Detection of Covid-19 using Cough Sounds

Harsh Taneja[*], Abhinav, Apoorv, Himanshu Mangal, Naman Agarwal
Bharati Vidyapeeth's College of Engineering, New Delhi, India.
Emails; harshtaneja.cse@gmail.com; abhi2000.rtk@gmail.com;
apoorvpeb1@gmail.com;himanshumangal09@gmail.com; aggarwalnaman025@gmail.com

***** Correspondence: harshtaneja.cse@gmail.com

Abstract

Coronavirus, the pandemic due to which about 4 million have lost their lives and counting, is still on. Many scientists and researchers are trying to find ways to detect coronavirus as soon as possible in the human body so that they can start their medication and precaution as soon as possible. Still, due to lack of lab facilities, the RT-PCR is taking more than three days to give the report, and in the meanwhile, patients get serious and life in danger. So in this paper, we proposed an audio-based coronavirus detection technique in which we can get results in minutes. Coronavirus is a respiratory disease, and the sound produced while breathing can tell us about the presence of coronavirus. Audio-based detection was already used for the detection of asthma, pneumonia. So, in this paper, we implemented a combination of machine learning and deep learning techniques to find the presence of Covid-19, and the model has an accuracy of 78% and an f1 score of 74%. This technique can be used as a starting point for just audio data to diagnose diseases and save lives.

Keywords: COVID-19, Cough, Voice, Machine Learning, VGG-19, Classifiers, KNN, Audio Analysis

1.Introduction

Covid-19, also known as Coronavirus [1], was declared as a global pandemic by the World Health Organization in March 2020 [2]. Since then, due to the devastating impact of Covid19 and the tragic loss of lives, leaders all across the world have been making policies trying to stop the spread of this deadly disease. It has become a topic of utmost concern to develop methods for early detection of Covid-19, which may help limit its spread. Medical workers have been working day and night to develop various vaccines and trying to improve their efficiency. In many countries like India, the USA, and the UK, mass production of vaccines has already begun, and people have been given vaccines for Covid-19. Meanwhile, scientists have been trying to come up with different ways to detect Covid-19 in patients at early stages using Machine Learning and Artificial Intelligence. Some of them have come up with methods that have very high accuracy, such as detection of Covid-19 using chest X-Rays and CT scans and respiratory sound data [3]. In this paper, we will be discussing our research on the Detection of Covid-19 using Audio analysis of respiratory data.

Patients' early symptoms include breathlessness, high fever, tiredness, and dry cough [1]. Out of these, dry cough is a symptom seen in over 65% of the cases at a very early stage. Cough audio sounds can be used to extract information on pulmonary health with the help of Machine Learning and Artificial Intelligence.

We are using nine different audio samples of individuals, including cough sounds(shallow and heavy) and breathing(shallow and deep), and others. The dataset used is provided by IISC Banglore [4]. These audio files in the dataset are cleaned using the Librosa library [5]. The audio data needs to be converted to images to perform feature extraction for better understanding. This is done using Mel Spectrogram [6]. We use a VGG model with 19 layers of

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

weight for feature extraction, which provides us with approximately 25,000 features [7]. To avoid overfitting, we use PCA(Principal Component Analysis) to reduce features from 25088 to 200 [7]. Finally, to detect whether the tested individual is infected or not we use different classification algorithms. XGBoost Classifier, Random Forest, KNN, SVM, Logistic Regression, Linear Discriminant Analysis, Bayes, Quadratic Discriminant, Decision Trees are the several classification algorithms we used. As a measure of choosing the best, we consider the F1 score to judge the algorithms out of which K-Nearest Neighbors Algorithm(KNN)[8]. KNN provides us with the highest F1 score of 74.02%. Further, by ensembling all the nine classifiers, we achieved an F1 score of 66.67% for hard ensembling, which is less than the F1 score for KNN. Hence we give our results with KNN.

This paper is organized as follows: Section II covers Related Works. Section III provides a brief about the Audio Dataset Used. Section IV discusses the Working Methodology and Pipeline Structure of the ML model. Finally, in Section V, we give the Results and Conclusion of our research.

2. Related Work

In the past few years, studies have shown that various respiratory diseases such as tuberculosis, pneumonia, asthma, and bronchitis can be detected using acoustic characteristics of audio voice samples. Some of the studies and related works have been discussed below.

An integrated model of audio and image processing [1][9] was used by the healthcare division for the detection of diseases like tuberculosis with high accuracy. It was achieved by recording verbal communication of patients. This model identifies the pain in a patient's voice using machine learning tools like CNN(Convolutional Neural Network)

Damage in pulmonary and vocal regions due to respiratory diseases can be easily identified by audio analysis of cough and breathing sounds [1]. The differences in cough sounds of patients suffering from asthma, bronchitis, and pneumonia can be analyzed using speech recognition techniques [10]. Using a combination of vocal features based on parameters obtained from cough sounds alone, a sensitivity of 94% and specificity of 75% can be achieved [10].

