APPLIED COMPUTER SCIENCE

Applied Computer Science, vol. 18, no. 4, pp. 96–115
doi: 10.35784/acs-2022-31

Submitted: 2022-11-01 | Revised: 2022-12-07 | Accepted: 2022-12-19

Keywords: COVID-19, cough recordings, machine learning, PCA, classification

Elmehdi BENMALEK [0000-0003-1078-1421]*,
Jamal EL MHAMDI [0000-0001-8219-3560]*, *Abdelilah JILBAB* [0000-0002-1577-9040]*,
Atman JBARI [0000-0002-1855-2503]*

A COUGH-BASED COVID-19 DETECTION SYSTEM USING PCA AND MACHINE LEARNING CLASSIFIERS

Abstract

In 2019, the whole world is facing a health emergency due to the emergence of the coronavirus (COVID-19). About 223 countries are affected by the coronavirus. Medical and health services face difficulties to manage the disease, which requires a significant amount of health system resources. Several artificial intelligence-based systems are designed to automatically detect COVID-19 for limiting the spread of the virus. Researchers have found that this virus has a major impact on voice production due to the respiratory system's dysfunction. In this paper, we investigate and analyze the effectiveness of cough analysis to accurately detect COVID-19. To do so, we performed binary classification, distinguishing positive COVID patients from healthy controls. The records are collected from the Coswara Dataset, a crowdsourcing project from the Indian Institute of Science (IIS). After data collection, we extracted the MFCC from the cough records. These acoustic features are mapped directly to the Decision Tree (DT), k-nearest neighbor (kNN) for k equals to 3, support vector machine (SVM), and deep neural network (DNN), or after a dimensionality reduction using principal component analysis (PCA), with 95 percent variance or 6 principal components. The 3NN classifier with all features has produced the best classification results. It detects COVID-19 patients with an accuracy of 97.48 percent, 96.96 percent f1-score, and 0.95 MCC. Suggesting that this method can accurately distinguish healthy controls and COVID-19 patients.

* E2SN, ENSAM de Rabat, Mohammed V University in Rabat, Morocco, elmehdi.benmalek@um5s.net.ma, mhamdi_jamal@yahoo.fr, a_jilbab@yahoo.fr, atman.jbari@ensam.um5.ac.ma

1. INTRODUCTION

The coronavirus disease 2019 (COVID-19) is a pandemic caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). As of July 2022, more than 560 million cases of COVID-19 have been reported in 223 countries, regions, and territories, resulting in more than 6.36 million deaths (World Health Organization). Symptoms of COVID-19 are nonspecific and the presentation of the disease can vary from no symptoms (asymptomatic patients) to severe pneumonia and death. In the majority of cases (about 80 percent), people infected with COVID-19 have mild to moderate symptoms (eg, cough, fever, fatigue) while 14 percent have severe symptoms (eg, dyspnea and hypoxemia), and 6 percent have critical clinical condition (eg, respiratory failure, septic shock, multi-organ failure) (Weng, Su & Wang, 2021).

To control the spread of COVID-19, effective patient screening is essential. So far, the gold standard screening method is the polymerase chain reaction-reverse transcription (RT-PCR) assay which was designed to detect SARS-CoV-2 genetically (Ismail, Deshmukh & Singh, 2021). Unfortunately, the RT-PCR takes time and necessitates the use of medical expertise, which may not be available. On the other hand, some investigations have found that RT-PCR testing has a high percentage of false-positive results (Ai et al. 2020; Yang et al., 2020). Also, the quick detection of this virus is very important, to reduce the likelihood and risk of COVID-19 spreading. As a result, researchers in the fields of virology, medicine, and artificial intelligence (AI) have stepped up to find creative ways to contain this issue. The AI community made major contributions and proposed computer-aided diagnostics systems that can help identify, forecast, and treat COVID-19. With the use of machine learning (ML) technology, computers can mimic human intelligence and find patterns and information in massive amounts of data to comprehend the spread of COVID-19, and speed up research and treatment.

Numerous researchers have suggested using speech signals and medical imaging to automatically detect patients suffering from COVID-19. A dataset comprised of 368 COVID-19 positive patients and 127 other pneumonia cases from two hospitals in China was used by (Wu et al., 2020) to apply ResNet50 with the multi-view fusion approach. They attained 76 percent accuracy, 81.1 percent sensitivity, 61.5 percent specificity, and 81.9 percent AUC. ResNet50 and CT scans were used by (Li et al., 2020) for an automatic coronavirus diagnosis (COVNet). 4536 chest CT samples were used in total (1296 COVID-19 cases, 1325 non-pneumonia observations, and 1735 community-acquired pneumonia cases). For COVID-19 instances, their approach achieved a sensitivity of 90 percent, specificity of 96 percent, and AUC of 96 percent. ResNet-18 architecture has the best overall precision and sensitivity of 98.5 percent and 98.6 percent, using CT scan images when (Benmalek, Elmhamdi & Jilbab, 2021) compared the performances of CT scan and CXR images in the diagnosis of COVID-19. To determine how distinguishable COVID-19 sounds are from those in asthma or healthy controls, (Brown et al., 2020) used coughs and breathing. Their model achieves an AUC of above 80 percent for all tasks. A voice-based approach has been suggested by (Han et al., 2021) to automatically identify those who have tested positive for COVID-19. AUC of 0.79, a specificity of 0.82, and a sensitivity of 0.68 have been achieved. A medical dataset comprising 328 cough sounds from 150 patients split into four classes (Healthy, Asthma, Bronchitis, and COVID-19) was used by (Pal & Sankarasubbu, 2020). With specificity and accuracy of 95.04 percent and 96.83 percent, respectively, the experiment findings show that their model captures a more robust feature embedding to distinguish between COVID-19 patient coughs and diverse sorts of coughs that are not COVID-19.

Based on several studies that suggest the voice of COVID-19 patients is infected by the disease (Han et al., 2021), we aim in this work to detect COVID-19 by applying the MFCC, different machine learning classifiers, and dimensionality reduction using the PCA. The evolution of the models was done by the confusion matrix, sensitivity, specificity, precision, f1-score, and Matthews correlation coefficient (MCC). The main contribution of this paper is to explore the effectiveness of the ML and AI algorithms in improving disease screening, diagnosis, and monitoring of the COVID-19 pandemic and reducing the need for human involvement in a way that lessens the burden on the healthcare industry.

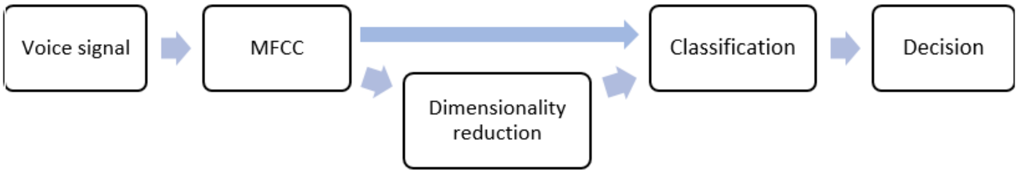


Fig. 1. COVID-19 diagnosis with cough recording diagram

2. METHODS

2.1. Dataset

The data is collected from the Indian Institute of Science Bangalore's Coswara Project (Sharma et al., 2020). The dataset is a collection of vowel (/a/, /e/, and /o/) sustained phonation, a counting exercise, breathing sounds, and cough recordings that we employed in this study. The gathering of records began on April 13th, 2020.

