UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

ESTIMACIÓN DE DIFICULTAD RESPIRATORIA EN ENTORNO DE INTERACCIÓN HUMANO-ROBOT

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA, MENCIÓN ELÉCTRICA

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO

EDUARDO ALEXIS ALVARADO GUTIÉRREZ

PROFESOR GUÍA:
NÉSTOR BECERRA YOMA

MIEMBROS DE LA COMISIÓN:
FERNANDO HUENUPÁN QUINAN
CLAUDIO ESTÉVEZ MONTERO

Este trabajo ha sido parcialmente financiado por Fondecyt Regular N°1211946

SANTIAGO DE CHILE
2023

RESUMEN DE LA TESIS PARA OPTAR
AL GRADO DE MAGÍSTER EN CIENCIAS
DE LA INGENIERIA, MENCIÓN ELÉCTRICA
Y MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELÉCTRICO
POR: EDUARDO ALEXIS ALVARADO GUTIÉRREZ
FECHA: 2023
PROF. GUÍA: NÉSTOR BECERRA YOMA

ESTIMACIÓN DE DIFICULTAD RESPIRATORIA EN ENTORNO DE INTERACCIÓN HUMANO-ROBOT

En este trabajo se propone el primer sistema de estimación de dificultad respiratoria en un entorno de interacción humano-robot (HRI) basado en *Deep learning*. Esta investigación utiliza la voz como fuente de información para detectar el nivel de disnea de las personas, aportando comodidad y usabilidad para los usuarios.

El procedimiento para entrenar los modelos de estimación de disnea, se basan en la simulación del entorno acústico HRI con respuestas de impulsos reales (estimadas con un robot PR2) y ruido aditivo. Los datos de entrenamiento y evaluación se procesaron mediante tres técnicas de *speech enhancement*: *delay-and-sum*, MVDR y cRF.

Los resultados sugieren que es posible reducir significativamente la degradación de la precisión en la estimación de dificultad respiratoria en un escenario real HRI. Donde un entrenamiento alineado entre las bases de datos de entrenamiento, evaluación y sus *speech enhancement* correspondientes, permiten entregar en promedio una mejora de 14 % y 6 % en precisión y AUC, respectivamente, frente al caso base de entrenar con datos telefónicos y evaluar el caso ruidoso real sin filtrado espacial.

Dedicado a mi familia,
amigos y cercanos.

Agradecimientos

Agradecer el inmenso apoyo brindado por mi familia durante toda mi vida, tanto de mis padres como mis hermanos. Además, siempre tendré en cuenta a mis amigos y personas con las que he compartido durante estos años, los que me han ayudado a ser mejor persona.
También quiero agradecer las correcciones brindadas por los profesores guías, quienes aportaron bastante a mejorar mi trabajo de titulación.

Para culminar, me encuentro muy agradecido de la ayuda y la buena disposición de todos los integrantes del LPTV, los que se han transformado en una verdadera familia, donde me he hecho amigos muy cercanos.

Tabla de Contenido

Índice de Tablas

Índice de Ilustraciones

Capítulo 1

Introducción

En este capítulo se presentará una breve introducción al proyecto de investigación. Primero, se expondrá la motivación que impulsó el estudio. Luego, se presentará la hipótesis en la que se basa la tesis. Finalmente, se describirán los objetivos generales y específicos del trabajo.

1.1. Motivación

En los próximos 10 o 20 años la comunicación entre seres humanos y robots tendrá un papel fundamental en diversas aplicaciones. Inclusive ahora, varios sectores han estado avanzando en la inclusión de robots para mejorar y optimizar sus procesos [1]. La mayoría de los robots utilizados se consideran únicamente herramientas, ya que se encargan de tareas específicas que no requieren mucha deliberación [2]. Por otro lado, existen los robots sociales, los que están diseñados para comunicarse y coordinarse con las personas para alcanzar objetivos comunes. Este tipo de robots son mucho más relevantes en ámbitos como la educación o la sanidad [3].

Para emular con éxito la comunicación humana, es necesario que el robot caracterice el perfil del usuario, ya sea física, cognitiva y/o socialmente [4]. De este modo, el robot puede adaptar su respuesta en función del comportamiento y las necesidades del usuario. Caracterizar físicamente a la persona es una tarea bastante complicada, ya que la obtención de la información necesaria para ello suele ser muy invasiva, como mediciones de la presión arterial, análisis de sangre o mediciones de la capacidad pulmonar. Otras opciones menos invasivas corresponden al uso de *wearables*, que permiten realizar mediciones del sueño, del movimiento, neurológicas, cardiovasculares, etc., de forma rápida y cómoda para el usuario [5]. Sin embargo, los *wearables* no son muy precisos ni robustos, por lo que mejorar sus prestaciones sigue siendo un desafío importante [6] [7].

En este contexto, el uso de la voz se alza como una alternativa importante para la elaboración de perfiles de usuario. La voz además de incluir información lingüística y paralingüística (prosodia) muy útil en diversas aplicaciones [8], también brinda información útil para detectar problemas respiratorios [9].

Centrándose en este último punto, los sistemas sanitarios mundiales soportan una pesada carga de enfermedades respiratorias crónicas. Las más comunes son el asma bronquial (esti-

mada en 262 millones de personas) y la enfermedad pulmonar obstructiva crónica (EPOC), que afecta a más de 200 millones de personas. La EPOC causa casi 3 millones de muertes al año, es decir, el 6 % de todos los fallecimientos [10]. Aunque la EPOC no se puede curar, su tratamiento mejora la calidad de vida de los pacientes al permitir un mejor control de los síntomas [11]. Por lo mismo, el ofrecer más oportunidades para detectar o cuidar esta enfermedad puede ser muy útil para las personas, tales como sistemas automáticos y/o más cupos de atención en los centros de salud. De hecho, desde el inicio de la pandemia mundial COVID-19, la automatización de los procedimientos de diagnóstico ha sido ampliamente estudiada. Sin embargo, el foco de estos trabajos se ha puesto en el COVID-19 [12], aunque algunas excepciones también incluyen la detección de otras enfermedades como el asma, la bronquitis o la tos ferina [13]. La mayoría de estos trabajos detectan afecciones respiratorias utilizando sonidos producidos por el sistema respiratorio, como la tos, la voz y la respiración [14] [15]. Siendo la voz, la fuente de información que permite la conexión entre la caracterización del perfil del usuario y la detección de dificultad respiratoria.

Si bien los confinamientos provocados por la pandemia parecen haber llegado a su fin, los problemas en los centros sanitarios aún persisten, como lo son la falta de suministros, escasez de profesionales y el crecimiento de la población vulnerable. Aquí es donde surge un entorno ideal para que los robots sociales se inmiscuyan en este mundo, atendiendo, diagnosticando y/o tratando a los pacientes de una forma tan similar a como lo haría un humano. Los robots sociales ya se han ido extendiendo en el ámbito sanitario debido a estos problemas [16], pero en general realizando tareas administrativas o de cuidado de niños, ancianos y personas con movilidad reducida [17] [18]. Sorprendentemente, el uso de robots sociales para estimar la disnea es nulo, sobre todo si se tiene en cuenta el auge en los últimos años de los estudios de dificultad respiratoria y de interacción humano-robot (HRI) por separado.

En base a lo anterior, surge la motivación para impulsar el desarrollo de esta tesis, la que consiste en sentar un precedente en el estado del arte al proponer un sistema de estimación de dificultad respiratoria de forma automática en un entorno HRI, con lo que se espera que estudios como este empiecen a extenderse y por que no, sean aplicados en los centros sanitarios del mundo. De esta forma, se permitirá aportar en la reducción de la congestión y sobrecarga de trabajo en los centros médicos, pero sin dejar de lado la necesidad de interacción y comunicación que requiere el ser humano. Esta necesidad podrá ser suplida en parte gracias a los robots sociales, que además de interactuar lo más parecido a un humano, también podrá atender consultas diagnosticando y tratando a pacientes, y en particular para este trabajo, estimando el nivel de disnea.

1.2. Hipótesis

Entrenar un sistema de estimación de dificultad respiratoria basado en *Deep Learning* simulando condiciones reales de evaluación acústica y de reducción de ruido, permitiría reducir la degradación de rendimiento al evaluar condiciones reales HRI frente a la grabación original telefónica.

1.3. Objetivo General

En base a lo visto en la motivación, un sistema de detección automática de dificultad respiratoria en un entorno de interacción humano-robot podría traer grandes beneficios a la sociedad, específicamente a la reducción de carga sobre los recintos asistenciales, mejorando la calidad de vida de tanto el personal de salud como de los pacientes. Es por esto que se define como objetivo general de este estudio lo siguiente.

- Mejorar la exactitud y robustez del sistema de detección de dificultad respiratoria en un entorno de interacción humano-robot estático y dinámico con respecto a un sistema *baseline* sin técnicas de reducción de ruido.

1.4. Objetivos específicos

El cumplimiento de los objetivos específicos permiten alcanzar el objetivo general de esta tesis, los cuales son enumerados a continuación:

- Generar una base de datos con voces de personas con y sin disnea en condiciones reales HRI para realizar esta y futuras investigaciones.
- Mejorar la exactitud en la estimación de dificultad respiratoria en HRI para un caso real estático con respecto al *baseline* sin técnicas de reducción de ruido.
- Mejorar la exactitud en la estimación de la dificultad respiratoria en HRI para un caso real dinámico con respecto al *baseline* sin técnicas de reducción de ruido.

Capítulo 2

Estimación de dificultad respiratoria e interacción humano-robot

En este capítulo se presentarán los conceptos clave y una exhaustiva revisión del estado del arte de los diferentes temas que se abordan en esta tesis. Estos temas incluyen robots sociales y el perfil del usuario, estimación de la dificultad respiratoria y la interacción Humano-Robot (HRI).

