University of Mumbai

Detection of Covid 19 using Cough Sound

Submitted at the end of semester VI in partial fulfillment of requirements

For the degree of

Bachelors in Technology

by

Baldev Sankhe
Roll No: 1913040
Manaswini Peram
Roll No: 1913041
Rudrani Sinare
Roll No: 1913056
Harshal Soman
Roll No: 1913057

Guide

Mrs. Sangeeta Kulkarni, Mrs. Jayshree Kundargi

Department of Electronics and Telecommunication Engineering
K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

Batch 2019 -2023

K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

Certificate

This is to certify that this report entitled **Detection of Covid 19 using Cough Sound** submitted by Ripal Doshi at the end of semester VI of TY B. Tech is a bonafide record for partial fulfillment of requirements for the degree of Bachelors in Technology in Electronics and Telecommunication Engineering of University of Mumbai

__________________ ______________________

Guide/Examiner1 Examiner2

Date:

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

BALDEV SANKHE ______________________ **Signature of the Student** **1913040** ______________________ **Roll No.**	**MANASWINI PERAM** ______________________ **Signature of the Student** **1913041** ______________________ **Roll No.**
RUDRANI SINARE ______________________ **Signature of the Student** **1913056** ______________________ **Roll No.**	**HARSHAL SOMAN** ______________________ **Signature of the Student** **1913057** ______________________ **Roll No.**

Date:

Place: Mumbai-77

Abstract

The emergence of a pandemic is rare event.Major pandemics and epidemics such as plague, cholera, flu, severe acute respiratory syndrome coronavirus (SARS-CoV) and Middle East respiratory syndrome coronavirus (MERS-CoV) have already afflicted humanity. When the world was hit with CoronaVirus disease in 2019, it wasn't prepared to deal with it. Originating in Wuhan, it spread rapidly across all nations of the world and was declared Public Health Emergency of International Concern on 30 January 2020, and a pandemic on 11 March 2020, shutting down the world within months.

The disease is identified by the presence of SARS-CoV-2. This detection was conducted by a PCR test to detect the genetic material of the coronavirus. PCR (polymerase chain reaction) tests are a fast, highly accurate way to diagnose certain infectious diseases and genetic changes. The tests work by finding the DNA or RNA of a pathogen (disease-causing organism) or abnormal cells in a sample.

In this project we have worked on a new method of detection of Covid-19 in patients,by analyzing the sound of the cough, which is a common symptom of the disease .We have used Machine Learning.

***Key words*:** Logistic Regression, SVM, CNN, MLP, Random Forest, MFCC

Contents

List of Figures………………………………………………………………………………..			Viii
List of Tables………………………………………………………………………………			Ix
Nomenclature……………………………………………………………………….….....			X
1	**Introduction**…………………………………………………………………...		1
	1.1	Background……...………………………………………………………..	1
	1.2	Motivation ………………………………………………………………..	4
	1.3	Scope of the project ………………………………………………………..	4
	1.4	Brief descriptions of project undertaken………………………………….	5
	1.5	Organization of the report………………………………………………..	5
2	**Literature Survey**……………………………………………………………….		6
3	**Project design** …………………………………………………………………		9
	3.1	Introduction……………………………………………………………….	9
	3.2	Problem statement…………………………………………………………	9
	3.3	Dataset………………………………………….	12
	3.4	Pre-processing……………….………………………………………………	15
	3.5	Block Diagram……………………………………..………………	16
4	**Implementation and experimentation….**…………………………………….		28
	4.1	Audio Preprocessing…………………	28
	4.2	Feature Extraction……………………………..	32
	4.4	MFCC…………………………………..	34
	4.5	Machine Learning Models 1. Logistic Regression…………………………………………………	35

		2. Support Vector Machine……………………………………………. 3. Random Forests ……………………………………………………. 4. MLP………..……………………………………………………….. 5. CNN……………………………………………………………….	
	4.8	Results ………………..……………………………………………..	36
5	**Conclusions and scope for further work**……….………………………………		55
	5.1	Conclusions………………………………………………………….	55
	5.2	Scope for further work…………………………………………………….	56
Bibliography ……………………………………………………………………...			57
Appendix A……………………………………………………………………………..			
Author's Publications……………………………………………………………….			
Acknowledgements…………………………………………………………………			

List of Figures

1. COVID-19 Tests in India ……..……………………………………………………..

2.1 Coswara dataset ………………………………………………………………..

2.2 *Coswara dataset summary and cleaning……………………………………………………….*
2.3 Test status and Covid status ……………………………….
2.4 Cleaned Dataset after reduction of columns
2.5 Final Count of classes in Dataset

3 Block Diagram ……………………………….

4.1 MFCC procedure ……………………………………………………………………
4.2 Function to Extract features ……………………………………………………….
4.3 Extracting Features using
MFCCs………………………………………………………………………..