Reaching out to the global human population was the main goal of the data collection strategy. To do so, a website application was developed with a straightforward and interactive user interface. Open the application in a web browser on a computer or mobile device, enter the necessary metadata, without identification data, and then start recording sound samples with the device's microphone. The program is used on average for 5 to 7 minutes. The user was instructed to use a personal device, clean it with sanitizer before and after recording, and keep it 10 cm away from their mouths while they were recording. The dataset is divided into positive COVID-19 cases with 77 observations and 82 healthy controls representing the true negatives. The demographic information, symptoms, and comorbidities are represented in Tab. 1 and 2 for each class.

Tab. 1. Sex and age group of the participants for each class

		Age (years old)				
	Sex	**[20-29]**	**[30-39]**	**[40-49]**	**[50-59]**	**>=60**
Positive	45 Males 32 Females	48	9	6	11	3
Negative	54 Males 28 Females	44	11	7	12	8

Tab. 2. Disease symptoms and comorbidities of the participants for each class

	asthma	cold	cough	loss_of_smell	diabetes	fever	pneumonia
Positive	2	13	19	5	1	14	1
Negative	0	0	0	0	2	0	0

2.2. Mfcc

Mel Scale Frequency Spectral Coefficients are the most widely used parameters in speech processing for automated voice speaker, and language recognition, as well as the detection and classification of speech pathologies (Zheng, Zhang & Song, 2001). The principle for calculating MFCCs comes from psychoacoustic research on the perception of different frequency bands by the human ear. The main interest of these coefficients is to extract relevant information in a limited number based on both production (Cepstral theory) and speech perception (Mels scale) (Muda, Begam & Elamvazuthi, 2010). For this study, we used 14 MFCCs.

MFCCs are used in various speech processing techniques. The basic procedure for developing them is represented in Fig. 2:

- Convert Hertz to Mel Scale,
- Take the logarithm of the Mel representation of the audio,
- Take a logarithmic magnitude and use the discrete cosine transformation,
- This result creates a spectrum on Mel frequencies as opposed to time, thus creating MFCCs.

According to the MFCC calculation, the number of filters, their shape, how they are spaced and whether or not they overlap each other, and how the power spectrum is warped can all have an impact on how well the MFCC performs (Han et al., 2006).

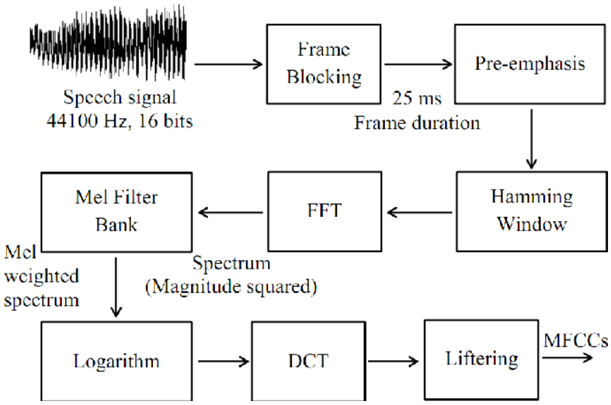


Fig. 2. Block Diagram of the computation steps of MFCC

2.3. Classification algorithms

a) K-NN

the K-NN (K-nearest neighbors) algorithm is a supervised learning method. It can be used for both regression and classification (Indyk & Motwani, 1998). Its operation can be likened to the following analogy “tell me who your neighbors are, I will tell you who you are…”.

To make a prediction, the K-NN algorithm will not calculate a predictive model from a training set as is the case for other machine learning classifiers. Indeed, K-NN does not need to build a predictive model. Thus, for K-NN there is no actual learning phase. This is why it is sometimes categorized as Lazy Learning. To be able to make a prediction, K-NN relies on the dataset to produce a result.

The importance of K is that it affects how accurate and efficient the algorithm is. Other KNN algorithm extensions that have been proposed include the weighted KNN classifier, the K-means KNN classifier, the Shared Nearest Neighbor KNN classifier, and the SVM KNN classifier. These reduce execution time and increase accuracy. A KNN example can be seen in Fig. 3, which includes training samples with two classes: "blue square" and "red triangle." The green circle designates the test sample. These samples are set up in two-dimensional feature spaces, where each feature has its own dimension. To identify whether a test sample belongs to the "blue square" or the "red triangle" class, KNN uses a distance function to identify the test sample's K closest neighbors. The test sample's class can be predicted by finding the majority of classes among the k closest neighbors. Due to the presence of two red triangles, the test sample in this instance is classified to the first class “red triangle” when k = 3. However, when k = 5, it is classified as the second class, "blue square," since there are two red triangles and three blue squares.

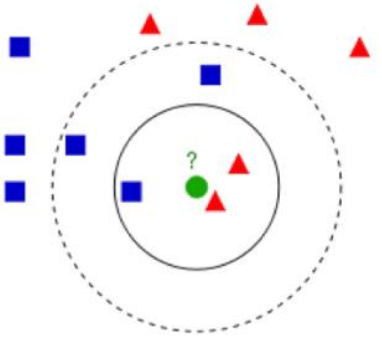


Fig. 3. An example of KNN classification with K neighbors K = 3 (solid line circle) and K = 5 (dashed line circle), the distance measure is Euclidean distance.

b) SVM

Support vector machines or wide-margin separators are a set of supervised learning techniques designed to solve regression or classification problems. They were introduced by V.Vapnik in 1995 in his book "The nature of statistical learning theory", but their first appearance was in 1992 after they were published by (Boser, Guyon & Vapnik, 1992).

The dimensionality of the data and its increased power of generalization, make the SVM more advantageous. SVM is widely used as a binary classifier in most fields. Its objective is to find the optimal boundary that separates two classes with the largest margin between the separation boundary and the support vectors (Fig. 4). SVM could surpass more sophisticated classifiers like deep neural networks for some classification problems, where the model selection in supervised ML should typically be based on the model's suitability for answering the specific question at hand, despite a natural intuition dictates that model complexity equates to model superiority (Pisner & Schnyer, 2020).

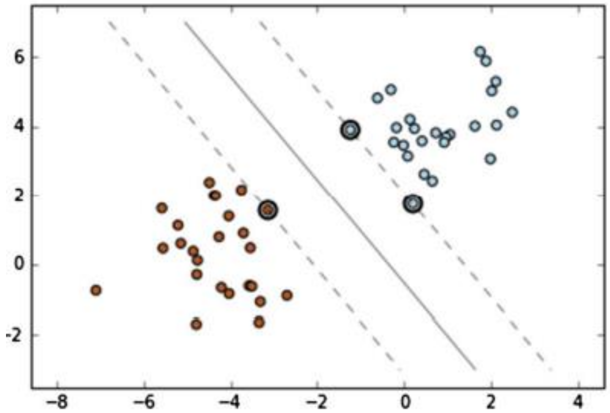


Fig. 4. An infinite number of classifiers can be drawn for the given data but SVM finds the classifier with the largest gap between support vectors. Circles represent the support vectors

When compared to other types of classifiers, SVM's strength and appeal largely come from its capacity to deliver balanced performance – high accuracies that are generalizable – even in situations where the dimensionality of the feature space significantly exceeds the number of observations available for training.

c) Decision tree

Decision trees represent one of the best-known and most-used techniques in classification (Quinlan, 1986). Their success is notably due to their ability to deal with complex classification problems. Indeed, they offer a representation that is easy to understand and interpret, as well as an ability to produce logical classification rules.

A decision tree is made up of:

- decision nodes each containing a test on an attribute,
- branches generally corresponding to one of the possible values of the selected attribute,
- sheets including objects that belong to the same class (Fig. 5).