2.1. Robots sociales y perfil del usuario

La comunicación entre seres humanos y robots tendrá un papel fundamental en un sin fin de aplicaciones en los próximos 10 o 20 años. De hecho en la actualidad, sectores como la salud, educación, defensa, seguridad y la industria han estado avanzando en la inclusión de robots para mejorar y/u optimizar sus procesos [1]. De igual forma, en su gran mayoría los robots utilizados son considerados solo como herramientas, debido a que están programados para tareas específicas que no requieren de mucha deliberación, como por ejemplo, aplicaciones teleoperadas [2]. Por el lado contrario, los robots sociales tienen una finalidad completamente distinta, donde al estar diseñados para interactuar con personas, deben comunicarse y coordinarse con ellas para lograr objetivos comunes. En base a lo anterior, estos robots cobran una relevancia aún más importante en áreas como la educación o salud, donde deben asistir u orientar a las personas de la forma más parecida a como lo hace el ser humano [3].

Para mejorar la efectividad de la comunicación entre las partes, el robot debe tener la capacidad de adaptar la forma en la que responde en base al comportamiento y necesidades del usuario. Esto con la intención de emular la comunicación humano-humano, la que por definición es multimodal y permite entender distintos contextos a partir de información que obtiene del exterior, como lo es la actitud y el estado anímico del otro interlocutor [19].

Si se desea emular con éxito la comunicación de las personas, es necesario que el robot pueda caracterizar el perfil del usuario, obteniendo información que permita modelarlo en base a su estado físico, cognitivo y/o social [4]. Uno de los problemas que surgen es que obtener información para "perfilar" físicamente a las personas suele ser altamente invasivo, donde mediciones de presión, sangre o pruebas de capacidad pulmonar estándar requieren de una complejidad elevada. Otras opciones menos invasivas corresponden al uso de sensores

"vestibles" o *wearables*, los cuales ya están siendo utilizados en ambientes sanitarios, gracias a que permiten realizar mediciones de sueño, movimiento, neurología, cardiovasculares, etcétera [5]. De igual forma, los *wearables* aún no poseen una precisión estable, y entre distintas mediciones sus resultados pueden variar considerablemente, por lo que mejorar la robustez de estos dispositivos es uno de los grandes desafíos que se presentan [6] [7].

En este contexto, el utilizar la voz surge como una de las alternativas más cómodas y útiles para realizar el perfilamiento del usuario. La voz es el canal de comunicación por excelencia entre las personas, ya que permite capturar información lingüística y paralingüística (prosodia), como lo son las palabras que se comunican y el contexto en las que se dictan [8]. Además de esto, la voz ha sido ampliamente utilizada para detectar dificultad respiratoria [9], por lo que no solo permite mejorar la comunicación entre el humano y robot, si no que también permite diagnosticar y/o monitorear el estado de salud de los personas.

Esto abre la posibilidad de que el robot no solo sea útil para orientar o asistir a las personas en los centros médicos, si no que también podría tener la capacidad de atender en consultas, mejorando considerablemente la calidad de los centros de salud: reduciendo los tiempos de espera para el público; evitando la toma de exámenes invasivos; y apoyando en la disminución de carga de trabajo sobre el personal médico.

2.2. Estimación de dificultad respiratoria

En esta sección se explicarán distintos mecanismos de estimación de dificultad respiratoria, tanto los tradicionales, como los más modernos que se pueden encontrar en el estado del arte.

2.2.1. Antecedentes y mecanismos tradicionales de estimación

Las enfermedades respiratorias crónicas (CRDs por sus siglas en inglés, *Chronic respiratory diseases*) generan una alta carga sobre los sistemas de salud alrededor del mundo. Se estima que 262 millones de personas sufren de asma bronquial y más de 200 millones de personas padecen enfermedad pulmonar obstructiva crónica (EPOC), siendo ambas las CRDs más comunes. Cada año fallecen sobre 3 millones de personas a causa de EPOC, representando un 6 % de los fallecidos al año [10]. Si bien las CRDs no son curables, el tratamiento de estas permiten controlar de mejor forma los síntomas y por tanto, mejorar la calidad de vida de las personas que las padecen [11].

Los pacientes con enfermedades cardiopulmonares suelen tener problemas respiratorios, al que los médicos le denominan disnea. La disnea se puede definir como "una sensación de incomodidad al respirar" [20]. La escala modificada del Medical Research Council (mMRC), es la escala validada más utilizada para evaluar la disnea en la vida diaria de las personas. Esta posee 5 niveles de severidad que son detallados en la tabla 2.1 [21].

Tabla 2.1: Escala de Disnea Modificada del Medical Research Council (mMRC)

Grado	**Descripción de la falta de aire**
0	Sólo me quedo sin aliento con el ejercicio extenuante
1	Me falta el aire cuando me apresuro en un terreno llano o cuando subo una pequeña colina.
2	En terreno llano, camino más despacio que las personas de la misma edad porque me falta el aire, o tengo que parar para respirar cuando camino a mi ritmo en el nivel
3	Me detengo para respirar después de caminar unos 100 metros o después de unos minutos en terreno llano.
4	Me falta el aire para salir de casa o para vestirme

Uno de los mecanismos más utilizados para detectar y controlar afecciones respiratorias corresponden a los rayos X, debido a su rapidez, accesibilidad y bajo costo. Otro examen ampliamente empleado es la tomografía computarizada, la cual permite visualizar y detectar cuantitativamente la severidad de la enfermedad [22]. La prueba de espirometría igualmente permite detectar desórdenes pulmonares, cuyos resultados deben ser interpretados cuidadosamente por el médico especialista [23]. Expertos biomédicos también utilizan el análisis de sonidos derivados del sistema respiratorio (sonido de pulmones, tos, respiración, voz, sonidos del corazón), para detectar cuadros respiratorios como Asma, bronquitis, Pertusis y SARS-CoV-2 [24].

Desde el surgimiento de la pandemia mundial COVID-19 a finales del 2019, el foco de estudio se ha centrado en la automatización de los diagnósticos. Desde la fecha, han sido propuestas un gran número de soluciones basadas en inteligencia artificial para detectar de forma automática SARS-CoV-2 [25], dejando en un segundo plano otras enfermedades respiratorias.

De los métodos disponibles en el estado del arte, los que se encargan de la detección automática de COVID-19 (u otras enfermedades respiratorias) utilizando como entrada los exámenes de rayos X y tomografía computarizada de los pulmones, han sido ampliamente estudiados [26]. Si bien este mecanismo permite que no se necesite un radiólogo analizando cada examen, si requiere de la toma de muestras que puedan resultar incómodas y engorrosas para el paciente.

Otros estudios apuntan a la detección automática de SARS-CoV-2 u otras enfermedades respiratorias por medio de sonidos generados por el sistema respiratorio, como lo son la tos, la voz y la respiración [14] [15], aunque en la gran mayoría el foco está puesto en el análisis de la tos forzada para clasificar. La tos es un síntoma frecuente tanto en resfriados comunes como en cuadros respiratorios, donde este síntoma por sí solo representa entre un 10 % y 38 % de solicitudes de trastornos respiratorios [27]. Si bien puede considerarse a la tos como una fuente importante de información para el modelo de *machine learning* (ML), no es replicable de forma natural, por lo que debe ser forzada para realizarla en el momento solicitado, generando incomodidad para la persona que quiera ser diagnosticada. Además, existen estudios que catalogan que utilizar la tos no es lo más efectivo para clasificar enfermedades respirato-

rias como COVID-19, obteniendo peores resultados que con una vocal sostenida o la lectura de un texto [28].

Es aquí donde la voz toma un papel fundamental, permitiendo la obtención de información fundamental para conocer el estado de las enfermedades respiratorias, evitando la toma de exámenes de mayor complejidad o pruebas incómodas como forzar la tos. Para obtener información de la voz, se hace necesaria realizar una extracción de características que representen el comportamiento dinámico de esta.

2.2.2. Extracción de características de la voz

A partir de la señal de voz, se pueden extraer distintas características que permiten representar su comportamiento a lo largo del tiempo. En particular para este trabajo, las principales características se detallan a continuación.

I. Espectrograma

Un espectrograma es una herramienta básica y ampliamente utilizada en el análisis de voz, ya que permite el análisis tiempo-espectral de la señal. Este se define como un gráfico de intensidad (generalmente en escala logarítmica) de la magnitud de la *Short-Time Fourier Transform*. Para graficar el espectrograma, lo primero que se debe hacer es "enventanar" la función, definiendo el tamaño de la ventana y el solapamiento de estas. Una vez se tenga "enventanada" la señal (asumiendo periodicidad en dicho segmento) se calcula la FFT. El gráfico consta de 3 ejes, donde el eje x corresponde al eje temporal de los *frames*, en el eje y se ubican las frecuencias en Hertz y en el eje z (en general en una escala de colores) se representa la magnitud de la STFT [29].

Este gráfico tridimensional permite realizar un análisis temporal y frecuencial de forma directa, ya que para cada *frame* de tiempo es posible determinar la energía presente para las distintas frecuencias. De esta forma se entrega información valiosa para realizar un análisis de la voz, ya que muestra claramente el comportamiento de los distintos armónicos y componentes frecuenciales a lo largo de la señal. En la Figura 2.1, se muestra un extracto de la señal de voz de una persona contando del uno al treinta, tanto en el dominio temporal como su espectrograma.