4.4 Logistic Regression Implementation …………………………………………………
4.5 Logistic Regression Confusion Matrix ……………………………………………….
4.6 Logistic Regression Results …………………………………..……………….
4.7 SVM Implementation ………………………………….………………….
4.8 SVM Results ………………………………….………………….
4.9 Random Forest Implementation ……………………………….
4.10 Random Forest Results ……………………………….
4.11 MLP Implementation …………………………………….………….
4.12 MPL Results ……………………………….
4.13 CNN Architecture ……………………………….

2 Literature Survey……………………………………………………….

3 Project design ……………………………………………………….

3.1 Introduction……………………………………………………….
3.2 Problem statement……………………………………………………….
3.3 Dataset……………………………………….
3.4 Pre-processing………………….……………………………………
3.5 Block Diagram……………………………………..………………

List of Tables

1. Comparative Analysis of ML Models

Nomenclature

SVM: Support Vector Machine

MLP: Multilayer Perceptron

CNN: Convolutional Neural Network

RBF: Radial Based Function

CHAPTER 1

INTRODUCTION

1.1 Background

Corona Virus or Covid-19 disease originated in December 2019 in Wuhan, China. Within months the virus spread across nations around the globe and was declared a pandemic by the World Health Organization.

Symptoms of COVID-19 include fever, cough, headache, fatigue, breathing difficulties, loss of smell, and loss of taste.81% of patients develop mild to moderate symptoms,14% develop severe symptoms(dyspnea, hypoxia, or more than 50% lung involvement on imaging) whilst 5% suffer critical symptoms(respiratory failure, shock, or multiorgan dysfunction). Aged people and people with comorbidities(pre-existing lung diseases, diabetes etc). Some people experience prolonged effects of the disease and damage to organs.

With the capability of spreading rapidly and with little to no contact, this virus led to the shutting down of the world and yet managed to take millions of lives and affect even more.

1.2 Motivation

A reverse transcription-polymerase chain reaction (RT-PCR) test and Rapid Antigen test have been used to detect the presence of the virus in the human body. Both the tests involve taking a nasopharyngeal and throat swab as a sample test. RT-PCR is deemed as the most reliable and accurate Covid-19 detection test. However, RT-PCR test results are not available immediately. On average it takes a day to obtain the results. Furthermore, laboratory technicians and test kits are required for the test. A shortage of test kits and facilities can be observed. It is also difficult to make the facility available to remote areas and control the high demand for tests in cities. Patients find it arduous to wait in lines for tests and live in the uncertainty of the results until they arrive. Meanwhile, technicians and staff involved in the process also come in contact with the virus despite taking precautions.

1.3 Scope of the Project

Fifty to 70 per cent of people with symptomatic COVID-19 will develop a dry cough, according to William Checkley, MD, PhD, an associate professor in the division of pulmonary and critical care medicine at Johns Hopkins Medicine in Baltimore. A dry cough is prevalent and can be felt in the chest during the disease and sometimes even after it.

The cough tends to come on quickly, says Dr Checkley, beginning about a day or so after the onset of illness, but it doesn't typically subside quickly, especially for people who aren't vaccinated. An April 2021 study in The Lancet Respiratory Medicine found that cough lasts an average of 19 days for most people with COVID-19 and up to four weeks in about 5 per cent of patients.

Cough is different for every disease. The human ears might be incapable of distinguishing the same, but a machine can do the same. The sound of cough could be used to check if a patient is Covid-19 positive or not. An RT-PCR test takes time to get the results; but conducting tests via the sound of a cough would generate immediate results, require no setup and could be done by an individual from any location via their smartphone.

1.4 Brief description of the project undertaken

In this project, Machine learning techniques have been used to analyze the sound of a cough and detect if the person is COVID positive or not. Preprocessing has been done on the Coswara data to extract the necessary information and perform feature extraction using Mel-frequency cepstral coefficients (MFCCs). Support Vector Machine, Multilayer Perceptron, Logistic Regression and Random Forests models and a three-layer Convolutional Neural Network models have been used for further recognition and analysis of the data.

The objectives of this project are:

- To classify the cough recordings efficiently and help in detection of covid 19.
- To understand audio spectrograms and MFCC's.
- To understand and implement a CNN model.
- To compare the performance of CNN model to other machine learning classifiers.

1.5 Organization of Report

This project begins with an introduction to COVID-19, followed by a literature survey which helped gain knowledge about the correct implementations required. Furthermore, information of the entire process is specified in Implementation along with code snippets. Results and Conclusion are towards the end.

