## **COVID-19 Cough Detection Using Mel Spectrum**

## **and Convolutional Neural Networks**

| Tommy Chien  Student - CST Diploma  British Columbia Institute of Technology  Burnaby BC, Canada  tchien@my.bcit.ca | Nathan Dong  Student - CST Diploma  British Columbia Institute of Technology  Burnaby BC, Canada  ndong5@my.bcit.ca | Tegvaran Sooch  Student - CST Diploma  British Columbia Institute of Technology  Burnaby BC, Canada  tsooch2@my.bcit.ca | Jack Shih  Student - CST Diploma  British Columbia Institute of Technology  Burnaby BC, Canada  jshih14@my.bcit.ca |
| --- | --- | --- | --- |
|  |  |  |  |

***Abstract*—In this paper, we outline the problem of Covid-19 diagnosis through the analysis of cough audio using Neural Networks. The process of preprocessing the data included separating out usable cough audio samples and segmenting samples with multiple coughs, in order to gain a set of usable, single-cough data samples adequate for use in training a neural network. Through the usage of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNN’s), as well as other techniques, we developed models suitable for classifying Covid-19 cough audio samples from non-Covid audio samples up to an accuracy rate of ~91%.**

# INTRODUCTION

More than a year after the COVID-19 pandemic began, communities around the world are still feeling the effects. Though numerous Covid-19 testing methods have been developed, many take hours or days to get results with some results still not being 100% accurate. Furthermore, these testing methods required trained technicians and specialized equipment, infeasible for people at home who want to informally self-test themselves. Through the usage of Neural Networks, mainly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNN’s), we can develop a model able to classify coughs as Covid-19 positive or negative, at an extremely higher accuracy rate than humans can.

# DATA EXPLORATION

## A. *Dataset Used*

As our dataset we used COUGHVID, a crowdsourced dataset with 25,000 cough recordings[2]. Though samples could be taken from anywhere in the world, samples specifically labeled as symptomatic and COVID-19 were ensured to originate from countries with high infection rates. The dataset included many features, including location, age, gender and existing respiratory conditions, as well as some samples having additional features labeled by physicians who listened to the cough and labeled specific cough qualities. The reason we chose COUGHVID over other datasets was the vast amount of samples available.

Few samples from another crowdsourced cough dataset called COSWARA from the Indian Institute of Science (IISc) Bangalore [1]. It contains a total of 1,600 samples with This dataset was solely used as unseen data to further test the trained model.

*B. Audio Files*

The dataset consists of audio files in the webm format or a few in the ogg format. Webm is a compressed video format commonly used on the Internet. Ogg is a compressed audio format.

*C. Symptomatic labels*

The data’s labels were split into three categories: healthy, covid and symptomatic. The first two being clearly defined labels. Our team could not find the clear definition of symptomatic labels and decided to omit the samples labeled as symptomatic. This was done to ensure that we are not misclassifying any of the samples and we knew that it would not impact the results as it only represented 9% of the total data as shown in Fig. 1.

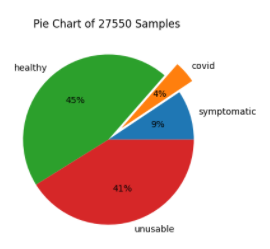


Fig. I. Percentage of different categories of samples in the dataset

# METHODS

Cough Recordings are segmented into individual cough samples as a measure to both standardize the samples and increase sample size to combat data imbalance. The same approach can be seen in other published reports [2][3].

*A. Data Conversion*

Sound samples from the Coughvid Dataset are first converted from webms (compressed video format) to wavs (uncompressed audio format), separated by categories of Covid status: healthy and covid. These wav files are then converted to a numpy array.

*B. Data Downsampling*

Samples were collected at 48kHz. The final processed samples were downsampled by a factor of 4 to be 12kHZ. This was done to save storage space and to make the training process faster.

*C. Data Segmentation*

Numpy arrays are then segmented twice with different parameters to produce individual sound files for each cough in a recording. Coughs are detected when there is a sudden spike in the signal power.

In the first segmentation, the minimum length of a cough is at least 0.2 seconds. 1 second of padding is then added to both sides of this spike, resulting in a segmentation of 2 seconds. The same process is repeated again for the 2nd segmentation, with a minimum cough length of 0.01 seconds and a padding of 0.2 seconds.

*D. Data Cleaning*

Each sound file is further cleaned and trimmed to isolate a single cough by finding the two least prominent peaks of size 1 surrounding the most prominent cough as shown in Fig. 2.

