C19-Audit: Detecting COVID-19 Cases from Coughs Recordings with SVM and CNNs

1nd Renato Avellar Nobre
Computer Science Department
Università degli Studi di Milano
Milano, Italy
email address or ORCID

***Abstract*—During the COVID-19 pandemic, one of the main challenges to contain the virus was the testing methods. This challenge could be reduced if we had a widely available pre-screening method. A possible approach would be to use machine learning techniques to analyze the patient's cough. Therefore this work proposes C19-Audit, a framework to detect COVID infection using cough sounds and machine learn models. C19-Audit tries to classify audio using two approaches: an SVM with standard audio features and a CNN model with Mel spectrograms. Therefore, C19-Audit was designed to evaluate if cough audio is sufficient to detect COVID and analyze whether the CNN approach can overcome the SVM. Experimental results favored the traditional audio features techniques with an SVM compared to the CNN approach. One of the SVM models classified the problem with a 77% precision, 72% recall, and 72% f1-score, setting a good baseline for approaching the problem.**

***Index Terms*—COVID-19, audio processing, convolutional neural networks, support vector machines**

I. Introduction

The coronavirus disease, popularly known as COVID-19, has been declared a pandemic by the World Health Organization (WHO). It first appeared in Wuhan, China, and quickly spread to the whole world [1]. The COVID-19 is a severe acute respiratory syndrome (SARS-CoV2). It causes severe respiratory infections with very high mortality and poses a severe threat to humans [2]. The most common symptoms of the virus are severe fever, dry cough, and difficulty in breathing [2]. By the end of 2021 the virus accumulated approximately 5.7 million deaths and 393 million cases [1].

During this time, one of the main challenges to contain the virus was the knowledge of contamination [3]. The testing methods were time-consuming and not widely available due to overwhelming demand [1]. This challenge could be reduced if we had complementary pre-screening methods, allowing the patient to have easy access to a preliminary test [4].

A possible approach to a pre-screening method would be to analyze the patient's cough [4]. This approach is advantageous because every person with a microphone could be able to test. Cough audio signal classification has already been successfully used to diagnose a variety of respiratory conditions [5], and there has been a significant interest in machine learning studies to provide a widespread COVID-19 screening using audio information [4], [6]–[8].

Recent works have been employing significant effort to achieve pre-screening with audio tracks. Schuller *et al.* [7] uses a Convolutional Neutral Network (CNN) model as a ubiquitous, low-cost, pre-screening method for detecting COVID. It utilizes raw breathing and coughing audio together with spectrograms to classify if the patient is infected or not. The proposed method performed hyperparameter tuning using Bayesian Optimization, achieving an Area Under ROC Curve (AUC) of 80.7%. Tena *et al.* [6] designed a quick and efficient methodology for the automatic detection of COVID-19 in raw audio files. It performed automated extraction of time-frequency cough features and selected the more significant ones to be used on a supervised machine-learning algorithm. Random Forest achieved the best performance among multiple supervised algorithms with an accuracy close to 90%. Imran *et al.* [8] developed a machine learning solution in a smartphone app, named AI4COVID-19. The app records and sends three 3 second cough sounds to an AI engine to classify. Its methods use transfer learning from a multi-pronged mediator-centered risk-averse AI architecture. Results show AI4COVID-19 can distinguish between COVID-19 coughs and several types of non-COVID-19 coughs, with an accuracy of 88.76%. Finally, Erdoğan *et al.* [4] aimed to detect COVID-19 patients with cough acoustic data. The features were obtained from empirical mode decomposition (EMD) and discrete wavelet transform (DWT). Deep features were also obtained using pre-trained ResNet50 and pre-trained MobileNet models. The study obtained an accuracy of 98.4% and an F1-Score of 98.6%. Its results determined that the features obtained by traditional feature approaches show higher performance than deep features.

Keeping in mind the necessity for fast and widely accessible pre-screening methods and the recent efforts to classify COVID-19 with audio data, this report proposes C19-Audit, a framework for detecting COVID-19 in cough data. C19-Audio binary classifies a cough between COVID infected or non-infected. Therefore the remaining of this report, a non-infected COVID-19 person might be referred to as healthy. C19-Audit uses two approaches: a Support Vector Machines (SVM) and a Convolutional Neural Network (CNN) model. While the SVMs have been widely used and achieved outstanding results in traditional audio classification tasks, the Convolutions Neural Networks have gained space approaching the problem as an image to be classified [8].

