Modelos de aprendizaje automático para caracterizar la señal de la tos de pacientes con COVID-19

Christian Salamea-Palacios, Ph.D.[1], Tarquino Sánchez-Almeida, Ph.D.[2], Xavier Calderón-Hinojosa, M.Sc.[2], Javier Guaña-Moya, Ph.D.[3], Paulo Castañeda-Romero, M.Sc.[2] y Jessica Reina-Trávez, Ing.[2]
[1]Universidad Politécnica Salesiana, Ecuador, csalamea@ups.edu.ec.
[2]Escuela Politécnica Nacional, Ecuador, tarquino.sanchez@epn.edu.ec, xavier.calderon@epn.edu.ec, paulo.castaneda@epn.edu.ec, jesslore_1@hotmail.com.
[3]Pontificia Universidad Católica del Ecuador, Ecuador, eguana953@puce.edu.ec.

Resumen– ***El reconocimiento automático de las señales de audio es una tarea desafiante debido a la dificultad de extraer los atributos importantes de dichas señales que dependen en gran medida, de las características acústicas discriminatorias para determinar el tipo de señal de audio de tos provenientes de pacientes con COVID-19. En este trabajo, se investiga la utilidad de modelos preentrenados de última generación y una red neuronal convolucional para la tarea de extracción de características de la señal de la tos COVID-19.***

Se ha propuesto una comparación de tres modelos de aprendizaje automático para extraer las características que contienen información relevante, que conducen al reconocimiento de la señal de tos COVID-19. El primer modelo está basado en una red neuronal convolucional básica, el segundo, en un modelo pre-entrenado YAMNet y el tercero se trata de un modelo preentrenado VGGish. Los resultados experimentales realizados con una base de datos ComPare 2021 CCS, demuestran que, de los tres modelos utilizados, VGGish proporciona un mejor rendimiento al momento de extraer las características de las señales de audio de la señal de tos COVID-19, teniendo como resultados las métricas de rendimiento f1 score y exactitud con valores de 30.76% y 80.51%, representando una mejoría de 6.06% y 3.61% frente a la modelo YANMet, y las matrices de confusión, que validan el modelo mencionado.

Palabras Clave-- Red neuronal convolucional, YAMNet, Vggish, Transfer Learning, caracterización, COVID-19.

Abstract– ***Automatic recognition of audio signals is a challenging signal task due to the difficulty of extracting important attributes from such signals, which relies heavily on discriminating acoustic features to determine the type of cough audio coming from COVID-19 patients. In this work, the use of state-of-the-art pre-trained models and a convolutional neural network for the extraction of characteristics of a cough signal from patients with COVID-19 is analyzed.***

A comparison of three machine learning models has been proposed to extract the features containing relevant information, leading to the recognition of the COVID-19 cough signal. The first model is based on a basic convolutional neural network, the second is based on a YAMNet pre-treatment model, and the third is a VGGish pre-trained model. The experimental results carried out with a ComPare 2021 CCS database show that models, of the three, used, VGGish to provide better performance when extracting the characteristics of the audio signals of the COVID-19 cough signal, having as results the performance metrics f1 score and accuracy with values of 30.76% and 80.51%, representing an improvement of 6.06% and 3.61% compared to the YANMet model, and the confusion matrices, which validate the mentioned model.

Keywords-- Convolutional Neural Network, YAMNet, Vggish, Transfer Learning, characterization, COVID-19.

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

I. INTRODUCCIÓN

El COVID-19 es una enfermedad contagiosa producida por el virus del SARS-CoV-2 y considerada como pandemia mundial por la Organización Mundial de la Salud (OMS) a inicios de marzo del 2020 y que sigue siendo una amenaza crítica y urgente de atender a nivel mundial. El brote masivo del COVID-19 ha afectado globalmente, con más de 313 millones de pacientes infectados y 5.3 millones de muertes alrededor del mundo. Esta pandemia sigue siendo un desafío para los sistemas de salud de todo el mundo en varios aspectos como el rebrote de contagiados por las nuevas variantes y el considerable aumento de la demanda de camas en las unidades de cuidados intensivos (UCI) [1].