### 

Fig. 2. Example of one of 9 segmented cough recordings

*E. Data Padding*

Sound Data is trimmed to remove all leading 0s and then padded to give them all the same shapes for model training. The final segmented coughs have a shape of 7500.

*F. Wav to MFCC*

Sound files are converted using the Fast Fourier Transform method which can then be displayed as a power spectrum density diagram as shown in Fig 3.

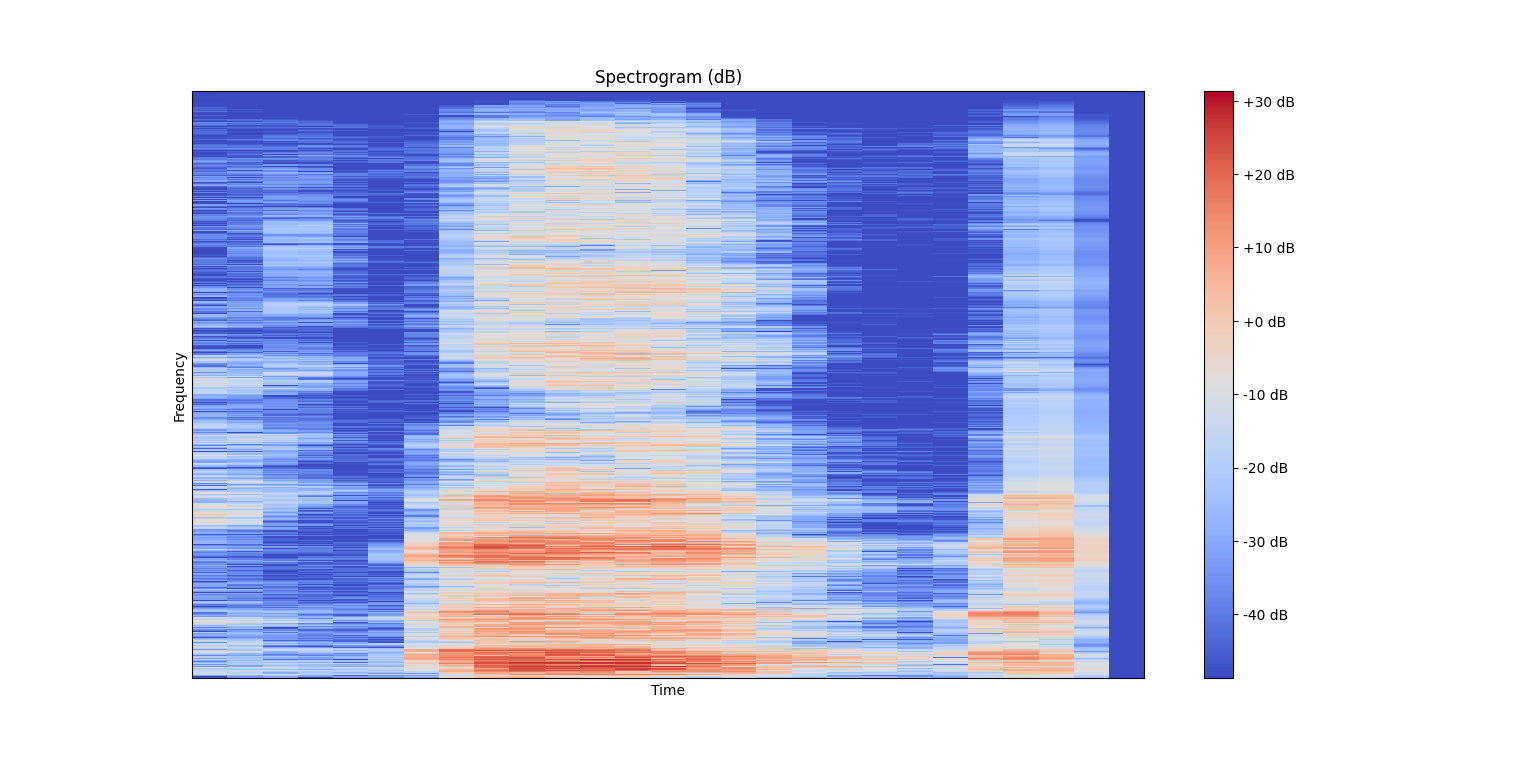


Fig. 3. Example of one of wav files with power spectral density

In addition, using (MFC) Mel-frequency cepstrum which is a representation of the short-term power spectrum of a sound. It focuses on the range of human auditory. Afterward, let a sound data transfer to MFCCs (Mel-frequency cepstral coefficients) as shown in Fig 4.

