Covid-19 Cough Classification Using Machine Learning

Third Year Mini-Project

Submitted in partial fulfilment of

The requirements for the

6th SEMESTER

IN

B.TECH INSTRUMENTATION AND CONTROL

Abhijeet Ramesh Deshmukh

MIS: 111909002

Anisha Bapurao Raut

MIS: 111909009

under the guidance of Dr. Amruta Deshpande.

Department of
Instrumentation and Control
College of Engineering, Pune-411005

(2021-22)

Contents

Abstract **1**

1 Introduction **2**

1.1 Literature Survey 2
1.2 Data 3
1.3 Data Extraction And Pre-processing 4
1.3.1 Spectrogram of Cough Audio Sample 5
1.3.2 Mel Power Spectrogram 5

2 Feature Extraction And Dataset Balancing **7**

2.1 Feature Extraction 7
2.1.1 chroma stft 7
2.1.2 RMS 7
2.1.3 Spectral Centroid 7
2.1.4 Spectral Bandwidth 8
2.1.5 Roll off 8
2.1.6 Zero Crossing Rate 8
2.1.7 MFCC coefficients 8
2.2 Dataset Balancing 9

3 Machine Learning Models **11**

3.0.1 Catboost Classifier 11
3.0.2 Gradient Boosting Classifier 11
3.0.3 Light Gradient Boosting Machine 12
3.0.4 Extreme Gradient Boosting 12
3.0.5 Random Forest Classifier 13
3.0.6 Extra Trees Classifier 14
3.0.7 K Neighbors Classifier 14
3.0.8 Naive Bayes 15
3.0.9 Decision Tree Classifier 15

3.0.10 Dummy Classifier 15
3.1 Hyperparameter Tuning 15

4 Result **18**

5 Conclusion and Future Scope **20**
5.1 Conclusion 20
5.2 Future Scope 21

Abstract

In this project report, We present a machine learning based COVID-19 cough classifier which can discriminate COVID-19 positive coughs from both COVID-19 negative and healthy coughs recorded on a smartphone. This type of screening is non-contact, easy to apply, and can reduce the workload in testing centres as well as limit transmission by recommending early self-isolation to those who have a cough suggestive of COVID-19. The datasets used in this study include subjects from five continents and contain both forced and natural coughs, indicating that the approach is widely applicable. The publicly available Coswara dataset contains 482 COVID-19 positive and 1794 healthy subjects. The datasets indicate that COVID-19 positive coughs are 15%–20% shorter than non-COVID coughs. Dataset skew was addressed by applying the synthetic minority oversampling technique (SMOTE). The machine models implemented on the dataset include Catboost Classifier, Gradient Boosting Classifier, Light Gradient Boosting Machine, Extreme Gradient Boosting, Random Forest Classifier, Extra Trees Classifier,K Neighbors Classifier, Naive Bayes, Decision Tree Classifier, Dummy Classifier. Our results show that although all classifiers were able to identify COVID-19 coughs, the best performance was exhibited by the Catboost Classifier, which was best able to discriminate between the COVID-19 positive and the healthy coughs with an area under the ROC curve (AUC) of 0.9256. Since this type of cough audio classification is cost-effective and easy to deploy, it is potentially a useful and viable means of non-contact COVID-19 screening.

Chapter 1

Introduction

Cough and respiratory sound processing can assist in the early diagnosis of infections such as Covid-19. Even asymptomatic Covid-19 patients can be diagnosed early enough if appropriate speech modeling and signal-processing is applied. Covid-19 affects various speech subsystems that are involved in respiration, phonation and articulation.Clinically, coughs are identified by an underlying cause which can be due to common cold, bacterial infection, a hereditary disease like cystic fibrosis or viral infection such as influenza or coronaviruses. However, few coughs are also idiopathic like an acid reflux cough. These are presented in various forms like dry cough. SARS-CoV-2, a new coronavirus, has a trademark dry cough which is an important marker for analysis of cough to classify the disease correctly. Dry cough has a characteristic sound that comes from the upper respiratory tract while wet or chesty cough involves lower respiratory tract with mucus activity. In this project MFCC(Mel Frequency Cepstral Coefficients) features of audio samples have been used to identify positive and negative cough using various machine learning models.

The thesis is organized in five chapters. chapter 1 introduces the paper,Along with Data extraction from dataset and data preprocessing 2. Feature Extraction and dataset balancing 3. Various Machine Learning Models and Hyperparameter Tuning 4 Results obtained from implementation of Machine Learning Models in **??**