La herramienta fundamental para contrarrestar el avance de la pandemia es el diagnóstico temprano en los pacientes que han sido contagiados con COVID-19 y de esta manera controlar el avance de la pandemia y evitar que se propaguen más contagios. Además, la detección eficaz permite un diagnóstico rápido y eficiente del virus y puede reducir la carga sobre los hospitales. Así mismo, los sistemas de predicción inteligente y modelos de machine learning han aportado una valiosa contribución al dotar de una herramienta que permita obtener un diagnóstico temprano de COVID-19. [2]

El diagnóstico de COVID-19 en la etapa inicial también posibilita el aislamiento apropiado del paciente, la atención pronta de los pacientes crónicos en los hospitales y el monitoreo de la evolución de propagación del virus. Sin embargo, el diagnóstico temprano de COVID-19 es una tarea desafiante, debido no solamente al costo que significa realizarse las pruebas clínicas de RT-PCR (Reacción en Cadena de la Polimerasa) y de antígenos, sino también, por la rápida propagación de las nuevas mutaciones del virus SARS-CoV-2 que se han manifestado como variantes Delta y Ómicron. [2]

La escasez de los métodos de diagnóstico temprano, la falta de interés de la población por inocularse y las nuevas mutaciones del COVID-19, son razones que han provocado la rápida propagación del mismo. Sin embargo, los métodos de detección del virus por lo general tienen un costo considerablemente alto y debido a la situación económica de países en vías de desarrollo, muchas personas no pueden acceder a los mismos. Por todo ello, se ha visto la necesidad de emplear modelos de aprendizaje automático como una solución oportuna en el diagnóstico de detección temprana de COVID-19 [3].

De esta manera, las técnicas de aprendizaje automático se han desarrollado para la creación de modelos de programación que juegan un rol importante en el área de la salud, implementando sistemas de apoyo a la decisión clínica, no solamente para la clasificación de enfermedades médicas como el cáncer de mama, la tuberculosis, la neumonía y las enfermedades musculares, sino también, es útil para la detección, diagnóstico, y clasificación de la enfermedad de COVID-19 [4] [5].

Los modelos de aprendizaje automático son tecnologías emergentes en el campo de medicina y debido a su capacidad para generar resultados predictivos, aportan un papel esencial para el avance tecnológico en el sistema de salud. Sin embargo, debido a la complejidad que significa trabajar con señales de audio es difícil implementar un modelo que provea un alto rendimiento para detectar COVID-19 en la señal de la tos, por lo que, es necesario considerar modelos de machine learning capaces de extraer las características de los audios de tos para después predecir si se tratan de tos COVID-19 positivo o negativo, en conjunto con un análisis del especialista medico [3].

Es de gran importancia presentar un sistema de predicción que sea capaz de detectar y diagnosticar con una alta *precisión* los contagios de COVID-19. En este estudio, el enfoque principal es realizar una comparación entre varios modelos de aprendizaje automático para la extracción de características importantes de la señal de tos y así predecir si corresponde a personas con COVID-19 positivo o no. Los modelos han sido utilizados con la base de datos ComPare2021, la cual dispone de datos de entrenamiento y prueba [6].

II. Trabajos Relacionados

La gran cantidad de estudios para el reconocimiento de señales de audio se han centrado en extraer características para clasificarlos de una manera precisa [7] [8]. Hoy en día, cada vez más investigaciones se han centrado en los métodos de aprendizaje automático debido a su importancia para la extracción de características de las señales de audio [5]. Lakomkin et al., presenta varios modelos que utilizan representaciones neuronales inferidas por entrenamiento empleando grandes bases de datos de voz, para la extracción de características en torno al reconocimiento de emociones [9].

Gaowei et al., demostró que la combinación de *CNN* (Convolutional Neural Network) con clasificadores de Random Forest funciona mejor que las tradicionales *CNN* de extremo a extremo, para la extracción de características de una señal de audio. Las múltiples capas convolucionales permitieron extraer las características de la señal y luego estas alimentaron a tres clasificadores Random Forest independientes. El autor propone utilizar clasificadores multinivel y encontró que la combinación de una red CNN multi capa con los clasificadores Random Forest funciona mejor que la *CNN* tradicional considerando sólo las características más importantes. Los resultados sugieren que las características generadas por las capas multinivel proporcionan una mejor generalización del modelo y que las funciones de *CNN* con Random Forest funcionan mejor que el modelo *CNN* de extremo a extremo [10].

Niu y Suen reconocen dígitos escritos a mano usando un nuevo método. Con este enfoque, se entrena un modelo tradicional de *CNN* y luego la salida de la capa oculta se extrajo del modelo *CNN* preentrenado y se utiliza para entrenar un clasificador SVM (Support Vector Machine). Los autores utilizan *CNN* como extractor de características y SVM como un clasificador. Los resultados muestran que la tasa de error del modelo híbrido es más baja que el propio modelo de *CNN* [11].

