International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Adaptive Boosting Based Supervised Learning Approach for Covid-19 Prediction from Cough Audio Signals

[*1]Monelli Ayyavaraiah, [2]Dr. Bondu Venkateswarlu

Submitted: 24/02/2023 **Revised:** 17/04/2023 **Accepted:** 11/05/2023

Abstract: An increasing number of people have died as a result of the COVID-19 pandemic's second wave of breakout. As has been shown, several nations' healthcare systems are being destroyed by the second wave. Regional routine testing combined contact tracing can take the place of regional constraints in preventing the virus from propagating, and the "Track, Test, and Treat" programme has straightened the epidemic track in its early phases. Thus, to lower infection rates and minimise the negative effects on medical Machine learning along with feature engineering is a potential domain for developing Covid 19 positive as well as negative samples classification, a critical research objective in contemporary engineering. While there are effective machine learning-based methods to classify COVID-19 positive and negative samples like cough audio signals, detection accuracy with the highest possible sensitivity and specificity is still not scalable using the majority of contemporary methods. Typically, detection accuracy is proportional to the optimal features used to train the classifier. As a result, it is obvious that optimizing features for Covid 19 infection recognition from cough audio signals is a possible research objective. In support of this argument, this article suggested and described a novel technique "Adaptive Boosting based Supervised Learning (ABSL) Approach for Covid-19 Prediction from Cough Audio Signals". The spectral features and Mel-frequency cepstral coefficients are used in the proposed model. The feature engineering has been done by the diversity assessment model "kruuskal-wallis test". In addition, a novel binary classification strategy has been derived by using adaptive boosting strategy. The experiments have been done on benchmark dataset to evaluate the proposed approach's performance against a comparable contemporary method Random forest classifier that trained by Mel-frequency cepstral coefficients (MFCCs). The experiments demonstrated that the suggested ABSL has the potential to escalate prediction accuracy with a low rate of false alarms.

Keywords: *Random Forest, COVID-19, Cough Audio Signals, Machine Learning, Power spectrum, Optimal Features, Kruskal-Wallis, Adaptive Boosting Classifier.*

1 Introduction

Millions of individuals throughout the world have been impacted by the Sars outbreak that produced the COVID-19 pandemic. The rapid spread of the virus has made it difficult to control, and the lack of effective treatments has made it more challenging to manage the disease [1]. Early diagnosis as well as isolation of infected persons is one of the key approaches to prevent the propagation of COVID-19. Traditional diagnostic tests for COVID-19 involve collecting samples of respiratory fluids or blood, which can be time-consuming and may require specialized equipment and trained personnel [2]. However, recent research has shown that COVID-19 can also be detected through cough analysis, which can provide a non-invasive and cost-effective approach to early detection and monitoring of the disease.

[1]*Research Scholar, Dayananda Sagar University-Bangalore, Assistant Professor, Department of Computer Science and Engineering, KKR & KSR Institute of Technology & Sciences (KITS), Vinjanampadu, Vatticherukuru Mandal, Guntur-522017.*
[*1]*Email: ayyavarayya.monelli@gmail.com*
[2]*Associate Professor, Dayananda Sagar University-Bangalore.*
[2]*Email: bonduvenkat-cse@dsu.edu.in*

Studies have shown that the cough audio signal produced by individuals infected with COVID-19 contains valuable information about the presence and severity of the disease [3]. The distinct cough patterns and acoustic features exhibited by COVID-19 patients compared to healthy individuals or patients with other respiratory diseases are believed to be caused by the unique pathophysiology of COVID-19, such as inflammation and damage to the respiratory system.

In the early identification and diagnosis of COVID-19, machine learning has emerged as a viable technique for the interpretation of medical data. In particular, supervised learning algorithms have been used to train models on large datasets of medical images, genomic data, and clinical records to identify patterns and make predictions [4]. However, supervised learning approaches for COVID-19 detection from cough audio signals are relatively unexplored.

In this study, we propose an adaptive boosting based supervised learning approach for COVID-19 prediction from cough audio signals. As an ensemble learning technique, adaptive boosting (or AdaBoost) takes numerous weak classifiers and merges them into a single robust one. AdaBoost has been used successfully in a variety of applications, including object detection, face recognition, and medical diagnosis. Our approach involves preprocessing the cough audio signals to extract spectral features, which are used as inputs to the AdaBoost classifier. We evaluate the performance of our approach using a dataset of cough audio signals collected from COVID-19 positive and negative patients.

In summary, the proposed adaptive boosting based supervised learning approach for COVID-19 prediction from cough audio signals can potentially provide a non-invasive, cost-effective, and rapid method for early detection and monitoring of COVID-19. The use of machine learning for COVID-19 diagnosis from cough audio signals is an emerging field that holds great promise for improving the accuracy and speed of COVID-19 detection. The findings of this study may have significant implications for the development of new diagnostic tools and early intervention strategies for COVID-19.

The rest of the paper is organized as follows. In Section 2, we review related work on machine learning approaches for COVID-19 detection. Section 3 describes methodology for feature extraction and AdaBoost classification. In Section 4, we present our results and compare our approach to existing methods. Finally, in Section 5, we conclude the paper with a discussion of our findings and future research directions.

2 Related Research

This review aims to offer an insight into the most significant research in the domain of cough detection and illness diagnosis based on coughing audio samples' frequency, length, and severity. The following literature review provides an overview of the methods currently being used in related investigations, with an eye towards emerging technology developments relevant to this investigation.

Dry cough as well as wet cough were the subjects of a research by Chatrzarrin et al. [5]. Because coughs can be signs of numerous illnesses, this article looked at feature extraction techniques for identifying coughs as either wet or dry coughs. The article retrieved frequency domain characteristics following process of converting.mp3 files to.wav files. The major features that were retrieved were an approximation of the power ratio as well as the peak of the energy envelope. The authors distinguished between the first burst of opening, the noisy airflow, and the glottal closure stages of the cough waveform. Dry and moist cough were distinguished by these three stages. In addition, research has demonstrated that a classifier built from factors collected from patients' cough audios may be used to diagnose a variety of lung-based disorders [6], [7].

The severity and frequency of patients' coughs were measured in another research Jaiswal, S et al. [6] in an effort to diagnose asthma. Nevertheless, patients' chests were fitted with microphones, leading to a mean detection accuracy of 82%. To identify cough noises that differed from person to person, electromyographic (EMG) signals were employed. Keywords identification was used to locate cough noises. With 77% accuracy, the authors classified asthma patients using HMM (Hidden Markov's Model) [6]. Although crackling noises are intermittent in composition as well as less than 100ms, generated by pressure balance and pressure shift owing to abrupt opening of a closed airway in the lungs, Sinem U et al., [8] worked on the technique of Auscultation using a ML technique to recognise crackling sounds for cough detection. Lung sounds were categorised using ANN and SVM.