The difference between dry and wet cough can be identified using audio signal spectral energy, temporal envelope, and time-independent waveform [11]. When trained against 536 samples, a recall of 55% and specificity of 93% were obtained. In the same way, audio cough samples [12] show that patients who have asthma have higher energy signatures when compared to non-asthmatic patients. Reference [13] provides a comprehensive review of the detection and analysis of respiratory diseases using audio and speech analysis so far.

As the surge of Covid-19 cases increased, researchers started studying different methods to detect the presence of Covid-19 at early stages to prevent the virus from spreading. They gave a brief overview of initiatives taken by researchers so far to detect Covid-19 using Machine Learning and AI.

As the number of cases of Coronavirus disease increased, it was visible that cough was a common symptom in 65% of cases. Hence, some research work has been done on the detection of Covid-19 using cough sound analysis. But coughing is also a symptom for over Thirty more non-COVID-19 diseases [14]. Therefore this made it even more challenging, use cough as a differentiator for COVID-19. A literature review of work done on the detection of Covid-19 using cough analysis is presented in[15].

Ali Imran et al. [14] began to investigate the pathomorphological alterations in the respiratory system caused by the COVID-19 infection and compared them to other non-COVID 19 diseases. In order to deal with the shortage of data, they used transfer learning [16]. They employed three different models along with a mediator to avoid false-positive results. They used Deep Learning-based classifiers [14][16] and Classical ML-based classifiers under the hood [14]. The model also has a mediator which returns inconclusive results if any of the output from three classifiers mismatches. Their model is able to identify covid 19 coughs from several other coughs. Due to the lack of audio data, they shifted to a domain-aware approach and began to investigate the distinctness of the pathomorphological alterations, thus overcoming the lack of data [14].

Neeraj Sharma et al. [4] created a database called Coswara, a respiratory system for cough, breath, and voice sounds. They used a website application for collecting sound samples via worldwide crowdsourcing. The dataset is divided into nine different categories these are i)breathing shallow, ii)breathing deep iii)cough shallow iv) cough heavy v) sustained vowel phonation a, vi) sustained vowel phonation e, vii) sustained vowel phonation-o, viii) one to twenty digits counting in normal pace vii) one to twenty digit counting in fast-paced. They have used a random forest classifier [17] with default parameters and 30 trees, and the model was able to achieve an accuracy of 66.74% on the test dataset.

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

Brown, Chloë, et al. [18] started collecting voice sounds(cough and breath sounds) for the detection of covid -19. So they came up with a cross-platform application for the collection of crowdsourced data. They built a dataset consisting of voice sounds to distinguish between COVID-19, asthma, and healthy people [18]. They used three binary classifiers to distinguish between COVID-19 patients and healthy people, distinguish COVID-19 patients having cough from healthy people who have a cough, and distinguish between COVID-19 patients with cough from those having asthma who had a cough. Their dataset consisted of around 7000 unique people, out of which more than 200 were detected as COVID-19 positive [18]. Due to the smaller amount of data, the researchers applied standard audio augmentation techniques to increase the sample size of the dataset. They tried out three classifiers - Logistic Regression [19], Gradient Boosting Trees [20], and SVM [21] and got more than 70% in AUC scores in all three binary classification tasks. When they used only breath sounds for classification, they got an AUC score of about 60%, but when the researchers combined the cough and breathing sound for classification, the AUC score improved to 80% [18].

Junaid Shuja et al. [22] introduced a new approach in which they used a portable smartphone enabled with a spirometer with automated disease classification using CNN. This approach was found to have good results for the classification of other diseases, so the authors tried to use the same technique for COVID-19 detection. They proposed a system consisting of three basic modules [22]. A fleisch type airflow tube for capturing the breathing sound using a differential pressure-based approach, a Bluetooth-enabled microcontroller for data processing, and lastly, an Android application with a pre-trained CNN model to classify breath sounds. The authors examined various classifiers such as stacked AutoEncoders, Long Short Term Memory Network, and CNN and used them for lung diseases. In some cases, 1D CNN classifiers performed well with higher accuracy than other ML classifiers, so they thought to use the same concept for the classification of COVID-19. In Order to improve the results of the existing applications, more audio data is required from COVID-19 patients [22].

3. Description of Audio data

In this paper, we have used the Open Source Coswara dataset, which is uploaded by IISC Banglore [4]. To collect this dataset they have made a website that supports both laptop and mobile applications. The average time a user devotes to recording audio on this website is about 5-7 minutes. Figure 1 shows the categories in which the coswara dataset was divided. There are seven sound categories: Healthy, positive mild, positive asymptomatic, positive moderate, respiratory illness not identified, no respiratory illness exposed, and recovered, full of which a user has to choose their condition. In every subcategory, there are nine audio classes, namely, cough heavy, cough shallow, breathing shallow, breathing deep, sustained vowel, vowel e, vowel o, one to twenty digit counting normal, one to twenty digit counting fast-paced. These audio samples are recorded at a sampling frequency of 48 kHz [4].