CHAPTER 2

LITERATURE SURVEY

Paper	Model used	Dataset	Results	Remarks
[1] Alkhodari M, Khandoker AH (2022) Detection of COVID-19 in smartphone-based breathing recordings: A pre-screening deep learning tool. PLoS ONE Journal	Training of the CNN-BiLSTM neural network	Coswara Dataset	• Overall discrimination accuracy of 94.58% and 92.08% using shallow and deep recordings • Overall AUC ROC of 0.90 • Overall Accuracy 94.58%	• 1D Signal Data Augmentation • MFCC • Highest Precision of shallow-95% & lowest-deep-90.83%. • 100.00% accuracy in predicting asymptomatic COVID-19 subjects.

[2] End-to-end convolutional neural network enables COVID-19 detection from breath and cough audio: a pilot study , BMJ Journal	CNN & Resnet	University of Cambridge Dataset	• AUC ROC of 0.846	• linear kernel SVM • threefold cross-optimisation+train folds. • CIdeR was better able to distinguish the 19 participants with asthma and a cough from the 23 who were COVID positive with a cough (AUC-ROC 0.909).
[3] Automatic diagnosis of COVID-19 disease using deep convolutional neural network with multi-feature channel from respiratory sound data: Cough, voice, and breath, Elsevier Journal	Deep CNN(DCNN) & VGG classification	University of Cambridge Dataset	• 95.45% accuracy	• DAE, GFCC< IMFCC - Data Augmentation • This technique can only distinguish COVID-19 positive, Non-COVID-19 Asthma, Pertussis, Bronchitis, and Healthy from respiratory sounds • 7% more accuracy than SVM &VGG models.

[4] Using Deep Learning with Large Aggregated Datasets for COVID-19 Classification from Cough, Cornell University.	SSL(Self-Supervised Learning), CNN,(SVM)	(Virufy, COUGHVID, Coswara,IATOS dataset,)	• SSL, CNN, and SVM approaches achieve an AUC of 0.807, 0.802, and 0.75 on the validation set, • SSL & CNN approaches achieve an AUC of 0.791 and 0.775	• Sufficient Data • There is a small difference between the models' AUCs, with the SSL approach having a 0.016 higher AUC on the test set, however the SSL model is slightly skewed toward sensitivity. • More focus on Data Aggregating and then training
[5] COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features, Elsevier Journal	CNN, LSTM and Resnet50, LR,SVM,KNN, MLP Different Combinations are used.	• Coswara, Sarcos, ComParE, • TASK, Brooklyn and Wallacedene • Google Audio Set & Freesound and Lib-rispeech	• ROC AUC of 0.98, 0.94 and 0.92 respectively for all three sound classes: coughs, breaths and speech • Resnet 50 is the optimal classifier.	• Sufficient Data • Deep Transfer Learning & extracting Bottleneck features. • LR,SVM,KNN,MLP are trained using bottleneck features as input.

[6] COVID-19 cough classification using machine learning and global smartphone recordings, Elsevier Journal	Logistic regression, k-nearest neighbour, support vector machine, multilayer perceptron, CNN, LSTM, Resnet50.	Coswara Dataset (Samples from 5 continents) Sarcos Dataset (Samples from South Africa)	• Resnet50 AUC - 0.976. • CNN AUC-0.953 • LSTM AUC–0.942 • MLP -0.897. • LR - 0.736 , SVM- 0.815. • CNN,LSTM,Resnet50 (Sarcos dataset testing)- 0.755,0.779,0.742	• Sufficient data. • All types of models. • LSTM performance on Sarcos was improved using SFS(Sequential Feature extraction). AOC- 0.938
[7] PANACEA cough sound-based diagnosis of COVID-19 for the DiCOVA 2021 Challenge, Cornell University Journal	LightGBM classifier, Logistic regression , Multilayer perceptron and Random Forest.	DiCOVA dataset COUGHVID	• AUC of -0.7631 on the test set	• LightGBM- tree based machine learning classifier(better than XGBoost) • Tried to implement transfer learning method for DiCOVA dataset but couldn't get good results.

[8] Audio feature ranking for sound-based COVID-19 patient detection, Cornell University	AdaBoost, KNN, LR, RF, and SVM	Cambridge Dataset Coswara Dataset	• The cepstral and spectral feature categories encode particularly informative data compared to time domain and tonal. • ROC-AUC for Cambrige(SVM)-87.68 • ROC-AUC for Coswara(SVM)-77.15	• More focus on feature extraction. • Cambridge accuracy improved by 17% and Coswara 10% just because of feature extraction.
[9] Automatic COVID-19 disease diagnosis using 1D convolutional neural network and augmentation with human respiratory sound based on parameters: cough, breath, and voice, AIMS Press Journal	1D CNN	Cambridge dataset	• Best tuned 1D CNN - accuracy of 89.48%.	• Data augmentation techniques. • Data De-noising Auto Encoder instead of conventional MFCC