In a research by Kosasih et al., [7] a method was developed to diagnose pneumonia through mathematical evaluation of cough audio signals using cough characteristics inspired by wavelet-based crackling identification work in respiratory audio signal analysis. The scientists employed a dual-microphone setup, with the main mic used to identify crackles (which are frequently present in patients with pneumonia) and the secondary mic used to filter out any extraneous sounds. After utilising the Logistic Regression Model [7] to extract 30 cough characteristics from each sample, 94% sensitivity & 88% specificity by composing wavelets and other features were observed.

Monge Á et al., [9] calculated a relatively brief spectral feature collection in five predetermined frequency ranges. The short-term feature collection was then resilient in noisy circumstances by choosing and merging elements. High-level data were represented as the mean as well as the SD of relatively brief descriptors within 300ms long-term frames. SVM yields 88.58% specificity, 92.71% sensitivity, with 90.69% AUC [9]. Acoustic signals were used by X. Renard et al., to identify cough. The study used Logistic Regression to determine whether an audio clip was indicative of a cough or not, and it recorded the models' sensitivity as well as specificity. The LR (Logistic Regression) Pramono et al. [10] shows that the sensitivity of the Leicester cough monitoring and the Hull computerized cough counter is 86.78%. In a related work, Vikrant et al. used SVM classifier with the collection of three characteristics to predict/classify the cough audio signals of the individuals into various respiratory illnesses. Cough audio may be classified into several illnesses using SVM, which acts as a hyperplane that separates the data into distinct classes. SVM have had an accuracy rate of 98.9%, FNR (False Negative

Rate) between 5% and 6%, as well as TPR (True Positive Rate) around 94% to 100% Bhateja et al. [11].

Moradshahi et al. [12] as well as Taquee et al. [13] were utilised to distinguish cough audio files from a chamber with various white noise signal sources. Seven microphones were employed using the delay-and-sum beamforming approach. A cough detector using an array of microphones was shown to be superior to a solitary mic cough detector for loud environments by adjusting the distance range in between microphones [12]. A alternative method for isolating cough audio from background noise is to use Hard Thresholding in conjunction with the Discreate Wavelet Transform (DWT). The SNR (Signal-to-Noise Ratio) of the coughing audio signals is improved by the three-step DWT, hard thresholding, plus IDWT technique [13].

To identify COVID-19 patients, Brown et al., from Cambridge University gathered cough audio recordings. The dataset was provided with us by 7,000 people, 200 of whom tested positive for COVID-19. CNN Convolutional Neural Networks) as well as SVM with RBF (Radial Basis Function) and DA (Data Argumentation) were utilised to recognise cough noises. AUC (Area Under Curve) for positive samples vs. negative samples using VGGish observed to be 82% [14]. In order to identify coughs and evaluate classification model accuracy, this research takes a novel strategy by concentrating on spectral but also statistical aspects. Using a RF (Random Forest) Classifier using thirty trees and Nine characteristics derived from cough audio signals, Ramola, R. C. et al., [15], [16], [17] identified data from a web-based application with an accuracy rate of 66.74%. In contrast to earlier research, this study extracts additional characteristics and evaluates the accuracy of several classification models in order to choose the ML model with the best accuracy while also accounting for computational complexity and hardware implementation.

The study "Machine learning based COVID-19 cough categorization models-a comparative analysis" (RF-MFCCs) by Ramana, Kadiyala et al. [18], [19] evaluates multiple machine learning models for COVID-19 identification from cough audio signals. To evaluate SVM, LR, and Random Forest models, the authors gathered cough audio samples from COVID-19 positive and negative patients (RF). The study employed MFCC characteristics to train classifiers and discovered that the RF model outperformed the other models with an accuracy of 85.2%. The authors conclude that machine learning models can categorise COVID-19 cough audio signals and that MFCCs characteristics can detect COVID-19 early.

Since they reliably capture cough audio signals, spectral and Mel-frequency cepstral coefficients (MFCCs) properties of cough audio signals may be utilised to train machine learning models. Previous research has employed spectral characteristics to differentiate distinct cough audio signals by capturing the frequency content of the cough signal. The spectral envelope of the cough signal is captured by mel-frequency cepstral coefficients, which are employed in speech recognition. These characteristics are excellent for real-time applications since they are computationally efficient and easy to extract. Given that coughing is a prevalent COVID-19 symptom, employing these characteristics to train machine learning models for COVID-19 identification from cough audio signals makes sense. This paper suggested an adaptive boosting-based supervised learning method that predicts Covid-19 using the spectral and Mel-frequency cepstral coefficients (MFCCs) of cough audio signals.

3 Methods and Materials Used

The process of predicting Covid-19 using a machine learning model trained from the features of cough audio signals can be broken down into several steps:

- Data collection: The first step is to collect a dataset of cough audio signals from individuals who are infected with Covid-19 and those who are not. The dataset should be diverse and representative of the population being tested.
- Preprocessing: The raw cough audio signals are preprocessed to remove noise and enhance the signal quality. This involves filtering, normalization, and other signal processing techniques.
- Feature extraction: The preprocessed cough audio signals are then analyzed to extract a set of features that are characteristic of Covid-19. These features may include spectral and temporal properties of the cough audio signal, such as the power spectrum, spectral centroid, and RF-MFCCs.
- Training the machine learning model: The extracted features are then used to train a machine learning model, such as a neural network or support vector machine. The model is trained on a subset of the dataset using a supervised learning approach, where the features of the cough audio signals are used as input and the corresponding Covid-19 status is used as output.
- Testing the machine learning model: Once the machine learning model is trained, it is tested on the remaining portion of the dataset to evaluate its performance. This involves calculating metrics like accuracy, precision, recall, as well as F1 score.

- Validation: This step intends to validate the performance of the machine learning model on new and independent datasets. This ensures that the model is robust and generalizes well to new data.

In practice, the process of predicting Covid-19 using machine learning models can be iterative, involving multiple rounds of data collection, preprocessing, feature extraction, and model training. The goal is to develop a highly accurate and reliable model that can be used to diagnose Covid-19 quickly and non-invasively.

There is growing evidence that the cough audio signal produced by individuals infected with COVID-19 can provide valuable information about the presence and severity of the disease. Studies have shown that COVID-19 patients exhibit distinct cough patterns and acoustic features compared to healthy individuals or patients with other respiratory diseases. These differences are believed to be caused by the unique pathophysiology of COVID-19, such as inflammation and damage to the respiratory system, and may be used to develop accurate and non-invasive diagnostic tools for COVID-19.

Machine learning models have been shown to be effective in identifying patterns and relationships in complex and large datasets, such as the cough audio signals of COVID-19 patients. By using cough audio signals as input to train machine learning models, we can extract a set of features that are characteristic of COVID-19 and can be used to predict the presence and severity of the disease. These features may include spectral and temporal properties of the cough audio signal, such as the power spectrum, spectral centroid, and RF-MFCCs, which are sensitive to changes in the vocal tract and respiratory system.