Therefore, C19-Audit was designed to evaluate if cough audio is sufficient to detect COVID using machine learning

techniques and to analyze whether the CNN approach can overcome the SVM.

Experimental results also favored the traditional audio features techniques with an SVM compared to the CNN approach. From the observable results we can infer that the SVM with the Coswara dataset was the most capable of classifying the problem, with a 77% precision, 72% recall, and 72% f1-score. Even though its values are not competitive compared to recent literature, it can be considered a good baseline for approaching the problem.

The remainder of this work is organized as follows. Section II proposes and explains the classification system while laying out the theoretical foundation necessary for understanding its behavior. Section III explains the methodology used to validate the proposed solution and presents the results obtained from the experimental procedure. Finally, Section IV presents the main conclusions and future research.

II. Method

This section describes the C19-Audit framework, a pre-screening method that performs binary classification on cough audio files, aiming to differentiate between a COVID-19 infected and a healthy patient. For that, the C19-Audit takes advantage of two different approaches, an SVM and a CNN model. On the one hand, SVMs have been widely used and achieved outstanding results in traditional audio classification tasks using standard audio features [9]. On the other hand, the Convolutions Neural Networks have gained space in audio classification tasks, using time-frequency representations such as the spectrograms and the Mel-spectrograms as an image to be classified [8]. Therefore, C19-Audit was designed to evaluate two different research questions:

- Can we detect COVID-19 only from cough audio of an infected person, using either traditional or deep learning methods?
- Is the CNN architecture feed with log-Mel spectrograms capable of overcoming the results of a traditional SVM classification with traditional feature extraction?

A. Overview

Figure 1 shows an overview of the C19-Audit framework. C19-Audit receives as input 16-bit audio files in the ".wav" extension. It starts by opening the audio file, normalizing it between $[-1, 1]$, and fixing its minimum length to a predetermined value denominated AUDIO_LEN (measured in sampling frames). If the audio is smaller than the minimum length, zero padding is applied at the end of the audio. After this, we can have a fixed length waveform. This waveform can be used to compute the audio spectrogram and the audio's time-domain features (see Figure 1).

The spectrogram is a representation of the signal and presents the evolution of the signal in the time-frequency domain [9]. It is calculated with Short-Time Fourier Transform (STFT), which breaks the signal into overlapping frames using a moving window technique and computes the Discrete Fourier Transform (DFT) at each frame [9]. The STFT is therefore represented as a matrix of coefficients, where the column

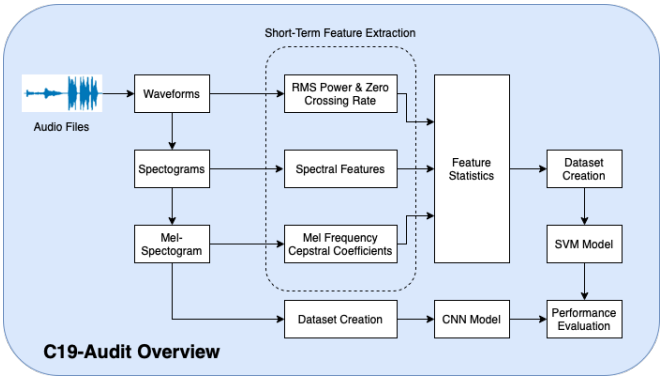


Fig. 1: Overview of C19-Audit framework

index represents time, and the row index is associated with the frequency of the respective DFT coefficient [9]. After that, we compute the magnitude of each coefficient, obtaining the spectrogram.

The length of the moving window to compute the spectrogram plays an important role. It defines the frequency resolution of the spectrum, given the sampling frequency [9]. Longer windows lead to better frequency resolution at the expense of decreasing the quality of time resolution. In comparison, shorter windows provide a more detailed representation in the time domain, but with a poor frequency, resolution [9]. Therefore, C19-Audit controls the window length with a variable denominated N_FFT and the distance of the overlapping frame with a variable called HOP_LENGTH. Its values were explored in the system to allow for better results in the classification process.

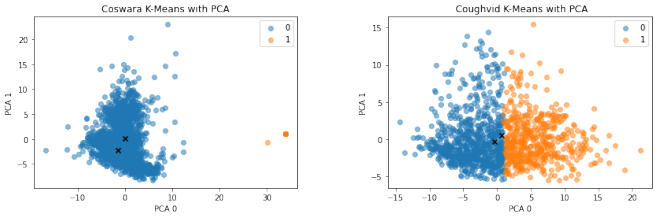
After the spectrogram generation, another representation was created, the log Mel spectrogram, a representation of the spectrum where the frequencies have been set in the Mel Scale [9]. A MEL scale is a unit of pitch proposed by Stevens *et al.* [10]. The scale is based on the way humans distinguish between frequencies which makes it very convenient to process sounds [9]. Therefore, the MEL scale is a scale of pitches judged by listeners to be equal in distance one from another. Because of how humans perceive sound, the MEL scale is non-linear, and the distances between the pitches increases with frequency [10]. The Mel spectrogram, together with the magnitude spectrogram and the waveforms, was used to calculate multiple audio features.