[10] Automated detection of COVID-19 cough, Elsevier Journal	Time-Frequency Representation Analysis, YAMNet, Random Forest (RF), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), Logistic Regression (LR) and Naïve Bayes (NB).	University Hospital Arnau de Vilanova of Lleida, University of Cambridge , Coswara and Virufy datasets,Pertussis dataset,	From results of C Vs N,C Vs NC, C vs. NNC , C vs PT Random forests shows best overall performance	10-fold cross-validation Detected positive COVID-19 coughs but did not work so well for classifying pertussis coughs(Specificity = 85% for LR and LDA)
[11] COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features, Elsevier Journal	GeMap, ComParE,random forests, boosted and bagged decision trees	Via CDCVA study , 1103 participants,using five vocal tasks	Performance evaluation random forests, boosted and bagged decision trees: audio feature extraction approach with ComParE acoustic feature set showed better performance	

[12] COVID-19 detection with traditional and deep features on cough acoustic signals, Elsevier Journal	ReliefF algorithm SVM	re access site https://virufy.org/. The data was provided by a mobile application developed by Stanford University, (Virufy)	For SVM linear 98.4% accuracy, 99.5% recall, 97.3% specificity, 97.4% precision and 98.6% F1-score values were obtained.	Traditional :16 of those in the COVID-19(+) were detected incorrectly, only 3 of those in the COVID-19(Deep learning:7 of those in the COVID-19(+) class were detected incorrectly, 9 of those in the COVID-19(

CHAPTER 3

METHODOLOGY

3.1 Problem Statement

During the Pandemic of COVID-19, millions of people around the world got infected and suffered tremendously. These troubles were increased by the long time taken to confirm the presence of the virus in the patient and the process they had to go through to get tested. This project finds an easier and user-friendly method to ease the process of testing the presence of COVID-19. It aims to detect Covid-19 using audio signals of voice, cough and breathing patterns with the help of Deep Learning.

3.2 Block Diagram

Figure 2.1: Block Diagram

3.3 Dataset

Project Coswara by the Indian Institute of Science (IISc) Bangalore is an attempt to build a diagnostic tool for Covid-19 based on respiratory, cough and speech sounds. The project is in the data collection stage now. It requires the participants to provide a recording of breathing sounds, cough sounds, sustained phonation of vowel sounds and a counting exercise which takes around 5-7 minutes of your time. No personally identifiable data is collected from the participants.

This Coswara dataset, with 2374 entries, has been used in this project. This dataset from its source consists of information such as date of sound recording, COVID-19 status, gender, country, region, vaccination status, co-morbidities and sound.

	id	a	covid_status	record_date	ep	g	l_c	l_l	l_s	rU	...	vacc	bd	others_resp	ftg	st	ihd	asthma	others_preexist	cld	pneumonia
0	iV3Db6t1T8b7c5HQY2TwxlhjbzD3	28	healthy	2020-04-23	y	male	India	Anantapur	Andhra Pradesh	n	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	AxuYWBN0jFVLINCBqlW5aZmGCdu1	25	healthy	2020-04-20	y	male	India	BENGALURU URBAN	Karnataka	n	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	C5elsssb9GSkaAglfsHMHeR6fSh1	28	healthy	2020-04-24	y	female	United States	Pittsburgh	Pennsylvania	n	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	YjbEAECMBlaZKyfqOvWy5DDlmUb2	26	healthy	2020-04-23	y	male	India	Bangalore	Karnataka	n	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	aGOvk4ji0cVqlzCs1jHnzlw2UEy2	32	healthy	2020-04-22	y	male	India	Nalanda	Bihar	n	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 36 columns

Figure 2.2: Coswara dataset

3.4 Data Pre-processing

Step 1: Filter required columns

In this project, we only require the ID, cough sound and COVID-19 status to assess our models. Therefore, only the required information was kept in the .csv file during EDA.

	id	wav1_path	covid_status
0	iV3Db6t1T8b7c5HQY2TwxlhjbzD3	./Extracted_data/20200424/iV3Db6t1T8b7c5HQY2Tw...	healthy
1	AxuYWBN0jFVLINCBqlW5aZmGCdu1	./Extracted_data/20200424/AxuYWBN0jFVLINCBqlW5...	healthy
2	C5elsssb9GSkaAglfsHMHeR6fSh1	./Extracted_data/20200424/C5elsssb9GSkaAglfsHM...	healthy
3	YjbEAECMBlaZKyfqOvWy5DDlmUb2	./Extracted_data/20200424/YjbEAECMBlaZKyfqOvWy...	healthy
4	aGOvk4ji0cVqlzCs1jHnzlw2UEy2	./Extracted_data/20200424/aGOvk4ji0cVqlzCs1jHn...	healthy
...	...	...	...
1345	eAH6zemkWAfsP96K8tMyZJIDvgR2	./Extracted_data/20210714/eAH6zemkWAfsP96K8tMy...	positive_moderate
1346	gGgG540lBOZoSK1vyqNVlqBzQBI1	./Extracted_data/20210714/gGgG540lBOZoSK1vyqNV...	healthy
1347	nx6RCgD0OjdzLa07UHBdHfz54ML2	./Extracted_data/20210714/nx6RCgD0OjdzLa07UHBd...	healthy
1348	Ft82dZ2O4PMuq64dPIEVik9upMr2	./Extracted_data/20210714/Ft82dZ2O4PMuq64dPIEV...	positive_moderate
1349	A5KW9PXraNgckln1gnNaJnX6DvB2	./Extracted_data/20210714/A5KW9PXraNgckln1gnNa...	positive_moderate