1.1 Literature Survey

COVID-19 (COrona VIrus Disease of 2019), caused by the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV2) virus, was declared a global pandemic on February 11, 2020 by the World Health Organisation (WHO). It is a new coronavirus but similar to other coronaviruses, including SARS-CoV (severe acute respiratory syndrome coronavirus) and MERS-CoV (Middle East respiratory syndrome coronavirus) which caused disease outbreaks in 2002 and 2012, respectively. The most common symptoms of COVID-19 are fever, fatigue and dry cough. Other symptoms include shortness of breath, joint pain, muscle pain, gastrointestinal symptoms and loss of smell or taste. The scale of the pandemic has caused some health systems to be overrun by the need for testing and the management of cases. Several attempts have been made to identify early symptoms of COVID-19 through the use of artificial intelligence applied to images. The 50-layer residual neural network (Resnet50) architecture has been shown to perform better than other pre-trained models such as AlexNet, GoogLeNet and VGG16 in these tasks. For example, it has been demonstrated that COVID-19 can be detected from computed tomography (CT) images with an accuracy of 96.23architecture.Coughing is one of the predominant symptoms of COVID-19 and also a symptom of more than 100 other diseases, and its effect on the respiratory system is known to vary . For example, lung diseases can cause the airway to be either restricted or obstructed and this can influence the acoustics of the cough. Respiratory data such as breathing, sneezing, speech, eating behaviour and coughing can be processed by machine learning algorithms to diagnose respiratory illness. Simple machine learning tools, like binary classifiers, are able to distinguish COVID-19 respiratory sounds from healthy counterparts with an area under the ROC curve (AUC) exceeding 0.80. Detecting COVID-19 by analysing only the cough sounds is also possible. Data collection from COVID-19 patients is challenging and the datasets are often not publicly available. Nevertheless, efforts have been made to compile such datasets. For example, a dataset consisting of coughing sounds recorded during or after the acute phase of COVID-19 from patients via public media interviews.The Coswara dataset is publicly available and collected in a more controlled and targeted manner [31]. At the time of writing, this dataset included useable 'deep cough' i.e. loud coughs recordings from 438 COVID-19 positive and 1794 healthy subjects. As the dataset is imbalanced SMOTE technique have been used for balancing the dataset. Also hyperparameter tuning is done on machine learning models to obtain the best results.

1.2 Data

Coswara Dataset is publicly open dataset hence was easily available. The pre-processed Coswara dataset, used for feature extraction and classifier training contains total 2278 audio samples out of which 482 are positive.The Subjects in dataset are from five continents: Asia (Bahrain, Bangladesh, China, India, Indonesia, Iran, Japan, Malaysia, Oman, Philippines, Qatar, Saudi Arabia, Singapore, Sri Lanka, United Arab Emirates), Australia, Europe (Belgium, Finland, France, Germany, Ireland, Netherlands, Norway, Romania, Spain, Sweden, Switzerland, Ukraine, United Kingdom), North America (Canada, United States), and South America (Argentina, Mexico). Age, gender, geographical location, current health status and pre-existing medical conditions are also recorded. Health status includes 'healthy', 'exposed', 'cured' or 'infected'. Audio recordings were sampled at 44.1 KHz.

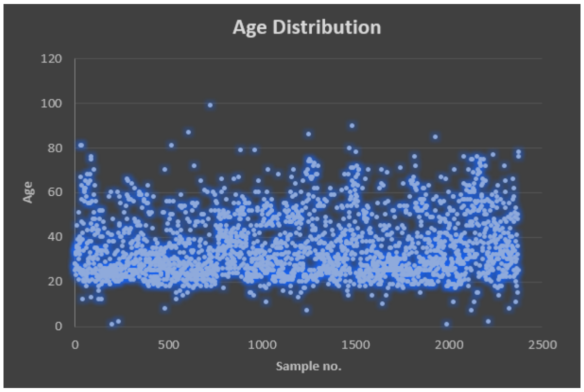

Figure 1.1: Age Distribution of Dataset

The Data contains 704 audio samples from females and 2373 samples from males.

1.3 Data Extraction And Pre-processing

The raw cough audio recordings from dataset have the sampling rate of 44.1 KHz and is subjected to some simple pre-processing steps, described below. We aim at finding the mfcc coefficients of each sample. In sound processing, the mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.

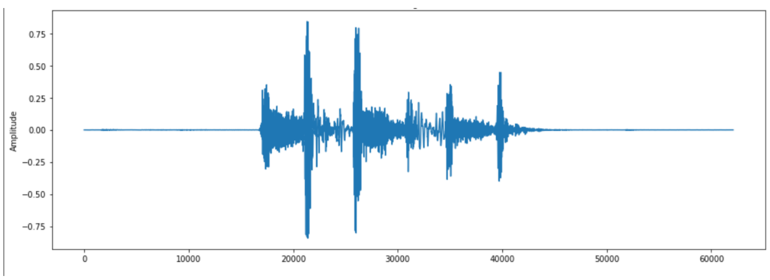

Figure 1.2: Raw wave of Cough Audio Sample

1.3.1 Spectrogram of Cough Audio Sample

A spectrogram is a visual way of representing the signal strength, or “loudness”, of a signal over time at various frequencies present in a particular waveform. Not only can one see whether there is more or less energy at, but one can also see how energy levels vary over time.A spectrogram is a detailed view of audio, able to represent time, frequency, and amplitude all on one graph. A spectrogram displays changes in the frequencies in a signal over time. Amplitude is then represented on a third dimension with variable brightness or color.