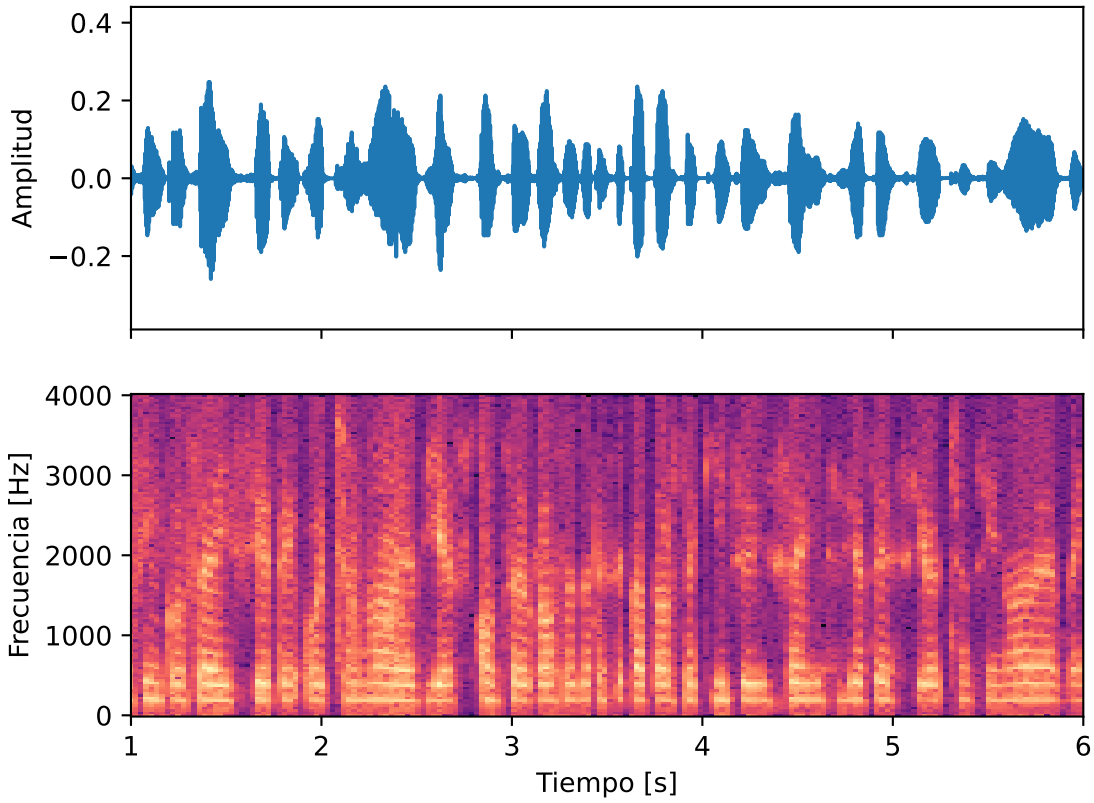


Figura 2.1: Señal de voz en el dominio temporal y su espectrograma.

II. Coeficientes Cepstrales en las Frecuencias de Mel (MFCC)

Los *Mel Frequency Cepstral Coefficients* o MFCCs son coeficientes que se encargan de representar características especificas del habla en base a la percepción auditiva humana. A partir del espectrograma, se debe aplicar un banco de filtros en la escala Mel y sumar las energías correspondientes, luego a cada frecuencia Mel se le calcula el logaritmo y se le aplica la transformada coseno discreta [30].

Como se dijo anteriormente, los MFCCs permiten representar el espectrograma en una escala con menor dimensionalidad, adaptando la señal de forma equivalente a como lo hace la capacidad auditiva humana [31]. Como consecuencia, diversas aplicaciones de procesamiento de voz, como lo son el reconocimiento automático de voz y la identificación o clasificación de hablantes funcionan bastante bien. En la Figura 2.2, se muestra un espectrograma normal y un espectrograma MFCC para la misma fonetización anterior (conteo del 1 al 30), donde es fácil notar la reducción de definición espectral al pasar de los 257 *bins* frecuenciales de la Figura 2.2 (gráfico inferior) a los 14 filtros de Mel utilizados (gráfico superior).

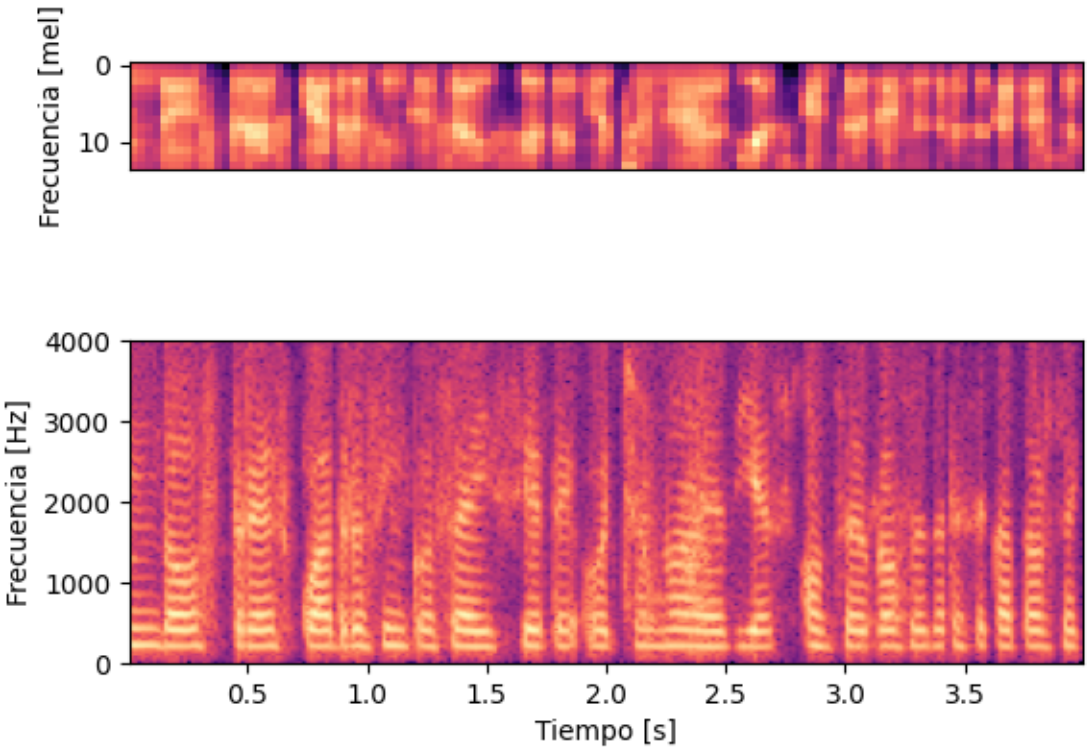


Figura 2.2: Espectrograma y MFCCs de una señal de voz.

Al igual que a los espectrogramas, a los MFCCs también se les puede calcular la primera y segunda derivada, lo que es útil para evaluar el comportamiento dinámico de la señal de voz [32].

III. Tono (f_0)

El tono se define como la altura o elevación de la voz que resulta de la frecuencia de las vibraciones de las cuerdas vocales. Si existe un número elevado de vibraciones por segundo, hay mayor tensión en las cuerdas vocales y por tanto esta será más aguda. Por el contrario, si existe un menor número de vibraciones, existe menor tensión en las cuerdas, implicando un tono de voz más grave. La frecuencia fundamental de oscilación de la señal, indicará el tono de la voz de la persona, por lo que en este trabajo, se llamará indiscriminadamente al tono de la voz como frecuencia fundamental, f_0 o *pitch*. Los múltiplos de esta frecuencia fundamental corresponden a los armónicos.

IV. Jitter y Shimmer

El *jitter* se define como el parámetro que mide la variación de la frecuencia fundamental entre ciclo y ciclo, mientras que el *shimmer* como la variación de la amplitud de la onda sonora [33]. En la Figura 2.3, se muestra donde se miden las perturbaciones del *Jitter* y *Shimmer* en una señal de voz (con bastante acercamiento) de la elocución controlada compuesta por la repetición de los fonemas /sa-sa/.

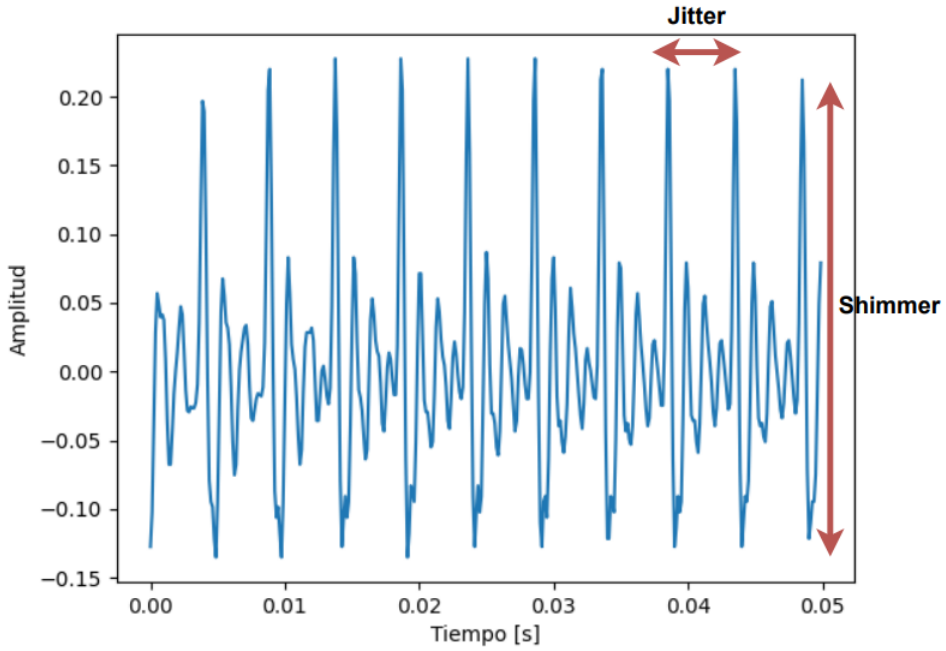


Figura 2.3: *Jitter* y *Shimmer* en una señal de voz.

2.2.3. Inteligencia artificial para la estimación de dificultad respiratoria

Como ya se ha mencionado, los estudios de identificación automática de afecciones respiratorias se han centrado principalmente en el COVID-19 [12], aunque también se han incluido enfermedades como el asma, la bronquitis o la tos ferina en algunos estudios [13]. Sin embargo, apenas se ha abordado la gravedad de los síntomas respiratorios, siendo que la gravedad de los síntomas es una métrica muy importante para el seguimiento de los pacientes, ya que en caso contrario, la detección solo sería útil como un primer diagnóstico. Una excepción se presenta en [34], donde se propone un método para clasificar a los pacientes en diferentes grados de EPOC en una escala de uno (leve) a cuatro (muy grave) según el FEV1 (volumen espiratorio forzado).