Basly et al., combina el método basado en el aprendizaje profundo y un extractor de características artesanal basado en clasificadores tradicionales para reemplazar el método de extracción de características artesanales por uno nuevo. En este enfoque, se extraen las características aprendidas con la red *CNN* de un modelo preentrenado basado en ResNet y las características luego se usan para entrenar un modelo SVM en el reconocimiento de la actividad humana. El modelo *CNN* se utiliza como extractor de características y el modelo SVM se utiliza como predictor o clasificador [12].

Liu et al., realiza una combinación de *CNN* y SVM en reconocimiento de Género basado en la marcha. El modelo VGGNet-16 se utiliza a través del aprendizaje de transferencia para la tarea de reconocimiento de género.

Los autores emplean diferentes métodos para sintonizar el modelo VGGNet-16 y las características son extraídas de tres capas diferentes completamente conectadas. La capa softmax se reemplaza por un clasificador SVM y los resultados muestran que el modelo CNN-SVM funciona mejor que el modelo tradicional de la *CNN* [13].

Cao et al., utiliza un enfoque híbrido de combinar una *CNN* con un algoritmo de Random Forest para segmentar imágenes de microscopía electrónica. Con este enfoque, un modelo *CNN* que consta de varias capas convolucionales, capas de agrupación, capas totalmente conectadas y una capa softmax se entrena con imágenes de entrada. Luego se usa el modelo *CNN* entrenado para extraer características de las imágenes. La salida de la última capa convolucional del modelo *CNN* se extrae y se introduce en un clasificador Random Forest. Los resultados muestran que el método

híbrido es el mejor comparado con un modelo tradicional de *CNN* en la segmentación de electrones imágenes de microscopía [14].

III. Metodología

A. Conjunto de datos (Dataset)

En este estudio, se ha empleado el conjunto de datos proporcionado por la Universidad de Cambridge, obtenidos en el programa ComParE 2021 COVID-19, que se denomina "Cough Sub-Challenge—CCS", el mismo que ha servido para entrenar el modelo para la extracción de características de la señal de tos COVID-19 [6].

La base de datos obtenida a través de la aplicación de sonidos COVID-19 grabadas a una tasa de muestreo de 16kHz, recopilada desde abril de 2020, tuvo el objetivo de obtener datos para entrenar modelos de aprendizaje automático, que pueden servir para detectar el diagnóstico preliminar de COVID-19 basado principalmente en la voz, la respiración y la tos.

En la Tabla 1, se muestra la información del conjunto de datos en CCS, el cual contiene grabaciones de tos, obtenidas de 397 participantes; además, dispone de etiquetas que muestran el estado que define, si el audio de tos es COVID-19 positivo o negativo. Estas grabaciones están distribuidas en dos categorías, 286 de entrenamiento y 231 de desarrollo.

TABLA I
CONJUNTO DE DATOS CCS [4]

Datos	Positivos	Negativos	Total
Entrenamiento	71	215	286
Desarrollo	48	183	231

B. Red Neuronal Convolucional (CNN)

Las *CNN* inicialmente se enfocaron en tareas de clasificación, detección de objetos y reconocimiento de imágenes. Las imágenes se emplean como entrada y se utiliza la base de datos ImageNet (http://www.image-net.org/, accesible en 30 de junio de 2021), que cuenta con 14 millones de imágenes, la cual se emplea para entrenar y evaluar algunas de las redes convolucionales más utilizadas para la extracción de características. Las *CNN* están evolucionando en términos de tamaño (número de capas) y estructura (conexión y tipo de capas) [15].

La red neuronal VGG fue la evolución de AlexNet, la cual emplea la función de activación ReLU, pero los campos receptivos fueron reemplazados por grupos más pequeños (3×3 en lugar de 11×11 y 5×5 en AlexNet) con un paso fijo de 1 creando bloques convolucionales, lo que conduce a un mejor rendimiento. VGG-16 y VGG-19 son típicamente usados, ambos con tres capas completamente conectadas y 16 capas convolucionales. Ellos tienen 138 millones y 144 millones de parámetros, respectivamente, y su entrada consiste en imágenes de 224×224×3 [16].

Para limitar la necesidad de recursos, Google introdujo MobileNet para dispositivos móviles y aplicaciones integradas. La red disminuye el número de parámetros (volumen en el disco) y la complejidad de las operaciones (latencia y potencia). MobileNet utiliza una reducción de 7 veces en el número de parámetros y solo un 1 % menos de *precisión* en comparación con los modelos de convolución completa. MobileNet-v1 toma un tamaño de imagen de entrada de 224×224×3, tiene 28 capas de profundidad y tiene 4.2 millones de parámetros [16].