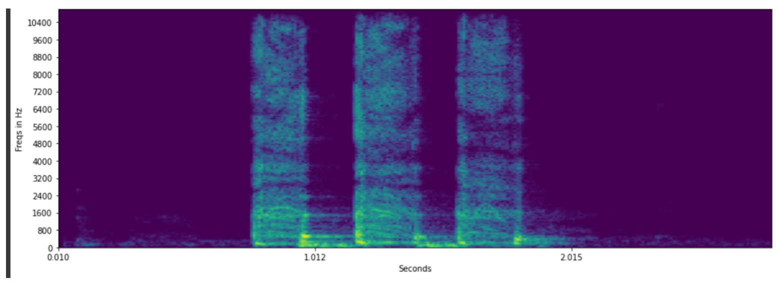

Figure 1.3: Spectrogram of Cough Audio Sample

1.3.2 Mel Power Spectrogram

The mel spectrogram remaps the values in hertz to the mel scale.The mel scale is a scale of pitches that human hearing generally perceives to be equidistant from each other. As frequency increases, the interval, in hertz, between mel scale values (or simply mels) increases. The linear audio spectrogram is ideally suited for applications where all frequencies have equal importance, while mel spectrograms are better suited for applications that need to model human hearing perception. Studies have shown that humans do not perceive frequencies on a linear scale. We are better at detecting differences in lower frequencies than higher frequencies. For example, we can easily tell the difference between 500 and 1000 Hz, but we will hardly be able to tell a difference between 10,000 and 10,500 Hz, even though the distance between the two pairs are the same. In 1937, Stevens, Volkmann, and Newmann proposed a unit of pitch such that equal distances in pitch sounded equally distant to the listener. This is called the mel scale. We perform a mathematical operation on frequencies to convert them to the mel scale.

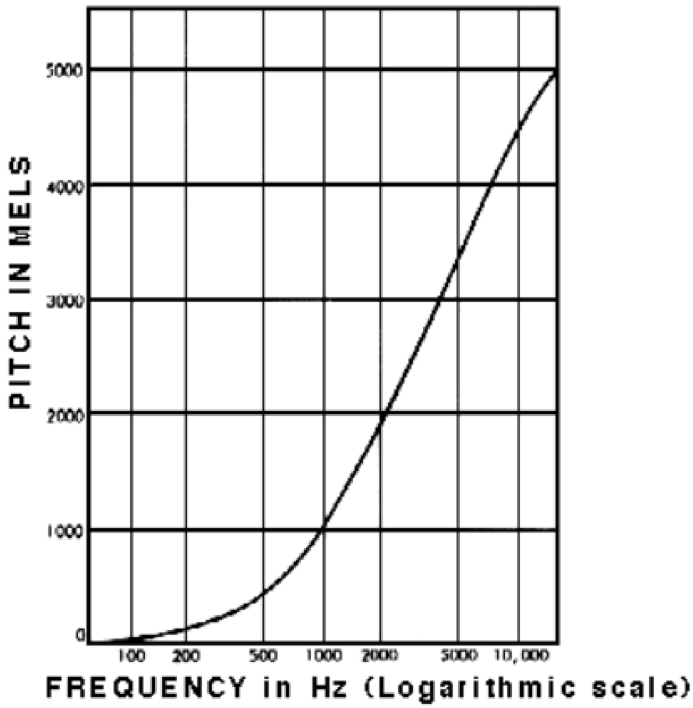

Figure 1.4: Mel Power Spectrogram of Cough Audio Sample

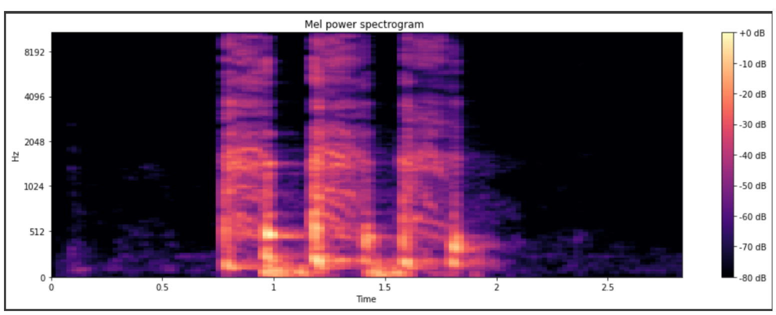

Figure 1.5: Mel Power Spectrogram of Cough Audio Sample

Chapter 2

Feature Extraction And Dataset Balancing

2.1 Feature Extraction

The features extracted from all the audio files of 'heavy cough' include chroma stft, RMS, Spectral centroid, Spectral Bandwidth, Roll off, Zero Crossing Rate, MFCC coefficients.

2.1.1 chroma stft

Chroma STFT The Chroma value of an audio basically represent the intensity of the twelve distinctive pitch classes that are used to study music. They can be employed in the differentiation of the pitch class profiles between audio signals. Chroma STFT is short-term Fourier transformation to compute Chroma features. STFT represents information about the classification of pitch and signal structure.

2.1.2 RMS

Compute root-mean-square (RMS) value for each frame, from a spectrogram. Computing the RMS value from audio samples is faster as it doesn't require a STFT calculation.

2.1.3 Spectral Centroid

The spectral centroid is a measure used in digital signal processing to characterise a spectrum. It indicates where the center of mass of the spectrum is located.

2.1.4 Spectral Bandwidth

The spectral bandwidth or spectral spread is derived from the spectral centroid. It is the spectral range of interest around the centroid, that is, the variance from the spectral centroid.

2.1.5 Roll off

Spectral roll off is the frequency below which a specified percentage of the total spectral energy lies.

2.1.6 Zero Crossing Rate

The zero-crossing rate (ZCR) is the rate at which a signal changes from positive to zero to negative or from negative to zero to positive.