Las bases de datos de acceso abierto utilizadas para entrenar modelos de *machine learning* como COSWARA [35], DICOVA [36] o COUGHVID [37], entre otras, así como las bases de datos privadas, muestran algunas similitudes en los audios que las componen, como lo es el uso de vocales sostenidas, respiración, lectura de frases o la tos forzada. A pesar de que el uso de vocales sostenidas es bastante común, es importante tener en cuenta que los esquemas de supresión de ruido de los teléfonos móviles pueden atenuar las señales estacionarias. Estudios como [38] emplean los mismos micrófonos para todos los participantes con el fin de evitar cualquier desajuste en el preprocesamiento de audio. Además, las bases de datos públicas o privadas suelen ser pequeñas porque son difíciles de producir, lo que a su vez obliga a optimizar el procedimiento de entrenamiento para maximizar la precisión y robustez final. Para aumentar el número de ejemplos de entrenamiento, se suelen adoptar métodos de aumento de datos (*data augmentation*) como el desplazamiento temporal [12] y el entrenamiento por validación cruzada *k-fold* [28].

Los esquemas basados en *machine learning* que emplean el habla como entrada, suelen extraer características como los coeficientes cepstrales de frecuencia Mel (MFCC) y los espectrogramas de frecuencia Mel, que han sido ampliamente empleados en el reconocimiento automático del habla (ASR) [39] y también se han propuesto en [40] para la detección de dificultades respiratorias. Además, la primera y segunda derivada de estos coeficientes permiten evaluar la dinámica de la señal de voz [41]. Otras características como el *pitch*, el *jitter* y el *shimmer* se propusieron en estudios como [42] también para la detección de COVID-19.

La optimización de arquitecturas y parámetros de *machine learning* es una práctica común, como puede verse en [12] [43] [44] [45] [15], donde los problemas de detección de COVID-19 o de problemas respiratorios se abordaron empleando capas de redes neuronales convolucionales (CNN) para obtener características profundas. Las características resultantes se concatenan y se introducen en un clasificador basado en redes neuronales que se entrenó *end to end* para combinar los parámetros. También se ha adoptado el entrenamiento por etapas: primero, los módulos de clasificación se entrenan de forma independiente con cada conjunto de características; después, la salida de los clasificadores se combinan para obtener la decisión final del sistema. Este tipo de estrategia permite optimizar la información proporcionada por cada conjunto de características y explorar métodos de fusión de clasificación, lo que a su vez no es posible con una arquitectura de red neuronal única. Por ejemplo, en [15], las salidas de los módulos de clasificación (es decir, softmax) se introducen en un *Support Vector Machine* para obtener la decisión final. En [46], la decisión final se obtiene aplicando la regla del voto mayoritario a las salidas del clasificador. En [47], las probabilidades de salida se ponderan para obtener la decisión final de clasificación.

Sorprendentemente, el estudio de la optimización de la complementariedad que pueden proporcionar los distintos tipos de fonetizaciones no se ha abordado de forma exhaustiva. En algunos casos, como en [28], se empleó la arquitectura CNN VGG19 para encontrar la vocalización que podía proporcionar la mayor precisión en la identificación de pacientes post COVID-19. En otros estudios, como en [13], las características extraídas de las fonetizaciones se concatenan y se introducen en una red neuronal que se espera que aprenda a combinarlas.

Como se ha mencionado anteriormente, los métodos basados en *machine* y *deep learning* han tomado una relevancia importante en el estado del arte. En la próxima sección se detallarán conceptos básicos para comprender las soluciones que se abarcan en la literatura y en este trabajo.

I. Machine Learning

El *machine learning* o aprendizaje automático es un algoritmo computacional que es capaz de aprender de los datos en vez de recibir una programación explícita. Este mecanismo que forma parte de la inteligencia artificial, necesita entrenar con datos de entrenamiento y así dar una salida a datos externos o de evaluación que no ha visto anteriormente. Comúnmente se utilizan diversos tipos de modelos, como lo son los de clasificación, regresión, *clustering*, entre otros [48].

Existen diversas categorías de *machine learning* dependiendo del objetivo del modelo y de los datos utilizados. El aprendizaje supervisado consta de entrenar el sistema con datos etiquetados, es decir, que para cada ejemplo utilizado se tiene su respectiva respuesta, de

esta forma el modelo aprende a obtener patrones en los datos e ir "entendiendo" que para dichas características se tiene una respectiva salida. El aprendizaje no supervisado se aplica cuando existen datos sin etiquetar (y generalmente son muchos), por lo que etiquetarlos no sería efectivo, en estos casos los modelos también encuentran patrones, pero con la diferencia de que les sirven para asociar los datos a distintos grupos o *clusters*. El aprendizaje semi-supervisado es una combinación de los métodos anteriores, donde se tienen datos con y sin etiqueta para aprender.

Dentro del *machine learning* se encuentra el *deep learning*, que en los últimos años ha tenido un desarrollo importante debido a la gran mejora a nivel de rendimiento que ha tenido en diversas tareas por sobre los métodos de aprendizaje automático tradicionales. Estos modelos poseen múltiples capas de procesamiento, las cuales permiten aprender y extraer características con distintos niveles de abstracción a partir de los datos crudos (*raw data*).

II. Perceptrón multicapa (MLP)

El perceptrón multicapa o MLP (por sus siglas en inglés, *Multi Layer Perceptron*), es un sistema de neuronas simples interconectadas con distinto número de capas. La unidad básica del MLP corresponde al perceptrón, la cual emula a una neurona humana [49]. El diagrama de la Figura 2.4 muestra los componentes del perceptrón, los cuales se explican a continuación [50].

- Entrada: Puede ser un valor o un vector de datos.
- Pesos: Los pesos multiplican a los datos de entrada uno a uno y se actualizan en cada iteración del entrenamiento.
- Suma Ponderada: Es una suma de los valores obtenidos al multiplicar la entrada con los pesos sinápticos. Además se agrega la suma de una constante denominada sesgo o *bias*.
- Función de activación: Es una función que se le aplica a la salida de la neurona, esta puede ser lineal o no lineal. El objetivo de esta función es mapear la salida en un rango de valores dependiendo del tipo de función utilizada, generalmente en valores dentro de un rango de -1, 0, 1, etc. Las funciones de activación más utilizadas son la sigmoide, ReLU, tanh, PReLU, etc.

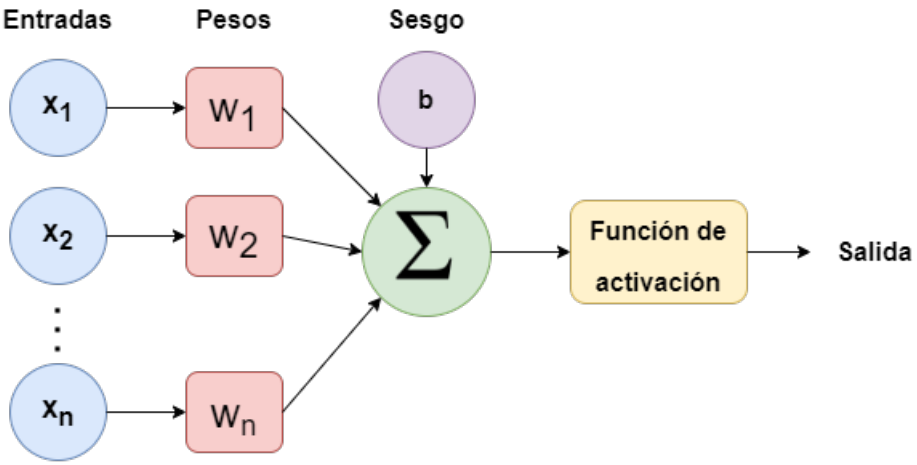


Figura 2.4: Esquema de perceptrón.

El utilizar distinto número de capas y neuronas, brinda la capacidad al MLP de resolver problemas no linealmente separables, generando un mapeo no lineal entre un vector de entrada y un vector de salida, solucionando así un problema intrínseco del perceptrón, el cual solo funciona para problemas lineales [51]. Un ejemplo de estructura MLP, es presentado en la Figura 2.5, donde se tiene una red con una capa de entrada de tres neuronas, dos capas ocultas de cinco neuronas y una capa de salida de dos neuronas.

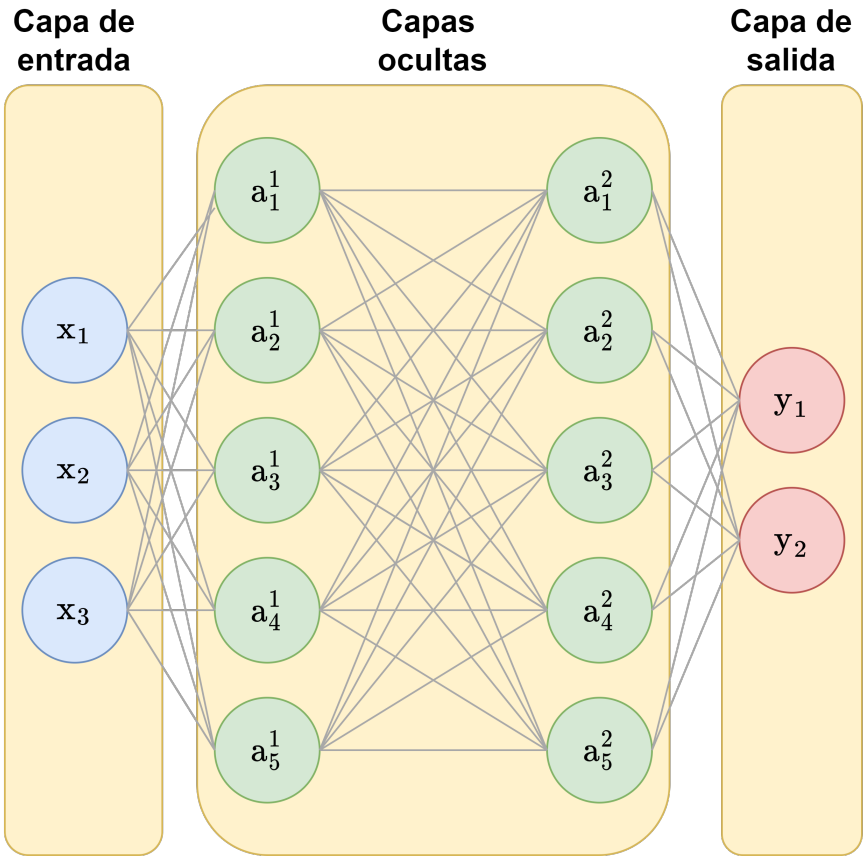


Figura 2.5: Diagrama de un perceptrón multicapa.