The use of decision trees in classification problems is done in two main steps:

- the construction of a decision tree from a learning base,
- classification or inference consists in classifying a new instance from the decision tree built in the first step.

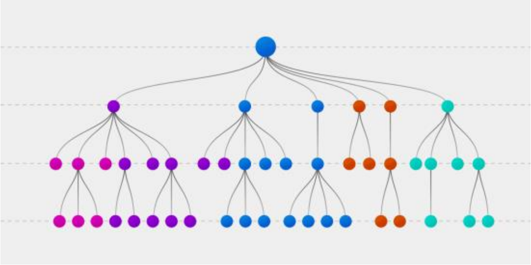


Fig. 5. Decision Tree Structure

With the use of algorithms like decision trees Iterative Dichotomiser 3 (ID3), C4.5, C5.0, and classification and regression trees (CART), problems relating to the classification, regression, clustering, and optimization can be resolved (Anuradha & Velmurugan, 2014). The ID3 and C4.5 algorithms were both created by Ross Quinlan; the latter is an enhanced variation of the former. Only categorical type features are applicable in ID3 decision trees; no numerical type features are allowed. The usage of information gain ratio (IGR) rather than information gain as in ID3 is one of the enhancements in C4.5. Second, pruning can be carried out both during and after tree building. Thirdly, C4.5 is capable of handling attributes with continuous features. Also can handle missing data (Adhatrao et al., 2013).

d) DNN

Multilayer neural networks are used in deep learning to design supervised and unsupervised learning mechanisms. In these mathematical architectures, each neuron performs simple calculations but the input data passes through several layers of calculation before producing an output. The results of the first layer of neurons are used as input for the calculation of the next layer and so on (Fig. 6). It is possible to play on the different parameters of the network architecture: the number of layers, the type of each layer, and the number of neurons that make up each layer (Bengio, 2009).

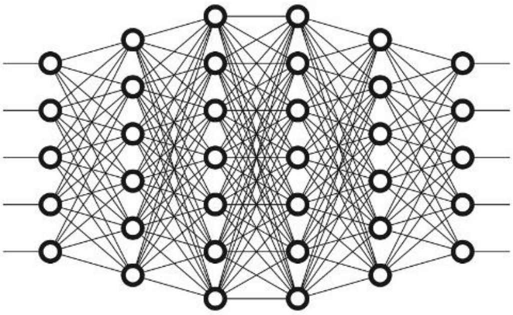


Fig. 6. Deep neural network structure

The multi-layer perceptron, capable of processing nonlinear phenomena, is an example of this type of network. Concretely, the first layers will make it possible to extract simple characteristics, which the following layers will combine to form increasingly complex and abstract concepts. Deep learning algorithms have mostly been used to improve computer capabilities so that they can comprehend what humans can accomplish, including speech recognition. Since the advent of artificial intelligence five decades ago, speech in particular – the primary means of human communication – has attracted a lot of attention (Singh & Bathla, 2013; Anusuya & Katti, 2009). As a result, it comes as no surprise that speech was one of the first applications of deep learning. To date, a large number of research papers have been published on the use of deep learning in speech-related applications, notably speech recognition (Singh & Bathla, 2013; Anusuya & Katti, 2009; Nassif et al., 2019).

2.4. Dimensionality reduction

Dimension reduction is a process studied in mathematics and computer science. It consists of taking data in a high-dimensional space and replacing them with others in a lower-dimensional space, which still contains most of the information contained in the large set. In other words, we seek to construct fewer variables while retaining as much information as possible.

Principal component analysis, or PCA, is a dimensionality reduction method that is often used to transform a large set of variables into a smaller set that still contains most of the information in the large set. The idea is to transform correlated variables into new uncorrelated variables by projecting the data in the direction of increasing variance. The variables with the maximum variance will be chosen as the principal components. To do this, we must first find a new orthonormal basis in which we will represent our data, such that the variance of these data along these new axes is maximized.

2.5. Model evaluation

The dataset is divided into two groups. For each label, 80 percent of the data are utilized for training, while the remaining 20 percent are used for testing. For the training set, we have 63 positives and 66 negatives. As for the test set, we used 14 positives and 16 negatives. To improve our model's performance, we took all the MFCC extracted from the cough records. Meaning we have done the classification based on the frames of the signals, and the frames are labeled based on the original records. Resulting in 12684 observations, which were divided as follows:

- 7424 negatives (5940 training and 1484 test),
- 5260 positives (4208 training and 1052 test).

The acoustics features of the test set were extracted separately from the training set, so no test frame was used while forming our models, to avoid data leakage.

a) k-fold validation

We use the k-fold cross-validation method to perform cross-validation. In k-fold cross-validation, the input data is divided into k subsets of data (in this work we set k to 5). The machine learning model is trained on all but one subset (k-1), then evaluates the model on the subset that was not used for training. This process is repeated k times, with a different subset reserved for evaluation (and excluded from training) each time. After validating the models with the training set using 5-fold cross-validation, we tested the models with the test set, firstly by frames, then by subjects.

b) Confusion matrix

The confusion matrix is like a summary of the prediction results for a particular classification problem. It compares the actual data for a target variable to that predicted by a model. Correct and false predictions are revealed and distributed by class, which allows them to be compared with defined values.

Also known as a contingency table, the confusion matrix is used to evaluate the performance of a classification model. It shows how confusing a certain model can be when making predictions. In its simplest form, it is a 2×2 matrix. For more complex classification problems, it is always possible to add rows and columns to the basic form.

c) Evaluation metrics

Sensitivity is a measure of how many actual positives have been correctly classified as such (true positive rate, recall):

$$Sensitivity\ = \frac{TP}{TP + FN} \tag{1}$$

A measure of how many of the negative observations in the data are classified negative is called specificity (also known as true negative rate or selectivity):

$$Specificity\ = \frac{TN}{TN + FP} \tag{2}$$

Precision is the ratio of true positives to all the subjects predicted to be positive:

$$Precision\ = \frac{TP}{TP + FP} \tag{3}$$

Accuracy is the proportion of true positives and true negatives, to all the subjects:

$$Accuracy\ = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

F1 score: a recall and precision-based indicator of test accuracy:

$$F1\ score\ = \frac{(2 * TP)}{(2 * TP + FN + FP)} \tag{5}$$

The binary classification of a sample's expected and actual class are correlated and measured by the Matthews correlation coefficient (MCC). The scale of this coefficient is defined as: (+1) indicates an accurate prediction. A score of (-1) indicates that the prediction and result are completely inconsistent, whereas a score of (0) provides no useful information.

$$MCC\ = \frac{[(TP * TN) - (FP * FN)]}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{6}$$

The ROC curve is a graphical representation of the relationship between the sensitivity and specificity of a test for all possible cut-off values. The ordinate represents the sensitivity and the abscissa corresponds to the quantity (1 – specificity). The calculation of the numbers

VP, FP, VN, and FN makes it possible to deduce the sensitivity and specificity of the test for each value obtained. The couples {1 – specificity, sensitivity} are then placed on the curve. Joining them with straight lines leads to a stepped path connecting the lower left corner of the graph (Se = 0 and Sp = 1) to the upper right corner (Se = 1 and Sp = 0).

3. RESULTS

In this section, we present the results achieved by the models using the test set (30 observations/2536 frames):

- 16 negatives (1484 frames),
- 14 positives (1052 frames).