1350 rows × 3 columns

Figure 2.3: Cleaned Data(Keeping required columns)

Step 2: Covid status consists of multiple classes indicating: Healthy, Mildly Positive, Moderately Positive, Fully Recovered, Positive, No Respective illness exposed and No Respective Illness identified.

```
healthy                       1404
positive_mild                  325
no_resp_illness_exposed        192
resp_illness_not_identified    153
positive_moderate              127
recovered_full                 103
positive_asymp                  70
Name: covid_status, dtype: int64
```

Figure 2.4: Coswara dataset classes

The only classes required to test models in this project are healthy and moderately positive as they would lead to precise results. This reduces the dataset to 1531 entries with healthy entries being 1404 and positive being 127. This concludes that the dataset is imbalanced.

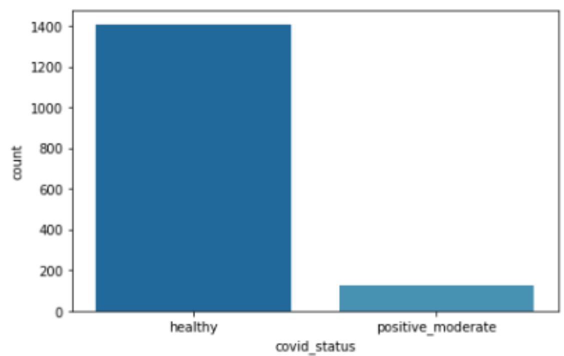


Figure 2.5: Classes required

Now, for feature extraction, the sampling rate of sound/audio files should be above 22050 Hz as it is the standard sampling rate for the Librosa library used here. Therefore entries that fail to meet the sampling rate are dropped. Thus, the final number of entries in our dataset is 1321. Now the dataset is ready for feature extraction.

```
healthy              1194
positive_moderate     127
Name: covid_status, dtype: int64
```

Figure 2.6: Final Count of classes in Dataset

CHAPTER 4

Implementation

4.1 Audio Preprocessing

The following functions have been performed on the data:

- Normalization: converting to a specific range for eg. -1 to 1
- Downsampling: Reducing the sampling rate while taking into consideration the aliasing effect. Reduces the audio to a manageable size.
- Segmentation Audio segmentation divides the audio into smaller meaningful segments.

4.2 Feature Extraction

To train a Machine Learning model for any audio-based algorithm such as classification, speech recognition, audio generation, tagging, audio segmentation, source separation, audio denoising, and more, it is necessary to extract information from the audio in a format recognized by the model. This information is called features and the process of feature extraction. Audio features are basically a description of sound or an audio signal. Different features capture different aspects of sound.

The traditional Machine Learning approach considers all or most of the features from both time and frequency domains as inputs into the model. Features had to be selected based on the requirements of the model. Commonly used features include Amplitude Envelope, Zero-Crossing Rate, Root Mean Square Energy, Spectral Centroid, Band Energy Ratio, and Spectral Bandwidth.

Deep Learning methods use unstructured audio representations such as spectrograms of MFCCs. They extract the pattern on their own. This increased the ease of feature extraction as it became automatic.

Commonly used features or representations that are directly fed into neural network architectures are spectrograms, Mel-spectrograms, and Mel-Frequency Cepstral Coefficients (MFCCs).

In this project, MFCCs have been used for feature extraction.

4.3 MFCC – Mel Frequency Cepstral Coefficients

The information on the rate of change in spectral bands of a signal is given by its cepstrum. A cepstrum is basically a spectrum of the log of the spectrum of the time signal. The resulting spectrum is neither in the frequency domain nor in the time domain and hence, it was named the quefrency (an anagram of the word frequency) domain. The Mel-frequency Cepstral Coefficients (MFCCs) are nothing but the coefficients that make up the Mel-frequency cepstrum.

The cepstrum conveys the different values that construct the formants (a characteristic component of the quality of a speech sound) and timbre of a sound. MFCCs thus are useful for deep learning models.