III. Redes Convolucionales

Las redes convolucionales o CNNs (por sus siglas en inglés, *Convolutional Neural Network*), han tenido un amplio desarrollo en los últimos años en muchos campos como: visión computacional, reconocimiento de voz y procesamiento de lenguaje natural. Esto en base a los buenos resultados que ha demostrado tener sobre otros algoritmos de aprendizaje automático [52]. La gran ventaja que presentan este tipo de redes, es que permiten realizar la extracción de características de forma automática durante el entrenamiento, generando así características que no son directamente observables o replicables por el ser humano. Otras ventajas importantes frente a las redes totalmente conectadas, son la reducción de parámetros y la aceleración de convergencia, además de la reducción de dimensionalidad, lo que permite disminuir así la cantidad de data con información redundante [53]. La arquitectura típica de una CNN se muestra en la Figura 2.6.

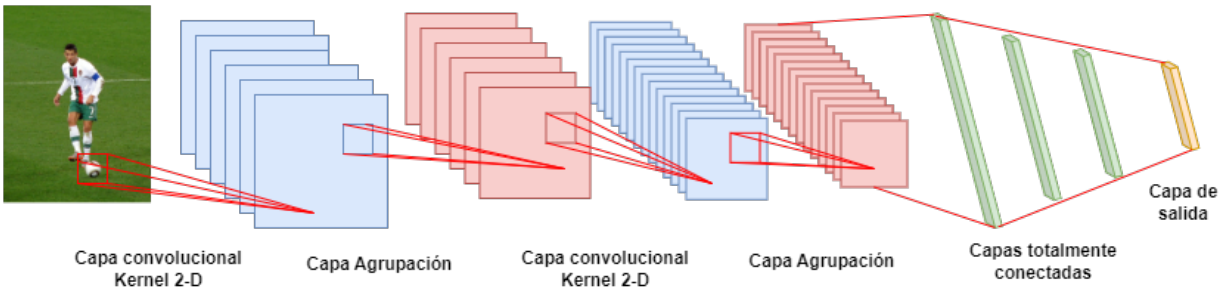


Figura 2.6: Esquema de red convolucional.

A grandes rasgos, la red convolucional se compone de 3 partes; las capas convolucionales, capas de agrupación y por las totalmente conectadas.

Capas convolucionales

Estas capas son las encargadas de extraer las características mediante la operación de la convolución. Para esto se utilizan *kernels*, los cuales operan punto a punto con los datos de entrada y se van deslizando hasta convolucionar todos los datos de entrada, además pueden ser de una o dos dimensiones. El *kernel* al ser convolucionado con todos los datos de entrada, genera un vector o matriz de características, por lo que si se utilizan varios filtros, se obtienen matrices de tres dimensiones que son conocidos como mapas de características. En la Figura 2.7, se muestra como se realiza la operación de la convolución para un kernel 3x3 (dos dimensiones).

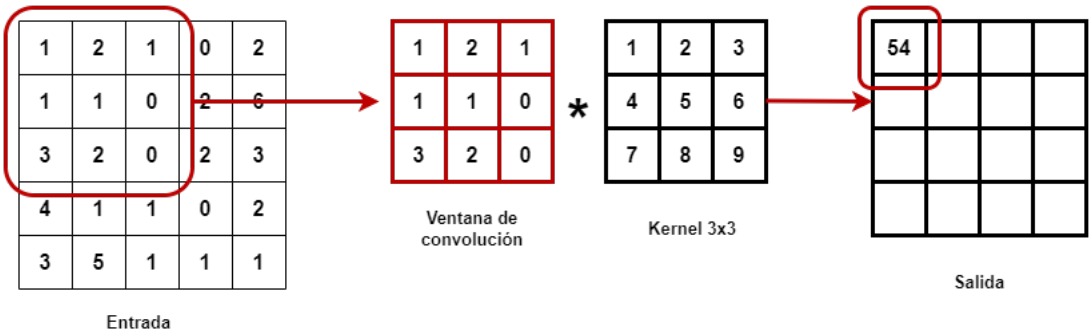


Figura 2.7: Proceso de convolución para *kernel* de 3x3.

La forma tradicional de realizar el deslizamiento del *kernel*, es ir avanzando muestra por muestra, aunque también es posible utilizar un "paso" mayor, lo que permite avanzar más muestras en cada desplazamiento del filtro. Otra variación utilizada corresponde a la convolución dilatada, en la cual se utiliza un *kernel* que no convoluciona con los datos contiguos al punto central, si no que selecciona puntos más distantes en base a un tasa de dilatación, obteniendo mayor contexto de la vecindad del dato principal [54]. En la Figura 2.8, se muestra la diferencia entre un *kernel* de 3x3 para distintas tasas de dilatación, donde es claro como se alcanza un mayor contexto de las vecindades a medida que la tasa aumenta.

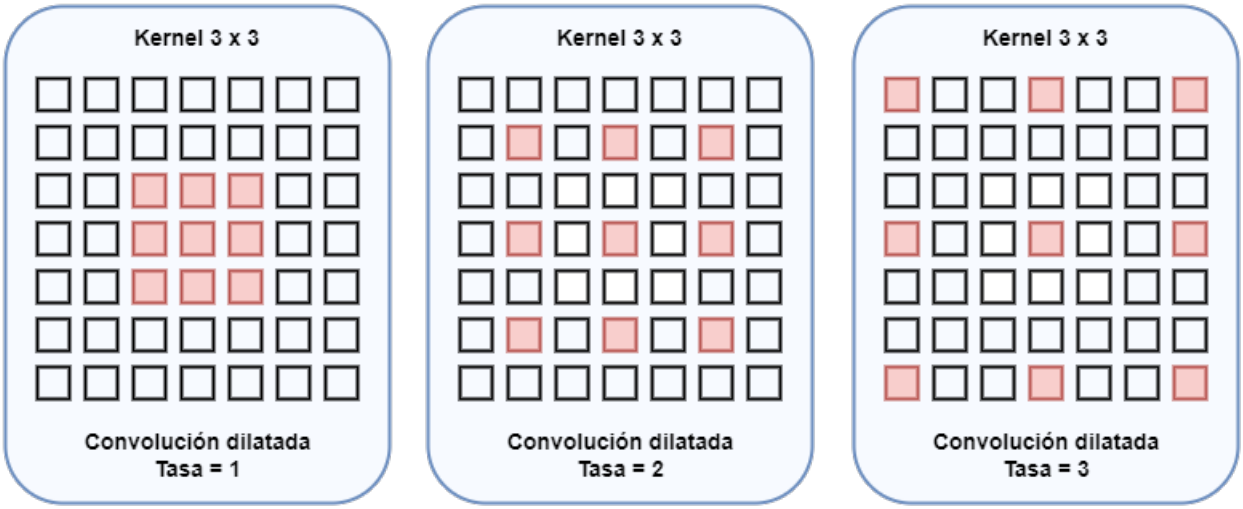


Figura 2.8: Kernel con distintas tasas de dilatación.

Capas de agrupación

El convolucionar los distintos *kernels* con todos los datos de entrada generan mucha información redundante, que ralentiza el entrenamiento y complejiza la convergencia de este. En base a lo anterior, las capas de agrupación permiten reducir la dimensionalidad de los mapas de características, conservando la información más relevante y desechando la redundante [55]. El proceso consta de tomar un subconjunto de datos y quedarse con solo un valor, generalmente la operación utilizada es el promedio o el máximo. En la Figura 2.9 se muestra el proceso del *max pooling* 2x2 ejemplificado.

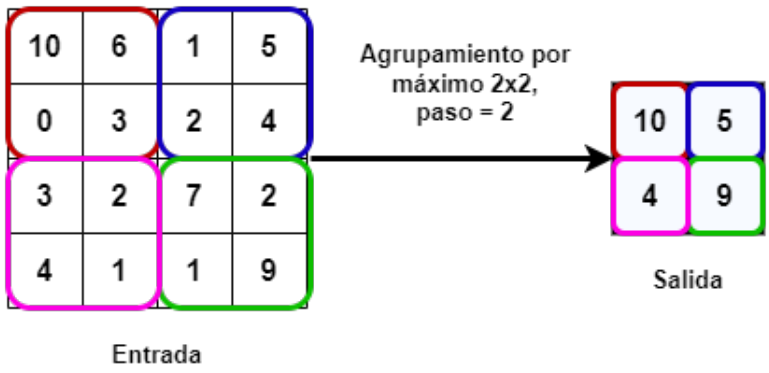


Figura 2.9: Ejemplo de agrupamiento por máximo de 2x2, paso igual a 2.

.

Capa totalmente conectada

Estas capas corresponden a la última parte de la red convolucional, tal como se aprecia en la Figura 2.6. Aquí se reciben los mapas de características extraídos por las capas convolucionales, los cuales son "aplananados" (para dejarlos de una dimensión) y posteriormente entregados a las neuronas completamente conectadas (MLP) para realizar la clasificación. Es debido a esto, que se intenta reducir previamente la dimensionalidad de los mapas de características, evitando así un número muy grande de parámetros al momento de aplanar el vector de características.