En 2020, el desempeño de las *CNN* en la clasificación de imágenes inspiró la idea de crear arquitecturas de redes neuronales profundas nuevas o adaptando las existentes para clasificar las señales de audio, caracterizadas como *CNN* que trabajan con señales de audio (en contraste con la arquitectura *CNN* centrada en la imagen). *VGGish* y *YAMNet* son dos *CNN* para la extracción de características, siendo la primera una red de 24 capas de profundidad, basada en VGG, y la segunda (*YAMNet*) es una red de 28 capas de profundidad que emplea la arquitectura MobileNet-v1.

El entrenamiento y la inferencia de las *CNN* de sonido están empleando conjuntos de datos de sonido, después de la transformación de la señal sonora en una imagen o conjunto de imágenes. El rendimiento de la clasificación está relacionado con la cantidad de datos disponibles y el desequilibrio de las clases dentro de dicho conjunto de datos. Este problema se aborda con algunos ajustes o utilizando el método de transferencia de aprendizaje [6].

C. Transfer Learning

Transferir el aprendizaje aplica el conocimiento obtenido de un origen (fuente) a un destino (objetivo) para aprovechar el modelo entrenado en el origen y con esto poder compensar los datos limitados o inadecuados en el dominio de destino (objetivo) [15].

Las técnicas de aprendizaje por transferencia se han utilizado en clasificación, regresión y en problemas de agrupación. Para emplear transferencia de aprendizaje en la clasificación de sonidos, se utilizan representaciones de sonido basadas en imágenes entre las redes entrenadas y los datos en bruto (sonidos). Específicamente, los extractos de sonido (después de la segmentación adecuada) se transforman en imágenes, como espectrogramas como un paso de preprocesamiento [15].

Las *CNN* normalmente constan de la parte convolucional y de agrupación, responsable de extraer las características, y las capas de clasificación conectadas con la primera parte. Los modelos entrenados son reutilizados de acuerdo con el conjunto de datos de destino.

Para emplear una red de transferencia se pueden generar tres opciones de configuración, la primera es entrenar la red general nuevamente, aprovechando la arquitectura y la

topología, para adaptar la existentes o calcular nuevos pesos para ambas partes del convolucional y el clasificador, la segunda opción es entrenar solamente una parte de la convolución, y así mismo la parte del clasificador. Esto puede incluir cambios potenciales de la arquitectura, con la inclusión, remoción o reformulación de capas. Finalmente, la tercera opción que consta en reentrenar el clasificador de la *CNN*, sin adaptaciones en el convolucional y parte de agrupación.

Volver a entrenar partes de la red permite una mayor flexibilidad y permite obtener una mejor *precisión* de clasificación, mientras que la tercera opción mantiene la mayoría de las ponderaciones y reserva los recursos computacionales. La tercera opción cuenta con la ventaja de que estandariza la red empleada y la desventaja de que solo permite entradas similares a las utilizadas por la red entrenada inicialmente [17].

1) YAMNet (Yet another Audio Mobilenet Network): Es un modelo preentrenado que predice 521 eventos de audio basados en el corpus AudioSet. Este modelo está disponible en TensorFlow Hub, incluidas las versiones TFLite y TF.js, para ejecutar el modelo en dispositivos móviles y en la web. El código se puede encontrar en su repositorio [18]. En la Fig. 1, se presenta la arquitectura de una red *YAMNet*.

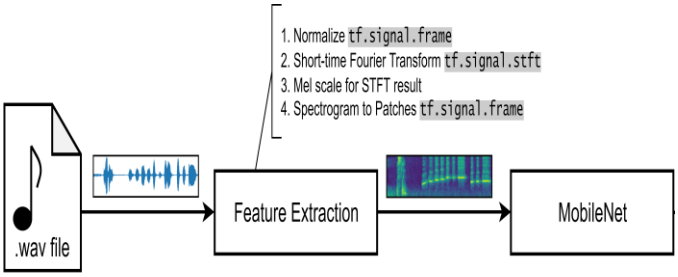

Fig. 1 Arquitectura de YAMNet [19].

El modelo generalmente dispone de 3 salidas, la primera relacionada con las puntuaciones de clase que usaría para la inferencia o predicción, los embebidos, que son la parte importante para la transferencia de aprendizaje y el espectrograma log de mel para proporcionar una visualización de la señal de entrada. El modelo toma una forma de onda representada como muestras de 16 kHz en el rango [-1.0, 1.0], la enmarca en ventanas de 0.96 segundos y saltos de 0.48 segundos, y luego ejecuta el núcleo del modelo para extraer las características de la señal de sonido [18].