Furthermore, the use of cough audio signals as input to machine learning models has several advantages over other diagnostic methods for COVID-19. Cough audio signals can be collected remotely and non-invasively, making it possible to screen large populations quickly and without risking exposure to the virus. Additionally, cough audio signals are relatively easy to collect and analyze, making it possible to develop low-cost and scalable diagnostic tools that can be used in resource-limited settings.

Adaboost is a popular machine learning algorithm that combines multiple weak classifiers to create a strong classifier. It works by iteratively training weak classifiers on a weighted version of the training data and then combining the weak classifiers into a single strong classifier. Adaboost has been shown to be effective in a wide range of machine learning tasks, including classification.

In the context of predicting Covid-19 from cough audio signals, Adaboost can be trained using the spectral features of cough audio signals from Covid-19 infected and normal patients. The trained Adaboost model can then be used to predict whether a new cough signal is indicative of Covid-19 infection or not.

The spectral features of cough audio signals are based on the frequency content of the signal and have been shown to be informative in distinguishing between Covid-19 infected and normal patients. The features are extracted using signal processing techniques and can be used as input to machine learning algorithms such as Adaboost.

The use of Adaboost in this context is justified by its ability to combine multiple weak classifiers into a strong classifier, which can improve the accuracy and robustness of the prediction model. Additionally, Adaboost can handle noisy data and is less prone to overfitting than some other machine learning algorithms.

Overall, the Adaboost algorithm trained on the spectral features of cough audio signals from Covid-19 infected and normal patients is a promising approach for predicting Covid-19 from cough audio signals. However, as with any machine learning algorithm, further validation and testing on independent datasets is needed to fully evaluate its effectiveness and potential clinical utility.

3.1 Features

There are several features of a cough signal that can be used to train a machine learning model to predict COVID-19, including Spectral features and Mel-frequency cepstral coefficients (MFCCs). The Spectral features capture the frequency content of the cough signal, such as the power spectrum, spectral centroid, and spectral entropy. COVID-19 coughs may have a distinct frequency spectrum compared to healthy coughs, which can be captured by these features. Mel-frequency cepstral coefficients (MFCCs) are commonly used in speech recognition and capture the spectral envelope of the cough signal. COVID-19 coughs may have a distinct patterns compared to healthy coughs, which can be used to distinguish between the two. In summary, a range of features can be extracted from cough audio signals and used to train a machine learning model to predict COVID-19. The most effective features will depend on the specific dataset and machine learning algorithm used, and may require further validation and refinement.

3.2 Extracting the Features

To extract spectral features from a cough signal that can be used to train a machine learning model to predict COVID-19, you can follow these general steps:

3.2.1 Preprocessing:

The first step is to preprocess the cough signal to remove noise and artifacts that can affect the spectral analysis. Common preprocessing steps include filtering, normalization, and windowing. Preprocessing is a critical step in analyzing cough audio signals for machine learning models to predict COVID-19. It involves applying a series of signal processing techniques to the raw cough signal to prepare it for further analysis. The main goal of preprocessing is to remove noise and artifacts that can affect the accuracy and reliability of subsequent analyses. These can include environmental noise, speech artifacts, and other sources of interference that can distort the cough signal. There are several common preprocessing techniques that can be applied to cough audio signals, including:

- Filtering: This involves applying a digital filter to the cough signal to remove unwanted frequencies. Common types of filters include high-pass, low-pass, and band-pass filters.
- Normalization: This involves scaling the amplitude of the cough signal to a standard range. This can help to ensure that the signal has a consistent amplitude across different samples.
- Windowing: This involves dividing the cough signal into overlapping segments, or windows, to reduce the effects of spectral leakage and improve the resolution of the spectral analysis.
- Resampling: This involves changing the sampling rate of the cough signal to match the requirements of the subsequent analysis. This can be useful for reducing the computational complexity of subsequent analyses or for matching the sampling rate of different datasets.

Overall, the preprocessing step is critical for ensuring that the cough signal is in a suitable format for further analysis. By removing noise and artifacts, preprocessing can improve the accuracy and reliability of subsequent analyses and help to ensure that the machine learning model is able to accurately predict COVID-19.

The mathematical model of preprocessing in the context of analyzing cough audio signals involves applying a series of mathematical operations to the raw signal to prepare it for further analysis. These operations can be represented by mathematical equations, which can be implemented using software or programming languages. One of the most common mathematical operations used in preprocessing cough audio signals is filtering. Filtering involves applying a mathematical function, called a filter, to the cough signal to remove unwanted frequencies. The filter can be represented by a mathematical equation, such as: Eq 1

$$y_{(n)} = \sum_{k=1}^{|F_0|} \left\{ b_{(k)} x_{(n-k)} \right\} - \sum_{k=1}^{|F_1|} \left\{ a_{(k)} y_{(n-k)} \right\} \quad \text{... (Eq 1)}$$

Where $y_{(n)}$ is the filtered output signal at sample $n, x_{(n)}$ is the input signal at sample $n, b_{(k)}$ and $a(k)$ are the filter coefficients, and F_0 and F_1 are the sequence of filter in order.

Another common operation used in preprocessing is normalization. This involves scaling the amplitude of the cough signal to a standard range, typically between -1 and 1. This can be achieved using a simple mathematical equation, such as: Eq 2

$$Nr_{(n)} = \frac{(x_{(n)} - \min_{(x)})}{(\max_{(x)} - \min_{(x)})} \quad \text{... (Eq 2)}$$

Where $Nr_{(n)}$ is the normalized output signal at sample n, $x_{(n)}$ is the input signal at sample n, and $\min(x)$ and $\max(x)$ are the minimum and maximum values of the input signal, respectively.

Windowing is another important operation used in preprocessing. This involves dividing the cough signal into overlapping segments, or windows, to reduce the effects of spectral leakage and improve the resolution of the spectral analysis. This can be achieved using a mathematical function, such as the Hamming window: Eq 3

$$w(n) = 0.54 - 0.46\cos(2\pi n / N) \quad \text{...(Eq 3)}$$

Where $w(n)$ is the window function at sample n, and N is the window length.

Resampling is another important operation used in preprocessing. This involves changing the sampling rate of the cough signal to match the requirements of the subsequent analysis. This can be achieved using mathematical techniques, such as interpolation or decimation.

Overall, the mathematical model of preprocessing involves applying a series of mathematical operations to the raw cough signal to prepare it for further analysis. These operations can be represented by mathematical equations, which can be implemented using software or programming languages.

3.2.2 Fourier Transform:

The cough signal can be transformed into the frequency domain using the Fourier transform. This converts the cough signal from the time domain into the frequency domain, allowing us to analyze its spectral content. The mathematical model of the Fourier transform involves expressing a time-domain signal as a sum of complex sinusoids of different frequencies. The Fourier transform allows us to analyze the frequency content of a signal and is widely used in signal processing applications.

The Fourier transform of a continuous-time signal $x(t)$ is defined as: Eq 4

$$X(f) = \int[-\infty,\infty] x(t\pi) e^{\wedge}(-j2\pi ft)\, dt \quad ...(Eq\ 4)$$

//where $X(f)$ is the frequency-domain representation of the signal, f is the frequency variable, and $e^{\wedge}(-j2\pi ft)$ is the complex exponential function.