IV. Conexiones residuales

Las redes convolucionales a pesar de tener un buen rendimiento en las tareas asignadas, son difíciles de entrenar, ajustar y optimizar. No es directo pensar en que el agregar más capas hará que el rendimiento de la red mejore, de hecho es probable que ocurra todo lo contrario [56]. Es por esto que en [57] se proponen las conexiones residuales, las cuales evitan que la red disminuya su rendimiento si es que se adicionan más capas a la arquitectura, haciendo más fácil optimizar los modelos convolucionales. La conexión residual conecta de forma directa la salida de una capa con la de otra capa no contigua. De esta forma, si es que el salida de una capa no esta aportando al aprendizaje, se le da más importancia a la salida de la capa anterior, evitando una degradación en el rendimiento. A continuación, la Figura 2.10 muestra un diagrama de una conexión residual sobre dos capas.

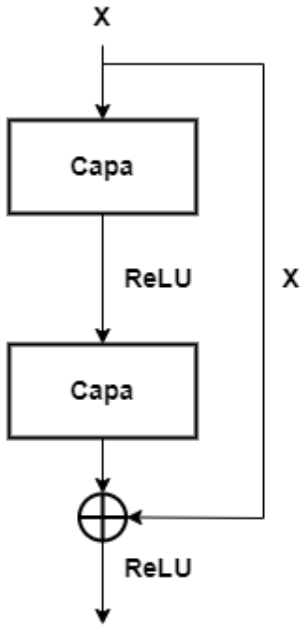


Figura 2.10: Esquema de conexión residual.

V. Redes recurrentes

Las redes recurrentes (RNNs, por sus siglas en inglés *Recurrent Neural Networks*), tienen la capacidad de aprender dependencias entre puntos, por lo que son muy útiles en series de

tiempo o en datos con dependencia. Aquí se diferencian de las redes tradicionales *feed-forward* (como las MLP o CNNs), que suponen independencia entre los datos que reciben. En base a dicha dependencia, surge el concepto de "memoria", ya que las RNNs para generar la salida actual, pueden utilizar sus estados internos que contienen información relevante de entradas pasadas [58]. La Figura 2.11 muestra el diagrama simple de una red recurrente, donde para cada tiempo t se tiene un estado h_t, el cual almacena información de entradas pasadas y se va propagando hacia el futuro, proceso al cual se le asocia el término de "memoria".

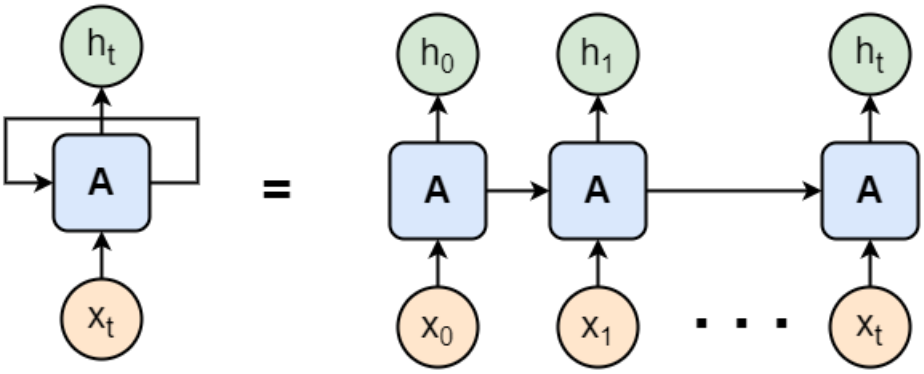


Figura 2.11: Diagrama de una RNN [59].

De todos modos, las RNNs no son perfectas, ya que presentan problemas como el desvanecimiento del gradiente, haciendo que la red no pueda aprender información relevante cuando el largo de la secuencia o serie temporal sea considerablemente extensa [60].

V.I. LSTM

La *Long Short-Term Memory* o LSTM por su siglas, es un tipo de red recurrente que busca solucionar el problema de las dependencias temporales largas mediante su arquitectura basada en celdas, evitando así en gran parte el desvanecimiento del gradiente [61]. De forma general, la salida para un tiempo en especifico de la LSTM va a depender de la memoria a largo plazo actual de la red (estado de la celda), salida del tiempo anterior (estado oculto) y la entrada actual a la celda. A continuación, se muestra el diagrama y las conexiones internas de la celda de LSTM.

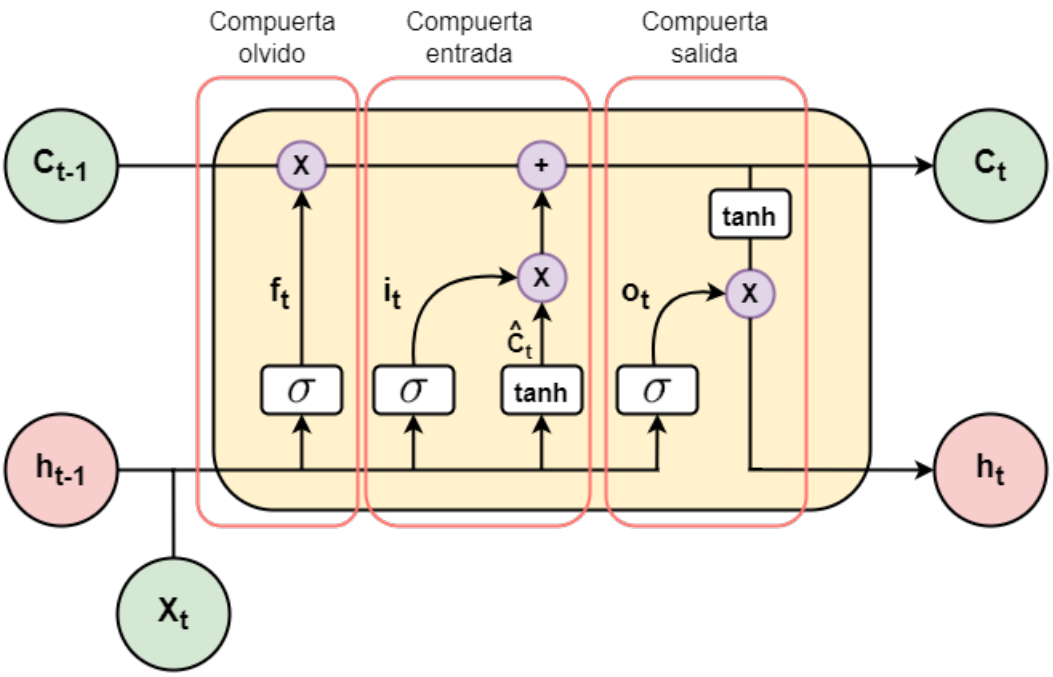


Figura 2.12: Diagrama de una celda LSTM [62].

Tal como se aprecia en la Figura 2.12, la celda de la LSTM puede ser separada en 3 segmentos o compuertas [63]:

Compuerta de Olvido

Esta compuerta se encarga a decidir que información es relevante y debe mantenerse en el estado de la celda o memoria de la LSTM. Aquí se realiza una multiplicación punto a punto entre el estado anterior de la celda y una concatenación entre, los datos de entrada y el estado oculto anterior. Esta compuerta define la relevancia de la información, con valores pertenecientes al rango de 0 y 1, donde 0 equivale a información irrelevante y 1 a relevante.

Compuerta de Entrada

En esta compuerta se define la "nueva" memoria de la celda, en la que se utiliza el estado oculto de la celda anterior concatenada con la entrada para el tiempo t. Este vector es activado por una sigmoide y una tangente hiperbólica, luego estas salidas son multiplicadas punto a punto y sumadas a la salida de la compuerta de olvido, definiendo así la nueva memoria o estado de la celda.

Compuerta de Salida

Aquí se define la salida de la celda o el estado oculto. La entrada es la misma que para la compuerta de olvido (estado anterior y entrada actual) con una función de activación sigmoide, además se utiliza la salida de la compuerta de entrada, pero activada con una función tanh, entre -1 y 1. Estos vectores son multiplicados punto a punto y con ello se obtiene la salida de la celda o el estado oculto.

Las conexiones que definen la celda LSTM son las siguientes:

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \tag{2.1}$$

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \tag{2.2}$$

$$\hat{c}_t = tanh(W_{\hat{c}h}h_{t-1} + W_{\hat{c}x}x_t + b_{\hat{c}}) \tag{2.3}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \hat{c}_t \tag{2.4}$$

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \tag{2.5}$$

$$h_t = o_t \cdot tanh(c_t) \tag{2.6}$$

VI. Función de pérdida de entropía cruzada

Cuando se trabaja con redes neuronales, el problema que se busca es minimizar la función de costo o pérdida. Mientras más bajo es este error significa que la red más ha "aprendido". Dependiendo del tipo de problema que se busca resolver, se utilizan distintas funciones de pérdida. En particular la más utilizada para clasificación consta de la pérdida de entropía cruzada. Esta se define matemáticamente como:

$$L_{EC} = -\sum_{k=1}^{N} t_k \text{Log}(P_K) \tag{2.7}$$

Donde N corresponde al número total de clases, t_k representa la clase verdadera y p_k a la probabilidad de que la salida de la red sea la clase k. Como se puede inferir de esta ecuación, solo se tiene en consideración el error cuando se encuentra en la clase correcta, sin importar los errores entre clases, ya que el t_k se hace igual a cero en los otros casos. Mientras mayor es la probabilidad de la clase correcta, más bajo es el error y por tanto, más ha aprendido la red.