2) VGGish: Es una variante del modelo VGG y se puede utilizar para extraer características de alto nivel de abstracción a partir de grabaciones de audio. En otros trabajos, se presenta un modelo de clasificación que puede lograr un mejor rendimiento con las características obtenidas a partir del modelo preentrenado VGGish [16].

En este trabajo, se ha usado VGGish para extraer las características de la señal de tos COVID-19 a partir de grabaciones de audio. La señal de audio es muestreada a 16 kHz. Luego, se extrae un tensor de espectrograma log mel (96×64) de cada segmento de un segundo. VGGish toma los tensores de espectrograma log mel como entrada. Por un segundo de audio, produce un vector de incrustación, este modelo ha sido preentrenado con YouTube-8M y publicado por Google [20]. En la Fig. 2, se observa la arquitectura del modelo VGGish.

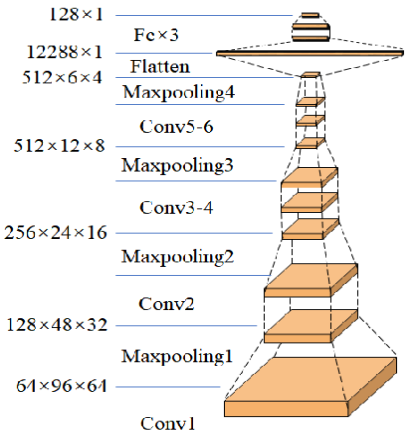

Fig. 2 Arquitectura de VGGish [21]

IV. Resultados

A. *Red Neuronal Convolucional (CNN)*

La base de datos inicial a usar se presenta en la Tabla II.

TABLA II
BASE DE DATOS UTILIZADA: ENTRENAMIENTO Y DESARROLLO

	Total	Positivos	Negativos	Duración promedio
Entrenamiento	286	71	215	6.47
Desarrollo	231	48	183	6.10

Con esta base de datos inicial se procede a entrenar un modelo inicial básico que está conformado de una red neuronal convolucional simple la cual toma como ingreso imágenes de los espectrogramas de cada uno de los audios de la base de datos. Los datos para la obtención de la imagen y el espectrograma se presentan a continuación:

Espectrograma: N_fft=512, Hop_Lenght = 128
Tamaño de Imagen: 64x64 pixeles

En la Tabla III, se puede observar en las métricas de *precisión* y *recall* que el modelo tiene un buen rendimiento en reconocer audios de personas negativas para COVID-19, teniendo un *recall* muy alto, pero tiene muchos fallos en la detección de positivos, lo cual se presenta en la *precisión* del

modelo. El *f1-score* es la métrica que muestra el rendimiento del modelo con la combinación de la *precisión* y el *recall*.

TABLA III
MÉTRICAS RESULTADOS MODELO CNN

Entrenamiento	Desarrollo			
Exactitud	Exactitud	Precisión	Recall	f1 score
77.27%	83.12%	18.75%	100%	31.57%

B. *Transfer learning YAMNet*

Entrenar una red neuronal desde cero requiere una gran base de datos para ayudar al aprendizaje de la red y evitar un sobre entrenamiento en los datos lo cual produce una mala generalización de la red neuronal. Una alternativa tomada por varias investigaciones en estos casos, es el uso de una red neuronal que ya haya sido entrenada para otro tipo de datos, y usar el conocimiento generado en la resolución de oras tareas, siendo este método llamado *Transfer Learning*.

Las pruebas iniciales realizadas utilizan una red neuronal llamada *YAMNet* que es una red neuronal entrenada para clasificar 521 clases de sonidos, en los que hay sonidos de animales, vehículos y también sonidos provocados por la tos. Debido a que esta red ya tiene conocimiento de que es una tos, el objetivo principal es usarla para entrenarla de nuevo y analizar su rendimiento en esta tarea. Esta red neuronal convolucional utiliza ventanas pequeñas de 0.96 s las cuales se desplazan por la imagen obteniendo un vector de características por cada una de ellas, las cuales son usadas posteriormente para realizar una clasificación del sonido, esto se presenta en la Fig. 3.

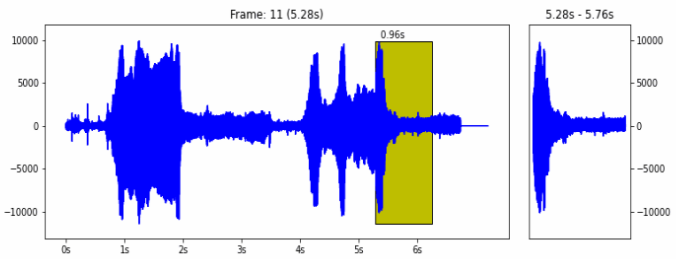

Fig. 3 Trama de la señal de voz, caracterizada con YAMNet.