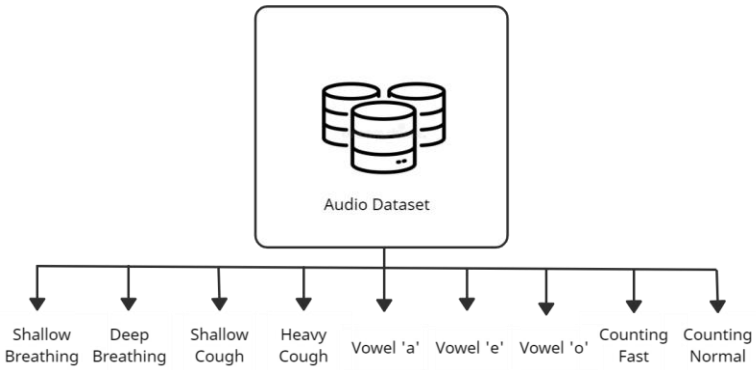


Fig. 1 Classification of Coswara Dataset

4. Methodology

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

The complete pipeline of our work can be divided into a total of 4 sets of phases: cleaning the audio, audio conversion into images using Mel spectrogram [23], image feature extraction using VGG [7], and finally applying image classifiers to get the results. Figure 2 shows the overall pipeline that we used in our experimentation.

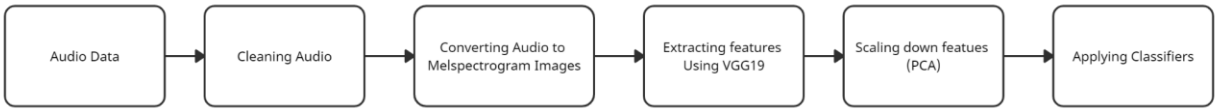


Fig. 2 Complete Pipeline of Experimentation

a. **Audio cleaning**

In the first phase of our methodology, the cleaning part of the audio is done. The overall functionality can be seen in Figure 3. We are using librosa library [5] to read the audio file. After reading, some basic audio cleaning techniques are applied, like making every audio file of the same length as 5 seconds. If the audio file size is greater than 5 seconds, then we trim the audio file, and if the audio file is less than 5 seconds, then we add a little bit of padding to it. The ultimate aim for this step is to make audio files of the same size.

Fig. 3 Steps to clean Audio Dataset

b. **Spectrogram generation**

In order to implement image classification techniques, raw audio data is first converted to a spectrogram representation. The Mel spectrogram acts like a transformation that details the frequency composition of a signal over time. In order to classify audio files, we are converting audio into image files and then using image classification techniques like CNN's to classify between positive and negative covid victims [24]. The recorded audio signal is the amplitude of air pressure over time. We can work on this waveform, but the resulting plot is a mess, and both humans and machines have difficulty understanding [22]. So to overcome this, we can use Fourier transform to transform the time domain into the frequency domain. The Fourier transform is computed on overlapping window segments of the signal, and we get a spectrogram [4]. The sampling frequency of the recording is 16000 kHz, and there is a well-known fact that humans can detect differences in lower frequency better than in higher frequency. So, there is a mel scale to convert f hertz into m mels [25].

$$m = 2595 log_{10}\left(1 + \frac{f}{100}\right) \qquad (1)$$

So after using this formula, we get mel spectrograms of each audio and then save each mel spectrogram in the respective image folder. Then these images are further used for later processing. After this, we're able to apply image classification techniques [23]. Consequently, the audio signal is represented as an image. It allows us to apply various image classifiers to it.

c. **Feature Extraction**

Further, We are using a pre-trained VGG model to extract features from the generated spectrogram images. A complete overview of the process can be seen in Figure 4. The Visual Geometry Group at the University of Oxford proposed the VGG convolutional neural network model for image recognition, where VGG16 refers to a VGG

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

model with 16 weight layers and VGG19 refers to a VGG model with 19 weight layers. The architecture of VGG 19 [26] is shown in Fig. 5: the input layer accepts an image of size (224 x 224 x 3), and the output layer is a 1000-class softmax prediction. The feature extraction part of the model runs from the input layer to the last max-pooling layer (labeled by 7 x 7 x 512), while the classification part of the model runs from the input layer to the previous max-pooling layer (labeled by 7 x 7 x 512). VGG is a convolutional neural network that is 50 layers deep. We can load a pre-trained version of the network trained on more than a million images from the ImageNet database. The pre-trained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals [4] [27]. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224. We need to convert every image into a fixed-sized vector which can then be fed as input to the final classifier.