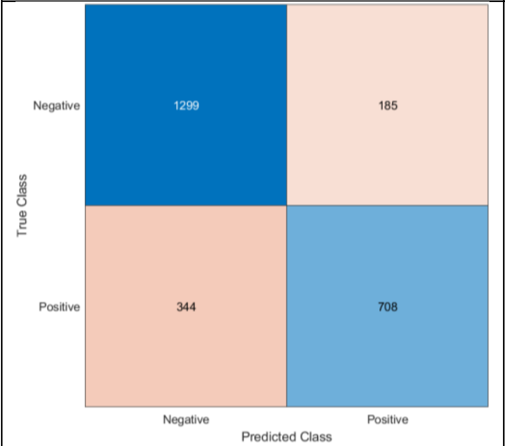
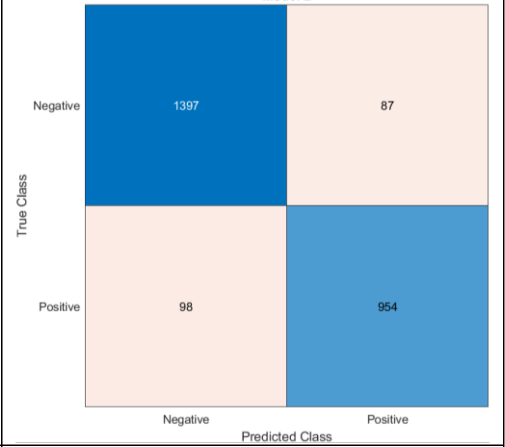
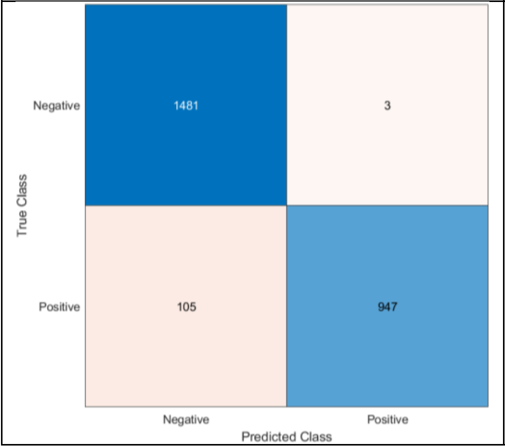
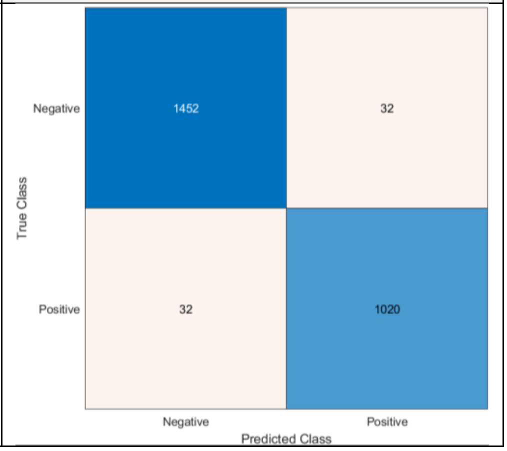
3.1. All features

Table 3 illustrates the confusion matrix obtained using all features. The 3NN, SVM, and DNN classifiers have achieved an excellent performance, they obtained an accuracy of 97.48 percent, 95.74 percent, and 92.7 percent, respectively, whereas the DT has 79.14 percent. The correlation between the sample's expected and actual class is calculated using the MCC; The 3NN and the SVM have an excellent prediction with 0.95 MCC for the 3NN and 0.91 for the SVM.

To evaluate the performances of the model in terms of false negatives and false positives, we calculated the sensitivity, specificity, and precision. With 99.8 percent specificity using SVM, and 96.96 percent for sensitivity using the 3NN, each model can detect the true positive of the respected class, but since in our application the false negatives could be fatal and may cause the spread of the virus, we believe the 3NN has the edge with 96.96 percent versus 90.02 percent achieved by SVM. The false alarms are quantified by the precision. The model with fewer false positives is the SVM with nearly perfect 99.68 percent precision. The F1-score values for each model were calculated to evaluate the overall performances. The 3NN has the best f1-score of 96.96 percent. For this model we have:

- for the negatives, the model has correctly classified 1452 observations from 1484, with 32 false positives,
- as for the class of the positives, 1020 observations have been detected, and 32 were misclassified and considered negatives.

Tab. 3. Confusion matrices using all features

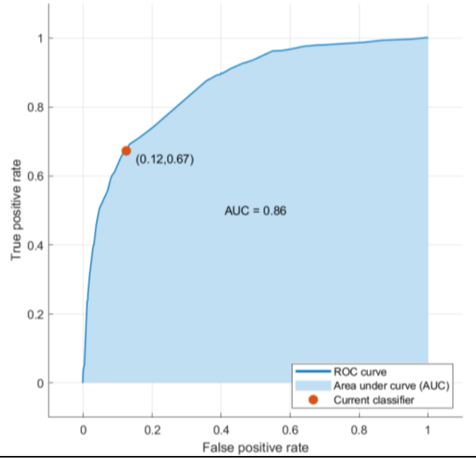
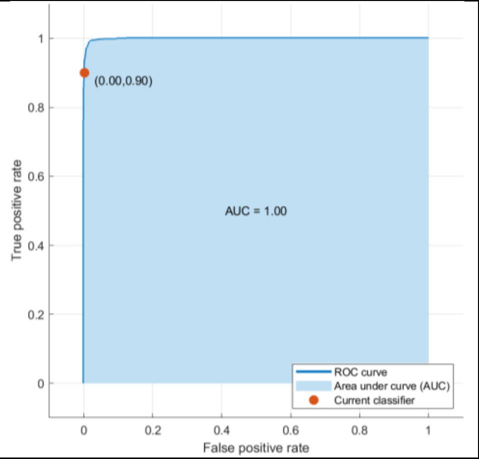
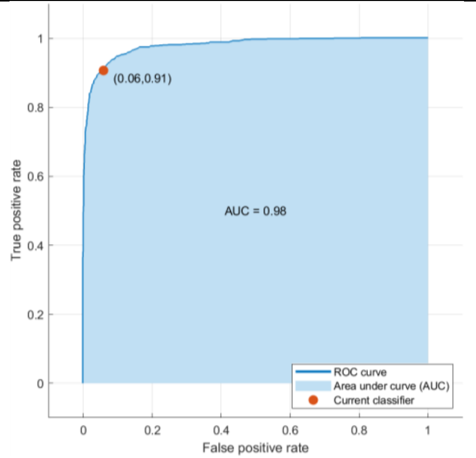
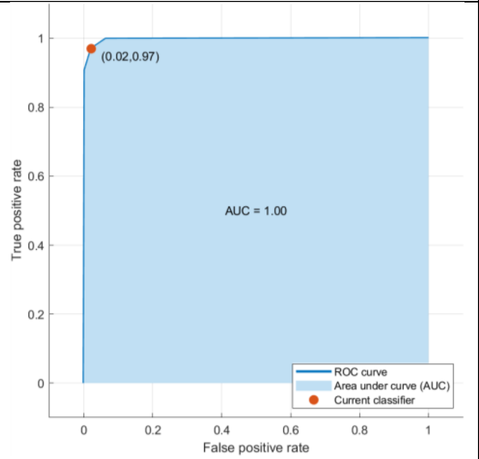
DT	DNN
	
SVM	**3NN**
	

Tab. 4 summarizes the evaluation metrics, and the ROC curves are illustrated in Tab.5 for all the classifiers.