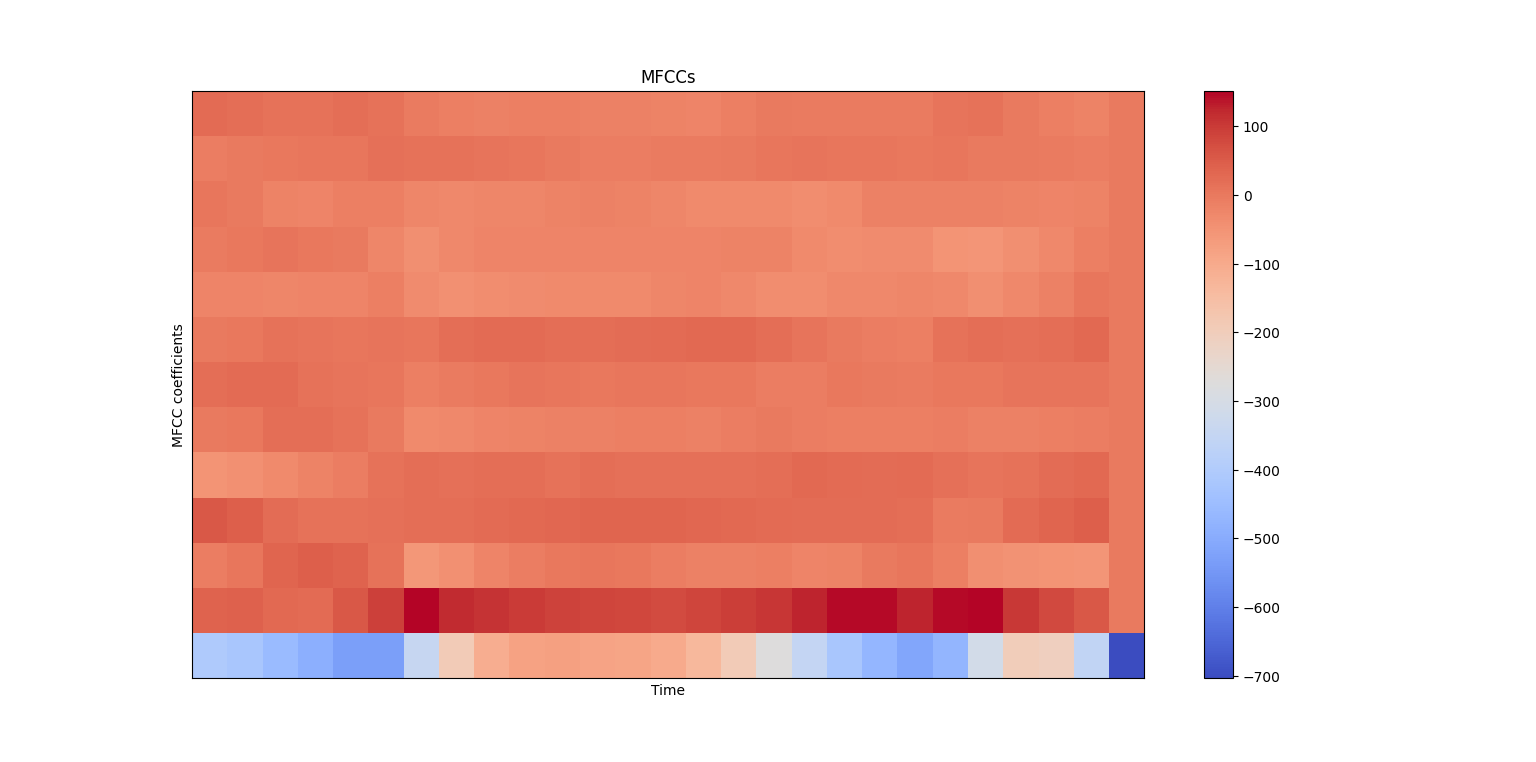


Fig. 4. Example of one of wav files with MFCC format

*G. Models*

For the model selection, according to the thesis (adding later), Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are the most popular methods for audio classification.

Both are using different approaches.

1. CNN with batch normalization
   1. CNN contains Convolutional layers, Pooling layers, flatten, and fully-connected neural networks with an image classification approach.

The architecture includes 3 convolution layers, 3 max pooling layers, 3 batch normalization layers, 2 fully-connected layers, 1 dropout layer and 1 flatten layer.