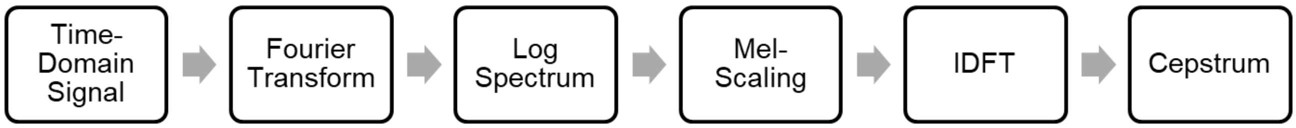


Figure 3.1 MFCC process

```
1it [00:00,  5.63it/s]0
1
3it [00:00,  5.39it/s]2
3
5it [00:00,  5.86it/s]4
5
7it [00:01,  8.22it/s]6
7
9it [00:01,  7.69it/s]8
9
11it [00:01,  8.58it/s]10
11
```

Figure 3.2: Features Extracted

4.4 Machine Learning Models

1. Logistic Regression

Logistic regression is a type of regression analysis. It is essentially used to calculate (or predict) the probability of a binary (yes/no) event occurring. Logistic regression is a classification algorithm. The factors or the independent variables that influence the outcome are independent of each other. In other words, there is little or no multicollinearity among the independent variables. In Logistic regression, we have used liblinear as the solver parameter. We also tried using the lbfgs solver but liblinear solver had more precision.

\

	precision	recall	f1-score	support
0	0.88	0.98	0.93	232
1	0.38	0.09	0.15	33
accuracy			0.87	265
macro avg	0.63	0.53	0.54	265
weighted avg	0.82	0.87	0.83	265

Figure 3.3 Logistic Regression Results

2. Support Vector Machine

Supervised Learning Algorithm

•Used for Classification and Regression purposes, in this project it has been used for the former.

•The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

•SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed a Support Vector Machine.

Here, we have used 'rbf' kernel which gave better precision than the sigmoid function. We also tried using various values of C like 0.1, 1, 10 and 100 but the model overfitted for all the values.

	precision	recall	f1-score	support
0	0.88	1.00	0.93	232
1	0.00	0.00	0.00	33
accuracy			0.88	265
macro avg	0.44	0.50	0.47	265
weighted avg	0.77	0.88	0.82	265

Figure 3.4: SVM Results

3. Random Forest Classifier

•Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result.

•Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

•The reason that the random forest model works so well is:

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

It predicts output with high accuracy, even for a large dataset it runs efficiently. The n_estimator parameter has been set to 100 after performing cross-validation.

	precision	recall	f1-score	support
0	0.88	1.00	0.93	232
1	0.00	0.00	0.00	33
accuracy			0.88	265
macro avg	0.44	0.50	0.47	265
weighted avg	0.77	0.88	0.82	265

Figure 3.5:: Random Forest Results

4. Multi-Layer Perceptron

MLP Classifier stands for Multi-layer Perceptron classifier which in the name itself connects to a Neural Network. In the Perceptron Algorithm, in the perceptron, we just multiply with weights and add Bias, but we do this in one layer only. In the Multilayer perceptron, there can be more than one linear layer (combinations of neurons).

If we take the simple example of a three-layer network, the first layer will be the input layer and last will be the output layer and the middle layer will be called the hidden layer. We feed our input data into the input layer and take the output from the output layer.

Activation function:

Activation functions also known as non-linearity describe the input-output relations in a non-linear way. This gives the model power to be more flexible in describing arbitrary relations. Here are some popular activation functions Sigmoid, Relu, and TanH. We have used the most efficient MLP architecture after cross-validation.

	precision	recall	f1-score	support
0	0.90	0.99	0.94	232
1	0.70	0.21	0.33	33
accuracy			0.89	265
macro avg	0.80	0.60	0.63	265
weighted avg	0.87	0.89	0.86	265

Figure 3.6: MLP Results

5. Convolutional Neural Network

CNNs or convolutional neural nets are a type of deep learning algorithm that does really well at learning images. CNN can also work with audio by feeding its features in the form of images.

Convolutional neural networks refer to a sub-category of neural networks: they, therefore, have all the characteristics of neural networks. However, CNN is specifically designed to process input images. Their architecture is then more specific: it is composed of two main blocks.

The first block makes the particularity of this type of neural network since it functions as a feature extractor. To do this, it performs template matching by applying convolution filtering operations. The first layer filters the image with several convolution kernels and returns “feature maps”, which are then normalized (with an activation function) and/or resized.

This process can be repeated several times: we filter the features maps obtained with new kernels, which gives us new features maps to normalize and resize, and we can filter again, and so on. Finally, the values of the last feature maps are concatenated into a vector. This vector defines the output of the first block and the input of the second.