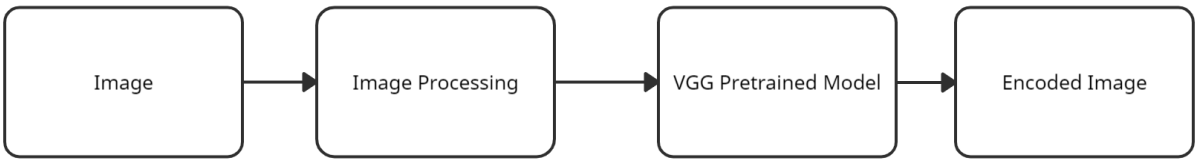


Fig. 4 Feature Extraction

Our purpose here is not to classify the image but to get a fixed-length informative vector for each image. After this, we encode our data, in which we first preprocess the image and then pass it from the model as input and then store it as a feature vector. Finally, we have to reshape our feature vector into a 4d tensor because the input given to the model is in the form of batches like this (1,224,224,3), and currently, it is 3d. While preprocessing, we also have to take care of Normalization, in which we use preprocess input as a function of Keras [28]. This pre-process input is done by resnet because resnet is trained on this kind of input; pixels are clipped in Range 0 to 255; it has subtracted the channel means from all of its pixels.

As we have used VGG19 as a feature extraction technique, we got approximately twenty-five thousand features from a single image, so to avoid overfitting and improve the results, we used PCA to reduce the features from 25088 to just 200 and maintain a variance of more than 80% [4]. PCA is an unsupervised linear transformation technique that is commonly used in a variety of fields, with the most common applications being feature extraction and dimensionality reduction [29]. PCA is also used for exploratory data processing and signal de-noising in stock market trading, as well as the analysis of genome data and gene expression levels in the field of bioinformatics. PCA seeks to find the highest variance directions in high-dimensional data and project them onto a new subspace with the same or fewer dimensions as the original one. We further use several classifiers like XGBoost classifier [7], KNN [29], Decision Trees [30], Random Forest Classifier [10], Support Vector Machines, Quadratic Discriminant Analysis [31], Logistic Regression, and Bayes Theorem on the extracted 200 features to classify the image between healthy and positive [29] [30].

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

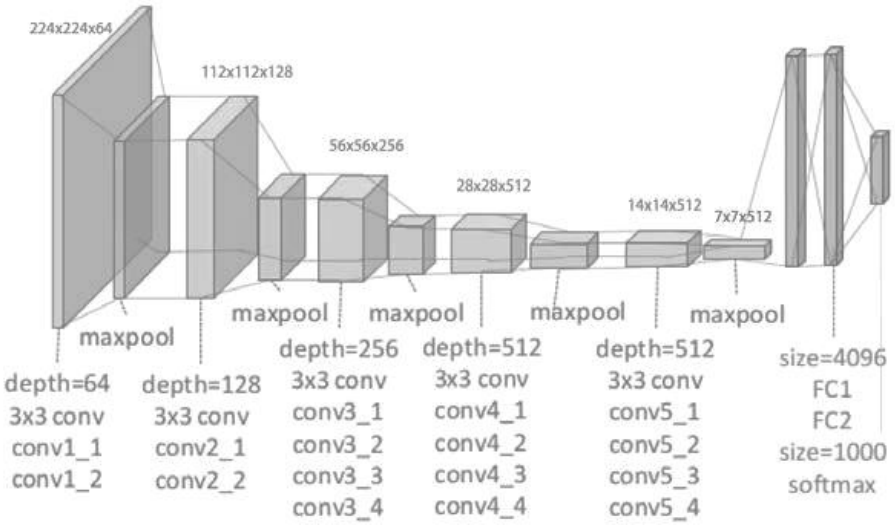


Fig. 5 Architecture of VGG19

d.**Classifiers**

The goal of the classification problem is to find the features that indicate which class each example belongs to. Due to the diversity and sophistication of image content, the image classification issue has become one of the most critical research directions in image processing and has been the subject of research for many years. Since current image classification models fail to fully exploit image information, this paper proposes a novel image classification approach that combines two outstanding techniques: the Convolutional Neural Network (CNN) [4] [32] for feature extraction and Machine Learning Algorithms like K- Nearest Neighbor, XGB classifiers and Logistic Regression. By combining pre-trained CNN as a trainable feature extractor to automatically obtain features from input and KNN as a recognizer in the top level of the network to generate performance, the presented CNN-KNN model provides more precise output [29]. This pattern can be used to evaluate existing data and anticipate the behaviour of new instances.