VII. Validación cruzada *K-fold*

Este método de entrenamiento es ampliamente utilizado para entrenar y ajustar parámetros de modelos basados en *machine learning*. *K-fold* consta de separar los datos de entrenamientos en k sub-conjuntos de entrenamiento y validación, procurando que los distintos conjuntos de validación no se solapen entre sí. Con esto, se puede entrenar el clasificador con distintos subconjuntos y evaluar su rendimiento, permitiendo optimizar de forma más transparente los parámetros en busca de un sistema con mayor capacidad de generalización. Generalmente es utilizado cuando la base de datos es pequeña, por lo que realizar una división tradicional de entrenamiento y test no es representativa [64].

2.3. Interacción Humano Robot (HRI) para dificultad respiratoria

En esta sección se presentarán antecedentes y mecanismos sobre como generar una conexión para potenciar la interacción Humano-Robot en la estimación de dificultad respiratoria. En el estado del arte prácticamente no existen trabajos que se centren en este tema, por lo que principalmente se detallarán cada uno de estos tópicos por separado.

2.3.1. Antecedentes y usos de robots sociales en salud

La interacción humano-robot o HRI, es un tema de investigación altamente estudiado en la actualidad, presentando grandes desafíos por la diversidad y complejidad de sus soluciones [65].

Como ya se comentó en la sección anterior, la investigación de estimación de dificultad respiratoria ha vivido un crecimiento abrupto en los últimos años debido a la pandemia del COVID-19, específicamente de forma automática reduciendo así la interacción entre personas (para evitar contagios). Y aunque los confinamientos provocados por la pandemia parecen haber llegado a su fin, los problemas en los centros sanitarios persisten, como lo son la falta de suministros, la escasez de profesionales y el crecimiento de la población vulnerable.

Si bien los robots sociales ya se han estado inmiscuyendo en el ámbito sanitario debido a estos problemas [16]. En general es realizando trabajos administrativos o de cuidado de niños, adultos mayores y personas con movilidad reducida [17] [18]. Aunque los robots brindan mucha ayuda en esta área, no se ha analizado en detalle otras aristas donde puedan aportar, como por ejemplo, que sean ellos los encargados de atender, derivar y/o tratar pacientes, lo que ayudaría considerablemente al descongestionamiento de los centros médicos. Sorprendentemente la estimación de dificultad respiratoria apoyada por robots sociales no ha sido estudiada en el estado del arte, siendo que los trabajos centrados afecciones respiratorias y en HRI, tuvieron un crecimiento importante de forma individual en los últimos años.

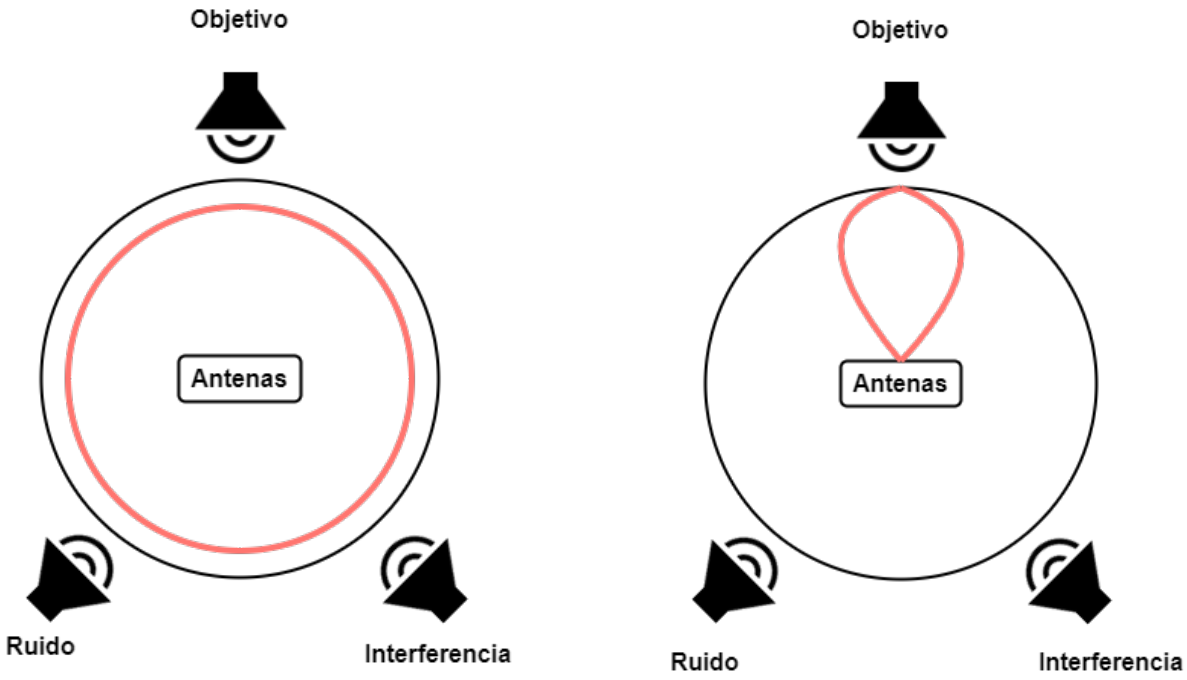
Como ya se ha mencionado, varios estudios postulan la voz como una importante fuente de información para estimar los trastornos respiratorios y caracterizar el perfil físico de las personas. Por ello, debe tenerse en cuenta lo susceptible que es esta a condiciones externas, como el ruido aditivo (*noisy cocktail effect*), la reverberación y/o el movimiento de los interlocutores. Estos efectos generan daños y cambios en la características del audio a lo largo del tiempo, afectando así aplicaciones que utilizan como entrada a la señal de voz. Debido a lo anterior, surgen técnicas de mejora de la calidad de voz o en inglés *speech enhancement*, los que permiten reducir el efecto del ruido, señales de voz interferentes y la revebereración, logrando así tener una señal de voz más limpia y focalizada en el objetivo (alguna persona o dirección en particular) [66].

Cuando se tiene un arreglo de micrófonos, es posible utilizar la información de estos para hacer filtrado espacial (*beamforming*), permitiendo obtener una mayor ganancia en la dirección deseada [67].

2.3.2. *Beamforming*

El *beamforming* o filtrado espacial es un método clásico de *speech enhancement*, el cual utiliza las muestras de un arreglo de sensores para mejorar la señal objetivo por sobre la señal recibida, la cual puede venir con ruido y/o señales interferentes. Para esto, se combina la información del conjunto de antenas de tal forma que para distintos ángulos existe una interferencia constructiva mientras que para otros hay una interferencia destructiva entre las señales que recibe cada micrófono, de esta forma se aumenta la directividad y ganancia del conjunto de sensores en la dirección de la señal objetivo [67].

La Figura 2.13.a muestra la diferencia entre un patrón de recepción omnidireccional que capta información en igual intensidad para todas las direcciones perpendiculares a un eje (recibiendo señal objetivo como ruido en igual magnitud), con un arreglo de micrófonos que puede orientar su directividad en función de la fuente objetivo (Figura 2.13.b), reduciendo así el efecto del ruido y/o señales interferentes.



(a) Patrón de captación omnidireccional.

(b) Patrón de captación con directividad.

Figura 2.13: Patrón de captación de antenas

Uno de los principales problemas que se presentan al momento de realizar *beamforming*, es que no necesariamente se conoce la ubicación y/o ángulo en la que se encuentra la fuente objetivo, el ruido o las señales interferentes.

I. Ángulo de dirección de arribo (DOA)

Como se había comentado anteriormente, el ángulo de la dirección de arribo (DOA, por sus siglas en inglés *Direction of Arrival*), es información muy importante para realizar el filtrado espacial, donde en algunos casos debe ser estimado o en otros es utilizado como dato. En la Figura 2.14, se muestra el arreglo lineal de micrófonos utilizado en este trabajo (Microsoft Kinect 360), donde se puede observar la dirección de llegada (DOA) con respecto al eje principal de captación del arreglo (MRA), lo que corresponde al ángulo θ. El objetivo del *beamforming* es "apuntar" la captación del arreglo en una dirección deseada, alineándolo con el DOA lo máximo posible.

Además, en la misma Figura 2.14 se puede apreciar la existencia de distintos desfases para cada micrófono, lo que se debe a que al estar en diferentes posiciones en el plano, algunos micrófonos reciben la señal antes y otros después, dependiendo de la dirección del ángulo de arribo.

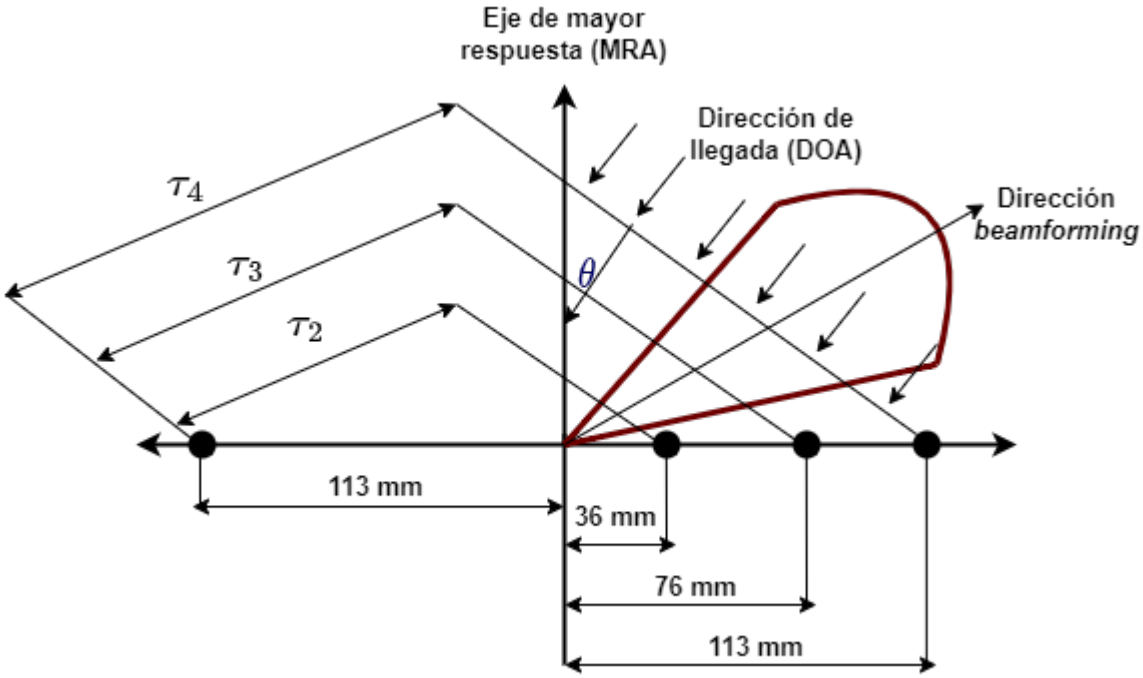


Figura 2.14: Geometría del arreglo lineal de micrófonos de la Microsoft Kinect, además se representa el MRA, DOA y dirección del *beamforming*.