C19 was implemented using the Python 3 programming language with the Pandas, Tensorflow, Scikit Learn, Librosa, and matplotlib libraries.

B. Feature Extraction

Using the generated signal representations (as seen in Figure 1), a set of time, frequency, and Mel frequency features were generated:

- **RMS Power:** Compute root-mean-square (RMS) value for each frame [9].
- **Zero Crossing Rate (ZCR):** Rate of sign-changes of the signal during the frame [9]. It is the number of times the signal changes value, from positive to negative and vice versa, divided by the length of the frame.



(a) Coswara Visualization Matrix (b) Coughvid Visualization Curve

Fig. 2: Visualizing Feature Space with Kmeans and PCA to assess the difficulty level of the problem.

- **Spectral Centroid & Spread:** The spectral centroid and the spectral spread are two simple measures of spectral position and shape [9]. The spectral centroid is the center of "gravity" of the spectrum, while the spread is the second central moment of the spectrum [9].
- **Spectral Entropy:** This feature is computed similarly to the entropy of energy, although, this time, the computation takes place in the frequency domain [9].
- **Spectral Flatness:** The Spectral Flatness is a measure to quantify how much noise-like a sound is, as opposed to being tone-like. A high spectral flatness (closer to 1.0) indicates the spectrum is similar to white noise [9].
- **Spectral Rolloff:** The frequency below which a certain percentage (usually around 90%) of the magnitude distribution of the spectrum is concentrated [9].
- **Mel Frequency Cepstral Coefficients (MFCCs):** The MFCC use the MEL scale to divide the frequency band into sub-bands and then extract the Cepstral Coefficients using Discrete Cosine Transform (DCT) [9].

Each feature yield a one or multi-dimensional vector with the size of the number of windows that can fit in the audio sample. After the features' calculation, statistical values of the vector are calculated (mean, standard deviation, maximum, minimum, and median) to serve as inputs to the desired model.

With the resulting feature vector a K-means clustering algorithm was employed, allowing to visualize the difficulty level of the problem. Figure 2 shows the obtained results using PCA to reduce the feature spaces. It is clear by observing the figures that the clustering results are not promising. Additionally, by evaluating the cluster silhouette score, we note that the clusters are not well defined, the Coswara achieved a 0.62 score while Coughvid achieved a 0.16 score.

C. SVM Model

Support Vector Machines (SVMs) [11] are state-of-the-art classifiers that have been successfully employed in numerous machine learning fields [9]. They are based on the critical observation that, for the simple case of linearly separable classes, the optimal decision hyperplane is the one that maximizes the 'margin' among the training data of the two classes [11]. C19-Audit modeled SVM is designed to receive a vector of audio pattern features. The features used for the input data were the following:

- **Time Domain:** RMS Power, Zero Crossing Rate
- **Frequency Domain:** Spectral Centroid, Spectral Bandwidth, Spectral Contrast (Size 7), Spectral Flatness, Spectral Rolloff
- **Mel Frequency Domain:** 13 MFCCs

The mean, standard deviation, maximum, minimum, and median statistics were calculated for each feature.

Additionally, for the SVM to classify correctly, we need to define its kernel function (order to map the feature vectors to the 'kernel space'), the constraint parameter C (related to the cost function of the SVM training procedure), and the gamma value (how a single training example influence the training [11]. The input features used and the values for the given parameter are discussed further in the experimental results.

D. CNN Model

Finally, C19-Audit uses a Convolutional Neural Network (CNN) structure as a second classification method. A CNN is a class of artificial neural networks, most commonly applied to analyze images [12]. They are based on convolving their input with learnable kernels [9]. A convolutional layer typically computes multiple feature channels, each from its corresponding kernel. Additionally, pooling layers can be used to down-sample the learned feature maps [12].