The second block is not characteristic of a CNN: it is in fact at the end of all the neural networks used for classification. The input vector values are transformed (with several linear combinations and activation functions) to return a new vector to the output. This last vector contains as many elements as there are classes: element represents the probability that the image belongs to class i. Each element is therefore between 0 and 1, and the sum of all is worth 1. These probabilities are calculated by the last layer of this block (and therefore of the network), which uses a logistic function (binary classification) or a softmax function (multi-class classification) as an activation function.

As with ordinary neural networks, the parameters of the layers are determined by gradient backpropagation: the cross-entropy is minimized during the training phase. But in the case of CNN, these parameters refer in particular to the image features.

In this project, Tensorflow has been used to create three layers of a network. We tried using 2 layers, the number of neurons and changing the dropout rate but the model overfitted for all the tried values.

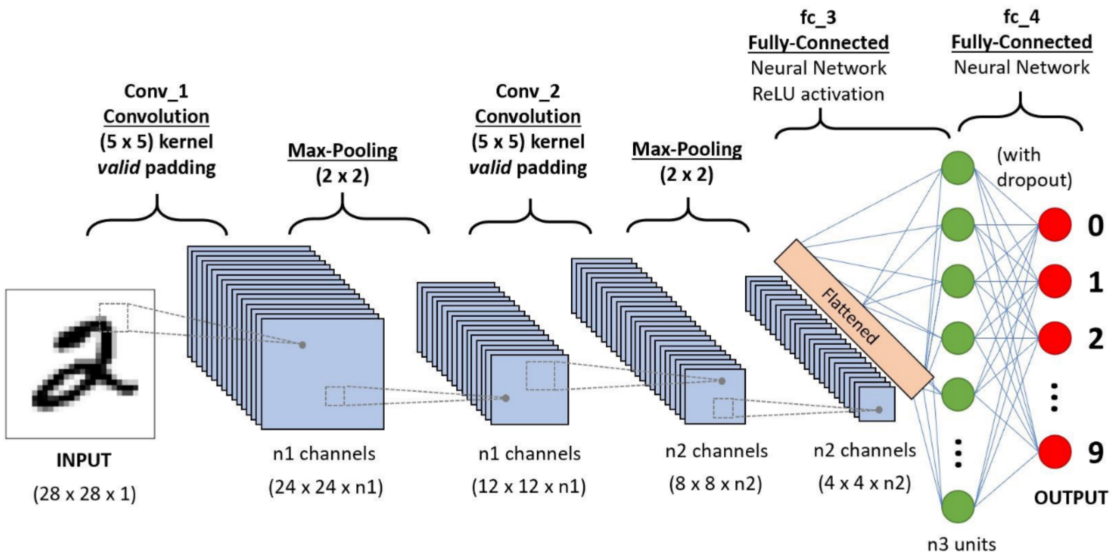
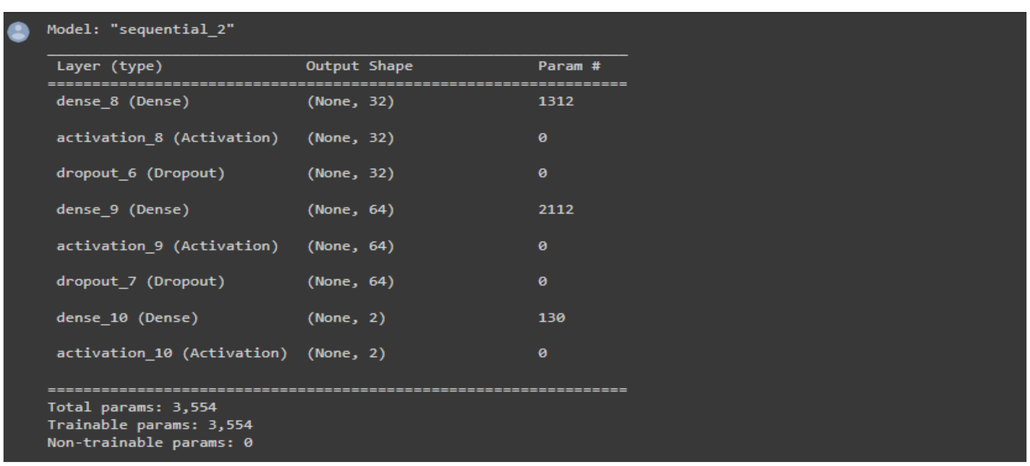


Figure 3.7: CNN Architecture



```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_8 (Dense)             (None, 32)                1312

 activation_8 (Activation)   (None, 32)                0

 dropout_6 (Dropout)         (None, 32)                0

 dense_9 (Dense)             (None, 64)                2112

 activation_9 (Activation)   (None, 64)                0

 dropout_7 (Dropout)         (None, 64)                0

 dense_10 (Dense)            (None, 2)                 130

 activation_10 (Activation)  (None, 2)                 0

=================================================================
Total params: 3,554
Trainable params: 3,554
Non-trainable params: 0
```

Figure 3.8: CNN Model Summary

```
              precision    recall  f1-score   support

           0       0.88      1.00      0.93       232
           1       0.00      0.00      0.00        33

    accuracy                           0.88       265
   macro avg       0.44      0.50      0.47       265
weighted avg       0.77      0.88      0.82       265
```

Figure 3.9: CNN Results

4.5 Results

Each model when implemented displayed the following results for the given parameters. We can see that MLP performed the best for all parameters followed by Logistic Regression for precision and f1-score and whilst all other models gave nearly the same results.