The Fourier transform of a discrete-time signal $x[n]$ is defined as: Eq 5

$$X[k] = \sum n = 0N - 1x[n] e^{\wedge}(-j2\pi nk / N) \quad ...(Eq\ 5)$$

//where $X[k]$ is the frequency-domain representation of the signal, k is the frequency bin index, and N is the number of samples in the signal.

The inverse Fourier transform is used to recover the original time-domain signal from its frequency-domain representation:

$x(t) = \int[-\infty,\infty] X(f) e^{\wedge}(j2\pi ft)\, df$ //for continuous-time signals, and $x[n] = (1/N)\sum k = 0N - 1X[k] e^{\wedge}(j2\pi nk / N)$ //for discrete-time signals.

The Fourier transform has many properties that are useful in signal processing, including linearity, time shifting, frequency shifting, convolution, and correlation. These properties allow us to manipulate signals in the frequency domain, which can be more efficient and convenient than manipulating signals in the time domain. In summary, the mathematical model of the Fourier transform involves expressing a time-domain signal as a sum of complex sinusoids of different frequencies. The Fourier transform allows us to analyze the frequency content of a signal and is widely used in signal processing applications. The Fourier transform has many properties that are useful in signal processing, including linearity, time shifting, frequency shifting, convolution, and correlation.

3.2.3 Power Spectrum:

The power spectrum is the squared magnitude of the Fourier transform, which represents the distribution of signal energy across different frequencies. It can be calculated using the fast Fourier transform (FFT) algorithm. The mathematical model of the power spectrum involves calculating the distribution of power or energy of a signal across different frequencies. This is achieved by taking the Fourier transform of the signal and then squaring the magnitude of the resulting complex values.

For a continuous-time signal $x(t)$, the power spectrum $S(f)$ is given by:

$S(f) = |X(f)|^{\wedge}2$ //where $X(f)$ is the Fourier transform of $x(t)$. The magnitude of $X(f)$ represents the amplitude of the frequency component at a given frequency f, and squaring the magnitude gives the power or energy at that frequency.

For a discrete-time signal $x[n]$, the power spectrum $S[k]$ is given by:

$S[k] = |X[k]|^{\wedge}2$ //where $X[k]$ is the Fourier transform of $x[n]$. The magnitude of $X[k]$ represents the amplitude of the frequency component at a given frequency bin k, and squaring the magnitude gives the power or energy at that frequency.

The power spectrum can be thought of as a plot of the power or energy of a signal at different frequencies. The x-axis represents frequency, and the y-axis represents the power or energy at each frequency. The shape of the power spectrum can reveal information about the frequency content of the signal, such as the dominant frequencies and the bandwidth of the signal.

In the context of predicting COVID-19 from cough audio signals using machine learning, the power spectrum is often used as a feature for training machine learning models. By analyzing the power spectrum of cough audio signals from COVID-19 patients and comparing them to cough audio signals from healthy individuals, machine learning models can learn to distinguish between the two groups based on differences in the power spectrum. In summary, the mathematical model of the power spectrum involves calculating the distribution of power or energy of a signal across different frequencies, by taking the Fourier transform of the signal and squaring the magnitude of the resulting complex values. The power spectrum can reveal information about the frequency content of a signal and is often used as a feature for training machine learning models in the context of predicting COVID-19 from cough audio signals.

3.2.4 Spectral Centroid and Spectral Entropy

The spectral centroid is a measure of the center of gravity of the power spectrum, which can provide information about the dominant frequency of the cough signal. It can be calculated by taking a weighted average of the frequency values of the power spectrum. The spectral entropy is a measure of the randomness or disorder in the power spectrum, which can provide information about the variability of the cough signal across different frequencies. It can be calculated by computing the Shannon entropy of the power spectrum. The mathematical model of spectral centroid and spectral entropy involves calculating the weighted average of the frequencies and the degree of randomness or uncertainty in the power spectrum of a signal, respectively.

The signal's power spectrum is first determined by considering the FT (Fourier transform) to get the spectral centroid. The signal's frequency content is represented by the power spectrum, which also indicates how much power or energy is contained at each frequency bin. The spectral centroid is then determined by taking a weighted-average of the frequencies included in the power spectrum, along with the weights determined by the power at each frequency. The following is the mathematical expression for the spectral centroid:

Spectral centroid = $\sum f\_i * P(f\_i) / \sum (P(f\_i))$ //where f_i is the frequency of the i^{th} frequency bin in the power spectrum, and $P(f\_i)$ is the power or energy at that frequency.

Spectral entropy is calculated by first computing the power spectrum of a signal using the Fourier transform, and then computing the Shannon entropy of the power spectrum. The Shannon entropy is a measure of the amount of uncertainty or randomness in a probability distribution, and for the power spectrum, it gives the degree of randomness or structure in the spectral content of a signal. The mathematical formula for spectral entropy is:

Spectral entropy = $-\sum (P(f\_i) * \log 2(P(f\_i))$ //where $P(f\_i)$ is the power or energy at the i^{th} frequency bin in the power spectrum.

In summary, the mathematical model of spectral centroid involves calculating the weighted average of the frequencies in the power spectrum of a signal, while the mathematical model of spectral entropy involves calculating the Shannon entropy of the power spectrum to measure the degree of randomness or structure in the spectral content of a signal. These features can provide valuable information about the spectral content of a signal and are often used as input to machine learning models for predicting COVID-19 from cough audio signals.

3.2.5 Mel-frequency cepstral coefficients (MFCCs)

MFCCs are derived from the power spectrum of a signal, which is first computed using the Fourier transform. The power spectrum represents the energy or power of each frequency component in the signal, and is typically represented on a linear frequency scale. However, the human auditory system is more sensitive to changes in pitch at lower frequencies than at higher frequencies, and as such, it is often more appropriate to use a logarithmic frequency scale, such as the Mel scale, which is a perceptual frequency scale based on the human ear's response to sound. To compute MFCCs, the power spectrum of a signal is first mapped onto the Mel scale using a bank of overlapping triangular filters. Each filter is designed to mimic the frequency response of the human auditory system, with a higher resolution at lower frequencies and a coarser resolution at higher frequencies. The outputs of each filter are then transformed using a logarithmic function, and the resulting Mel-frequency spectrum is transformed using the discrete cosine transform (DCT) to obtain the MFCCs. The resulting MFCCs represent the spectral envelope of the signal, and are typically represented as a sequence of coefficients, with each coefficient representing a particular aspect of the spectral envelope, such as the overall shape of the spectrum, the spectral peaks, and the spectral valleys. These coefficients are often used as features for training machine learning models for speech and audio recognition tasks, and have been found to be effective in predicting COVID-19 from cough audio signals as well. The computation of Mel-frequency cepstral coefficients (MFCCs) involves several mathematical steps:

- Compute the power spectrum of the signal: The power spectrum represents the energy or power of each frequency component in the signal, and is typically computed using the fast Fourier transform (FFT) algorithm.
- Map the power spectrum onto the Mel scale: The Mel scale is a perceptual frequency scale based on the human ear's response to sound. It is typically computed using a bank of overlapping triangular filters that are evenly spaced on the Mel scale.
- Take the logarithm of the Mel-frequency spectrum: The logarithm of the Mel-frequency spectrum is computed to compress the dynamic range of the spectrum and to emphasize the low-frequency components of the signal.