* 1. For the activation functions, only the final dense layer uses softmax, other layers use the Rectifier function [4].

1. CNN-VGG-16
   1. The VGG network architecture was introduced by Simonyan and Zisserman,It handles large scale image recognition. This network only uses 3 by 3 convolution layers, max pooling layers, and 2 fully-connected layers [5].
   2. The number 16 represents the number of layers. It’s a very deep convolutional network. Because it is a very dense architecture , it takes a long time for computing. For the computation consideration, using GPU to do calculation would be preferable instead of CPU.
2. RNN-LSTM
   1. Recurrent neural network is a model whose connections between nodes from undirected or directed graphs along a temporal sequence. It is developed based on neural networks. The difference is that each input is dependent on the node proprecessing and relies on prior elements with the sequence.
   2. Long short-term memory(LSTM) contributes to the feedback connections, it considers the whole sequence of data. There is a gate to identify which input is really key components to generate as an output. Even though it is a good solution to optimize the accuracy of predilection for RNN, LSTM still has a gradient exploding problem.
   3. With the current RNN model, the architecture includes 2 LSTM layers and 2 fully-connected layers and one dropout layer.
   4. According to Audio Scene Classification with Deep Recurrent Neural Networks [6], it showed that RNN has a better performance than CNN. The further study would do the general comparison.

# COUGH APPLICATION

A Web Application with a simple UI was made to incorporate our prediction model into an HTML page to predict the Covid status of either a live-recorded or uploaded cough recording as shown in Fig 5.

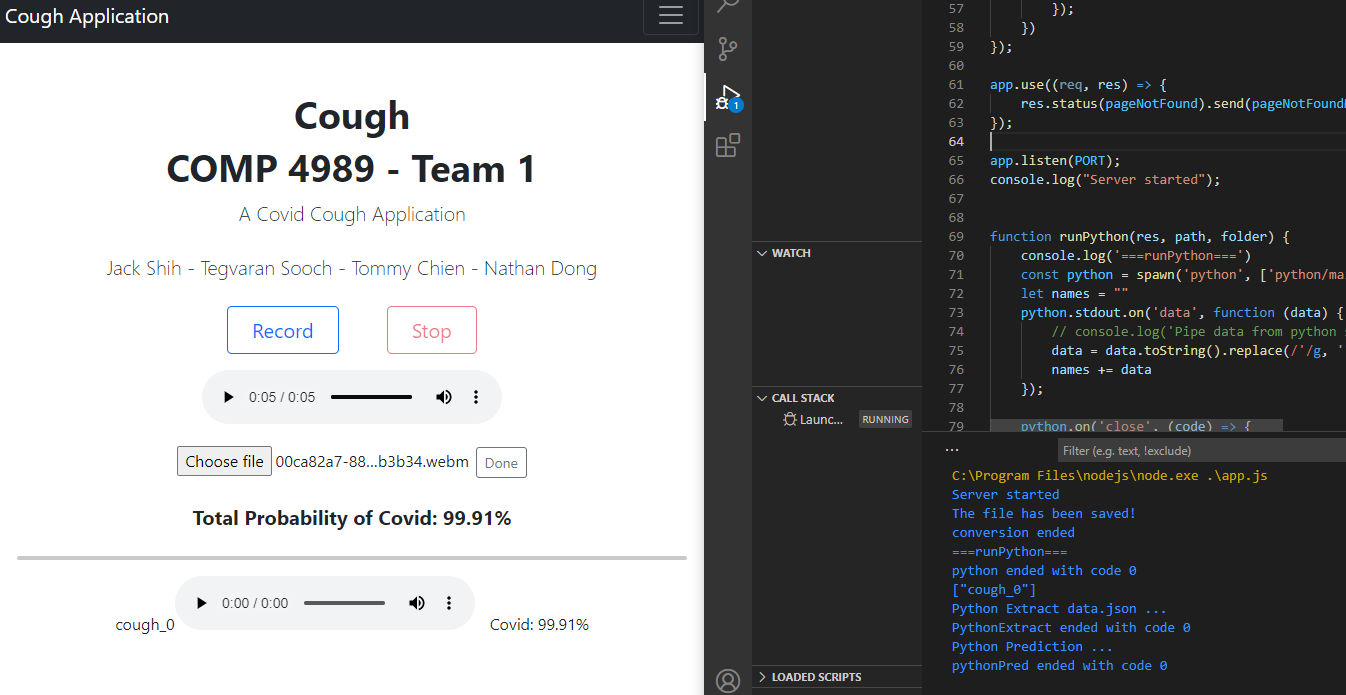


Fig. 5. Screenshot of the Covid Cough Application

The Web App uses a relaxed segmentation method than the one described above in that it only segments the user’s cough recording once, and would continuously increase the least-prominent-peak criterion until at least two peaks are found in the cleaning phase. This was necessary to ensure a high success rate in segmenting cough recordings.