Model	Precision	Recall	f1-score	Accuracy
Logistic regression	0.82	0.87	0.83	0.87
SVM	0.77	0.88	0.82	0.88
Random Forest	0.77	0.88	0.82	0.88
MLP	0.87	0.89	0.86	0.89
Deep Learning Model(CNN)	0.77	0.88	0.82	0.88

Figure 3.10:Comparative Analysis

CHAPTER 5

CONCLUSION AND SCOPE FOR FURTHER WORK

5.1 Conclusion

From the obtained results we conclude that the Coswara Dataset is highly imbalanced and causes overfitting for the models which were implemented. MLP gave surprisingly good results with a precision score of 0.90 for class 0 and 0.70 for class 1 which has a limited number of audio samples.

5.2 Scope for further work

- Model overfits due to imbalance in the datasets. This can be solved by data augmentation.
- Signal preprocessing techniques can be implemented to improve performance.
- CNN model hyper-parameters can be fine-tuned a little more to improve the accuracy.

Appendix:

CODE –

The coding part was carried out to prepare the image-dataset, visualize it, pre-processing, building the models and then evaluating them. The code has been written in Python programming language using Google Collab as IDE. The experiments and all the models building are done based on Python libraries. The code is available in the Google drive folder given in following link:

https://colab.research.google.com/drive/1URKDS-CR1fMY_ftgFtyKXnhMWgqagO_w

BIBLIOGRAPHY :

[1] Alkhodari M, Khandoker AH (2022) Detection of COVID-19 in smartphone-based breathing recordings: A pre-screening deep learning tool. PLoS ONE 17(1): e0262448. https://doi.org/10.1371/journal.pone.0262448

[2] Coppock H, Gaskell A, Tzirakis P, et alEnd-to-end convolutional neural network enables COVID-19 detection from breath and cough audio: a pilot studyBMJ Innovations 2021;7:356-362.

[3] Kranthi Kumar Lella, Alphonse Pja,
Automatic diagnosis of COVID-19 disease using deep convolutional neural network with multi-feature channel from respiratory sound data: Cough, voice, and breath,

Alexandria Engineering Journal, Volume 61, Issue 2, 2022, Pages 1319-1334, ISSN 1110-0168,

[4] arXiv:2201.01669 [eess.AS]

*[5] Madhurananda Pahar, Marisa Klopper, Robin Warren, Thomas Niesler,
COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features, Computers in Biology and Medicine, Volume 141, 2022, 105153,
ISSN 0010-4825,*

*[6] Madhurananda Pahar, Marisa Klopper, Robin Warren, Thomas Niesler,
COVID-19 cough classification using machine learning and global smartphone recordings,
Computers in Biology and Medicine,
Volume 135,
2021,
104572,
ISSN 0010-4825,*

[7] arXiv:2106.04423 [cs.SD]

[8] arXiv:2104.07128 [cs.SD]

[9] Lella, Kranthi Kumar, and Alphonse Pja. "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 vol. 8,2 240-264. 10 Mar. 2021

[10] Alberto Tena, Francesc Clarià, Francesc Solsona, Automated detection of COVID-19 cough, Biomedical Signal Processing and Control, Volume 71, Part A, 2022, 103175, ISSN 1746-8094,

[11]Madhurananda Pahar, Marisa Klopper, Robin Warren, Thomas Niesler, COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features, Computers in Biology and Medicine, Volume 141, 2022, 105153, ISSN 0010-4825,

[12] https://arxiv.org/abs/2010.09458

Acknowledgements

We thank our college, K. J. Somaiya College of Engineering for allowing us to work on this topic for our mini-project. We are grateful to our principal Dr. Shubha Pandit as well as our HOD Dr. Bharati Singh for providing such an opportunity to us. We would like to express our gratitude to Mrs. Jayshree Kundargi and Mrs. Sangeeta Kulkarni, our project guides, for guiding us in this project for the entire semester. We are extremely grateful to them for giving their valuable time and attention. We have learnt some very important skills in the duration of this mini project. Throughout the project work, their useful suggestions, constant encouragement has given the right direction and shape to our learning. Each of us have studied the concept through various information sources and tried our best to make the most from this opportunity.