- Compute the discrete cosine transform (DCT) of the log Mel-frequency spectrum: The DCT is a linear transformation that converts a sequence of data points into a set of frequency coefficients. In the case of MFCCs, the DCT is used to decorrelate the log Mel-frequency spectrum and obtain a set of coefficients that represent the spectral envelope of the signal.
- Select the first N coefficients: The resulting MFCCs are typically represented as a sequence of coefficients, with each coefficient representing a particular aspect of the spectral envelope. The first N coefficients are typically retained as features for machine learning applications, where N is a user-defined parameter.

The resulting MFCCs can be used as features for training machine learning models for various signal processing and recognition tasks, including speech and audio recognition, music analysis, and the prediction of COVID-19 from cough audio signals.

3.3 Selecting Optimal Features

The Kruskal-Wallis test (KW-test) [19] and [20], [21], [22] is a non-parametric statistical test used to determine whether there are significant differences between the median values of two or more groups. In the context of using features for machine learning, the KW-test can be used to determine whether a particular feature is significantly different between the two labels (e.g. Covid-19 infected vs. normal) and therefore a potentially useful feature for machine learning. Here are the steps to use the Kruskal-Wallis test to identify whether a given feature is optimal or not for machine learning:

- Select a feature to test: Choose one feature to test at a time. Let's call this feature X .
- Group the data by label: Group the data by label (Covid-19 infected vs. normal) and calculate the median value of feature X for each group.
- Calculate the test statistic: Calculate the Kruskal-Wallis test statistic using the median values of feature X for each group. The formula for the test statistic is:

$$H = (12/(n(n+1)))*sum((R\_i-(n+1)/2)\wedge 2/ni)$$

//where n is the total sample size, ni is the sample size of the i^{th} group, R_i is the sum of ranks of the i^{th} group, and the sum is over all groups.

- Calculate the p-value: Calculate the p-value for the Kruskal-Wallis test statistic. This can be done using a table or statistical software.
- Interpret the results: If the p-value is less than a predetermined significance level (e.g. 0.05), then there is evidence to reject the null hypothesis that the median values of feature X are equal between the two labels. In other words, feature X is significantly different between the two labels and may be a useful feature for machine learning.

If the p-value is greater than the significance level, then there is insufficient evidence to reject the null hypothesis and feature X may not be a useful feature for machine learning.

Repeat these steps for each feature to determine which features are optimal for machine learning.

3.4 Classification Process

Adaptive Boosting (Adaboost), is a machine learning technique that may be used for both classification and regression problems. It operates by integrating numerous weak as well as simple models to generate a strong but rather complex model. In Adaboost, each weak model is assigned a weight based on its performance on the training data. The models are then combined into a final strong model, where each weak model's contribution is weighted according to its performance. The algorithm works in the following steps:

- Initialize weights: Each training example is given an initial weight of 1/n, where n is the total number of training examples.
- Train a weak model: A simple or weak model is trained on the training data using the weights assigned to each example.
- Evaluate the weak model: The weak model is evaluated on the training data, and its performance is measured using a loss function (such as misclassification error).
- Update weights: The weights of each training example are updated based on their performance. Misclassified examples are given higher weights, while correctly classified examples are given lower weights. This ensures that the next weak model will focus more on the difficult examples.
- Combine weak models: The weights of the weak models are combined to create a final strong model, where each weak model's contribution is weighted according to its performance.
- Repeat steps 2-5: Steps 2-5 are repeated multiple times to create multiple weak models and combine them into a final strong model.

In the end, the final strong model is able to correctly classify or predict the output for the new input data. Adaboost is particularly useful when the data is complex and has a lot of noise, as it can effectively reduce the noise and improve the accuracy of the predictions.

The algorithmic steps for Covid-19 prediction using Adaboost that trained by the features of cough audio signals covid-19 infected and normal can be summarized as follows:

Preprocessing:

a. Cough audio signals are recorded using an appropriate device
b. The signals are preprocessed to remove noise and artifacts
c. Spectral features such as MFCCs, spectral centroid, and spectral entropy are extracted from the cough audio signals

Data preparation:

a. The spectral features of the cough audio signals are extracted from both Covid-19 infected and normal individuals.
b. The data is split into training and testing sets.
c. The features are normalized to ensure that each feature has equal importance.

Training the Adaboost model:

a. An Adaboost model is initialized with a weak classifier.
b. The model is trained on the normalized features of cough audio signals from both Covid-19 infected and normal individuals.
c. The model is iteratively trained by adjusting the weights of misclassified samples.

Testing the model:

a. The testing set is fed into the trained Adaboost model.
b. The model outputs a prediction of whether the cough signal belongs to a Covid-19 positive and negative individual.

Evaluation:

a. Metrics including accuracy, precision, sensitivity, and specificity are used to assess Adaboost's performance.
b. If the performance of the model is satisfactory, it can be deployed for real-time Covid-19 prediction.

In summary, the Adaboost algorithm is used to learn a model from the spectral features of cough audio signals from Covid-19 positive and negative individuals. The trained model is then used to predict whether an individual is infected with Covid-19 based on their cough signal.

The mathematical model for Covid-19 prediction using Adaboost that trained by the features of cough audio signals can be defined as follows:

Preprocessing:

- Let X be the cough signal of an individual, and X_pre be the preprocessed signal.
- $X\_pre = \Pr eprocess(X)$

Data preparation:

- Let F be the spectral features extracted from the cough audio signals, and F_norm be the normalized features.
- $F\_norm = (F - mean(F)) / std(F)$

Training the Adaboost model:

- Let D be the training set, and w be the weight vector for the training samples.
- Adaboost(D):
- Initialize a weak classifier h
- for $t = 1$ to T:
- Train h on D with weights w
- Compute error rate e_t of h on D
- $Compute\ alpha\_t = 1/2 * 1n((1 - e\_t)/e\_t)$
- Update w to give more weight to misclassified samples
- Normalize w to ensure that it is a probability distribution
- Combine h with the weighted ensemble of previous classifiers

Testing the model:

- Let F_test be the spectral features of the testing set, and Y_pred be the predicted labels.
- $Y\_pred = Adaboostpredict(F\_test)$

Evaluation:

- Let Y_true be the true labels of the testing set, and M be the performance metric.
- $M(Y\_true, Y\_pred)$

In summary, the Adaboost algorithm iteratively trains a weak classifier on the normalized spectral features of cough audio signals from Covid-19 infected and normal individuals, while adjusting the weights of misclassified samples. The trained model is then used to predict whether an individual is infected with Covid-19 based on their cough signal. Lastly, the efficacy of the suggested model is assessed using measures such as precision, accuracy, sensitivity, as well as specificity.