TABLA IV
MÉTRICAS RESULTADOS DEL MODELO YAMNET

Entrenamiento	Desarrollo			
Exactitud	Exactitud	Precisión	Recall	f1 score
75.17%	74.45%	33.31%	22.91%	27.15

Se observa que el desequilibrio de la base de datos, no favorece el aprendizaje del modelo. En la Tabla IV también se aprecia, que a pesar de que la red tiene un buen resultado de *exactitud* en las predicciones, este dato no representa totalmente los resultados, ya que, la base de datos al tener más datos negativos que positivos, provoca que la red tienda a predecir todos los negativos pero los positivos no los detecta, lo cual produce un buen resultado en *exactitud*, pero respuestas deficientes de *precisión* y *recall* que son métricas que expresan los resultados tomando en cuenta los falsos positivos y los negativos de las predicciones. La matriz de confusión se presenta en la Tabla V.

TABLA V
MATRIZ DE CONFUSIÓN: YAMNET

	Predicción	
Real	0	1
0	161	22
1	37	11

En esta matriz, se observa que el modelo es bueno prediciendo casos negativos y no tan bueno para predecir los positivos, lo cual, bien puede ser por el desequilibrio de la base de datos. Para resolver este desequilibrio se han analizado diferentes técnicas, una de ellas es la creación de nuevos datos. La que mejor ha resultado ha sido el aumento de la amplitud del sonido de las muestras positivas hasta 20 dB, con lo cual, se incrementa la cantidad de muestras con el fin de equilibrar la base de datos. Los resultados obtenidos luego de la aplicación de esta técnica se presentan en la Tabla VI y la matriz de confusión en la Tabla VII.

TABLA VI
MÉTRICAS MODELO YAMNET CON INCREMENTO DE AMPLITUD 20 DB

Entrenamiento	Desarrollo			
Exactitud	Exactitud	Precisión	Recall	f1 score
59.94%	46.75%	20.93%	56.25%	30.50%

TABLA VII
MATRIZ DE CONFUSIÓN: MODELO YAMNET CON INCREMENTO DE AMPLITUD 20 DB

	Predicción	
Real	0	1
0	81	102
1	21	27

C. *Transfer learning VGGish*

Luego de realizar pruebas usando la red de *YAMNet*, se pudo observar que la red no proporcionaba una gran flexibilidad en su código para realizar las pruebas respectivas. Por lo cual, se utilizó otra red neuronal llamada *VGGish*, cuyo modelo ha sido entrenado con una base de datos llamada AUDIOSET, la cual posee una clase de tos, por lo que es adecuada para realizar la técnica de *Transfer Learning*. Para estas pruebas se han utilizado los parámetros colocados en la Tabla VIII.

TABLA VIII
MÉTRICAS RESULTADO MODELO VGGISH

Entrenamiento	Desarrollo			
Exactitud	Exactitud	Precisión	Recall	f1 score
83.21%	80.51%	58.82%	20.83%	30.76%

La métrica de *exactitud* aumenta conforme se incrementa el número de épocas; sin embargo, esto crea un sobreajuste que no permitirá que el modelo generalice correctamente cuando se utilicen nuevos datos de evaluación. Para evitar este fenómeno, se decidió configurar el modelo con un número de épocas igual a 12, consiguiendo una *exactitud* del modelo con los datos de entrenamiento por encima del 80% evitando errores por sobre ajuste y falta de generalización. La matriz de confusión de la red *VGGish* se presenta en la Tabla IX.

TABLA IX
MATRIZ DE CONFUSIÓN: MODELO VGGISH

	Predicción	
Real	0	1
0	163	20
1	41	8

El rendimiento del sistema se evaluó en función de las métricas de rendimiento, especialmente en *f1 score*.

La métrica *f1 score* es la media armónica que combina las medidas de *precisión* y *recall* que permite comparar el rendimiento combinado de dichas medidas para analizar el rendimiento de los modelos de aprendizaje automático. En (1) se presenta la ecuación para calcular *f1 score*.

$$f1\ score = 2\frac{precision * recuperación}{precision + recuperación} \qquad (1)$$

En la Tabla X, se presentan los resultados de las métricas de rendimiento de los tres modelos implementados para la extracción de características de la señal de audio de la tos COVID-19.