Tab. 4. Evaluation metrics using all features

	Sensitivity (%)	Specificity (%)	Precision (%)	Accuracy (%)	F1 score (%)	MCC
DT	67.3	87.53	79.28	79.14	72.8	0.56
SVM	90.02	**99.8**	**99.68**	95.74	94.6	0.91
DNN	90.68	94.14	91.64	92.7	91.16	0.85
3NN	**96.96**	97.84	96.96	**97.48**	**96.96**	**0.95**

Tab. 5. ROC curve using all features

DT	SVM
	
DNN	**3NN**
	

3.2. PCA with 95 percent of the variance

We applied the PCA to explain the 95 percent variance. After training, 11 components were kept. Fig. 7 shows the variance per component.

When performing dimensionality reduction by the PCA with 95 percent variance, the 3NN and SVM have very close accuracy (96.92 percent using 3NN and 96.6 percent using the SVM), followed by DNN with an accuracy of 91.09 percent and the DT with 78.08 percent. The SVM has a slight improvement in the MCC with 0.94, whereas the 3NN has 0.95.

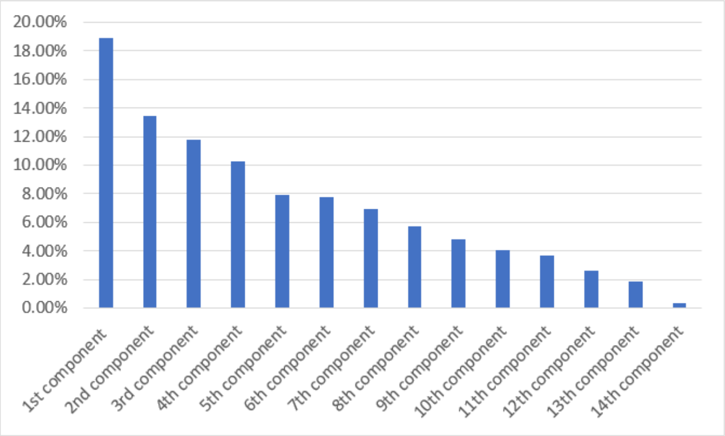
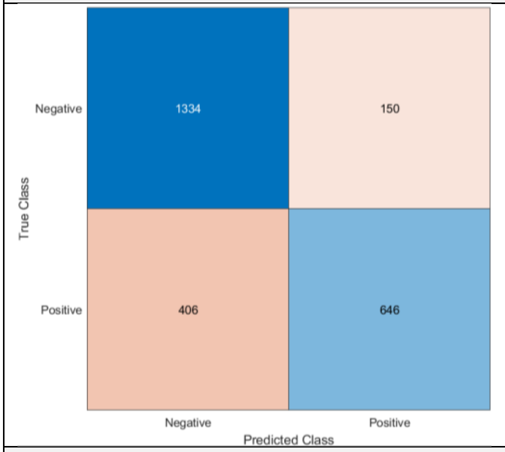
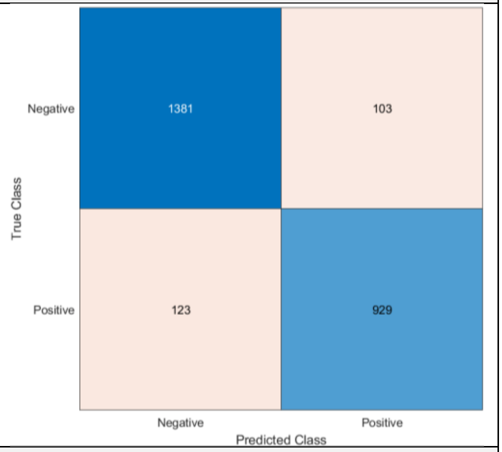
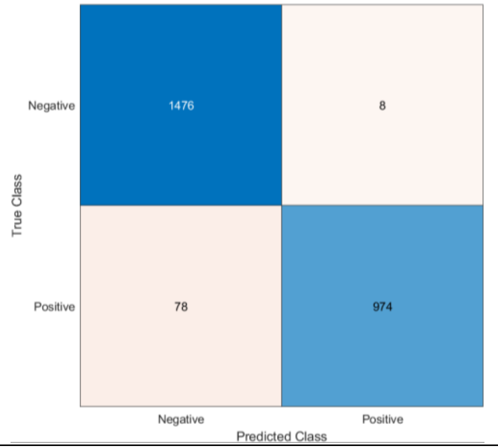
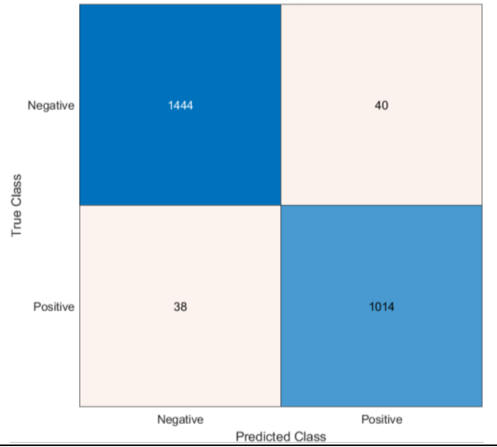


Fig. 7. Variance per component

Tab. 6. Confusion matrices using PCA with 95 percent of the variance

DT	DNN
	
SVM	**3NN**
	

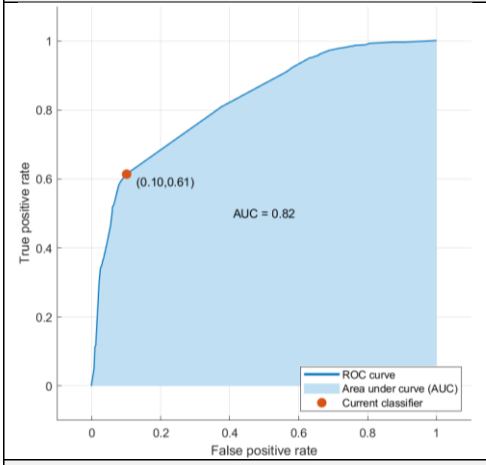
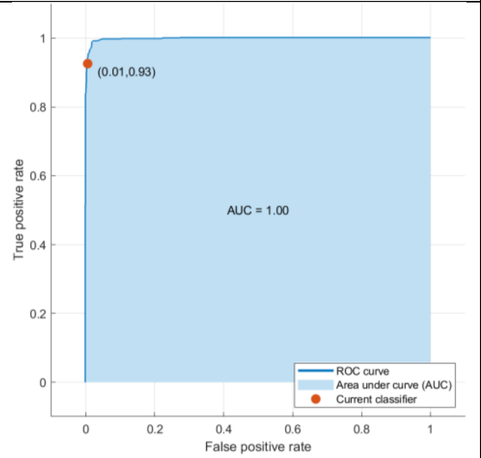
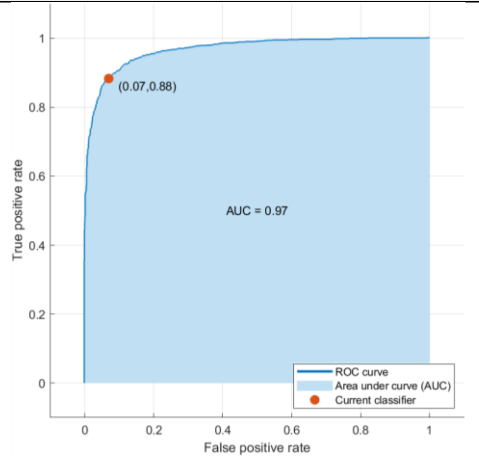
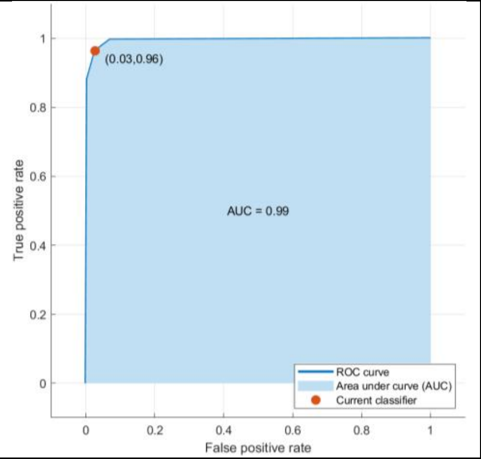
The F1-score values for each technique were calculated to evaluate the overall performances (Tab. 6). The 3NN has the highest f1-score of 97.3 percent. For this model we have:

