93.6% prediction on Bronchiolitis.

Table 4.2 : Performance Measure 2

Labels	Precision	Recall	Sensitivity	Specificity	F1-Score
Healthy	0.823	0.924	0.924	0.960	0.871
COPD	1.000	0.906	0.906	1.000	0.951
URTI	0.988	0.745	0.745	0.998	0.850
Bronchiectasis	0.830	1.000	1.000	0.964	0.907
Pneumonia	0.971	0.953	0.953	0.994	0.962
Bronchiolitis	0.854	0.936	0.936	0.972	0.893

Accuracy and Loss :

Training Accuracy: 0.912

Testing Accuracy: 0.906

Training loss: 0.281

Testing loss: 0.285

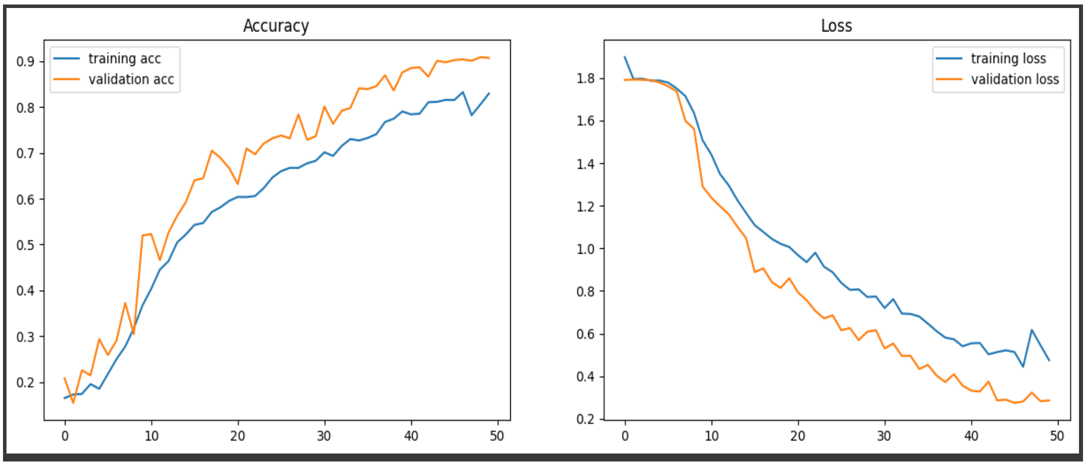

Fig. 4.18 Accuracy And Loss 2

This graph shows the accuracy and loss of Training data and Validation data. Here Validation accuracy is higher than Training accuracy at higher epochs and in case of loss , Validation loss is lower than Training loss at higher epochs.

3. Batch-size : 16, training-testing ratio : 80/20, dropout layer : 50%, epochs : 150, optimizer : Adam, Convolutional layer : 3 [Neurons(64,64,64)].