TABLA X
COMPARACIÓN DE MÉTRICAS DE RENDIMIENTO DE LOS MODELOS

	Entrenamiento	Desarrollo			
	Exactitud	Exactitud	Precisión	Recall	f1 score
CNN	77.27%	83.12%	18.75%	100%	31.57%
YAMNet	75.17%	74.45%	33.31%	22.91%	27.15
VGGish	83.21%	80.51%	58.82%	20.83%	30.76%

La comparación de las métricas de rendimiento muestra que los modelos *CNN* y *VGGish* presentan resultados con una *exactitud* mayor que el modelo de *YAMNet*, tanto en el proceso de entrenamiento como con los datos de prueba. Sin embargo, el modelo de *CNN* presenta un sobre ajuste esto quiere decir que se dificultará la tarea de predicción cuando se utilicen otros datos, esto debido al desbalance de los datos utilizados en el entrenamiento, con una mayor cantidad de casos de pacientes negativos.

Por otro lado, la métrica de rendimiento que mejor representa el desempeño del sistema *f1 score*, es mayor cuando se emplea el modelo de *VGGish*. Así mismo, la matriz de confusión presentada en la Tabla IX, presenta una mayor cantidad de detección correcta de los casos que son COVID-19 negativo, proporcionando una herramienta eficaz para la detección de casos negativos.

V. CONCLUSIONES

Al utilizar el modelo *CNN* se genera un sobre ajuste debido a la poca cantidad de audios y el desequilibrio de la base de datos. Además, las métricas de *precisión* y *recall*, a pesar de que muestran buen rendimiento en reconocer audios de personas negativas en COVID-19 considerando un *recall* del 100%, se tiene un problema de falta de generalización de los datos, lo cual no permitirá obtener buenos resultados al momento de utilizar nuevos datos como entrada ya que el modelo se encuentra sobre ajustado. Por lo tanto, el modelo *CNN* como extractor de características brinda un bajo desempeño dado que no generaliza adecuadamente y presenta sobreajuste debido a la falta de datos y al desbalance de los mismos.

Una base de datos desbalanceada es capaz de afectar el aprendizaje del modelo, en este caso el número de negativos es mayor a los casos positivos por lo que afectará al sistema al momento de realizar las extracciones de características de la señal de tos, provocando una mayor detección en los casos de COVID-19 negativo y por otro lado un desempeño menor cuando se considere detectar los casos de COVID-19 positivo.

Para el entrenamiento de una red neuronal como extractor de características, desde el inicio es necesario una base de datos con una considerable cantidad de datos, y que a su vez el mismo se encuentre balanceado lo máximo posible, para de esta manera ayudar al aprendizaje de la red, evitar la falta de generalización y el sobre ajuste del modelo.

El uso de modelos de *Transfer Learning* para la extracción de características de la señal de la tos COVID-19 permiten obtener mejores resultados que las redes neuronales convolucionales simples, debido a que los modelos pre entrenados como *YAMNet* y *VGGish* han sido implementados con conjuntos de datos extensos y su configuración ha sido implementada con expertos de Google. En este trabajo se ha validado que el mejor modelo para la extracción de características de la señal de la tos COVID-19 es *VGGish* por sus métricas de rendimiento en especial el *f1 score* de 30.76% y su matriz de confusión que presenta 163 casos de verdaderos negativos, lo que permite concluir que el modelo *VGGish* tiene un alto desempeño al momento de detectar casos que son COVID-19 negativo.

AGRADECIMIENTO

A la Corporación Ecuatoriana para el Desarrollo de la Investigación y Academia, CEDIA, por el financiamiento brindado a la investigación, desarrollo e innovación a través de los proyectos CEPRA, en especial el proyecto CEPRA-XV-2021-011: Caracterización de la tos provocada por el COVID - 19 en pacientes con diagnóstico positivo.

Los autores agradecen a la Escuela Politécnica Nacional, la Universidad Politécnica Salesiana y la Pontificia Universidad Católica del Ecuador.

REFERENCIAS

[1] Statista, "Número de personas fallecidas a consecuencia del coronavirus a nivel mundial a fecha de 16 de enero de 2022, por continente", 2022. https://es.statista.com/estadisticas/1107719/COVID-19-numero-de-muertes-a-nivel-mundial-por-region/

[2] Organización Mundial de la Salud (OMS), "Pruebas diagnósticas para el SARS-CoV-2",2020. https://apps.who.int/iris/bitstream/handle/10665/335830/WHO-2019-nCoV-laboratory-2020.6-spa.pdf

[3] R. Solera-Ureña, C. Botelho, F. Teixeira, T. Rolland, A. Abad, I. Trancoso, "Transfer Learning-Based Cough Representations for Automatic Detection of COVID-19". *Proc. Interspeech 2021*, 436-440, doi: 10.21437/Interspeech.2021-1702

[4] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data*," in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA*, 2020, p. 3474–3484.

