Diagnosing the Stage of COVID-19 using Machine Learning on Breath Sounds

Chinmayi Ramasubramanian[1]

[1] *Grade 9, Sri Kumaran Children's Home, Bangalore, Karnataka, India*
chinmayi.ramasubramanian@gmail.com

Abstract

With rapidly increasing COVID-19 cases, patients with mild and moderate symptoms are being asked to home-isolate themselves to save hospital resources for more severe patients. Such patients have been asked to self-monitor themselves and seek medical attention if their condition worsens. COVID-19 affects the respiratory system and home-isolated patients must monitor their lung condition continuously before it quickly deteriorates. But this is difficult to monitor by oneself, and the patient may not notice his worsening lung condition before it is too late. A machine-learning based approach is proposed to monitor lung condition by analyzing the breath sounds of a patient for respiratory sounds like wheezes, crackles and tachypnea, which in turn can identify the stage of COVID-19. Data from a respiratory sound database with recordings from 226 patients was split into 6898 respiratory cycles and pre-processed. In this paper, two approaches are evaluated. The first approach is demonstrated using Google Cloud AutoML with the recordings of respiratory cycles which were converted to spectrograms to train the model. In the second approach, Log Mel filter-bank features were extracted from the breath sounds and used to train multiple CNN models to hierarchically classify breath sounds. This ensemble-learning with hierarchical-model approach achieved a better accuracy of 78.12%. This model can be integrated with a mobile application to record and analyze breath-sounds. This will enable the patient to admit himself sooner if he is progressing to a severe stage of COVID-19.

1. Introduction

SARS CoV-2 or COVID-19 has affected various people around the globe. This disease has been declared as a pandemic and the fatality rates are high. As of May 2021, 163 million people have contracted this disease across the globe (Cascella, Rajnik, Aleem, Dulebohn, & Di Napoli, 2021). COVID-19 has stages of increasing severity, from a slight fever, cough and sore throat to severe pneumonia and death. However, only 14% of all cases are severe while 5% of all cases are critical.

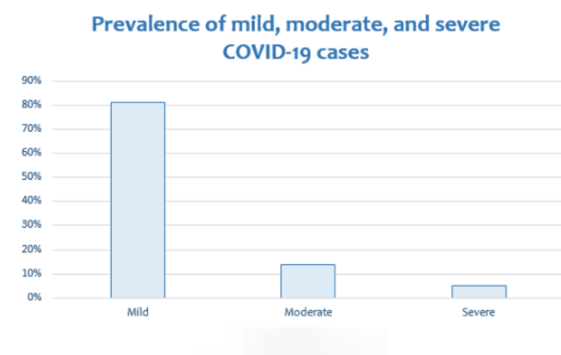


Figure 1. Percentage distribution of mild, moderate and severe COVID-19 cases across the globe.

As seen in Figure 1, mild patients make up the majority of all COVID-19 cases. Home-isolating more people with mild symptoms can free up a lot of resources for patients who need hospital resources more. So, devising personalized methods of continuously and easily monitoring lung condition and the severity of the COVID-19 of the home-quarantined patient is highly beneficial.

Stages of COVID-19

A two-level CNN model is built to detect these respiratory sounds/markers in the recorded breath sounds of COVID-19 patients.

These respiratory markers are mapped to stages of COVID-19 (mild, moderate, severe) according to past research, and verified with a pulmonologist and a pathologist.

Currently, progressive lung condition deterioration can only be detected by doctors using stethoscopes or by taking CT-scans of the lungs. Both these options are resource-intensive, costly and put increased pressure on health personnel and health equipment who are already strained by the pandemic.

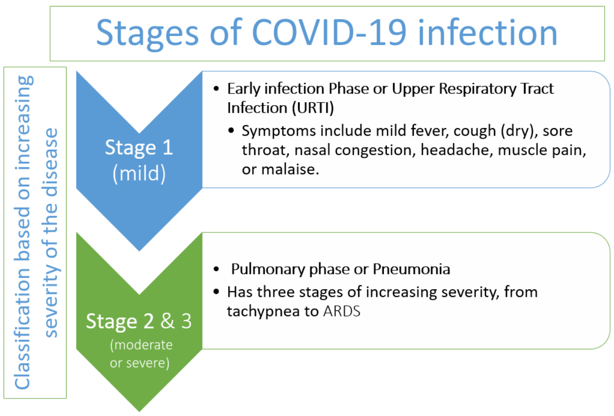


Figure 2. Stages of COVID-19 infection – Stage 1 and Stage 2

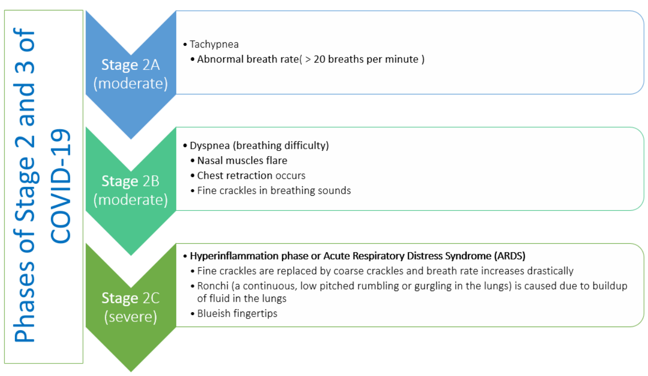


Figure 3. Further classification of Stage 1 and Stage 2 into Pulmonary phase and Hyperinflammation phase

These methods may also need home-quarantined patients to visit clinics or hospitals or testing centers which could increase the spread of the virus. There has been research (Huang et al., 2020) relating some respiratory sounds to COVID-19. However there is no automated method to diagnose the severity of COVID-19.