C19-Audit uses the log Mel Spectogram as an input feature for CNN. Therefore, the CNN will analyze the spectrogram considering it as an image. This approach has demonstrated increased usage in recent literature. The implemented CNN had the following layers in sequential order: resizing (to downsample to 32 by 32), normalization, 32 and 64 filters convolutions with kernel size three and "relu" activation, a max-pooling layer, a dropout layer of 0.25, a flatten layer, a dense layer with 128 neurons and "relu" activation, a dropout layer of 0.5, and finally a dense classifier layer with one neuron and sigmoid activation.

III. Performance Evaluation and Methodology

This section presents the methodology adopted and the results obtained to validate the proposed methods. For this, the validation was made by training and evaluating the validation results for each technique with each of the selected datasets, aiming to answer our proposed research questions.

A. Datasets

To evaluate our proposed methodology, two different datasets were selected.

1) Coswara: One of the dataset used was the Coswara[1] [13]. This dataset contained multiple metadata and respiratory sounds: cough, breath, and voice. The sound samples were collected via worldwide crowdsourcing, using a website application [13].

The audio samples were recorded at a sampling frequency of 48 kHz. All sound files were manually curated. These helped verify the category label and the quality of the audio file. In total, the dataset has 6507 clean, 1117 noisy, and remaining highly degraded audio files corresponding to respiratory sound samples from more than 941 participants [13].

[1] Available at: https://github.com/iiscleap/Coswara-Data

2) Coughvid: The Coughvid[2] is a dataset that provides over 25,000 crowdsourced cough recordings representing a wide range of participant ages, genders, geographic locations, and COVID-19 statuses [5]. These recordings were preprocessed by lowpass filtering, and downsampling to 12 kHz [5].

Additionally, the Coughvid dataset has a subset of 2,800 recordings labeled by four experienced physicians to diagnose medical abnormalities present in coughs. Thereby contributing one of the most extensive expert-labeled cough datasets in the current literature that can be used for a plethora of cough audio classification tasks [5].

B. Data Preprocessing

For the datasets to be used adequately in the C19-Audit, a set of preprocessing and cleaning techniques were used.

In the Coswara dataset, multiple audio files were removed. Since C19-Audit main objective is to try to detect COVID using only cough sounds, the files related to breathing and voice were removed. Regarding the Coughvid data set, the subset of 2.800 specially curated files was selected for the classification task. This choice was motivated both for performance and data trustworthy issues. The Coughvid files were also transformed from ".egg" and ".webm" extensions to ".wav", and their bits were reduced from 32 to 16. Therefore allowing the data to be processed by the technologies used.

Additionally, none of the metadata was used for either of the datasets. Some corrupted audio was removed, and audios regarding unclear classification between healthy and COVID (such as other respiratory diseases) were removed.

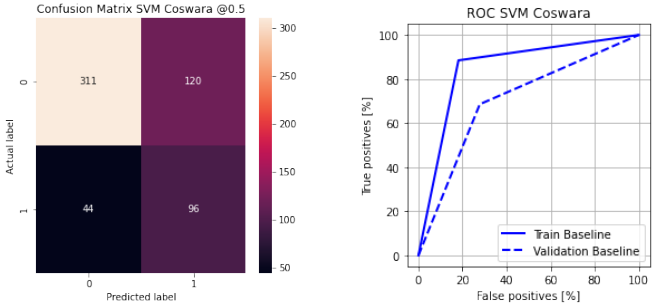
C. Evaluation Metrics

To analyze the experiments, the following metrics were used: (i) accuracy represents how close the models are to the correct value; (ii) precision represents how correct the positive predictions of the models were classified as correct, excluding false negatives; and (iii) recall represents how correct the positive predictions of the models were rated correct, excluding false positives. Additionally the F1-Score and the Area Under Curve (AUC) for the Receiver Operating Characteristic ROC (AUC-ROC) was also used for classification evaluation. Those metrics were specially chosen due to the nature of the problem. Both datasets are unbalanced, with a higher negative class value (healthy). Therefore accuracy alone is not a good metric to be used [14].

D. Experimental Results

To generate and evaluate the models in C19-Audit, each database was divided into training and validation. The training corresponds to 70% of the database. The other 30% was used for the validation phase. The experiments evaluated each dataset (Coswara and Coughvid) with each model (SVM and CNN). The dataset features were calculated in short-term using a window size N_FFT of 1024. A HOP_LENGHT of 256 and a "hann" window shape.