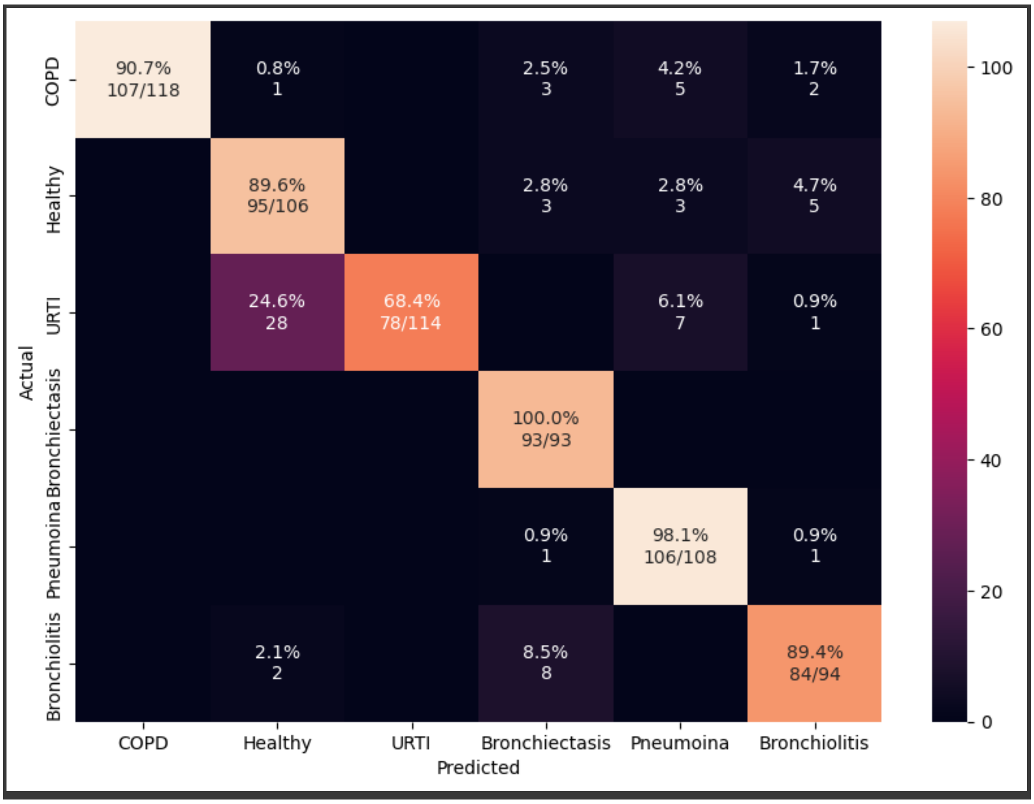

Fig. 4.19 Confusion Matrix 3

This confusion matrix shows , the model has 90.7% prediction on COPD, 89.6% prediction on Healthy, 68.4% prediction on URTI, 100.0% prediction on Bronchiectasis, 98.1% prediction on Pneumonia, and 89.4% prediction on Bronchiolitis.

Table 4.3 : Performance Measure 3

Labels	Precision	Recall	Sensitivity	Specificity	F1-Score
Healthy	0.753	0.896	0.896	0.941	0.818
COPD	1.000	0.906	0.906	1.000	0.951
URTI	1.000	0.684	0.684	1.000	0.812

Bronchiectasis	0.861	1.000	1.000	0.972	0.925
Pneumonia	0.876	0.981	0.981	0.971	0.925
Bronchiolitis	0.903	0.893	0.893	0.983	0.898

Accuracy and Loss :

Training Accuracy: 0.882

Testing Accuracy: 0.889

Training loss: 0.376

Testing loss: 0.375

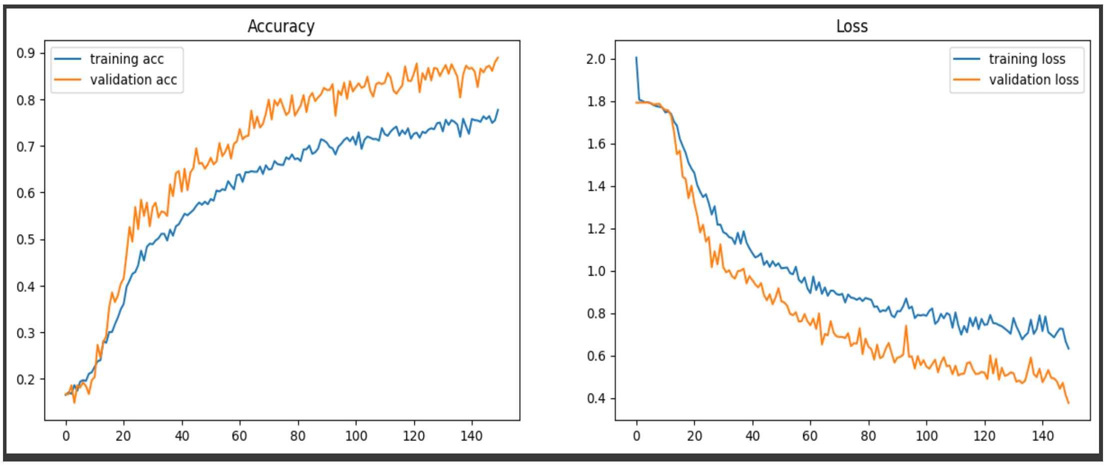

Fig. 4.20 Accuracy And Loss 3

This graph shows the accuracy and loss of Training data and Validation data. Here Validation accuracy is higher than Training accuracy at higher epochs and in case of loss , Validation loss is lower than Training loss at higher epochs.

Chapter 05: CONCLUSIONS

5.1 Discussion on the Results Achieved

The above presented models are simple and low-resource CNN based models that may help doctors identify COPD, URTI, Bronchiectasis, Bronchiolitis, Pneumonia using audio signals of Respiration. To do a deep study of the respiratory sounds dataset, we use features based on MFCC of the audio signal.

In this project there are many models that are trained and tested and between them four model results are presented above . We got one of the good accuracy and loss that are [Training Accuracy: 0.912, Validation Accuracy: 0.906, Training loss: 0.281, Validation loss: 0.285]

5.2 Application of the Project

The clear picture of the lungs is now done and utilized for clinical diagnosis by many medical practitioners. Respiratory illness categorization utilizing the sound of breathing is still limited in the research sector, although it is gradually becoming available. Respiratory sounds are used to realize extracting many characteristics from audios, developing different modeling methodologies, and identifying illnesses experimentally using suitable multi-class testing procedures. The model's precision allows for immediate illness prediction without X-Rays.

5.3 Future Work

We can enhance its features in the future to assist doctors in identifying different additional ailments, such as the likelihood of a heart attack based on beats of heartbeat, diagnosis of Asthma based on breathing audio, and so on. We can also improve the present technique to detect illness severity. Our technology may be paired with a Breath Monitoring System in aid to cure COPD.