There has also been some research in the past year about identifying the presence of COVID-19 using respiratory sounds (Sharma et al., 2020). This study focuses only on diagnosing the presence of COVID-19 using machine learning. Diagnosing the severity of COVID-19 in an already home quarantined patient has not been studied. Using only breathing sounds collected with a mobile/laptop microphone to diagnose COVID-19 has not been studied. Thus, this research provides novel insights in that regard.

This project has benefits for:

- COVID-19 patients in home isolation with mild or moderate symptoms
- Caregivers who want to record patient conditions can use this application/device

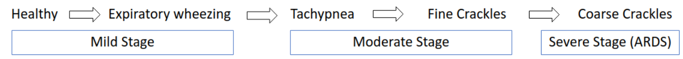


Figure 4. Mapping of respiratory markers to stages of COVID-19

- Doctors and Health care workers who want to remotely monitor lung condition of the COVID-19 patients.

A method is proposed for diagnosing the lung condition and the stage of COVID-19 of the patient by identifying respiratory markers (described above) in the breath Sounds of the patient. This method would be made available to the patient in the form of a mobile application which asks the patient to record and submit his/her breath sounds for preliminary determination of COVID-19 severity.

2. Methods and Materials

Two methods were evaluated for this research. Initially, an automated machine-learning solution called Google AutoML was used to obtain preliminary results on the validity of the approach. Based on the promising results shown by the model devoloped in Google AutoML, a unique machine learning model, combining two different CNN models to form a 2-level heirarchical model was developed. Both approaches are discussed here.

In the first method, the spectrograms were created using the command line-based tool ffmpeg and trained using AutoML. AutoML is a solution by Google that aims to reduce the time and resources needed to create a machine learning algorithm. It is based on Google's state-of-the-art research in image recognition using Neural Architecture Search (NAS) to find optimal Neural Networks for a given dataset. (He, Zhao, & Chu, 2021) However, Google AutoML performs best on real-world images and may not give the most optimal neural network in our case.

In the second method, log mel-filterbanks were used. The hypothesis was derived that mel-scale would give better results as the mel-scale simulates the way human ear works. Filterbank analysis was done to capture the spectral envelope and reduce the dimensionality of the features.

Log mel-filterbank features with a CNN models worked well, better than frequency spectrogram images with Google AutoML, giving a fair accuracy of 70.03% for the first-level model and 78.12% for the second-level model.

2.1. Data

A respiratory sound database was used, which was originally compiled to support the scientific challenge organized by Informatics in Biomedical Health ICBHI 2017 (Rocha et al., 2017). This database included 920 .wav files of the breathing of 226 different patients. The recordings were collected on heterogeneous equipment and their duration ranged from 10 to 90 seconds. The data included clean breathing sounds and noisy recordings that simulate real-life conditions, collecting sounds from seven chest locations. The patients cover all age groups: children, adults and the elderly. The data also included a label file with the start and end times of events of interest for each breath recording, like wheezes and crackles.

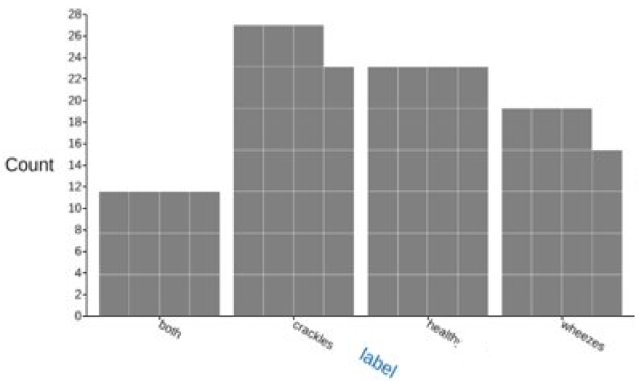
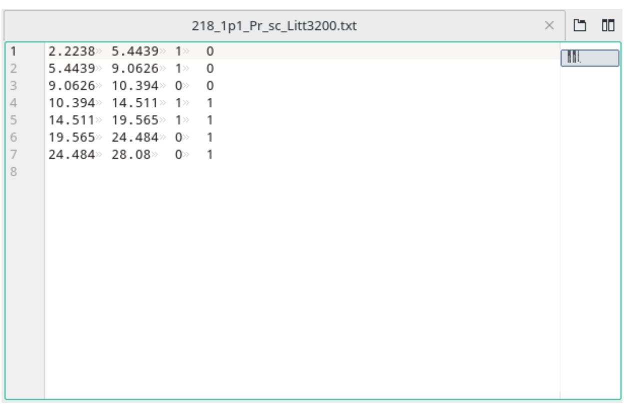


Figure 5. Original Label distribution



1	2.2238	5.4439	1	0
2	5.4439	9.0626	1	0
3	9.0626	10.394	0	0
4	10.394	14.511	1	1
5	14.511	19.565	1	1
6	19.565	24.484	0	1
7	24.484	28.08	0	1
8				

Figure 6. The label file format. The first two columns represent the start and end times of the sound clip. The third column indicates if the clip contains crackles and the fourth column indicates the presence of wheezes.