- for the negatives, the model has correctly classified 1444 observations, with 40 false positives,
- in the class of the positives, 1014 observations have been identified, whereas 38 were misclassified.

Tab. 7. Evaluation metrics using PCA with 95 percent of the variance

	Sensitivity (%)	Specificity (%)	Precision (%)	Accuracy (%)	F1 score (%)	MCC
DT	61.41	89.9	81.16	78.08	69.91	0.54
SVM	92.59	**99.46**	**99.19**	96.6	95.77	0.93
DNN	88.3	93.06	90.02	91.09	89.16	0.82
3NN	**96.39**	97.3	96.2	**96.92**	**96.3**	**0.94**

Tab. 8. ROC curve using PCA with 95 percent of the variance

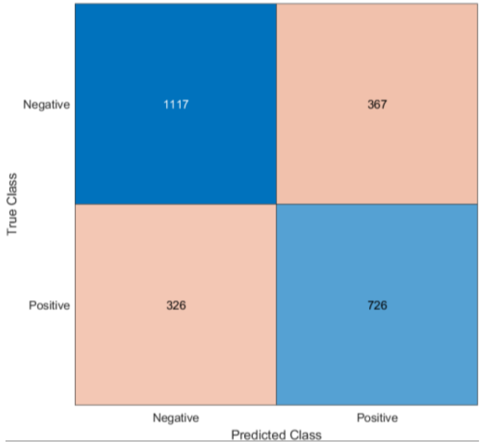
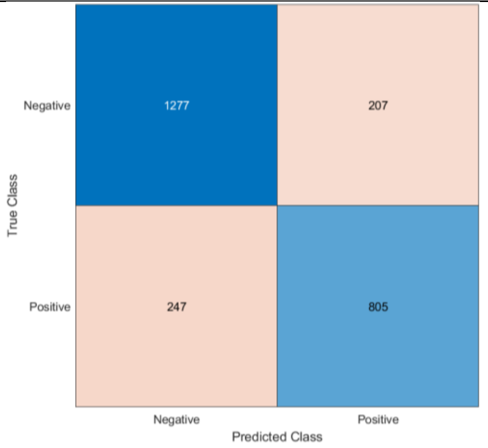
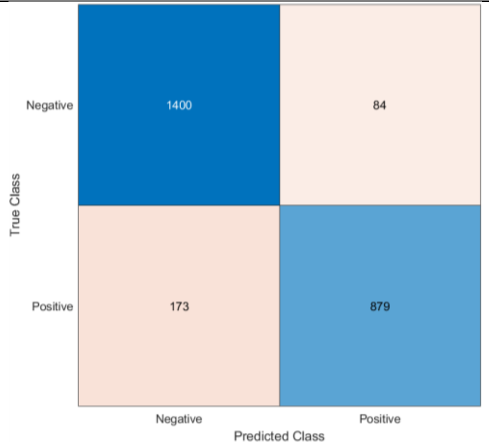
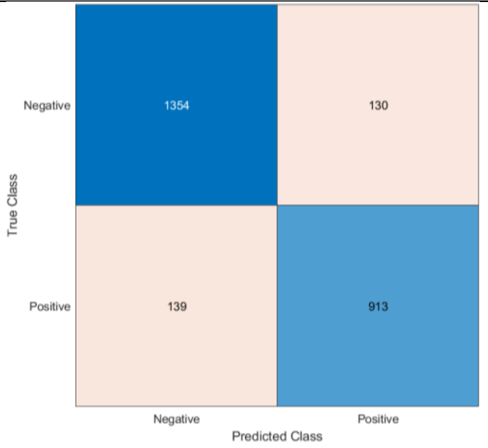
DT	SVM
	
DNN	**3NN**
	

The highest specificity and precision were achieved by the SVM above 99 percent. While the 3NN has a sensitivity and f1-score above 96 percent. The rest of the results are listed in Tab. 7. The ROC curves of each model are presented in Tab.8.

3.3. PCA with 6 components

We have selected the first six principal components calculated, for 70 percent variance. The classification results are given in Tab. 9.

Tab. 9. Confusion matrices using PCA with 6 principal components

DT	DNN
	
SVM	**3NN**
	

Reducing the space features by mapping only 6 principal components, the SVM has a satisfactory accuracy of 89.87 percent, and an F1-score of 87.25 percent, close to the 3NN (89.4 accuracies and 87.16 F1-score). With an MCC of 0.79 for the first and 0.78 for the latter. The SVM has:

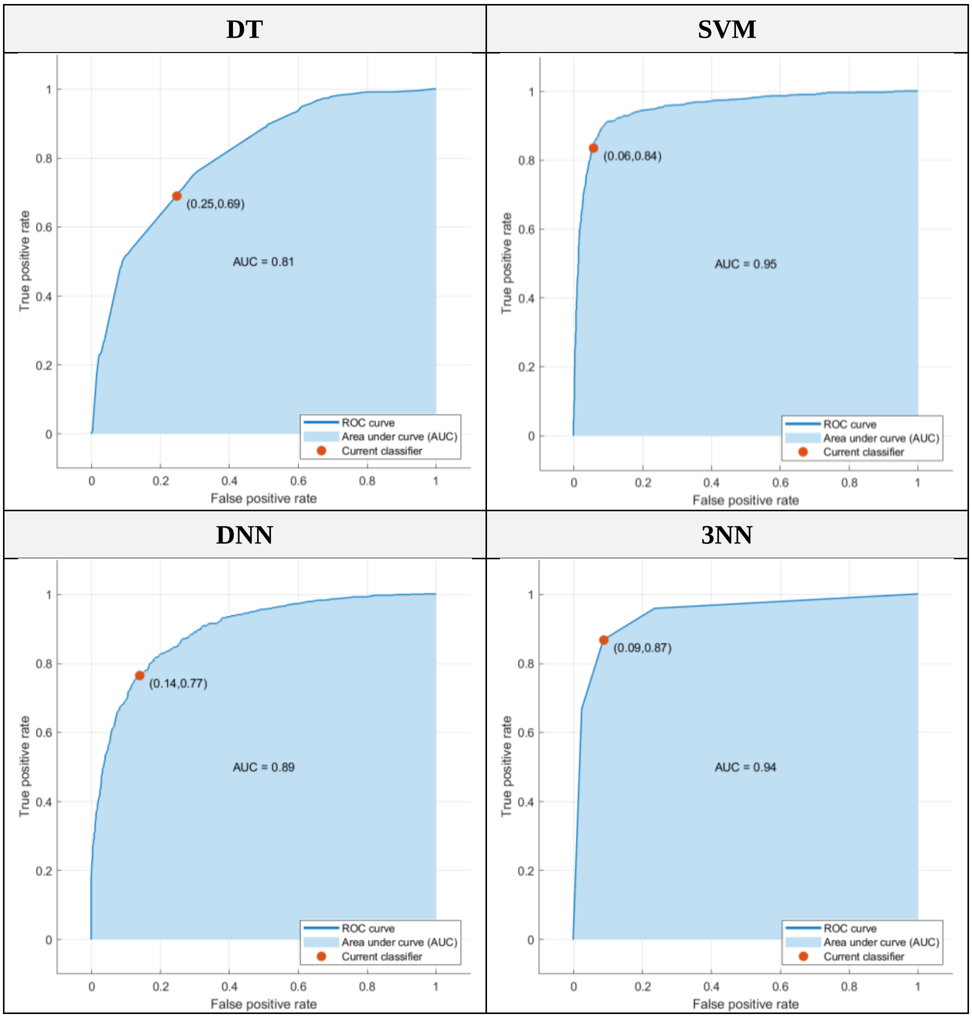
- 1400 observations negatives have been correctly classified, with 84 false positives,
- and 879 positive subjects have been detected, and 173 misclassified.