REFERENCES

[1] R. X. A. Pramono, S. Bowyer, and E. Rodriguez-Villegas, ‘‘Automatic adventitious respiratory sound analysis: A systematic review,’’ PLoS ONE, vol. 12, no. 5, May 2017, Art. no. e0177926.

[2] B.-S. Lin and B.-S. Lin, ‘‘Automatic wheezing detection using speech recognition technique,’’ J. Med. Biol. Eng., vol. 36, no. 4, pp. 545–554, Aug. 2016.

[3] M. Lozano, J. A. Fiz, and R. Jané, ‘‘Automatic differentiation of normal and continuous adventitious respiratory sounds using ensemble empirical mode decomposition and instantaneous frequency,’’ IEEE J. Biomed. Health Informat., vol. 20, no. 2, pp. 486–497, Mar. 2016.

[4] M. Hassaballah, H. A. Alshazly, and A. A. Ali, ‘‘Ear recognition using local binary patterns: A comparative experimental study,’’ Expert Syst. Appl., vol. 118, pp. 182–200, Mar. 2019.

[5] M. Aykanat, O. Kilic, B. Kurt, and S. Saryal, ‘‘Classification of lung sounds using convolutional neural networks,’’ EURASIP J. Image Video Process., vol. 1, pp. 65–74, Sep. 2017.

[6] J. Gu et al., ‘‘Recent advances in convolutional neural networks,’’ Pattern Recognit., vol. 77, pp. 354–377, May 2018.

[7] D. Bardou, K. Zhang, and S. M. Ahmad, ‘‘Lung sounds classification using convolutional neural networks,’’ Artif. Intell. Med., vol. 88, pp. 58–69, June. 2018.

[8] C. Wachinger, M. Reuter, and T. Klein, ‘‘DeepNAT: Deep convolutional neural network for segmenting neuroanatomy,’’ NeuroImage, vol. 170, pp. 434–445, Apr. 2018.

[9] B. M. Rocha et al., ‘‘A respiratory sound database for the development of

automated classification,'' in Precision Medicine Powered by pHealth and Connected Health. Singapore: Springer, 2018, pp. 33–37.

[10] N. Sengupta, M. Sahidullah, and G. Saha, ''Lung sound classification using cepstral-based statistical features,'' Comput. Biol. Med., vol. 75, pp. 118–129, Aug. 2016.

[11] Altan, Kutlu & Allahwardi (2019) Altan G, Kutlu Y, Allahwardi N. Deep learning on computerized analysis of chronic obstructive pulmonary disease. *IEEE Journal of Biomedical and Health Informatics.* 2019;24(5):1344–1350. doi: 10.1109/JBHI.2019.2931395.

[12] Aykanat et al. (2017) Aykanat M, Kılıç Ö, Kurt B, Saryal S. Classification of lung sounds using convolutional neural networks. *EURASIP Journal on Image and Video Processing.* 2017;2017(1):65. doi: 10.1186/s13640-017-0213-2.

[13] M. Zañartu, G. J. Brown, and R. D. H. C. Perera, "Automatic respiratory sound analysis: A review," in IEEE Reviews in Biomedical Engineering, vol. 6, pp. 37-50, 2013.

[14] Acharya, J.; Basu, A. Deep neural network for respiratory sound classification in wearable devices enabled by patient specific model tuning. IEEE Trans. Biomed. Circuits Syst. 2020, 14, 535–544.

[15] Rocha B.M., Filos D., Mendes L. et al. A Respiratory Sound Database for the Deve-lopment of Automated Classification. In: Precision Medicine Powered by pHealth and Connected Health, Singapore: Springer Singapore. 2018., p. 33–37. (IFMBE Proceedings; vol. 66).

[16] Andrès E., Gass R., Charloux A., Brandt C., Hentzler A. Respiratory sound analysis in the era of evidence-based medicine and the world of medicine 2.0. Journal of Medicine and Life, 2018, Volume 11, Issue 2, pp 89–106.

[17] J. Chen, Y. Gao, L. Li and Y. Zhang, "A hybrid deep learning approach for respiratory diseases classification using X-ray images," 2020 IEEE International Conference on Data Mining Workshops (ICDMW), 2020, pp. 108-115.

[18] R. Kumar, V. Rathi, G. Kaur and A. Bala, "An ensemble approach for classification of respiratory diseases using machine learning techniques," 2018 IEEE 5th International Conference on Computing for Sustainable Global Development (INDIACom), 2018, pp. 1313-1317.

[19] S. K. Das, D. Dey, P. C. Debnath and S. Chattopadhyay, "Machine learning-based classification of lung diseases using chest X-ray images," 2018 IEEE 4th International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, pp. 63-68.

[20] S. S. Bandyopadhyay, S. Sengupta and D. Bhattacharjee, "Classification of interstitial lung diseases using machine learning techniques," 2019 IEEE 7th International Conference on Smart Energy Grid Engineering (SEGE), 2019, pp. 368-373.