2.2. Pre-Processing

As explained before, each stage of COVID-19 contains specific a respiratory disease marker:

However, the respiratory sound database labels only instances of no respiratory markers, wheezes, crackles or both. The sound recordings with crackles were further split up into coarse and fine crackles, sounds with no respiratory markers were split up into healthy and tachypnea. In the process, any silent sound files and sound files with high background noise were removed. The reclassification was validated seperately by two different doctors.

Table 1. Relation between respiratory disease markers and COVID-19 stages

Mild	• No respiratory markers, the breath sounds healthy • Wheezes develop later
Moderate	• Tachypnea • Fine Crackles
Severe	• Coarse crackles, or both fine and coarse crackles

Using a python program, each .wav file was split according to the start and end times provided in its respective label file to get a total of 6898 .wav files. These were the files used for training the model.

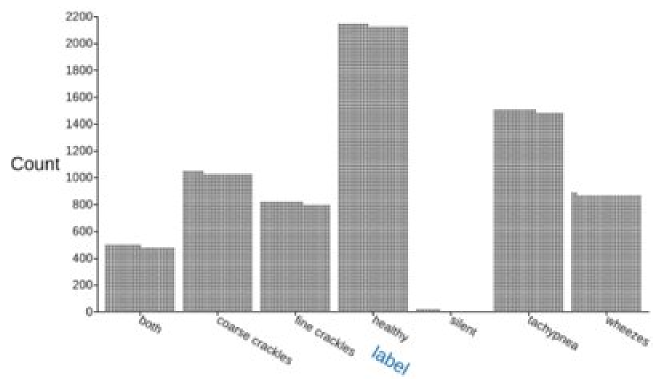


Figure 7. Label distribution after preprocessing

The dataset was split into a train-validation-test split of 60-30-10. This split was used to train both methods of machine learning models during experimentation.

The ML models were trained on this data to identify various respiratory markers seen in COVID-19. The respiratory markers identified were mapped to the appropriate stage of COVID-19 (mild, moderate, severe). Finally, the accuracy of this mapping was tested using the Coswara dataset.The Coswara dataset was collected using crowdsourcing by IISc Bangalore, India. It contains breath sounds of patients with no COVID-19, mild COVID-19 and moderate COVID-19. The breath sounds of 30 patients with mild COVID-19 and 9 patients with moderate COVID-19 was used for testing the model on actual COVID-19 patient data.

3. Experimentation

3.1. Method 1 - Google AutoML

In this method, Google AutoML was used to train on spectrogram images of the breathing sounds in order to test the validity of the idea, since it requires much less time and resources

3.1.1. Feature Extraction

Frequency spectrogram images of the breath sounds were used for training.

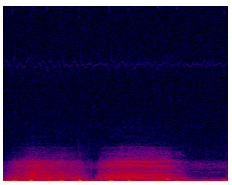


Figure 8. A healthy spectrogram

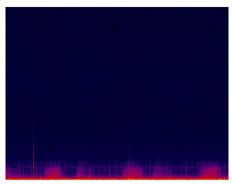


Figure 9. A tachypnea spectrogram

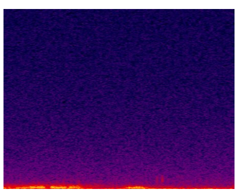


Figure 10. A coarse crackles spectrogram

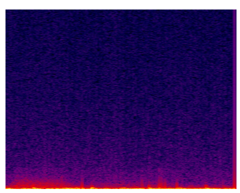


Figure 11. A fine crackles spectrogram

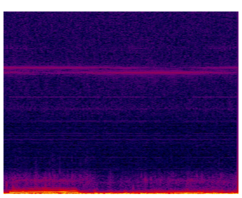


Figure 12. A wheezes spectrogram

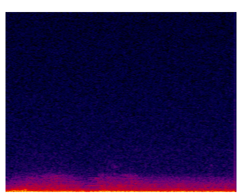


Figure 13. A spectrogram with both wheezes and crackles

3.1.2. Model Training

To train the AutoML model, wav files were used to create spectograms. All the spectrograms of breath sounds were uploaded to the Google Cloud Storage. Then, a .csv file was uploaded which had a link to each image and the respiratory condition it represents. Finally, the spectrogram pictures were reviewed to check if it had uploaded properly and start training of the model. The data had a 60-30-10 split for training, validation and testing.

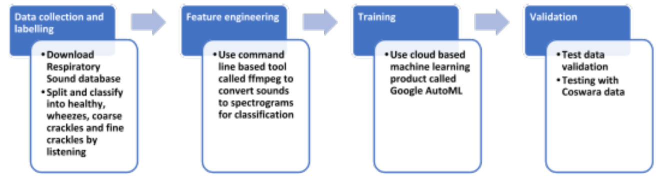


Figure 14. Machine learning pipeline for Google AutoML

3.1.3. Evaluation and Results

After evaluation, the finished model had a precision of 67.62% and a recall rate of 55.23%, which showed that the application was viable. This was calculated at a confidence threshold of 50%. By setting the threshold to 92%, the precision can be brought up to 91%. This ensures a false positive rate of only 9%. The average accuracy turned out to be 0.65 or 65%.