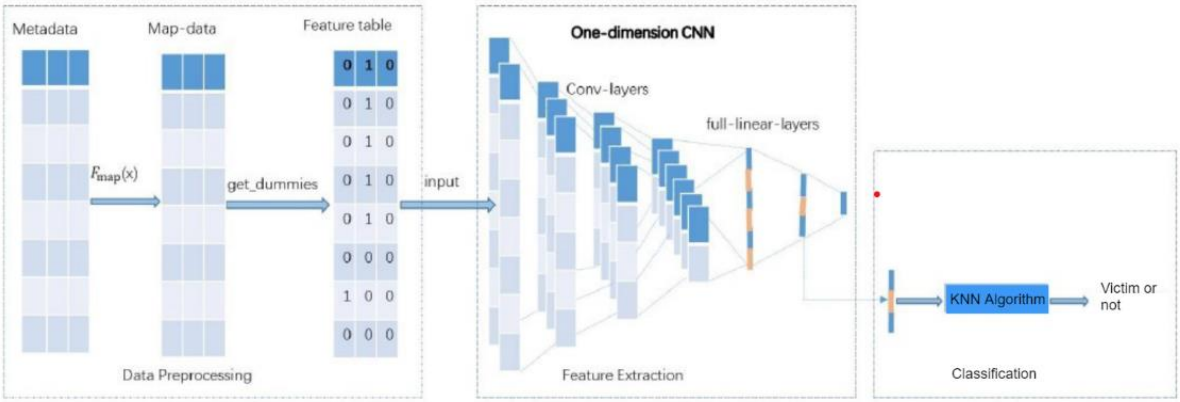


Fig. 6 Final Classification Model Architecture

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

As seen in Figure 6 above, our architecture comprises three sections: data preprocessing, feature extraction, and regression analysis. In the final phase, We used different classifiers like XGBoost, knn, logistic regression, linear discriminant analysis, SVM, bayes theorem, random forest, quadratic discriminant analysis and decision trees directly in the regression analysis process to make popularity predictions based on the high-level features extracted by pre-trained models [33]. We have used several classifiers to predict Covid19, as listed in Table 1. We cannot use accuracy solely to determine the best model, so we calculated several parameters like precision, F1 score, etc. This can be seen in Table 1. The KNN algorithm is a pattern recognition approach for classifying objects based on the feature space's nearest training samples. KNN is an example of instance-based learning, also known as lazy learning, in which the function is only approximated locally, and all computation is postponed until classification.

When there is little or no prior knowledge about the data distribution, the KNN is the most basic and straightforward classification algorithm. During learning, this rule keeps the entire training set and gives a class to each query based on the majority label of its k-nearest neighbors in the training set. When K = 1, the Nearest Neighbor rule (NN) is the simplest form of KNN.

A case is classified by a majority vote of its neighbors, with the case being allocated to the class with the most members among its K closest neighbors as determined by a distance function.

If K = 1, the case is simply allocated to the nearest neighbor's class.

$$\text{Euclidean } \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \qquad (2)$$

$$\text{Manhattan } \sum_{i=1}^{k} |x_i - y_i| \qquad (3)$$

$$\text{Minkowski } \left(\sum_{i=1}^{k} (|x_i - y_i|)^q\right)^{\frac{1}{q}} \qquad (4)$$

It's worth noting that all three distance measurements are only applicable to continuous variables. The Hamming distance must be employed when categorical variables are present. It also raises the issue of numerical variable normalization between 0 and 1 when the dataset contains both numerical and category variables.

$$D_H = \sum_{i=1}^{k} |x_i - y_i| \qquad (5)$$
$$x = y \Rightarrow D = 0$$
$$x \neq y \Rightarrow D = 1$$

Our method creates a kNN model for the data, which then replaces the data as the classification foundation. The value of k is taken to be five, and it is optimal in terms of classification accuracy.

5. **Analysis Of Results**

As explained in the last part of the paper, we used VGG19 to extract features from the image data(converted from audio) and then fed those features into different classifiers like XGB, Support Vector Machine, Random forest, to name a few. Due to the lack of sufficient data, we are using an unbalanced dataset, i.e., the number of covid positive cases is not in equal proportion to the number of covid negative cases. Thus we cannot explicitly use accuracy as a parameter to determine the best classifier out of all those mentioned in Table 1.

Table 1 Comparison of Classifiers

	Model	**Fitting time**	**Scoring time**	**Accuracy**	**Precision**	**Recall**	**F1_score**	**AUC_ROC**
1	**K-Nearest Neighbors**	0.012046	0.037497	0.783623	0.680773	0.570658	0.740233	0.630130

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

2	**Logistic Regression**	0.068904	0.009447	0.739831	0.616446	0.605142	0.733837	0.613913
3	**Linear Discriminant Analysis**	0.096662	0.010695	0.744324	0.614932	0.595233	0.733191	0.636763
4	**XGB Classifier**	1.067243	0.009896	0.787101	0.614124	0.533871	0.712665	0.626001
5	**Bayes**	0.004879	0.006084	0.716546	0.582561	0.557693	0.705452	0.598683
6	**Support Vector Machine**	1.082017	0.025556	0.782778	0.516248	0.511558	0.692992	0.707307
7	**Random Forest**	0.969429	0.023752	0.779493	0.41463	0.502500	0.683840	0.58244
8	**Quadratic Discriminant Analysis**	0.052469	0.011379	0.779493	0.389191	0.500000	0.681389	0.538613
9	**Decision Tree**	0.190894	0.005614	0.655966	0.533744	0.540233	0.664635	0.540233