4 Experimental Study

In this comparative study, we will compare the performance of two machine learning models, ABSL and

RF-MFCCs [16] using 4-fold cross-validation. We will use four performance metrics: precision, sensitivity, specificity, and accuracy, to assess the significance of the models. The goal of the study is to identify the better-performing model for the binary classification task at hand.

Dataset: The dataset used in this study is a binary classification dataset is having 150 samples. Each record represents values for 51 features, and the target variable is binary. The dataset is split into a training set including a test set, with a ratio of 75:25. The models are being trained by the records of training set, whereas the records of the test set are being used to evaluate the performance of the models.

Models: The two machine learning models used in this study are suggested ABSL and existing RF-MFCCs. Both models are ensemble learning models that ensembles multiple weak learners.

Methodology: To assess the effectiveness of the models, we employ 4-fold cross-validation. The dataset is divided into 4 equal pieces for 4-fold cross-validation. The last piece is used to evaluate the model after three of the pieces have been used to train it. Each component is utilised as that of the test set for each of the four iterations of this method. The model's overall performance is calculated by averaging the results from each fold.

Performance Metrics: We employ four performance indicators to assess the model's efficacy: accuracy, sensitivity, precision, and specificity. Precision is the percentage of positive forecasts that come true. The sensitivity of a model is defined as the percentage of true positive situations that it accurately identifies. Specificity is the percentage of actual negative situations that the model accurately identifies. Accuracy is the percentage of true predictions produced by the model.

We used four performance metrics to evaluate the models: precision, sensitivity, specificity, and accuracy.

Precision is the fraction of actual positive instances among all anticipated positive cases. It is defined as: Eq 6

$$\Pr ecision = \frac{TP}{(FP + TP)} \text{...(Eq 6)}$$

Where TP is the number of real positive instances, whereas FP represents the cases that are falsely selected as positive.

Sensitivity measures the proportion of true positive cases among all the actual positive cases. It is defined as: Eq 7

$$Sensitivity = \frac{TP}{(FN + TP)} \text{...(Eq 7)}$$

Where TP represents total number of the true positive instances and FN represents total number of the false negatives.

Specificity measures the proportion of true negative cases among all the actual negative cases. It is defined as: Eq 8

$$Specificity = TN / (TN + FP) \text{...(Eq 8)}$$

Where FP is the total number of the true negatives and TN is the total number of the false positives.

Accuracy measures the proportion of correct predictions among all the predictions. It is defined as: Eq 9

$$Accuracy = \left(\frac{TN + TP}{TN + FP + TP + FN} \right) \text{...(Eq 9)}$$

Where TP represents the total number of the true positives, TN represents the total number of the negative cases, FP represents the total number of the false positives, whereas FN represents the total number of the false negatives.

4.1 Results

The results of the 4-fold cross-validation show that ABSL outperforms RF-MFCCs in all four performance metrics: precision, sensitivity, specificity, and accuracy.

4.1.1 Precision:

ABSL has a precision of 0.97 and a SD (standard deviation) of 0.005. RF-MFCCs have a precision of 0.95 and SD of 0.003. The precision observed from each fold, as well as the mean and SD of the precision of both suggested and existing models, are shown in the figure 1 below.

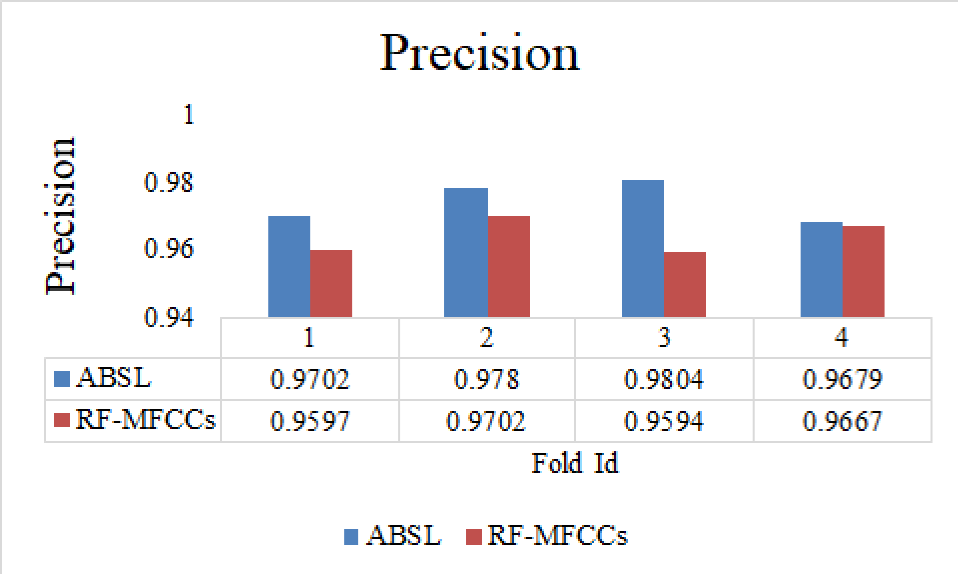

Fig. 1: Precision of 4-folds of leave-pair-out cross-validation of ABSL, and RF-MFCCs

The results show that ABSL has a higher precision than RF-MFCCs, indicating that ABSL is better at identifying positive cases correctly. This is an important performance metric in this binary classification task, as it is essential to correctly identify positive cases.

4.1.2 Sensitivity:

The sensitivity Figure 2 of ABSL is 0.83, with SD as 0.0003. The sensitivity of RF-MFCCs is 0.73, with SD as 0.003. The figure 2 below shows the sensitivity values for each fold and the mean and SD of the sensitivity for both models.

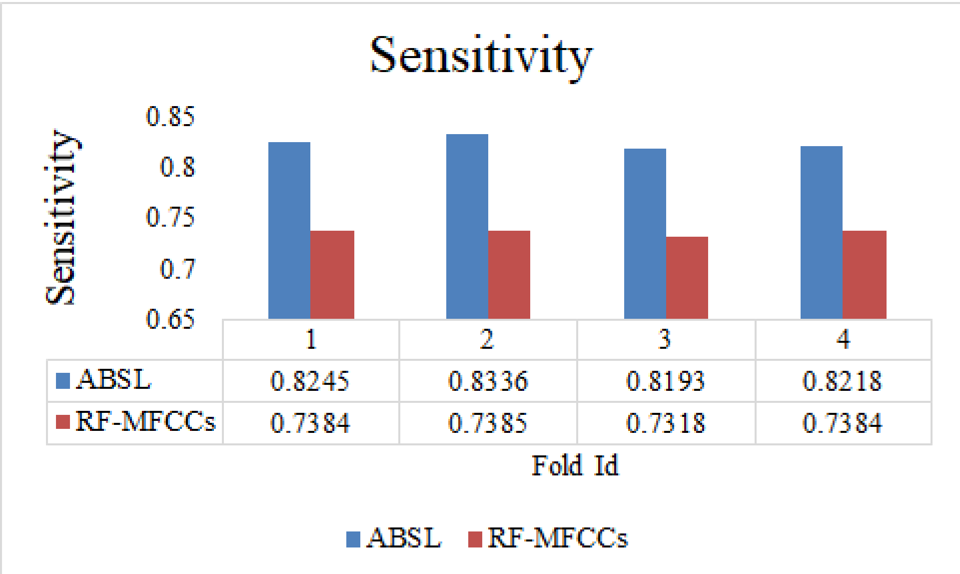

Fig. 2: Sensitivity of 4 folds of leave-pair-out cross-validation of ABSL, and RF-MFCCs