2.1.7 MFCC coefficients

MFCCs have been used very successfully as features in audio analysis and especially in automatic speech recognition. They have also been found to be useful in differentiating dry coughs from wet coughs and classifying tuberculosis coughs Mel Frequency Cepstral Coefficients has 39 features.The 39 MFCC features parameters are 12 Cepstrum coefficients plus the energy term. Here we considered 20 MFCC features.

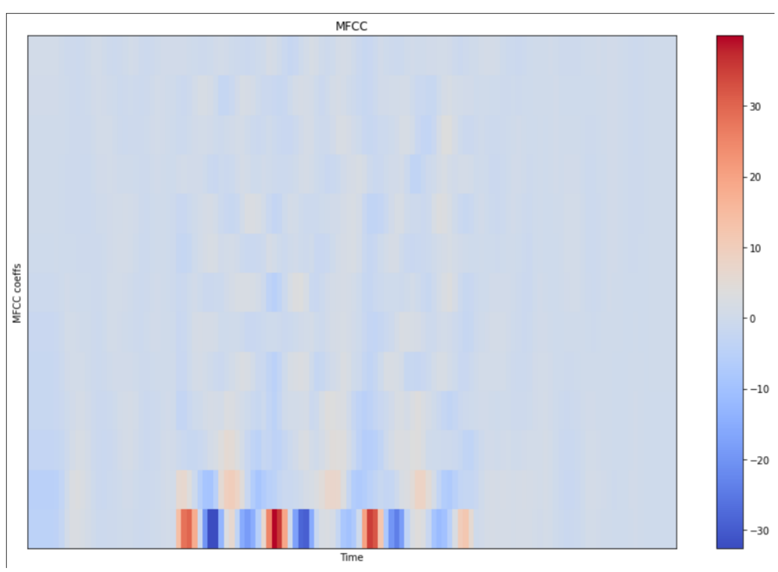

Figure 2.1: mfcc coefficients of cough audio samples

The following Steps are involved in finding out the coefficients: Frame the signal into short frames.

- For each frame calculate the periodogram estimate of the power spectrum.
- Apply the mel filterbank to the power spectra, sum the energy in each filter.
- Take the logarithm of all filterbank energies.
- Take the DCT of the log filterbank energies.
- Keep DCT coefficients 2-13, discard the rest. Also delta and delta-delta coefficients are calculated by differentiating the mfcc coefficients.

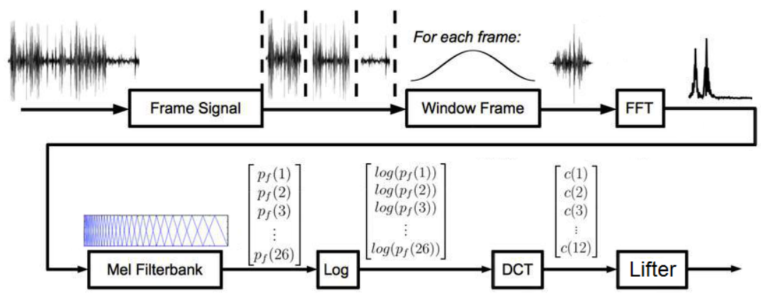

Figure 2.2: steps involved in finding mfcc coefficients

2.2 Dataset Balancing

The challenge of working with imbalanced datasets is that most machine learning techniques will ignore, and in turn have poor performance on, the minority class, although typically it is performance on the minority class that is most important. COVID-19 positive subjects are under represented in Coswara dataset. To compensate for this imbalance, which can detrimentally affect machine learning, we have applied SMOTE (Synthetic Minority Oversampling Technique) data balancing to create equal number of COVID-19 positive coughs during training. This technique has previously been successfully applied to cough detection and classification based on audio recordings.

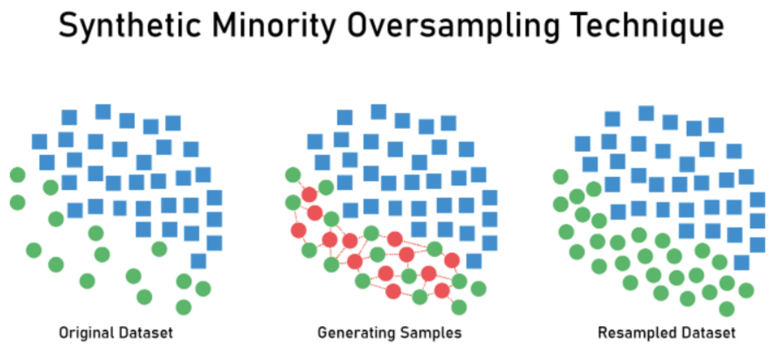

Figure 2.3: Dataset balancing using SMOTE

SMOTE oversamples the minor class by generating synthetic examples.SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space and drawing a new sample at a point along that line.

Chapter 3

Machine Learning Models

The training data must contain the correct answer, which is known as a target or target attribute. The learning algorithm finds patterns in the training data that map the input data attributes to the target (the answer that you want to predict), and it outputs an ML model that captures these patterns. Following are the various machine learning models we implemented on the dataset.

3.0.1 Catboost Classifier

CatBoost is based on gradient boosted decision trees. During training, a set of decision trees is built consecutively. Each successive tree is built with reduced loss compared to the previous trees. The number of trees is controlled by the starting parameters. To prevent overfitting, the overfitting detector is used. When it is triggered, trees stop being built.