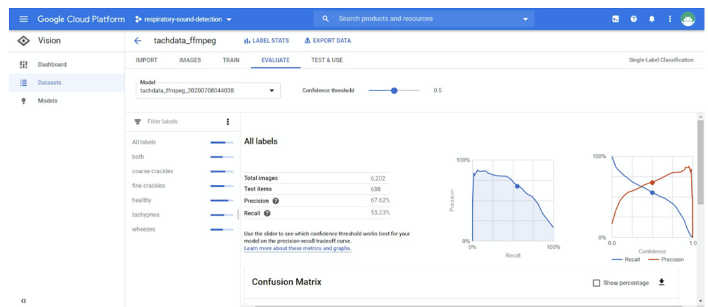


Figure 15. Evaluate screen on Google AutoML

True Label \ Predicted Label	fine crackles	both	wheezes	healthy	coarse crackles	tachypnea
fine crackles	61%	-	9%	12%	5%	13%
both	12%	45%	20%	8%	8%	6%
wheezes	2%	2%	61%	16%	6%	13%
healthy	6%	0%	4%	76%	3%	11%
coarse crackles	7%	3%	5%	15%	54%	16%
tachypnea	7%	-	9%	17%	11%	56%

Figure 16. Confusion matrix for the Google AutoML Model

3.1.4. Testing

1. The accuracy of the machine learning model in classifying the 6 respiratory markers was first evaluated.

Table 2. Validation testing results

Test category	Prediction Accuracy
wheezes	61%
coarse crackles	54%
fine crackles	61%
both wheezes and crackles	45%
healthy	76%
tachypnea	56%

These results were used as preliminary results to validate the concept that respiratory markers like crackles and wheezes could diagnose COVID-19.

2. From the Project Coswara dataset , a number of recordings of breath sounds of different patients were collected. Here the data was classified as healthy, no_resp_illness_exposed, resp_illnes_not_identified, positive_mild and positive_moderate. Only healthy, positive_mild and positive mild files were of importance to my testing. As mulitple respiratory markers like wheezes and tachypnea can show up in single stage of COVID-19 as classified in Coswara, a number of labels in the trained model can correspond to a single label in the Coswara dataset.

Table 3. Coswara testing results

Coswara Test file category	Output expected	Accuracy
mild	healthy/wheezes	75%
moderate	both/tachypnea/fine crackles	66.6%
severe	both/coarse crackles	NA

The preliminary training in Google AutoML showed that identifying the stage of COVID-19 by identifying recorded breath spectrograms was viable, but the accuracy obtained was not satisfactory. It was not possible to do any finetuning of the model to improve accuracies. Hence, this model was discarded.

3.2. Method 2 - CNN

Multiple iterations of Convolutional Neural Networks in TensorFlow were created by changing parameters like the method of feature extraction and data augmentation. All iterations conducted are shown in Table 6 in the appendix. Convolutional Neural networks were chosen as has been previously shown to have good accuracies in sound classification. (Thornton, 2019)

3.2.1. Final Model and methodology

The final iteration showed the best results. Here, two CNN models were trained to heirarchically classify the 6 respiratory markers. A total of 6898 files were used, which had a 60-30-10 split for training, validation and testing.

The first level consisted of one CNN model that classifies breath sounds as none, crackles, wheezes, or both.

The second level consisted of another CNN model and a function that calculated respiratory rate. The second level CNN model classifies sounds as fine crackles, coarse crackles or none. The second level also consists of a respiratory rate function that calculates the respiratory rate of the breath, based on which the breath is classified as healthy or tachypnea. Depending on the result of the first model, the sound file passes through either the second level model or the second level function. This model architecture to first classify, than subclassify respiratory markers gave better results than classifying all 6 respiratory markers at once.

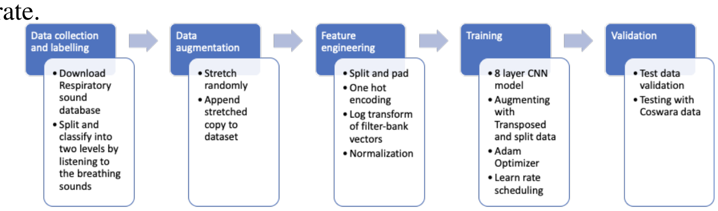


Figure 17. Machine learning pipeline for final CNN model

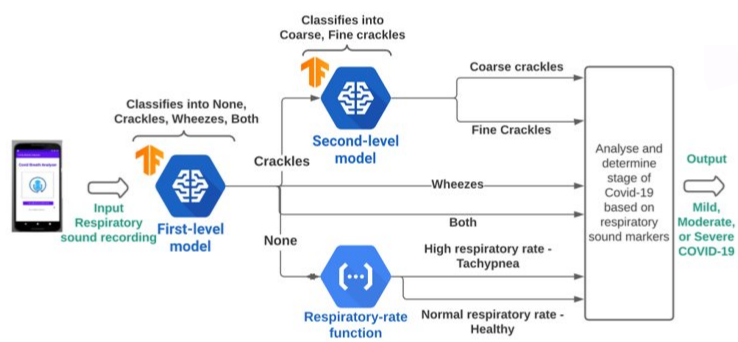


Figure 18. Final model flow (first level and second level)

The model architecture of the first and second level CNN models is shown in Fig. 22 in the appendix.