[5] J. Huang, Y. Li, J. Tao, J. Yi, "Multimodal Emotion Recognition with Transfer Learning of Deep Neural Network". *ZTE Commun.* 2017, 15, 1.

[6] B. W. Schuller, A. Batliner, C. Bergler, C. Mascolo, J. Han, I. Lefter, H. Kaya, S. Amiriparian, A. Baird, L. Stappen, S. Ottl, M. Gerczuk, P. Tzirakis, C. Brown, J. Chauhan, A. Grammenos, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, L. J. M. Rothkrantz, J. Zwerts, J. Treep, and C. Kaandorp, "The INTERSPEECH 2021 Computational Paralinguistics Challenge: COVID-19 Cough, COVID-19 Speech, Escalation & Primates," *in Proceedings of INTERSPEECH 2021, Brno, Czechia*, Sept. 2021.

[7] B. Schuller, "Recognizing affect from linguistic information in 3D continuous space". *IEEE Trans. Affect. Comput*. 2011, 2, 192–205.

[8] A. Severyn, A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks". *In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Santiago, Chile, 9–13 August 2015; pp. 959–962.

[9] E. Lakomkin, C. Weber, S. Magg, S. Wermter, "Reusing Neural Speech Representations for Auditory Emotion Recognition". *In Proceedings of the Eighth International Joint Conference on Natural Language Processing*, Taipei, Taiwan, 27 November–1 December 2017; pp. 423–430.

[10] X. Gaowei, L. Min, J. Zhuofu, S. Dirk, S. Weiming, 2019. "Bearing Fault Diagnosis Method Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning." *Sensors 19*, no. 5: 1088.

[11] N. Xiao-Xiao, Y. Ching Suen, "A novel hybrid CNN–SVM classifier for recognizing handwritten digits, Pattern Recognition", Volume 45, Issue 4, 2012, Pages 1318-1325, ISSN0031-3203, https://doi.org/10.1016/j.patcog.2011.09.021.

[12] H. Basly, W. Ouarda, F. Sayadi, B. Ouni, and A.M. Alimi, "CNN-SVM Learning Approach Based Human Activity Recognition", *in Editor (Ed.) 'Book CNN-SVM Learning Approach Based Human Activity Recognition' (Springer International Publishing, 2020, edn.)*, pp. 271-281

[13] T. Liu, X. Ye and B. Sun, "Combining Convolutional Neural Network and Support Vector Machine for Gait-based Gender Recognition," 2018 Chinese. *Automation Congress (CAC), Xi'an*, China, 2018, pp. 3477-3481, doi: 10.1109/CAC.2018.8623118.

[14] G. Cao, S. Wang, B. Wei, Y. Yin, G. Yang, "A Hybrid CNN-Rf Method for Electron Microscopy Images Segmentation", 2013. *J Biomim Biomater Tissue Eng* 18:114.

[15] F. Li, M. Liu, Y. Zhao, L. Kong, L. Dong, X. Liu and M. Hui, "Feature extraction and classification of heart sound using 1D convolutional neural networks", 2019. https://doi.org/10.1186/s13634-019-0651-3.

[16] E. Koh, S. Dubnov, "Comparison and Analysis of Deep Audio Embeddings for Music Emotion Recognition", 2021. *University of California, San Diego*. arXiv:2104.06517v1.

[17] W. Jiang, Z. Wang, J. S. Jin, X. Han and C. Li., "Speech Emotion Recognition with Heterogeneous Feature Unification of Deep Neural Network", 2019.

[18] TensorFlow Blog, "Transfer Learning for Audio Data with YAMNet", 2021. https://blog.tensorflow.org/2021/03/transfer-learning-for-audio-data-with-yamnet.html

[19] YAMNet-CodiMD, "YAMNet", 2019. https://codimd.mcl.math.ncu.edu.tw/prfY5nA6RV6AOV-MvQwu9A

[20] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," *in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 131–135.

[21] L. Shi, C. Zhang, H. Ma and W. Yan, "Lung Sound Recognition Algorithm Based on VGGish-BiGRU", *IEEE*, 2019. DOI:10.1109/ACCESS.2019.2943492.

[22] Y. Gu, K. Yang, S. Fu, S. Chen, X. Li, I. Marsic, "Multimodal Affective Analysis Using Hierarchical Attention Strategy with Word-Level Alignment". *In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, Australia, 15–20 July 2018; Volume 2018, p. 2225.