3.0.2 Gradient Boosting Classifier

Gradient boosting classifiers are a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. Decision trees are usually used when doing gradient boosting. The main idea behind this algorithm is to build models sequentially and these subsequent models try to reduce the errors of the previous model. This is done by building a new model on the errors or residuals of the previous model. When the target column is continuous, we use Gradient Boosting Regressor whereas when it is a classification problem, we use Gradient Boosting Classifier. Hence, here classifier is being used. The only difference between the two is the "Loss function". The objective here is to minimize this loss function by adding weak learners using gradient descent. Since it is based on loss function hence for regression problems, we'll have different loss functions like Mean squared error (MSE) and for classification, we will have different for e.g log-likelihood.

3.0.3 Light Gradient Boosting Machine

Light Gradient Boosted Machine, or LightGBM for short, is an opensource library that provides an efficient and effective implementation of the gradient boosting algorithm. LightGBM extends the gradient boosting algorithm by adding a type of automatic feature selection as well as focusing on boosting examples with larger gradients. LightGBM splits the tree leaf-wise as opposed to other boosting algorithms that grow tree level-wise. It chooses the leaf with maximum delta loss to grow. Since the leaf is fixed, the leaf-wise algorithm has lower loss compared to the level-wise algorithm. Leaf-wise tree growth might increase the complexity of the model and may lead to overfitting in small datasets.

3.0.4 Extreme Gradient Boosting

Gradient boosting is asupervised learning algorithm, which attempts to acccurately predict a target variable by combining theestimates of a set of simpler, weaker models. When using gradient boosting for regression, the weak learners are regression trees, and each regression tree maps an input data point to one of its leafs that contains a continuous score. XGBoost minimizes a regularized (L1 and L2) objective function that combines a convex loss function (based on the difference between the predicted and target outputs) and a penalty term for model complexity (in other words, the regression tree functions). The training proceeds iteratively, adding new trees that predict the residuals or errors of prior trees that are then combined with previous trees to make the final prediction. It's called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

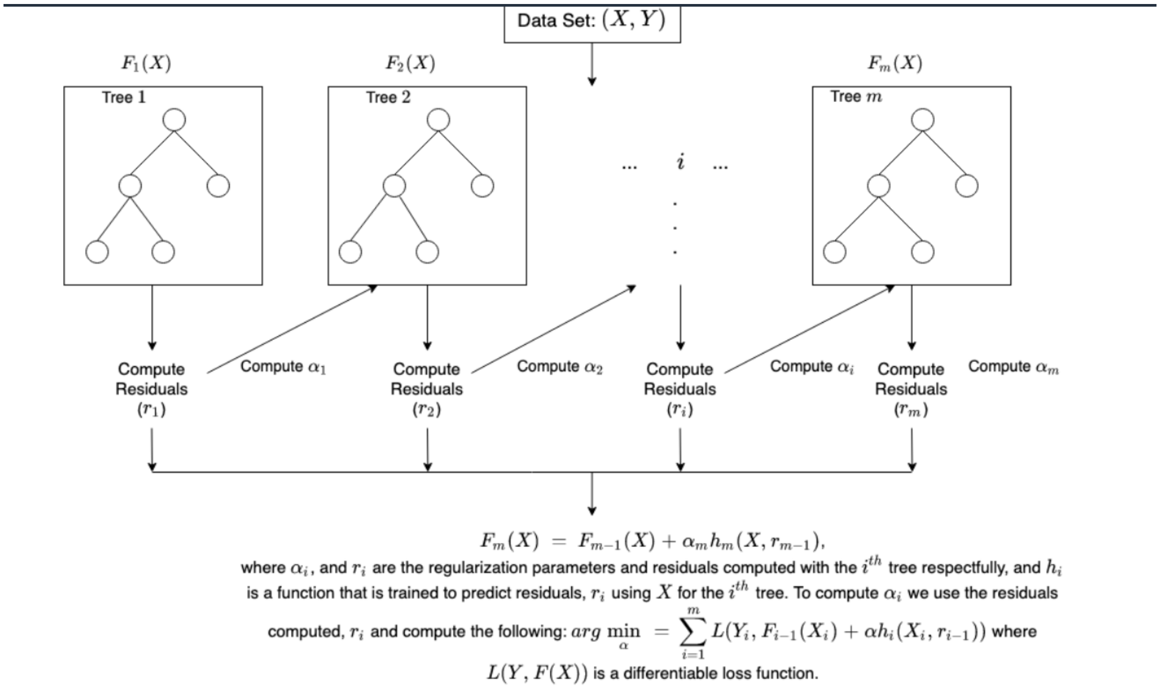

Figure 3.1: Brief description of how XGB works

3.0.5 Random Forest Classifier

It is suppervised learning algorithm and is widely used for classification and regression problems. Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. following are the steps involved in random forest classifier algorithm

- Step 1 First, start with the selection of random samples from a given dataset.
- Step 2 Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
- Step 3 In this step, voting will be performed for every predicted result.
- Step 4 At last, select the most voted prediction result as the final prediction result.