Thus to deal with this unbalanced data, we calculated a few parameters through which we can judge which classifier is the best one. As per our use case, one wants to maximize recall so that we can have the least number of false negatives, so our main priority is to maximize recall. Still, as we know, there is always a tug of war between precision and recall. We need to make sure that we have an even balance. Thus, we reach out to calculate the F1 score which takes an even balance of both of these parameters. Therefore finally, we used F1 scores to rank our models.

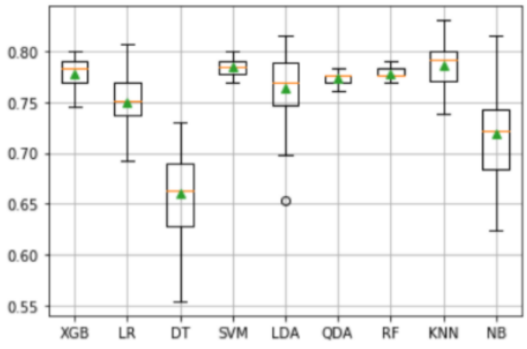


Fig. 7 Overall Classifier result

Figure 7 shows all the models tested on the data, out of all the models applied, the XGB classifier has the maximum accuracy, but considering the above facts, we rely on the F1 score, which is only 71.17%, comparatively lesser than other models.

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

Table 2 Ensembling Results

	Model	Accuracy	Precision	Recall	F1_score
0	**Ensembling_hard**	0.784615	0.750	0.034884	0.066667
1	**Ensembling_soft**	0.784615	0.625	0.058140	0.106383

According to the F1 score, the K-Nearest Neighbor classifier gives the best results; thus, relying on this fact, we can say it's the best classifier out of the others with an F1 score of 74.03%. Considering the fact that ensembling generally improves the model accuracy, we created an ensemble of all the models we have used so far. The results of the ensembling, both hard and soft, are shown in Table 2 and Figure 8.

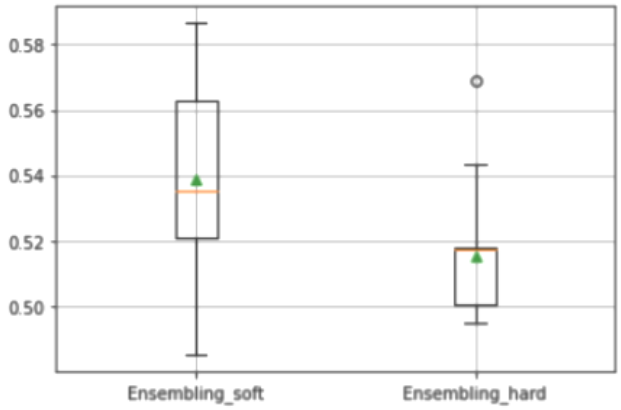
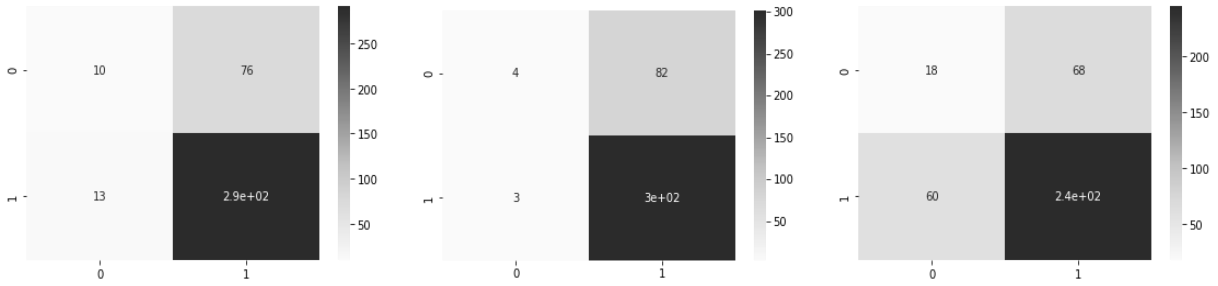


Fig. 8 Overall Ensembling Classifier Result

We can see that we can achieve an accuracy of about 78% by the ensemble, but the f1 score is terrible. So we can say that KNN is the best classifier if we keep the f1 score as the priority. Figure 9 to 17 shows the confusion matrix that we obtained on the coswara dataset.