[2]Available at https://www.kaggle.com/andrewmvd/covid19-cough-audio-classification



(a) Validation Set Confusion Matrix

(b) Receiver Operating Characteristic Curve

Fig. 3: Results for the Coswara SVM model

The SVM models were configured with a cubic polynomial kernel, L2, to normalize the data and class weights for handling imbalance in the dataset. For the Coswara SVM combination was used a C value of 10000 and a gamma value of 3, while for the Coughvid SVM, those values were set to 100000 and 2 respectively. For the CNN, in which the input consisted of an equal length Mel spectrogram, a maximum of $100,000$ samples were considered. Therefore, smaller samples were zero-padded at the end. The model configurations for both the Coswara CNN and Coughvid CNN combinations consisted of an Adam optimizer with 0.0001 learning rate, a binary cross-entropy loss, class weights, and 100 epochs with a 20 epochs patience early stop.

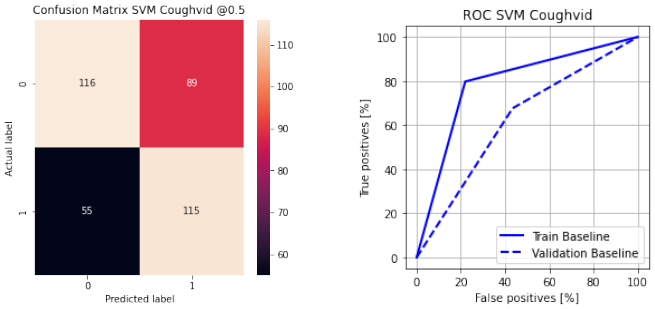
Table I summarizes the obtained results for each of the model-database combinations. Overall the SVM approach performed better than the CNN, and the Coswara dataset gave better models than the Coughvid. The Coswara SVM achieved the best precision 0.77, recall 0.72, and F1-score 0.72 while not under-performing in other metrics. The Coswara CNN reached the best accuracy 0.76 and AUC-ROC 0.73.

TABLE I: Results

Datasets	Training and Validation Metrics				
	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Coswara SVM	0.83/0.71	0.86/**0.77**	0.83/**0.72**	0.84/**0.72**	0.85/0.70
Coughvid SVM	0.78/0.61	0.78/0.62	0.78/0.61	0.78/0.61	0.78/0.62
Coswara CNN	0.83/**0.76**	0.72/0.52	0.68/0.50	0.70/0.51	0.91/**0.73**
Coughvid CNN	0.57/0.47	0.56/0.45	0.78/0.69	0.65/0.54	0.62/0.52

The SVM models had a performance that was better overall when compared to the CNN. Coswara SVM had the best results; Figure 3 shows the confusion matrix and the ROC curve of the classifier. Especially Figure 3b shows the best curve, in which we can achieve almost 70% true positives with around 25% false positives. For the Coughvid SVM, the true positive rate is about 70% with an almost 40% false positive, as seen on 4. Results worse than the Coswara SVM.

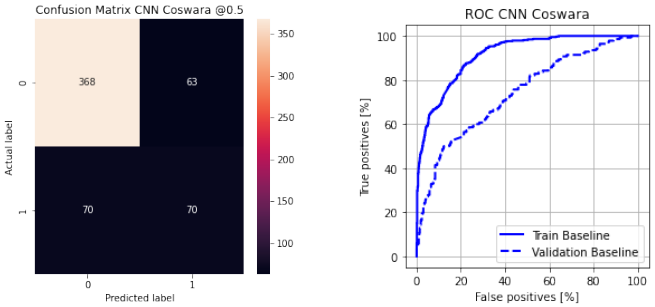
While the Coswara CNN model has the best overall accuracy, its values of precision-recall and F1-Score are not as significant as the Coswara SVM. It can be observed in Figure 5a that its results are mainly on predicting non-COVID relations. This is why accuracy alone should not be considered with unbalanced datasets. Finally, the Coughvid CNN presented average values on all considered metrics. Observing Figure



(a) Validation Set Confusion Matrix

(b) Receiver Operating Characteristic Curve

Fig. 4: Results for the Coughvid SVM model



(a) Validation Set Confusion Matrix

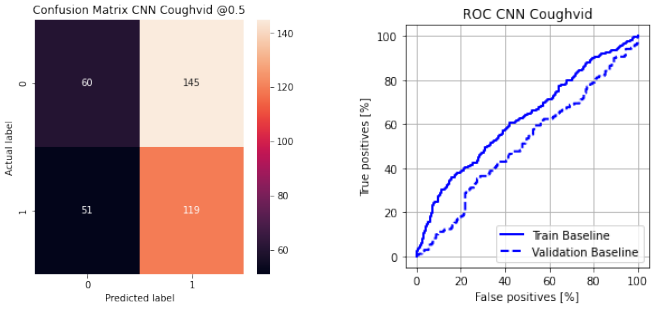
(b) Receiver Operating Characteristic Curve

Fig. 5: Results for the Coswara CNN model

6.b it is noted that the ROC is very near the random guessing line, which could indicate that the model had problems fitting the data.