In this method, the log Mel-Filterbank features of each of the sound files in the database was extracted. These features are increasingly used for Automatic Speech Recognition (ASR) and they model frequency information on the sensitivity on the human ear.

First, the linear frequency spectrogram was extracted by the

Short-Time Fourier Transform(STFT). This was done by splitting the time domain sound signal into short-time frames of length 25 ms and a stride of 10 ms, resulting in an overlap of 15ms between frames. The hamming window function was used to reduce discontinuities in the signal. This was done using the scipy python library. Then, frequencies above 7500Hz were removed, as they are unlikely to appear in a breath sound and were unnecessary.

Next, 50 triangular filters were applied on the frequency signal to split them into filterbanks. The linear frequency filter banks were warped according to the Mel scale.

To do this, the highest and lowest frequency in the linear frequency spectrum was first found and converted them to the mel scale. Then, 50 linearly spaced frequencies were generated between the 2 mel-frequencies. After converting the 50 values plus the highest and lowest frequencies back to hertz, a scale to convert the value of each linear frequency filter bank to a mel filterbank had been created. The following equations were used to convert the linear frequencies to the mel-scale and back:

$$m = 1125 \ln(1 + f/700)$$
$$f = 700(\exp(m/1125) - 1)$$

Using the scale of 50+2 values obtained, the following equation was used to calculate the mel-filter banks feature for the whole sound signal:

$$H_m(k) = \begin{cases} 0 & \text{if } k < f(m-1) \\ \dfrac{k - f(m-1)}{f(m) - f(m-1)} & \text{if } f(m-1) \le k \le f(m) \\ \dfrac{f(m-1) - k}{f(m+1) - f(m)} & \text{if } f(m) \le k \le f(m+1) \\ 0 & \text{if } k > f(m+1) \end{cases} \tag{1}$$

where, if M is the number of features needed, m is the mth number in the range (1, M+1), and f is the scale of linearly spaced M+2 Mel-spaced frequencies.

In both the first and second level model, there was a huge imbalance between the smallest and the largest class. So, the classes with the least data were up-sampled. The data was augmented by randomly stretching and squeezing a fraction of the smallest classes.

All sound files had to be of the same length. Since the majority of the breath sound cycles were less than or equal to 5 seconds, all breath cycles were resized to 5 seconds.

A custom data pipeline was also constructed to ensure uniform interleaving between classes. Each sound was rolled before feeding to the CNN model by a random amount on the time axis such that the elements that roll beyond the last position are re-introduced at the first, to reduce overfitting.

K-fold cross validation was used with k=7.

Finally, hyper parameter tuning was done to find best epochs, learning rates and batch size:

Table 4. Results of hyper parameter tuning on CNN model

Sl.No.	Hyper parameters	Results	Optimal (Y/N)
1	16 epochs; 0.001 learning rate; batch size = 128	- loss: 0.4223 - acc: 0.8350 - val_loss: 1.1062 - val_acc: 0.6687	No
2	30 epochs; 0.001 learning rate; batch size = 128	- loss: 0.1536 - acc: 0.9388 - val_loss: 1.5874 - val_acc: 0.7067	Yes
3	30 epochs; 0.001 learning rate; batch size = 256	- loss: 0.2915 - acc: 0.8894 - val_loss: 1.2188 - val_acc: 0.6690	Yes

3.2.2. Evaluation and results

The final model contains 2 CNN models - one in the first level and one in the second level.

First Level Model: Classifies as none (no respiratory markers), crackles, wheezes or both.
Total number of respiratory cycles after data augmentation:

Overall model accuracy: 70.03%
F1 score: 0.7
Precision: 0.71
Recall: 0.69

	precision	recall	f1-score	support
none	0.69	0.76	0.72	729
crackles	0.66	0.79	0.72	746
wheezes	0.74	0.69	0.71	750
both	0.74	0.53	0.62	612
accuracy			0.70	2837
macro avg	0.71	0.69	0.69	2837
weighted avg	0.71	0.70	0.70	2837

Second Level Model: Classifies as none, fine crackles or coarse crackles

Overall model accuracy: 78.12%
F1 score: 0.77
Precision: 0.71

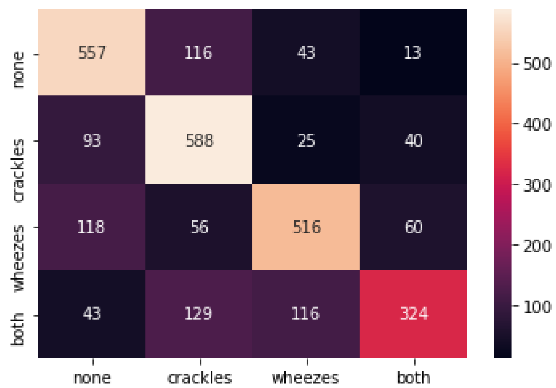


Figure 19. Prediction scores and confusion matrix for first-level model

Recall: 0.69

```
                 precision    recall  f1-score   support

           none       0.77      0.62      0.68       272
 coarse crackles       0.79      0.93      0.85       274
   fine crackles       0.78      0.79      0.78       273

        accuracy                           0.78       819
       macro avg       0.78      0.78      0.77       819
    weighted avg       0.78      0.78      0.77       819
```