The highest sensitivity was achieved by the 3NN (86.79 percent). While the SVM has better specificity and precision of 94.34 percent and 91.28 percent, respectively. The evaluation metrics are presented in Tab. 10, and the ROC curves are in Tab. 11.

Tab. 10. Evaluation metrics using PCA with 6 principal components

	Sensitivity (%)	Specificity (%)	Precision (%)	Accuracy (%)	F1 score (%)	MCC
DT	69.01	75.27	66.42	72.67	67.69	0.44
SVM	83.56	**94.34**	**91.28**	**89.87**	**87.25**	**0.79**
DNN	76.52	86.05	79.55	82.1	78	0.63
3NN	**86.79**	91.24	87.54	89.4	87.16	0.78

Tab. 11. ROC curve using PCA with 6 principal components



After testing our models for each frame of the signals, we performed the classification by subjects, using the 3NN with all features. The results show how accurate the model is in predicting the subjects based on the most frequent value predicted by frame for each subject (Fig. 8).

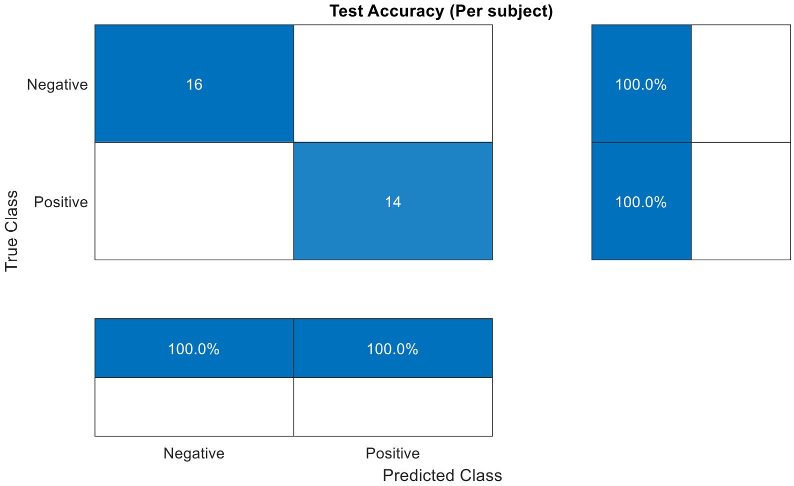


Fig. 8. confusion matrix by subject using 3NN

4. DISCUSSION

The 3NN classifier with all features has produced the best classification results. The model detects COVID-19 patients with an accuracy of 97.48 percent, 96.96 percent f1-score, and 0.95 MCC. This suggests that our approach can accurately distinguish healthy controls from COVID-19 patients. The importance of the AI-based screening method is limiting the spread of the virus, meaning building models with high sensitivity of the positive cases, in our case we were able to achieve 96.96 percent. The specificity of 97.84 percent, and the 96.96 percent precision. Performing PCA with 95 percent variance also gives an accurate prediction for the 3NN and the SVM classifiers. When choosing only 6 components the performance dropped but was still acceptable since the SVM and the 3NN obtained more than 89 percent accuracy, 87 percent f1-score, and 0.78 MCC.

We believe that when compared to previous studies, our findings are extremely encouraging (Tab. 12). Using less computationally expensive classifiers and focusing more on data augmentation through frame analysis, we were able to get excellent results.

Tab. 12. Comparison with other earlier studies for the COVID-19 sound-based diagnosis

Research	Dataset	Sound type	Features	Models /Classifiers	Results (Accuracy)
Coppock et al., 2021	Covid-19-sounds	Cough Breathing	Mel-spectrogram	ResNet	84.6%
Aly, Rahouma & Ramzy, 2022	Coswara Virufy	Coswara Virufy	MFCC – RMS - ZCR Spectral Rolloff/ Centroid/etc.	Deep Model Shallow classifiers	96.4%
Fakhry et al., 2021	Coughvid	Cough	Mel-spectrogram MFCC Clinical features	Multi-Branch Deep Learning Network	91%
Chaudhari et al., 2020	Coswara Coughvid Virufy	Cough	Mel-spectrogram MFCC Clinical features	Ensemble Deep Learning Model	77.1%
Laguarta, Hueto & Subirana, 2020	Opensigma	Cough	MFCC - other biomarkers	ResNet50	97%
Brown et al., 2020	Covid-19-sounds	Cough Breathing	MFCC - Tempo - RMS - ZCR - etc.	VGGish Shallow classifiers	80%
Pahar et al., 2021	Coswara SARCOS	Cough	MFCC - Log Energies - ZCR - Kurtosis.	Resnet50 LSTM	98% 94%
This study	Coswara	Cough	MFCC	3NN	97.48%

These results are encouraging and a controlled clinical trial by medical specialists is required to validate these findings. Furthermore, due to the pandemic's fast and current spread, there is still a scarcity of knowledge about the disease's etiology and progression, as well as the relationship between demographic and clinical data of COVID-19 patients. We focused solely on the effects of COVID-19 infection on voice quality in this study. However, in the future, we hope to investigate the effects of patient data, such as age and gender, the etiopathogenesis of the pandemic, whose symptoms, particularly in the early stages of the disease, are still frequently misinterpreted as other respiratory illnesses, and to identify COVID-19 disorders and enhance the model's accuracy. One of the biggest problems with COVID-19 is the lack of high-quality datasets. because of: (1) closed-source and unpublished datasets; (2) the scattered nature of COVID-19 datasets; and (3) privacy concerns that restrict data sharing. Therefore, cooperation amongst all medical organizations worldwide is necessary to increase the availability and quality of COVID-19 datasets. A link between COVID-19 infection and various medical comorbidities has been observed in the literature. Therefore, the COVID-19 prediction and detection processes must both take into account a patient's history of various illnesses (such as diabetes, liver, renal, heart disease, etc.) in order to create a precise and reliable prediction model. In contrast to working with IoT devices, building complicated ML models, analyzing, and interpreting massive data demand high computational resources. Edge computing and fog computing may therefore be useful in addressing this issue.