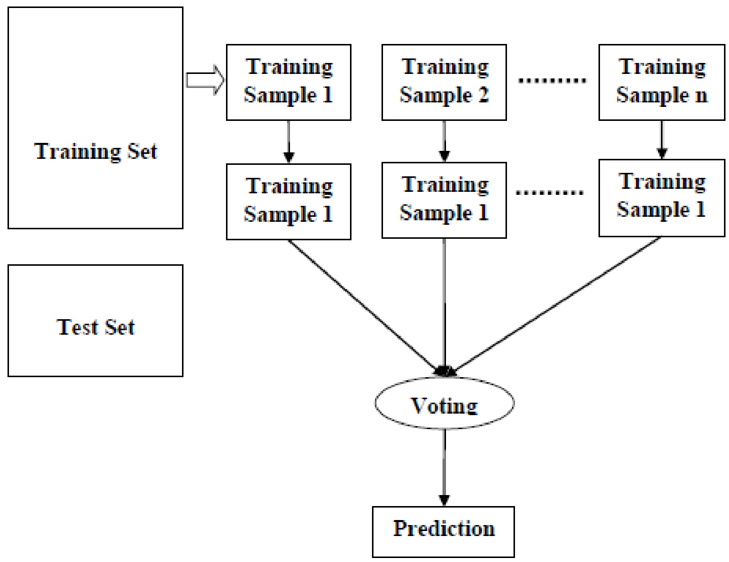

Figure 3.2: Brief description of how random forest classifier works

3.0.6 Extra Trees Classifier

Extremely Randomized Trees Classifier(Extra Trees Classifier) is a type of ensemble learning technique which aggregates the results of multiple de-correlated decision trees collected in a forest to output it's classification result.

3.0.7 K Neighbors Classifier

The k-nearest neighbors (KNN) algorithm is a data classification method for estimating the likelihood that a data point will become a member of one group or another based on what group the data points nearest to it belong to. The steps involved include:

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. -NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

3.0.8 Naive Bayes

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

3.0.9 Decision Tree Classifier

It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node.

3.0.10 Dummy Classifier

The dummy classifier gives you a measure of "baseline" performance–i.e. the success rate one should expect to achieve even if simply guessing.

3.1 Hyperparameter Tuning

For optimizing the the models, High end Optuna Hyperparameter optimization library is used on lightgbm model, the study is created for the optimization with 100 iterators, and the weight optimization over given features is as featured in the Parallel Coordinate plot. After performing the study of hyperparameter optimization the final product of optimum parameters is observed using the Optimization plot history Fig 7. The importance of each feature is determined using the feature plot. The best parameters that are obtained after tunning of LightGBM :

- boosting type: 'gbdt'.
- lambda l1: 2.9960439714443949.
- lambda l2: 0.0027925889876017735.
- colsample bytree: 0.5,
- bagging fraction: 0.8,
- feature fraction: 0.6,
- learning rate: 0.16714392495530678,
- max depth: 9,
- num leaves: 93,
- min child samples: 66.

the hyperparameter tunning improves the result of the model , the latency of the model to fail on test case data is highly optimized by applying the stratified K fold on validation set, same is done on Catboost and we got best result as discussed in result section .