From the observable results we can infer that the SVM with the Coswara dataset was the most capable of classifying the problem. However its results are not competitive when compared to recent literature. One of the possible reasons for these results is that cough information alone might not be sufficient for a clear classification. In Schuller *et al.* [7], the 80.7% ROC-AUC is achieved using both breathing and coughing audio, while in Tena *et al.* the 90% accuracy was achieved with deep automated feature extraction.

Experimental results also favored the traditional audio features techniques with an SVM compared to the CNN approach. This can be related to the simplicity of the employed CNN. It is possible that CNNs derived with transfer learning can present better results. When compared to recent literature both Imram *et al.* [8] and Erdogan *et al.* [4] uses transfer learning mechanism, which can favor the high accuracy obtained.



(a) Validation Set Confusion Matrix

(b) Receiver Operating Characteristic Curve

Fig. 6: Results for the Coughvid CNN model

IV. Conclusion

COVID-19 detection is an important aspect to solve in the current world situation. This work proposed C19-Audit, a framework for detecting COVID using two different models and datasets. Experimental hypotheses tried to discover if COVID-19 could be detected only with the cough audio and whether simple CNN methods can overcome classical SVM solutions. Results demonstrated a favor for the SVM approach, which achieved 77% precision, 72% recall, and 72% f1-score, setting a good baseline for approaching the problem.

Future works could approach the problem of classifying cough audio in a time-series approach. It might exist features in an infected cough related to how it reverberates in time. Additionally, future work could explore a mixed approach classification, considering both the audio file and other related symptoms features.

References

[1] W. H. Organization *et al.*, "Coronavirus disease (covid-19). 2021," 2021.

[2] C.-C. Lai, T.-P. Shih, W.-C. Ko, H.-J. Tang, and P.-R. Hsueh, "Severe acute respiratory syndrome coronavirus 2 (sars-cov-2) and coronavirus disease-2019 (covid-19): The epidemic and the challenges," *International journal of antimicrobial agents*, vol. 55, no. 3, p. 105924, 2020.

[3] G. Lippi, C. Mattiuzzi, C. Bovo, and M. Plebani, "Current laboratory diagnostics of coronavirus disease 2019 (covid-19)," *Acta Bio Medica: Atenei Parmensis*, vol. 91, no. 2, p. 137, 2020.

[4] Y. E. Erdoğan and A. Narin, "Covid-19 detection with traditional and deep features on cough acoustic signals," *Computers in Biology and Medicine*, vol. 136, p. 104765, 2021.

[5] L. Orlandic, T. Teijeiro, and D. Atienza, "The coughvid crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms," *Scientific Data*, vol. 8, no. 1, pp. 1–10, 2021.

[6] A. Tena, F. Clarià, and F. Solsona, "Automated detection of covid-19 cough," *Biomedical Signal Processing and Control*, vol. 71, p. 103175, 2022.

[7] B. W. Schuller, H. Coppock, and A. Gaskell, "Detecting covid-19 from breathing and coughing sounds using deep neural networks," *arXiv preprint arXiv:2012.14553*, 2020.

[8] A. Imran, I. Posokhova, H. N. Qureshi, U. Masood, M. S. Riaz, K. Ali, C. N. John, M. I. Hussain, and M. Nabeel, "Ai4covid-19: Ai enabled preliminary diagnosis for covid-19 from cough samples via an app," *Informatics in Medicine Unlocked*, vol. 20, p. 100378, 2020.

[9] T. Giannakopoulos and A. Pikrakis, *Introduction to audio analysis: a MATLAB® approach*. Academic Press, 2014.

[10] S. S. Stevens and J. Volkmann, "The relation of pitch to frequency: A revised scale," *The American Journal of Psychology*, vol. 53, no. 3, pp. 329–353, 1940.

[11] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 144–152.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[13] N. Sharma, P. Krishnan, R. Kumar, S. Ramoji, S. R. Chetupalli, P. K. Ghosh, S. Ganapathy *et al.*, "Coswara–a database of breathing, cough, and voice sounds for covid-19 diagnosis," *arXiv preprint arXiv:2005.10548*, 2020.

[14] S. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Pearson, 2002.