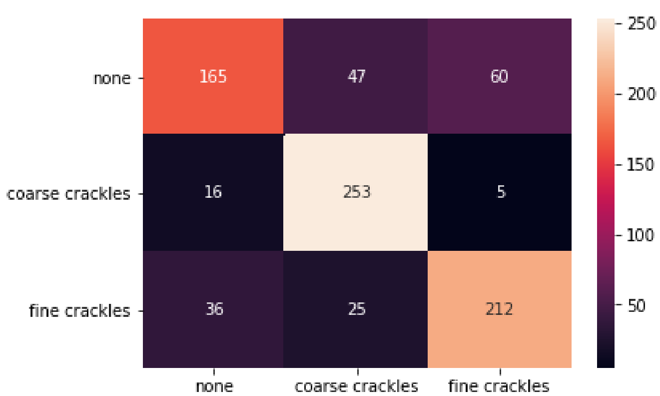


Figure 20. Prediction scores and confusion matrix for second-level model

3.2.3. Testing using COVID-19 patient data

Patient data from patients affected with COVID-19 was collected from the Coswara Dataset. The model was tested on 21 patients with mild COVID-19 symptoms and 9 patients with moderate symptoms. The results of 4 sample patients is shown in tables 7 and 8 in the appendix.

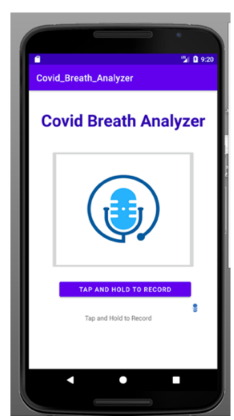
Table 5. Coswara testing results

COVID-19 Patient Test File Category	Accuracy
mild	95%
moderate	40%
severe - No data on Coswara	NA

4. Applications

The ability to monitor the current lung condition and diagnose the stage of COVID-19 can be made accessible to patients through a mobile application on smartphones. A patient can breathe into the inbuilt microphone of the mobile which feeds the recording to the machine learning model. The machine learning model identifies the respiratory markers contained in the recording and infers the severity. It then displays the current stage of COVID-19 in the mobile application. It can also compare the current condition of the patient with his/her history data and deduces if the patient's condition has worsened.

As part of this project, a prototype of the mobile application was created in Android Studio, that communicates with the machine learning model created in Google AutoML as it is automatically exposed to an API that infers from the model. The recording of the patient is uploaded to the Google Cloud Bucket, where a cloud function converts it to a spectrogram. It then calls the AutoML API to classify the spectrogram and send the results back to the mobile application. Integrating the mobile application with the model of experiment 2 which had the best results is still in progress.



This mobile app for self-diagnosing your lung condition with respect to COVID-19 would be most beneficial for patients in self-isolation. This would prevent the patient from admitting himself unnecessarily and using precious hospital resources, or too late, risking severe complications and death. This app also gives doctors or health workers more monitoring data to work with and can help the patient judge whether to admit himself at a hospital sooner.

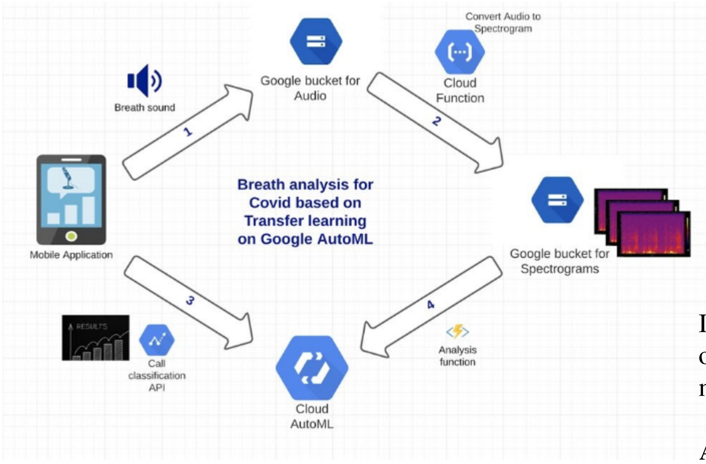


Figure 21. Prototype screen of mobile application to record breath sounds and flow diagram for integration

5. Next Steps

- Introduce more layers: Currently, the the final machine learning model consists of two layers. The first layers generally classifies respiratory markers as none, wheezes, crackles or both. The second layer further classifies crackles into fine crackles and coarse crackles, and none into healthy (normal respiratory rate) and tachypnea (high respiratory rate). More fine grained classification of breath sounds by increasing the number of levels in the ML model could increase the accuracy of the model in diagnosing COVID-19.
- Increase dataset size: Currently, data from only 226 patients were used. More data is required for more confident results.
- The machine learning model can also be used to diagnose the severity of any Influenza Like Illness (ILI) in the future. This is because the machine learning model diagnoses respiratory markers that are common to both COVID-19 and ILI. (Zayet et al., 2020; Jiang et al., 2020)
- The mobile application can be enhanced to use Edge computing for inference from the machine learning model by converting the model into a TensorFlowLite model.