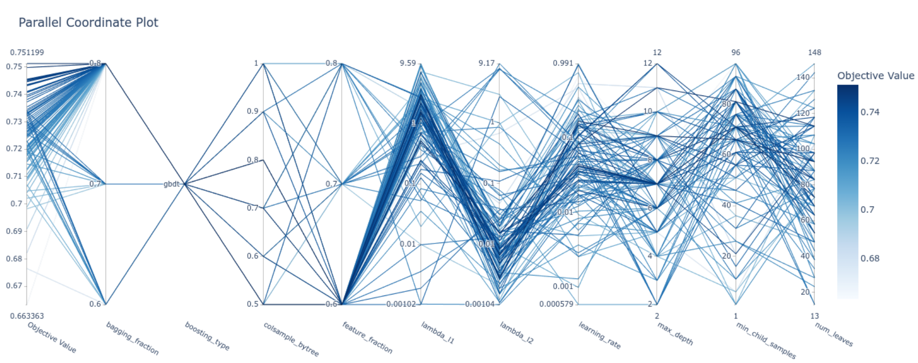

Figure 3.3: Parallel Co-ordinate Plot

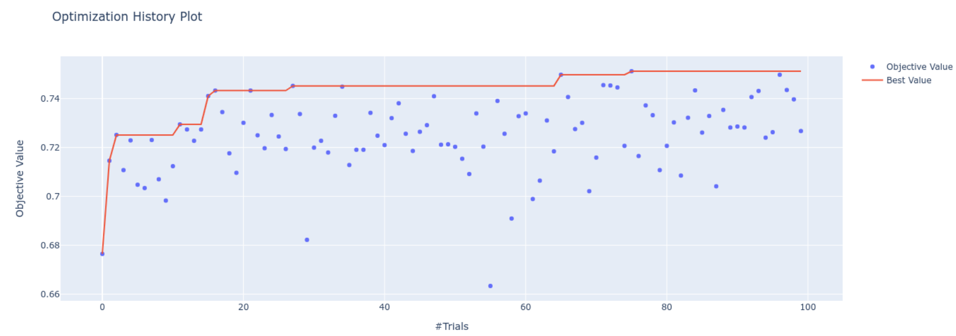

Figure 3.4: Optimization History Plot

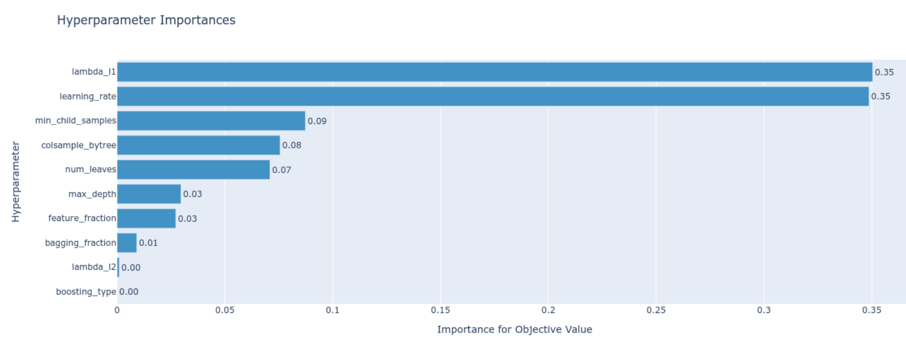

Figure 3.5: Hyperparameter Importance: feature plot

Chapter 4

Result

All the 10 machine learning models are capable of classifying the COVID-19 cough but the best accuracy is shown by Catboostclassifier followed by Gradient Boosting Classifier and Light Gradient Bossting Machine The Accuracy obtained by Catboost classifier Model is 89.75 with AUC score of 0.9256. Also significant increase is observed in the accuracy of models after hyperpaprameter tuning is performed. The AUC score and Accuracy obtained using each machine learning model is as follows:

Sr.no	Model	Accuracy(%)	AUC
1	Catboost classifier	87.95	0.9256
2	Gradient Boosting Classifier	87.51	0.9231
3	Light Gradient Boosting Machine	87.52	0.9226
4	Extreme Gradinet Boosting	87.01	0.9182
5	Random Forest Classifier	85.69	9043
6	Extra trees classifier	85.94	0.895
7	K-neigbors Classifier	82.86	0.8467
8	Naïve Bayes	79.91	0.7959
9	Decision Tree Classifier	84.8	0.7723
10	Dummy Classifier	78.53	0.5