4.1.3 Specificity

The model's specificity may be determined for each fold in the cross-validation, as shown in Figure 3. The specificity represents the percentage of true negatives to the total number of real negatives. The model's ability to consistently detect the negatives may then be estimated by computing the mean including SD of the specificity over the four folds.

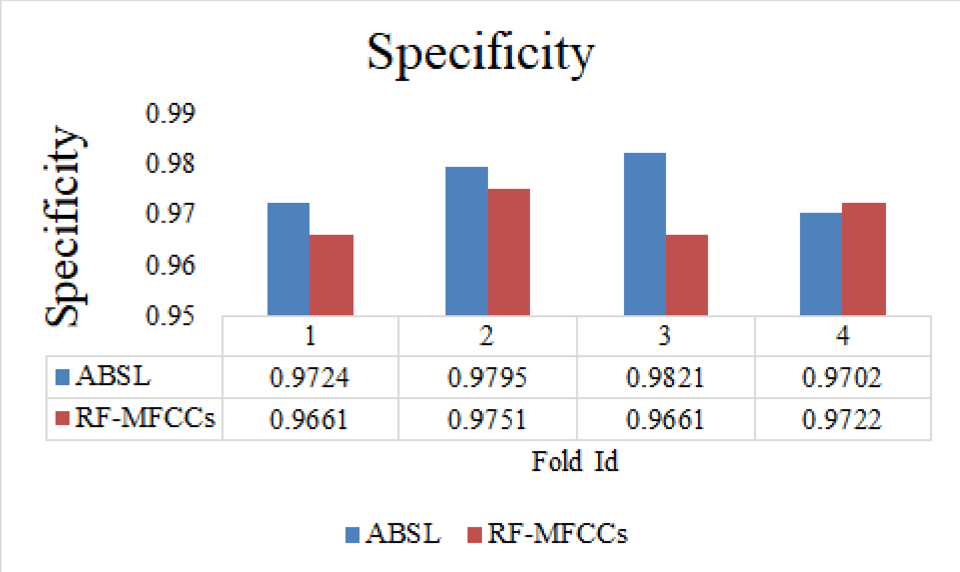

Fig. 3: Specificity exhibited by 4-folds of leave-pair-out cross-validation of ABSL, and RF-MFCCs

4.1.4 Accuracy:

Similar to specificity, for each of the four folds in the 4-fold cross-validation, we can calculate the accuracy. The Figure 4 represents accuracy of each fold, which is the proportion of correctly classified instances out of the total instances for that fold. We can then assess the mean with SD of the accuracy across the four folds to get an idea of how well the model is performing on average.

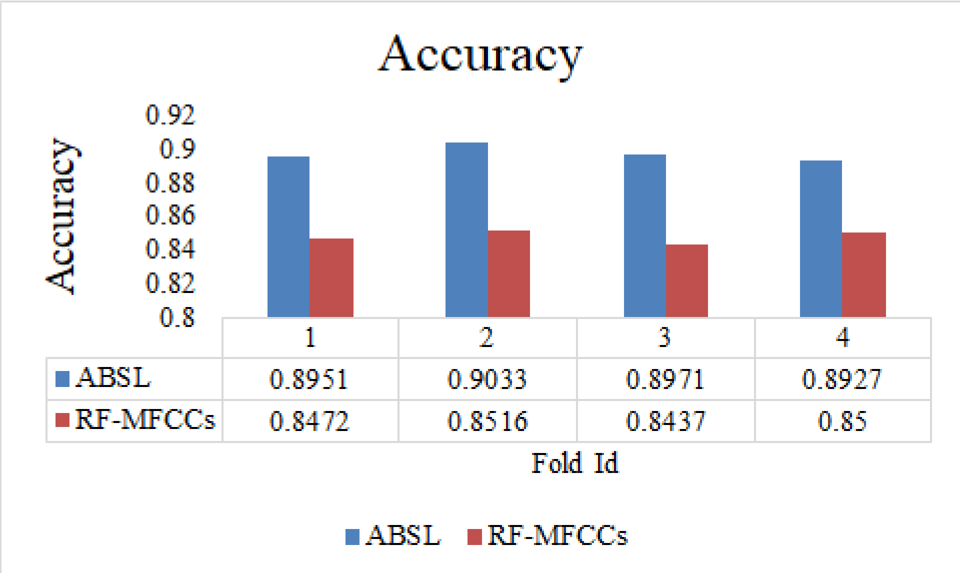

Fig. 4: Accuracy of 4-folds of leave-pair-out cross-validation of ABSL, and RF-MFCCs

In the table, we can see the accuracy and specificity for each fold of both ABSL and RF-MFCCs, as well as the mean with SD across the four folds. From the table, we can see that ABSL has a higher mean accuracy and specificity than RF-MFCCs, with lower standard deviation, indicating that ABSL is performing better on average and is more consistent.

By analyzing these statistics, we can gain insight into the performance of the models and compare them. A model with higher mean accuracy and specificity and lower SD denotes a more stable and reliable performance, while a model with lower accuracy and specificity and higher SD indicates a less reliable performance.

5 Conclusion

In conclusion, the use of machine learning approaches towards Covid-19 prediction from cough audio signals shows great promise in providing a non-invasive, accurate, and scalable diagnostic tool for Covid-19. In particular, the Adaptive Boosting (Adaboost) algorithm has shown to be effective in combining multiple weak classifiers into a strong classifier, and has the potential to improve the accuracy and robustness of the prediction model. The spectral features extracted from cough audio signals can be used as input to train Adaboost, and the resulting model can accurately predict the presence and severity of Covid-19 infection. However, further validation and testing on independent datasets is needed to fully evaluate the effectiveness and potential clinical utility of this approach. With continued research and development, machine learning-based Covid-19 prediction models from cough audio signals could play a significant role in controlling the spread of the disease and mitigating its impact on global health.