6. Conclusion

The goal of this project was to predict which stage of COVID-19 a home-quarantined patient was going through by analyzing his breath sound recordings using machine learning. This method would provide analysis of disease progression and guidance on whether to admit himself at a hospital.

Two ways of creating a machine learning model was experimented with - Using spectrogram images in Google AutoML, and hierarchical CNN models on log mel-filterbank features.

- It was found that a 2 level CNN model diagnoses the COVID-19 severity of an affected patient most accurately, with an accuracy of 70.03% for the first level model and 78.12% for the second level model.
- The model can be deployed into a portable mobile application to be used by COVID-19 patients
- This will help in early detection of worsening COVID-19 condition and lower mortality rate

In conclusion, the proposed method of identifying the stage of COVID-19 of a patient by identifying general respiratory markers like crackles and wheezes shows great promise.

Acknowledgements

I would like to thank my mentor Mr. Panish Ramakrishna for his valuable guidance. I thank pulmonologist, Dr. Vinod G and pathologist, Dr. Sandhya Sundaram for validating the medical research as well as for reviewing the further split of the ICBHI database used in this paper. Finally, I would like to thank Dr. Seema Chopra for reviewing my research and providing valuable advice.

References

Aykanat, M., Kılıç, Ö., Kurt, B., & Saryal, S. (2017). Classification of lung sounds using convolutional neural networks. *EURASIP Journal on Image and Video Processing*, *2017*(1), 1–9.

Cascella, M., Rajnik, M., Aleem, A., Dulebohn, S., & Di Napoli, R. (2021). Features, evaluation, and treatment of coronavirus (COVID-19). *StatPearls*.

He, X., Zhao, K., & Chu, X. (2021). AutoML: A Survey of the State-of-the-Art. *Knowledge-Based Systems*, *212*, 106622.

Huang, Y., Meng, S., Zhang, Y., Wu, S., Zhang, Y., Zhang, Y., ... others (2020). The respiratory sound features of COVID-19 patients fill gaps between clinical data and screening methods. *medRxiv*.

Jiang, C., Yao, X., Zhao, Y., Wu, J., Huang, P., Pan, C., ... Pan, C. (2020). Comparative review of respiratory diseases caused by coronaviruses and influenza A viruses during epidemic season. *Microbes and infection*, *22*(6-7), 236–244.

Mohamed, A.-r. (2014). *Deep neural network acoustic models for ASR.* (Unpublished doctoral dissertation). University of Toronto.

Rocha, B., Filos, D., Mendes, L., Vogiatzis, I., Perantoni, E., Kaimakamis, E., ... others (2017). A respiratory sound database for the development of automated classification. In *International conference on biomedical and health informatics* (pp. 33–37).

Sharma, N., Krishnan, P., Kumar, R., Ramoji, S., Chetupalli, S. R., Ghosh, P. K., ... others (2020).

Coswara–A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis. *arXiv preprint arXiv:2005.10548*.

Thornton, B. Z. J. L. S. (2019). *Audio recognition using mel spectrograms and convolution neural networks.*

Zayet, S., Lepiller, Q., Zahra, H., Royer, P.-Y., Toko, L., Gendrin, V., ... others (2020). Clinical features of COVID-19 and influenza: a comparative study on Nord Franche-Comte cluster. *Microbes and infection*, *22*(9), 481–488.

APPENDIX

Table 6. Iterations created to find the CNN model with the best accuracies

Iteration no.	Method				Accuracy (On validation split)
	Feature Extracted	Data Augmentation	Model Type	Model Architecture	
Iteration 1	Vectorized 64x128 Spectrogram Images	Horizontal and vertical Image shift	CNN	Flat	Overall accuracy: 23.4%
Iteration 2 (Later found to be errorneous method during feature extraction, result not valid)	Vectorized 200x431 Mel-Spectrogram Images	Horizontal and vertical Image shift	CNN	Flat	Overall accuracy: 12.5%
Iteration 3	chroma, rmse, spectral centroid, spectral bandwidth, rolloff, zero crossing rate, 20 mfcc coefficients	None	CNN	Flat	Overall accuracy: 22.64%
Iteration 4	13 mfcc coefficients	None	CNN	Flat	Overall accuracy: Fluctuating (Max: 73%, Cur: 46.88%)
Iteration 5 – Similar to iteration 2	Log Mel-Filterbanks	Randomly stretch and squeeze sound to balance all classes	CNN	2 level	First level model: 58.00% Second level model: 75.00%
Iteration 6 - Final Iteration	Log Mel-Filterbanks	Randomly stretch and squeeze sound to balance all classes Custom Data Pipeline to randomly transpose sound before feeding to reduce bias	CNN	2 level	First level model: 70.03% Second level model: 78.12%

Table 7. Testing, results and accuracy of training done with COVID positive mild patients from Coswara dataset on CNN