Figure 4.1: Results of various ML models

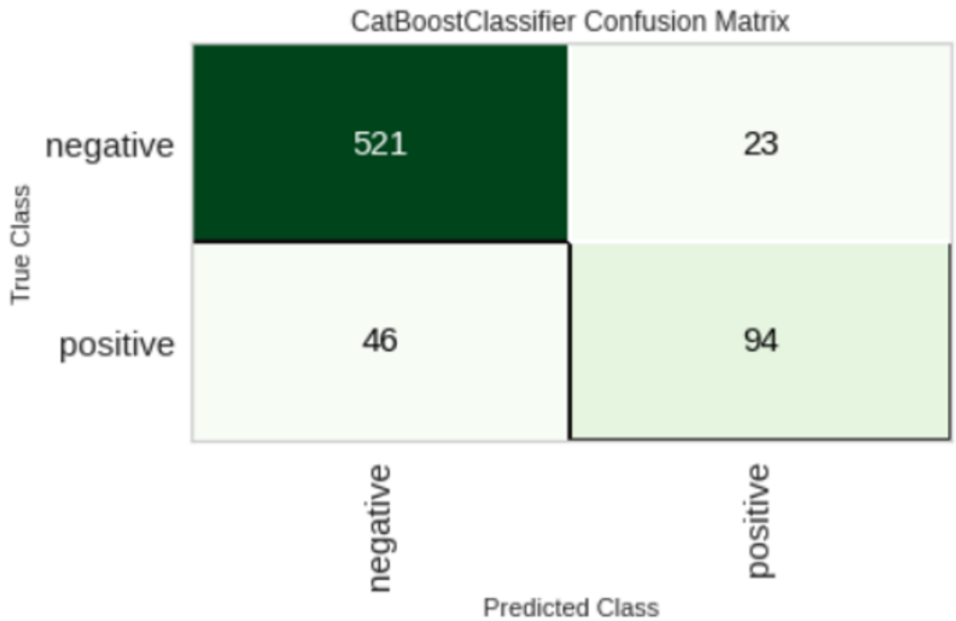

Figure 4.2: ROC curves for Catboost classifier

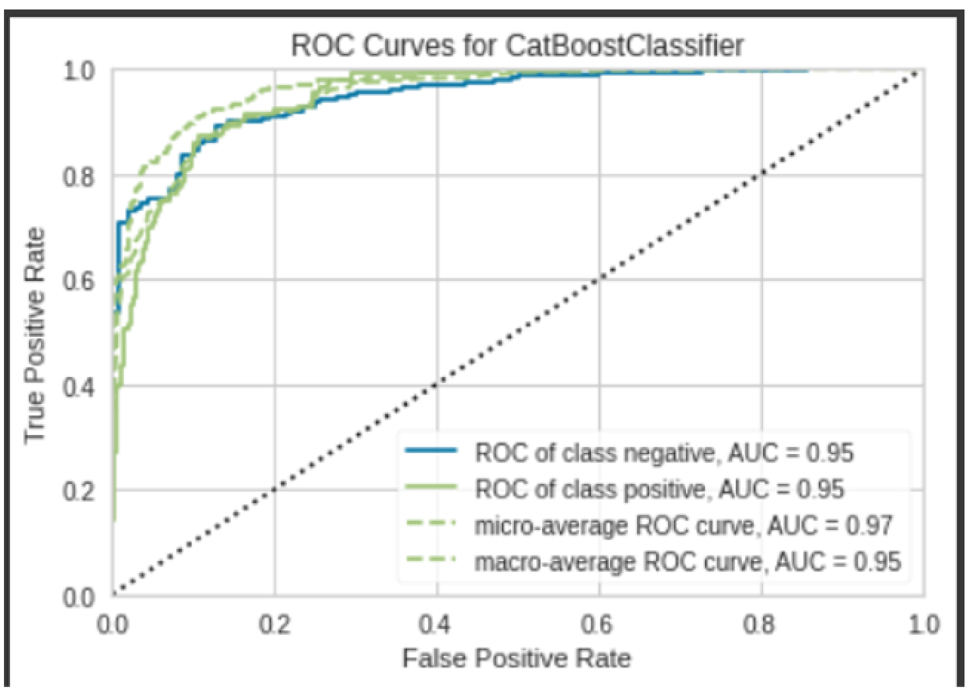

Figure 4.3: ROC curves for Catboost Classifier

Chapter 5

Conclusion and Future Scope

5.1 Conclusion

We have developed COVID-19 cough classifiers using smartphone audio recordings and ten machine learning models.To train and evaluate these models, we have used the Coswara dataset. Our bestperforming classifier is the catboost classifier architecture and is able to discriminate between COVID-19 coughs and healthy coughs with an AUC of 0.9256 on the Coswara dataset. The dataset is publicly available and contains data from 2277 subjects (483 COVID-19 postive and 1794 sujects healthy) residing on all five continents except Africa. since better performance is achieved using a larger number of MFCCs than is required to mimic the human auditory system, we also conclude that at least some of the information used by the classifiers to discriminate the COVID-19 coughs and the non-COVID coughs may not be perceivable to the human ear. Although the systems we describe require more stringent validation on a larger dataset, the results we have presented are very promising and indicate that COVID-19 screening based on automatic classification of coughing sounds is viable. Since the data has been captured on smartphones, and since the classifier can in principle also be implemented on such device, such cough classification is cost-efficient, easy to apply and deploy. Furthermore, it could be applied remotely, thus avoiding contact with medical personnel.

5.2 Future Scope

We have developed machine learning models for classifying Covid-19 positive and healthy samples. Further the dataset can be used to train CNN, DNN, ANN models such as Resnet50, VGG16 etc to get more accurate results. Also Hyperparameter tuning remains an important aspect while designing these models which can be implemented to get better results.

References

[1] 1. COVID-19 cough classification using machine learning and global smartphone recordings Computers in Biology and Medicine 135 (2021) 104572 (ELSEVIER) Madhurananda Pahar , Marisa Klopper , Robin Warren , Thomas Niesler

[2] 2. Cough Classification for COVID-19 based on audio mfcc features using Convolutional Neural Networks. 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON) Galgotias University, Greater Noida, UP, India. Oct 2-4, 2020 Vipin Bansal, Gaurav Pahwa ,Nirmal Kannan

[3] 3. Cough Sound Classification Based on Similarity Metrics 2021. 44th International Conference on Telecommunications and Signal Processing. IEEE Nikos Petrellis ,George K. Adam

[4] https://towardsdatascience.com/extract-features-of-music-75a3f9bc265d

[5] https://link.springer.com/content/pdf/bbm

[6] https://www.sciencedirect.com/science/article/pii/S0010482521003668

[7] https://medium.com/analytics-vidhya/bank-data-smote-b5cb01a5e0a2

[8] S. Bahl, R.P. Singh, M. Javaid, I.H. Khan, R. Vaishya, R. Suman. “Telemedicine technologies for confronting COVID-19 pandemic: A review,” Journal of Industrial Integration and Management, 2020, vol. 5(4), pp. 547–561, doi: https://doi.org/10.1142/S2424862220300057

[9] D. Dong et al., ”The role of imaging in the detection and management of COVID-19: a review,” IEEE Reviews in Biomedical Engineering, doi: 10.1109/RBME.2020.2990959.

[10] M. Pahar, I. Miranda, A. Diacon, T. Niesler, Deep neural network based cough detection using bed-mounted accelerometer measurements, in: ICASSP

2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 8002–8006, https://doi.org/10.1109/ICASSP39728.2021.9414744.

[11] L.L. Blagus, R, SMOTE for high-dimensional class-imbalanced data, BMC Bioinf. 14 (2013) 106, https://doi.org/10.1186/1471-2105-14-106.

[12] S. Walvekar, D. Shinde, et al., Detection of COVID-19 from CT images using Resnet50, in: 2nd International Conference on Communication Information Processing (ICCIP) 2020, 2020, https://doi.org/10.2139/ssrn.3648863. https://ssr n.com/abstract=3648863.

[13] N. Sharma, P. Krishnan, R. Kumar, S. Ramoji, S.R. Chetupalli, P.K. Ghosh, S. Ganapathy, et al., Coswara–A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis, arXiv preprint arXiv:2005.10548, 2020.