5. CONCLUSION

In this paper, we proposed a cough-based detection of the presence of COVID-19 using the main Machine learning classifiers and the PCA. Our objective is to realize a system capable of screening COVID-19 disorder, which could be useful as a pre-screening test as well as for the monitoring of patients' symptoms. The proposed models have been applied to the Coswara database, a crowdsourcing project from the Indian Institute of Science aiming to build a diagnostic tool for COVID-19 using audio recordings. The results have shown that the best accuracy in COVID-19 detection is achieved by mapping all the extracted features directly to the 3NN classifier.

Conflicts of Interest

Funding: Not applicable

Conflicts of interest/Competing interests: On behalf of all authors, the corresponding author states that there is no conflict of interest.

REFERENCES

Adhatrao, K., Gaykar, A., Dhawan, A., Jha, R., & Honrao, V. (2013). *Predicting students' performance using ID3 and C4. 5 classification algorithms*. arXiv preprint arXiv:1310.2071.

Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., Tao, Q., Sun, Z., & Xia, L. (2020). Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology*, *296*(2), E32-E40. https://doi.org/10.1148/radiol.2020200642

Aly, M., Rahouma, K. H., & Ramzy, S. M. (2022). Pay attention to the speech: COVID-19 diagnosis using machine learning and crowdsourced respiratory and speech recordings. *Alexandria Engineering Journal*, *61*(5), 3487–3500. https://doi.org/10.1016/j.aej.2021.08.070

Anuradha, C., & Velmurugan, T. (2014). A data mining based survey on student performance evaluation system. In *2014 IEEE International Conference on Computational Intelligence and Computing Research* (pp. 1–4). IEEE. https://doi.org/10.1109/ICCIC.2014.7238389

Anusuya, M. A., & Katti, S. K. (2010). *Speech recognition by machine, a review*. arXiv preprint arXiv:1001.2267.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, *2*(1), 1–127.

Benmalek, E., Elmhamdi, J., & Jilbab, A. (2021). Comparing CT scan and chest X-ray imaging for COVID-19 diagnosis. *Biomedical Engineering Advances*, 1, 100003. https://doi.org/10.1016/j.bea.2021.100003

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory* (pp. 144–152). The ACM Digital Library.

Brown, C., Chauhan, J., Grammenos, A., Han, J., Hasthanasombat, A., Spathis, D., Xia, T., Cicuta, P., & Mascolo, C. (2020). *Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data*. arXiv preprint arXiv:2006.05919.

Chaudhari, G., Jiang, X., Fakhry, A., Han, A., Xiao, J., Shen, S., & Khanzada, A. (2020). *Virufy: Global applicability of crowdsourced and clinical datasets for AI detection of COVID-19 from cough*. arXiv preprint arXiv:2011.13320.

Coppock, H., Gaskell, A., Tzirakis, P., Baird, A., Jones, L., & Schuller, B. (2021). End-to-end convolutional neural network enables COVID-19 detection from breath and cough audio: a pilot study. *BMJ innovations*, *7*(2), 356–362. https://doi.org/10.1136/bmjinnov-2021-000668

Fakhry, A., Jiang, X., Xiao, J., Chaudhari, G., Han, A., & Khanzada, A. (2021). Virufy*: A multi-branch deep learning network for automated detection of COVID-19*. arXiv preprint arXiv:2103.01806.

Han, J., Brown, C., Chauhan, J., Grammenos, A., Hasthanasombat, A., Spathis, D., Xia, T., Cicuta, P., & Mascolo, C. (2021). Exploring Automatic COVID-19 Diagnosis via voice and symptoms from Crowdsourced Data. In *ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 8328–8332). IEEE.

Han, W., Chan, C. F., Choy, C. S., & Pun, K. P. (2006). An efficient MFCC extraction method in speech recognition. In *2006 IEEE International Symposium on Circuits and Systems* (ISCAS) (pp. 4). IEEE.

Indyk, P., & Motwani, R. (1998). Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing* (pp. 604–613). The ACM Digital Library.

Ismail, M. A., Deshmukh, S., & Singh, R. (2021). Detection of COVID-19 through the analysis of vocal fold oscillations. In *ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1035–1039). IEEE.

Laguarta, J., Hueto, F., & Subirana, B. (2020). COVID-19 Artificial Intelligence Diagnosis using only Cough Recordings. In *IEEE Open Journal of Engineering in Medicine and Biology* (vol. 1, 275–281). IEEE. https://doi.org/10.1109/OJEMB.2020.3026928

Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X., Kong, B., Bai, J., Lu, Y., Fang, Z., Song, Q., Cao, K., Liu, D., Wang, G., Xu, Q., Fang, X., Zhang, S., Xia, J., & Xia, J. (2020). Using Artificial Intelligence to Detect COVID-19 and Community-acquired Pneumonia Based on Pulmonary CT: Evaluation of the Diagnostic Accuracy. *Radiology*, *296*(2), E65–E71. https://doi.org/10.1148/radiol.2020200905

Muda, L., Begam, M., & Elamvazuthi, I. (2010). *Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques*. arXiv preprint arXiv:1003.4083.

Nassif, A. B., Shahin, I., Attili, I., Azzeh, M., & Shaalan, K. (2019). Speech recognition using deep neural networks: A systematic review. *IEEE access*, *7*, 19143–19165. https://doi.org/10.1109/ACCESS.2019.2896880

Pahar, M., Klopper, M., Warren, R., & Niesler, T. (2021). COVID-19 cough classification using machine learning and global smartphone recordings. *Computers in Biology and Medicine*, *135*, 104572. https://doi.org/10.1016/j.compbiomed.2021.104572

Pal, A., & Sankarasubbu, M. (2021). Pay attention to the cough: Early diagnosis of COVID-19 using interpretable symptoms embeddings with cough sound signal processing. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing* (pp. 620–628). The ACM Digital Library. https://doi.org/10.1145/3412841.3441943

Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine.In *Machine learning* (pp. 101–121). Academic Press.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81–106.

Sharma, N., Krishnan, P., Kumar, R., Ramoji, S., Chetupalli, S. R., Ghosh, P. K., & Ganapathy, S. (2020). *Coswara--a database of breathing, cough, and voice sounds for COVID-19 diagnosis*. arXiv preprint arXiv:2005.10548.

Singh, H., & Bathla, A. K. (2013). A survey on speech recognition. *International Journal of Advanced Research in Computer Engineering & Technology*, *2*(6), 2186–2189.

Weng, L. M., Su, X., & Wang, X. Q. (2021). Pain symptoms in patients with coronavirus disease (COVID-19): a literature review. *Journal of Pain Research*, *14*, 147. https://doi.org/10.2147/JPR.S269206

Wu, X., Hui, H., Niu, M., Li, L., Wang, L., He, B., Yang, X., Li, L. Li, H., Tian, J., & Zha, Y. (2020). Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study. *European Journal of Radiology*, *128*, 109041. https://doi.org/10.1016/j.ejrad.2020.109041

Yang, Y., Yang, M., Shen, C., Wang, F., Yuan, J., Li, J., Zhang, M., Wang, Z., Xing, L. Wei, J., Peng, L., Wong, G., Zheng, H., Wu, W., Liao, M., Feng, K., Li, J., Yang, Q., Zhao, J., Zhang, Z., Liu, L., & Liu, Y. (2020). *Evaluating the accuracy of different respiratory specimens in the laboratory diagnosis and monitoring the viral shedding of 2019-nCoV infections*. MedRxiv. https://doi.org/10.1101/2020.02.11.20021493

Zheng, F., Zhang, G., & Song, Z. (2001). Comparison of different implementations of MFCC. *Journal of Computer science and Technology*, *16*(6), 582–589. https://doi.org/10.1007/BF02943243