The resulting issue here is that the preprocessing of inputs is no longer the same as that in the training phase.

The results we observed with this Web App on unseen data showed a high success rate in predicting healthy coughs. For Covid cough recordings, there is a good probability that out of all segmented coughs from one recording, at least one of them would be deemed Covid positive.

A preliminary assumption can be drawn that if any of the segmented coughs of a recording is detected to be Covid positive, then this cough likely belongs to someone who contracted Covid.

### RESULTS

For the classic CNN model with 30 epochs calculation, the test accuracy of cross validation is 0.65 as shown in Fig 6.

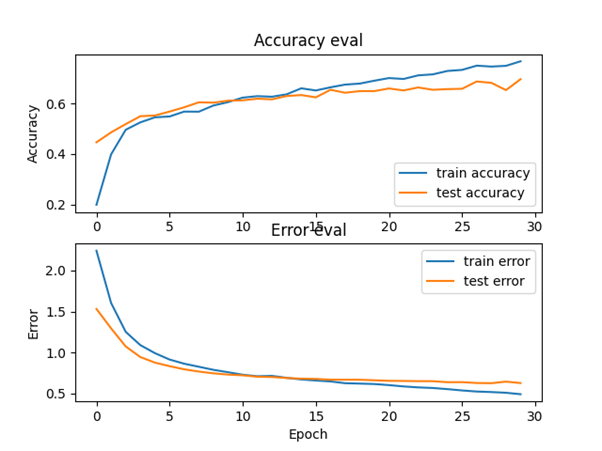


Fig. 6. The accuracy evaluation from CNN with 30 epochs

The CNN-VGG-16 model with 30 epochs training indicates that the bifurcation point between the train accuracy and the test accuracy starts at epoch 15m and the error and loss rises afterward. The final accuracy from the test accuracy is around 0.72. Compared to the classic CNN model, it shows a slightly better accuracy as shown in Fig7.

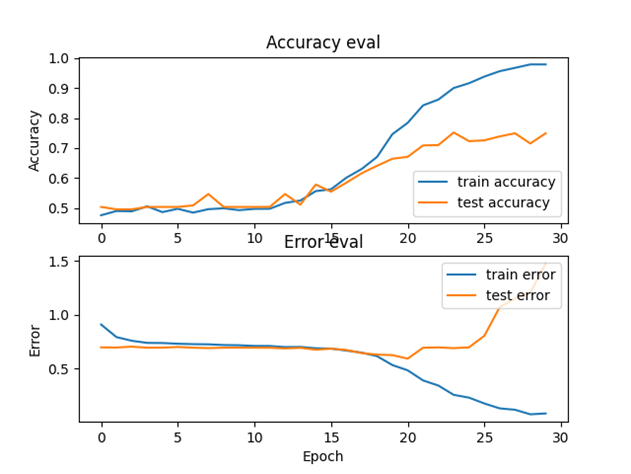


Fig. 7. The accuracy evaluation from CNN-VGG-16 with 30 epochs

The RNN-LSTM model with 30 epochs training, it got the earlier bifurcation point starting at 6 epochs. As we can see that the loss rate is quite high afterward. However, the testing accuracy maintains around 0.9 accuracy as shown in Fig 8. For the generalization, it could get lower accuracy with another testset, but according to the real testing though recording cough from test subject, it gave a quite high accuracy.