(a) (b) (c)

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

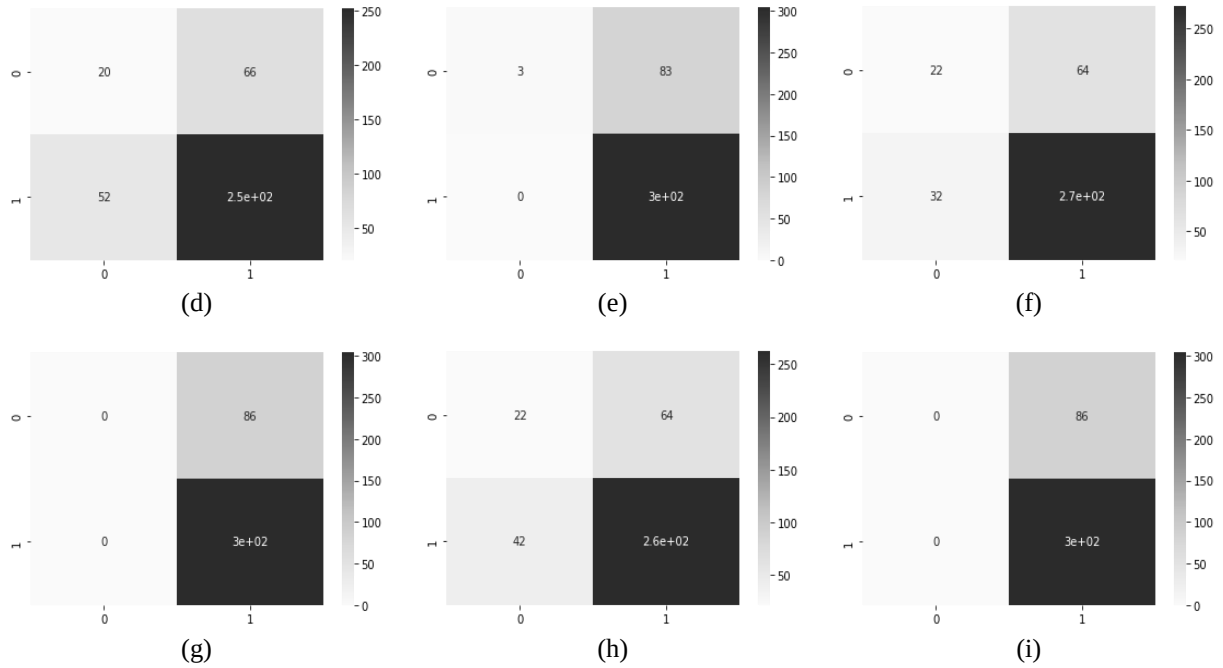


Fig. 9 Confusion Matrices of (a)KNN Classifier, (b)XGB Classifier, (c)Decision Tree Classifier, (d)SVC Classifier, (e)LDA Classifier, (f)Gaussian NB Classifier, (g)Random Forest Classifier, (h)QDA Classifier, (i)Logistic Classifier

6. **Conclusion and Future Scope**

This paper showcases the high potential cough sounds in the detection of Covid19. There have been various researches regarding the use of audio(cough) in detecting diseases like bronchitis, tuberculosis, pneumonia, in which coughing is a significant symptom. Considering the above fact, the cough sounds of other conditions can also generalize the model better and get better accuracy. Many models developed so far lack sufficient amounts of covid19 positive patient data; if the data can be increased, generalization can be made better. Selective extraction of features from the MFCCs can be used to get better conceptions for the model. This analysis can be further expanded to other speech sounds like vowel sounds. Moreover, the lack of data can be covered by the phone call data, but this requires large amounts of preprocessing as one needs to filter other sounds and capture only relevant audio data. Several research works have been done regarding the same and its feasibility. This research paper will ignite further research in this area.

Funding: "This research received no external funding."

Conflicts of Interest: "The authors declare no conflict of interest."

References

1. Mouawad, Pauline, Tammuz Dubnov, and Shlomo Dubnov. "Robust Detection of COVID-19 in Cough Sounds." SN Computer Science 2.1 (2021): 1-13.
2. World Health Organization. "World Health Organization coronavirus disease 2019 (COVID-19) situation report." (2020).
3. Deshpande, Gauri, and Björn Schuller. "An overview on audio, signal, speech, & language processing for covid-19." arXiv preprint arXiv:2005.08579 (2020).
4. Sharma, Neeraj, et al. "Coswara–A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis." arXiv preprint arXiv:2005.10548 (2020). https://github.com/iiscleap/Coswara-Data.