Coswara Data		Classification results				Expected output	
Anonymized patient_id	covid_status	First level classification	Label confidence	Second level classification	Label confidence	First level	Second level
1e8i6Q47ewbzrTiKqIeOLEvPv2Z21	positive_mild	none (no respiratory markers found)	0.8829330	Healthy	NA	none/wheezes	healthy
1e8i6Q47ewbzrTiKqIeOLEvPv2Z22	positive_mild	none (no respiratory markers found)	0.9998260	Healthy	NA	none/wheezes	healthy
1e8i6Q47ewbzrTiKqIeOLEvPv2Z23	positive_mild	none (no respiratory markers found)	0.7853060	Healthy	NA	none/wheezes	healthy
1e8i6Q47ewbzrTiKqIeOLEvPv2Z24	positive_mild	none (no respiratory markers found)	0.9995210	Healthy	NA	none/wheezes	healthy
31euepHD0deCxTd2nJ1wzXCk5EF31	positive_mild	none (no respiratory markers found)	0.9755290	Healthy	NA	none/wheezes	healthy
31euepHD0deCxTd2nJ1wzXCk5EF32	positive_mild	none (no respiratory markers found)	0.9999990	Healthy	NA	none/wheezes	healthy
31euepHD0deCxTd2nJ1wzXCk5EF33	positive_mild	none (no respiratory markers found)	0.9999020	Healthy	NA	none/wheezes	healthy
31euepHD0deCxTd2nJ1wzXCk5EF34	positive_mild	none (no respiratory markers found)	0.6940950	Healthy	NA	none/wheezes	healthy
31euepHD0deCxTd2nJ1wzXCk5EF35	positive_mild	none (no respiratory markers found)	0.9997150	Healthy	NA	none/wheezes	healthy

Coswara Data		Classification results				Expected output	
Anonymized patient_id	covid_status	First level classification	Label confidence	Second level classification	Label confidence	First level	Second level (any of)
b7mMUQm5bObj1jwGFaNyaR07pt831	positive_moderate	markers found)	1.0000000	Tachypnea	NA	none/crackles/both	tachypnea/fine crackles
b7mMUQm5bObj1jwGFaNyaR07pt832	positive_moderate	none (no respiratory markers found)	1.0000000	Tachypnea	NA	none/crackles/both	tachypnea/fine crackles
b7mMUQm5bObj1jwGFaNyaR07pt833	positive_moderate	none (no respiratory markers found)	1.0000000	Tachypnea	NA	none/crackles/both	tachypnea/fine crackles
b7mMUQm5bObj1jwGFaNyaR07pt834	positive_moderate	none (no respiratory markers found)	1.0000000	Tachypnea	NA	none/crackles/both	tachypnea/fine crackles
joORHxc0iCTwgP7Uh5SM5N7rnnf21	positive_moderate	none (no respiratory markers found)	0.9994900	Healthy	NA	none/crackles/both	tachypnea/fine crackles
joORHxc0iCTwgP7Uh5SM5N7rnnf22	positive_moderate	Crackles	0.8450522	Coarse Crackles	0.810477	none/crackles/both	tachypnea/fine crackles
joORHxc0iCTwgP7Uh5SM5N7rnnf23	positive_moderate	Crackles	0.8759342	Coarse Crackles	0.742215	none/crackles/both	tachypnea/fine crackles
joORHxc0iCTwgP7Uh5SM5N7rnnf24	positive_moderate	Crackles	0.9997760	Coarse Crackles	0.88538	none/crackles/both	tachypnea/fine crackles
joORHxc0iCTwgP7Uh5SM5N7rnnf25	positive_moderate	none (no respiratory markers found)	0.9154950	Tachypnea	NA	none/crackles/both	tachypnea/fine crackles

Table 8. Testing, results and accuracy of training done with COVID positive moderate patients from Coswara dataset on CNN

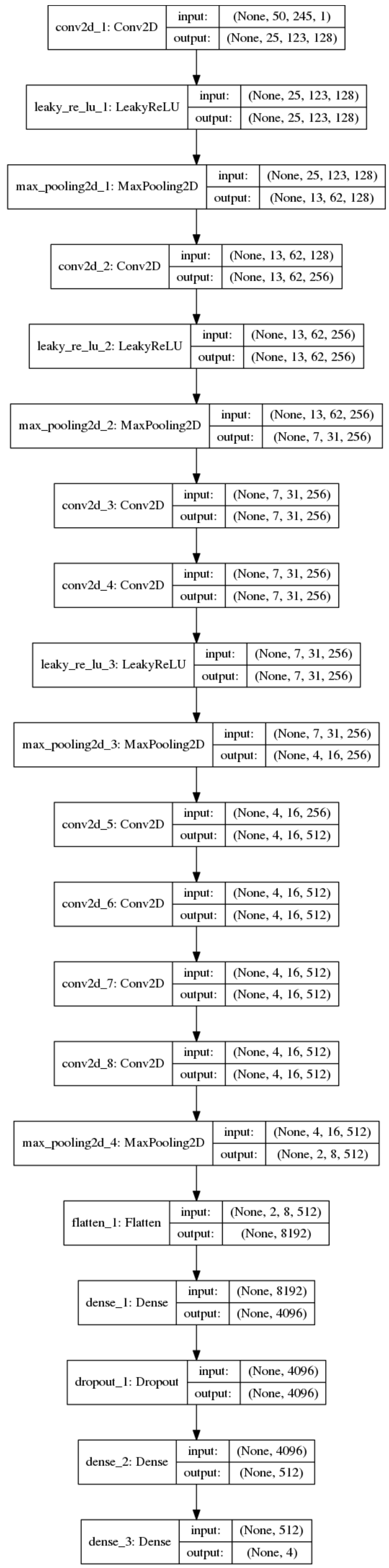


Figure 22. Model architecture of both first and second level CNN models