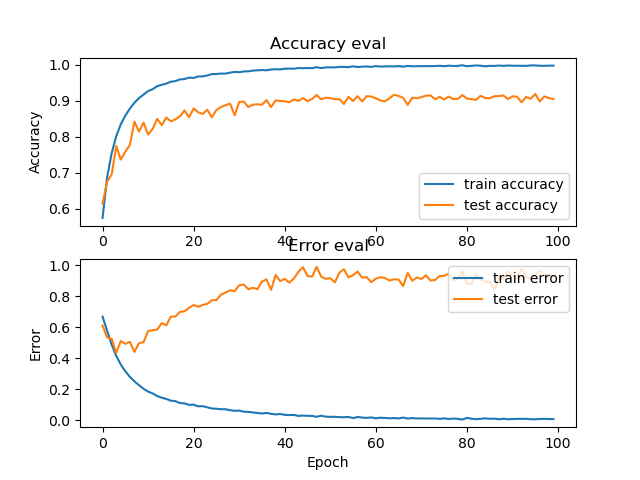


Fig. 8. The accuracy evaluation from RNN-LSTM with 100 epochs

### INSIGHTS

*A. Preprocessing*

Preprocessing is vital in training a model, however, it can lead to a false expectation for results. As the trained model is applied to another dataset, the same preprocessing pipeline may not be able to produce similar inputs to feed to the model. One can either reinforce the preprocessing pipeline to make it more robust, or design different preprocessing methods for different situations.

One way to ensure better data quality is to instruct users to conform to a predetermined guideline in submitting their cough samples, such as to only cough once for each recording.

*B. Cough Variability*

The current model is only trained on Healthy and Covid cough recordings. Other circumstances such as Asymptomatic Covid Positives, or Covid Negatives with pre-existing respiratory conditions are not considered. Data variability can improve both the applicability and robustness of our trained model. Perhaps an ensemble of AI models with curated datasets can be combined together to improve accuracy as well as cover a wider range of cases.

## C. *Sample Collection*

The success of any AI model is highly dependent on the data and how accurate they are labeled. Crowdsourced dataset may most easily provide a large number of samples, but there is also a higher likelihood of noise, or bias

.

To truly realize the full potential of a project such as this, a nation-wide collaboration with public health officials may be necessary.

# CONCLUSION AND FUTURE WORKS

Though we’ve utilized effective preprocessing and segmentation techniques that enabled us to develop an effective model to classify cough audio samples as Covid positive or negative, there are many areas in our work that are open to improvement , especially if we consider having our model accessible for general usage by the public. Namely, we hope that the model can have it’s accuracy increased as well as its ability to generalize well on non-optimal cough recording samples.

*A. Incorporating more features*

One avenue to explore to further improve model accuracy is to incorporate further features besides the cough audio itself into the model. Currently the model does not take into account a person’s age, gender and whether they have existing respiratory conditions, and we can theorize that people with different categories of these features may have certain qualities/patterns present in their cough audio samples. Examples of this could be females having generally high-pitched or shorter coughs compared to males, likewise with younger people and older people.

*B. Improving Preprocessing*

The preprocessing and segmentation process we currently used removed 41% of the samples, due to samples being low quality due to noise or whether we were not able to detect a cough. When removing samples of low quality, we reduce the generalization of our model (can’t recognize coughs in high noise recordings) and reduce the amount of training data for our model. By loosening these strict rules can help us improve our model’s generalization. Furthermore, by altering our segmentation process to better recognize individual coughs, we can also improve model performance and generalization. Future consideration should be given as to whether using noise reduction algorithms can rectify the considerations stated above.

*C. Usage Beyond the Covid-19 Pandemic*

Though our model has been trained to classify coughs that are Covid positive and negative, our model can be trained to classify other general respiratory conditions, such as pneumonia. Furthermore, our model pipeline may also be effective in diagnosing future pandemic coughs. Though the former and ladder may only be possible if suitable datasets are formulated.

##### REFERENCES

[1]"GitHub - iiscleap/Coswara-Data: Data repository of Project Coswara", GitHub, 2021. [Online]. Available: https://github.com/iiscleap/Coswara-Data. [Accessed: 26- Nov- 2021].

[2]"The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms", Nature,.com, 2021. [Online]. Available: https://doi.org/10.1038/s41597-021-00937-4. [Accessed: 10- Nov- 2021].

[3]”COVID-19 Detection in Cough, Breath and Speech using Deep Transfer Learning and Bottleneck Features” Cornell University, 2021 [Online]. Available: https://arxiv.org/abs/2104.02477 [Accessed: 12- Nov- 2021].

[4] “CNN Architectures for large scale audio classification”, IEEE. 2017

[Online] Available: https://ieeexplore.ieee.org/abstract/document/7952132

[Accessed: 16- Nov- 2021]

[5] “Large-Scale Weakly Supervised Audio Classification Using Gated Convolutional Neural Network”, [Online] Available: https://ieeexplore.ieee.org/abstract/document/8461975

IEEE. 2018 [Accessed: 18- Nov- 2021]

[6] “Audio Scene Classification with Deep Recurrent Neural Networks”, Cornell University, 2017 [Online] Available: https://arxiv.org/abs/1703.04770

[Accessed: 18- Nov- 2021]