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

5. McFee, Brian, et al. ”librosa: Audio and music signal analysis in python.” Proceedings of the 14th python in science conference. Vol. 8. 2015.
6. Dalmazzo, David, and Rafael Ramirez. ”Mel-spectrogram Analysis to Identify Patterns in Musical Gestures: a Deep Learning Approach.” MML 2020: 1.
7. Mateen, Muhammad, et al. ”Fundus image classification using VGG-19 architecture with PCA and SVD.” Symmetry 11.1 (2019): 1.
8. Cherif, Iyad Lahsen, and Abdesselem Kortebi. ”On using eXtreme gradient boosting (XGBoost) machine learning algorithm for home network traffic classification.” 2019 Wireless Days (WD). IEEE, 2019.
9. Santosh, K. C. ”Speech Processing in Healthcare: Can We Integrate?.” Intelligent Speech Signal Processing. Academic Press, 2019. 1-4.
10. Abeyratne, Udantha R., et al. ”Cough sound analysis can rapidly diagnose childhood pneumonia.” Annals of biomedical engineering 41.11 (2013): 2448-2462.
11. Swarnkar, Vinayak, et al. ”Automatic identification of wet and dry cough in pediatric patients with respiratory diseases.” Annals of biomedical engineering 41.5 (2013): 1016-1028.
12. Al-Khassaweneh, Mahmood, and Ra’ed Bani Abdelrahman. ”A signal processing approach for the diagnosis of asthma from cough sounds.” Journal of medical engineering & technology 37.3 (2013): 165-171.
13. Boyanov, Boyan, and Stefan Hadjitodorov. ”Acoustic analysis of pathological voices. A voice analysis system for the screening of laryngeal diseases.” IEEE Engineering in Medicine and Biology Magazine 16.4 (1997): 74-82.
14. Imran, Ali, et al. ”AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app.” Informatics in Medicine Unlocked 20 (2020): 100378.
15. Lella, Kranthi Kumar, and P. J. A. Alphonse. ”A literature review on COVID-19 disease diagnosis from respiratory sound data.” AIMS Bioengineering 8.2 (2021): 140-153.
16. Pan, Sinno Jialin, and Qiang Yang. ”A survey on transfer learning.” IEEE Transactions on knowledge and data engineering 22.10 (2009): 1345-1359.
17. Livingston, Frederick. ”Implementation of Breiman’s random forest machine learning algorithm.” ECE591Q Machine Learning Journal Paper (2005): 1-13.
18. Brown, Chloë, et al. ”Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data.” arXiv preprint arXiv:2006.05919 (2020).
19. Peng, Chao-Ying Joanne, Kuk Lida Lee, and Gary M. Ingersoll. "An introduction to logistic regression analysis and reporting." *The journal of educational research* 96.1 (2002): 3-14.
20. Feng, Ji, Yang Yu, and Zhi-Hua Zhou. "Multi-layered gradient boosting decision trees." *arXiv preprint arXiv:1806.00007* (2018).
21. Vishwanathan, S. V. M., and M. Narasimha Murty. "SSVM: a simple SVM algorithm." *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*. Vol. 3. IEEE, 2002.
22. Shuja, Junaid, et al. "COVID-19 open source data sets: a comprehensive survey." *Applied Intelligence* 51.3 (2021): 1296-1325.
23. Hwang, Yeongtae, et al. "Mel-spectrogram augmentation for sequence to sequence voice conversion." *arXiv preprint arXiv:2001.01401* (2020).
24. Nanni, Loris, et al. "An ensemble of convolutional neural networks for audio classification." *Applied Sciences* 11.13 (2021): 5796.
25. Godino-Llorente, Juan Ignacio, et al. "Support vector machines applied to the detection of voice disorders." *International Conference on Nonlinear Analyses and Algorithms for Speech Processing*. Springer, Berlin, Heidelberg, 2005.
26. Hemdan, Ezz El-Din, Marwa A. Shouman, and Mohamed Esmail Karar. "Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images." *arXiv preprint arXiv:2003.11055* (2020).
27. Pham, Tuan D. "A comprehensive study on classification of COVID-19 on computed tomography with pretrained convolutional neural networks." *Scientific reports* 10.1 (2020): 1-8.

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022

28. Subramanian, Vishnu. *Deep Learning with PyTorch: A practical approach to building neural network models using PyTorch*. Packt Publishing Ltd, 2018.
29. Ayesha, Shaeela, Muhammad Kashif Hanif, and Ramzan Talib. "Overview and comparative study of dimensionality reduction techniques for high dimensional data." *Information Fusion* 59 (2020): 44-58.
30. Quinlan, J. Ross. "Learning decision tree classifiers." *ACM Computing Surveys (CSUR)* 28.1 (1996): 71-72.
31. Ghojogh, Benyamin, and Mark Crowley. "Linear and quadratic discriminant analysis: Tutorial." *arXiv preprint arXiv:1906.02590* (2019).
32. Agarap, Abien Fred. "An architecture combining convolutional neural network (CNN) and support vector machine (SVM) for image classification." *arXiv preprint arXiv:1712.03541* (2017).
33. Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016.

Doi : https://doi.org/10.54216/FPA.070202

Received: January 09, 2022 Accepted: April 08, 2022