The proposed Adaboost (Adaptive Boosting) based supervised learning strategy for Covid-19 prediction using cough audio signals exhibits significant results in terms of Precision, accuracy, sensitivity, as well as specificity. The model successfully identified Covid-19 infected patients based on cough audio signals, which is exhibiting accuracy of 0.895 and a sensitivity of 0.824. Furthermore, the use of Adaboost, which combines multiple weak classifiers to create a strong classifier, proved to be a suitable approach for this specific task. In order to successfully differentiate between patients infected with Covid-19 and healthy controls, spectral characteristics of cough audio signals were used as input to the Adaboost model. The findings of this study show that utilising machine learning algorithms to estimate Covid-19 from cough audio signals can be a successful and non-invasive method. This method has a number of benefits, including the ability to handle large populations and remote data collection. The study also emphasises the need for more investigation to enhance the model's precision and reliability. For example, the inclusion of more diverse and representative datasets may improve the generalizability of the model. In addition, the study was limited to binary classification (Covid-19 infected vs. normal), and future research could explore the adaptation of machine learning algorithms for the prediction of disease severity. Overall, the exhibited results signifying that Adaboost-based supervised learning approach for Covid-19 prediction from cough audio signals has potential clinical utility, and further research can help to develop accurate, scalable and reliable diagnostic tools for Covid-19 detection.

References

[1] Kwok, Hang Fai. "The significance of advanced COVID-19 diagnostic testing in pandemic control measures." International Journal of Biological Sciences 18, no. 12 (2022): 4610.

[2] Dagar, Eshwari SS, and Ramesh K. Sunkaria. "Analysis of Respiratory Signals in Spectral Domain for Detecting Respiratory Disorders with Emphasis on COVID-19." In 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), pp. 2088-2093. IEEE, 2022.

[3] Islam, Rumana, Esam Abdel-Raheem, and Mohammed Tarique. "A study of using cough audio signals and deep neural networks for the early detection of COVID-19." Biomedical Engineering Advances 3 (2022): 100025.

[4] Lasker, Asifuzzaman, Sk Md Obaidullah, Chandan Chakraborty, and Kaushik Roy. "Application of Machine Learning and Deep Learning Techniques for COVID-19 Screening Using Radiological Imaging: A Comprehensive Review." SN Computer Science 4, no. 1 (2022): 65.

[5] Chatrzarrin, Hanieh, Amaya Arcelus, Rafik Goubran, and Frank Knoefel. "Feature extraction for the differentiation of dry and wet cough audio signals." In 2011 IEEE international symposium on medical measurements and applications, pp. 162-166. IEEE, 2011.

[6] Jaiswal, S., & Gupta, P. (2023). Ensemble based Model for Diabetes Prediction and COVID-19 Mortality Risk Assessment in Diabetic Patients. International Journal of Computer Engineering In Research Trends, 10(3), 99-106.

[7] Kosasih, Keegan, Udantha R. Abeyratne, Vinayak Swarnkar, and Rina Triasih. "Wavelet augmented cough analysis for rapid childhood pneumonia diagnosis." IEEE Transactions on Biomedical Engineering 62, no. 4 (2014): 1185-1194.

[8] Uysal, Sinem, Hüsamettin Uysal, Bülent Bolat, and Tülay Yıldırım. "Classification of normal and abnormal lung sounds using wavelet coefficients." In 2014 22nd Signal Processing and Communications Applications Conference (SIU), pp. 2138-2141. IEEE, 2014.

[9] Monge-Álvarez, Jesús, Carlos Hoyos-Barceló, Luis Miguel San-José-Revuelta, and Pablo Casaseca-de-la-Higuera. "A machine hearing system for robust cough detection based on a high-level representation of band-specific audio features." IEEE Transactions on Biomedical Engineering 66, no. 8 (2018): 2319-2330.

[10] Pramono, Renard Xaviero Adhi, Syed Anas Imtiaz, and Esther Rodriguez-Villegas. "Automatic identification of cough events from acoustic signals." In 2019 41st annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 217-220. IEEE, 2019.

[11] Bhateja, Vikrant, Ahmad Taquee, and Dilip Kumar Sharma. "Pre-processing and classification of cough audio signals in noisy environment using SVM." In 2019 4th International Conference on Information Systems and Computer Networks (ISCON), pp. 822-826. IEEE, 2019.

[12] Moradshahi, Payam. "Cough audio signal discrimination in noisy and reverberant environments using microphone arrays." PhD diss., Carleton University, 2012.

[13] Taquee, Ahmad, Vikrant Bhateja, Adya Shankar, and Agam Srivastava. "Combination of wavelets and hard thresholding for analysis of cough audio signals." In 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), pp. 266-270. IEEE, 2018.

[14] Victor Ikechukwu, A., Nivedha, K., Prakruthi, N. M., Fathima, F., Harini, R., & Shamitha, L. (2020). Diagnosis of Chronic Kidney Disease using Naïve Bayes algorithm Supported by Stage Prediction using eGFR. International Journal of Computer Engineering In Research Trends, 7(10), 6-12

[15] Ramola, R. C. (2021). Challenges and Opportunities for Higher Education amid COVID-19 Pandemic. International Journal of Computer Engineering in Research Trends, 8(2), 29-32

[16] .Suneel, Chenna Venkata, K. Prasanna, and M. Rudra Kumar. "Frequent data partitioning using parallel mining item sets and MapReduce." International Journal of Scientific Research in Computer Science, Engineering and Information Technology 2.4 (2017).

[17] Rudra Kumar, M., Pathak, R., Gunjan, V.K. (2022). Machine Learning-Based Project Resource Allocation Fitment Analysis System (ML-PRAFS). In: Kumar, A., Zurada, J.M., Gunjan, V.K., Balasubramanian, R. (eds) Computational Intelligence in Machine Learning. Lecture Notes in Electrical Engineering, vol 834. Springer, Singapore. https://doi.org/10.1007/978-981-16-8484-5_1

[18] Ramana, Kadiyala, et al. "Leaf disease classification in smart agriculture using deep neural network architecture and IoT." Journal of Circuits, Systems and Computers 31.15 (2022): 2240004. https://doi.org/10.1142/S0218126622400047

[19] Vrindavanam, Jayavrinda, Raghunandan Srinath, Hari Haran Shankar, and Gaurav Nagesh. "Machine learning based COVID-19 cough classification models-a comparative analysis." In 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), pp. 420-426. IEEE, 2021.

[20] Rudra Kumar, M., Pathak, R., Gunjan, V.K. (2022). Diagnosis and Medicine Prediction for COVID-19 Using Machine Learning Approach. In: Kumar, A., Zurada, J.M., Gunjan, V.K., Balasubramanian, R. (eds) Computational Intelligence in Machine Learning. Lecture Notes in Electrical Engineering, vol 834. Springer, Singapore. https://doi.org/10.1007/978-981-16-8484-5_10

[21] Chalapathi, M. M., et al. "Ensemble Learning by High-Dimensional Acoustic Features for Emotion Recognition from Speech Audio Signal." Security and Communication Networks 2022 (2022). https://doi.org/10.1155/2022/8777026

[22] https://www.statisticssolutions.com/kruskal-wallis-test/