II. Estimación de retardos

El desfase de cada señal viene dado por la siguiente ecuación [68].

$$\tau_l = \frac{\Delta_l \cdot sin(\theta)}{c} \tag{2.8}$$

Con Δ_l como la distancia entre el micrófono l y el micrófono de referencia, el ángulo θ corresponde al ángulo de dirección de arribo. Finalmente c es la velocidad de propagación del sonido en el medio.

A continuación, se explicarán dos de los métodos de *beamforming* más tradicionales y utilizados, *Delay-and-sum* y MVDR.

III. *Delay-and-Sum*

Este tipo de *beamforming* ha sido ampliamente utilizado en diferentes estudios [69], el cual consiste en que dado un DOA (Figura 2.14), se toman las señales obtenidas de cada micrófono y son desfasadas con el objetivo de alinearlas, es decir, ponerlas en fase. Posteriormente, las señales de los distintos canales son sumadas para realizar una suma constructiva de la señal objetivo (dada por el ángulo del DOA) y una suma destructiva para el ruido o señales interferentes provenientes de otra dirección distinta a la objetivo. Este proceso puede ser representado por la ecuación 2.9.

$$y(t) = \sum_{l=0}^{L-1} x_l(t - \tau_l) \tag{2.9}$$

Donde L representa el número de micrófonos, x_l la señal recibida por el micrófono l y τ_l

al desfase del micrófono l (calculados con la ecuación 2.8).

IV. MVDR

Respuesta sin distorsión de varianza mínima o MVDR (por sus siglas en inglés, *Minimum Variance Distortionless Response*), es un algoritmo de *beamforming* tradicional, que tiene como objetivo (tal como dice su nombre) en reducir al mínimo la potencia de las señales interferentes y del ruido, para mejorar la calidad de la señal objetivo [70].

Dada una señal ruidosa $y(t) = [y_1, y_2, .., y_L]$ (con el subíndice representando el micrófono correspondiente), esta puede ser descompuesta en la suma de una señal limpia con una señal de ruido, tal como en la ecuación 2.10.

$$y(t) = s(t) + n(t) \tag{2.10}$$

Donde s(t) representa a la señal de voz objetivo y n(t) al ruido. Al pasar al dominio tiempo-frecuencial, el vector pasa a ser una matriz Y(t,f), donde t representa a los *frames* (tiempo) y f a los *bins* (frecuencias). La señal de salida mejorada que retorna el MVDR puede ser representada con la siguiente ecuación.

$$\hat{Y}(t, f) = W(f)Y(t, f) \tag{2.11}$$

Donde W corresponde a los pesos del MVDR, los que son invariantes a lo largo del tiempo al solo tener dependencia de f, es decir, de los *bins* frecuenciales. MVDR busca minimizar la varianza del ruido sin causar distorsiones en la señal objetivo, lo que matemáticamente se define como [71]:

$$W_{\text{MVDR}} = \arg\min{}_W W^H \Phi_{NN} W \quad \text{s.a. } W^H v = 1 \tag{2.12}$$

Con ^{H} como el operador Hermético y v igual al *steering vector*, el cuál se puede representar matemáticamente como:

$$v(f) = [e^{-jw\tau_0} e^{-jw\tau_1} ... e^{-jw\tau_n}]^T \tag{2.13}$$

En la cual los τ son los desfases de los micrófonos que se calculan con la ecuación 2.8. Aunque existen varias soluciones para encontrar los pesos del MVDR, en este trabajo se utiliza la solución de la ecuación 2.12, basada en el *steering vector* [72]. Por su parte, los pesos del MVDR calculan como:

$$W = \frac{\Phi_{nn}^{-1} v(f)}{v^H(f) \Phi_{nn}^{-1} v(f)} \tag{2.14}$$

En la ecuación anterior, Φ_{NN} equivale a la matriz de covarianza del ruido, la cual puede ser estimada (en su caso más simple) como el promedio de $N(t, f)N^H(t, f)$. La complicación surge en obtener la señal de solo ruido a partir de una señal ruidosa, donde para solucionar este problema, se buscan segmentos de solo ruido y al concatenarlos se asume que se obtiene n(t), basados en la supuesta estacionaridad del ruido a lo largo del tiempo.

En este trabajo se calculó una matriz Φ_{NN} semi-dependiente del tiempo, en la que para las zonas de ruido se utilizó la media de las matrices de covarianza instantáneas, y para las zonas de habla se utilizó una interpolación entre la matriz de covarianza antes y después del

ruido.

2.3.3. Inteligencia artificial para *speech enhancement*

Con el auge de la inteligencia artificial, las principales propuestas del estado del arte sobre *beamforming* se basan en *deep learning* [73]. Los estudios proponen el uso de redes neuronales para estimar máscaras de tiempo-frecuencia, con las que se pueden calcular matrices de covarianza necesarias para el uso de *beamforming* [74] [75]. Por otro lado, el estudio del enmascaramiento mono-canal ha tenido mejoras considerables en cuanto a resultados, tanto en redes que trabajan en el dominio de la frecuencia como con algoritmos temporales (TasNets), que tienen la ventaja de evitar la sincronización de fases al reconstruir las señales [76] [77]. Este desarrollo mono-canal ha dado lugar a trabajos que proponen realizar *speech enhancement* en cada canal y posteriormente aplicar filtrado espacial, buscando reducir el efecto de los artefactos generados por las máscaras [78] [79] [80]. Otros estudios proponen realizar la mejora del habla multi-canal sin necesidad de *beamforming*, ya sea utilizando *autoencoders* [81] [82], redes neuronales gráficas [83] o redes completamente convolucionales [84].

En general, la evaluación de los sistemas propuestos en el estado del arte se realiza sobre bases de datos simuladas, lo que complica el análisis de la escalabilidad del modelo al mundo real. Además, es aún más complejo encontrar estudios que examinen el rendimiento de los modelos en condiciones dinámicas para bases reales, mientras que en HRI este problema es habitual. También hay excepciones, por ejemplo en [85] [68] [86] se evalúa el rendimiento de los sistemas en condiciones reales, además de intentar reducir la brecha entre los resultados para datos reales y simulados, aunque el principal objetivo de estos estudios es obtener mejoras a nivel de ASR.

De las soluciones anteriores, en este trabajo se implementó el *Complex filter estimator* de la propuesta de ADL-beamforming del trabajo [71]. Esta decisión se basó en las ventajas que presenta su arquitectura frente a los datos que se dispone para la realización de esta tesis. Como lo es el tener de entrada al sistema múltiples canales y el ángulo de incidencia de la señal objetivo. Que el DOA sea dato, permite reducir la complejidad del *beamforming* propiamente tal y su respectiva implementación. Para entender esta solución, a continuación se detallarán partes importantes de su arquitectura y se explicarán en detalle sus bloques.

I. ADL-beamforming

Esta red fue propuesta en [71] y la arquitectura se muestra en la Figura 2.15.

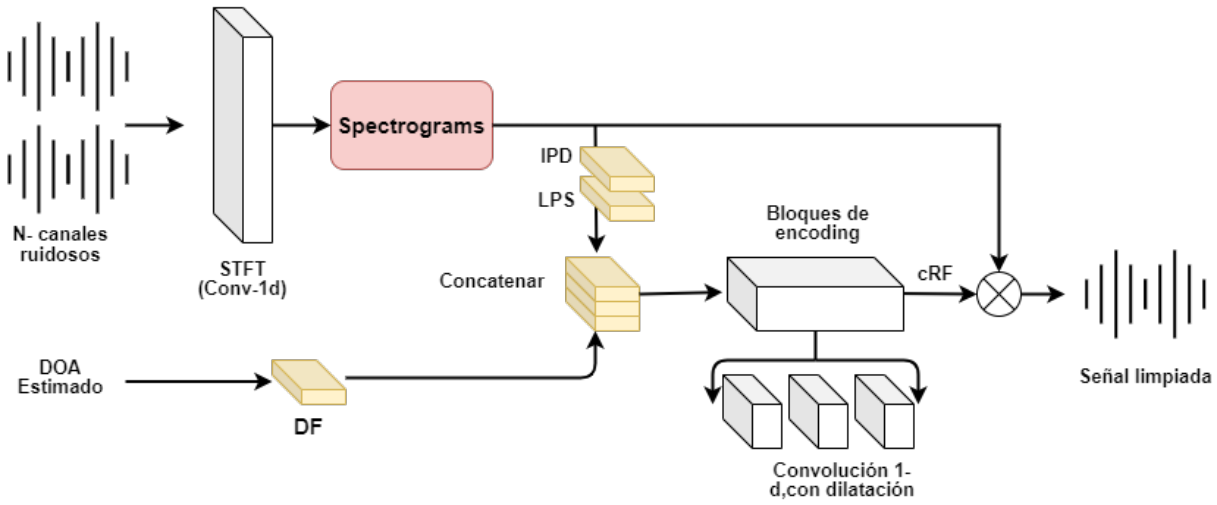


Figura 2.15: Arquitectura de *speech enhancement* ADL-Beamforming .

La arquitectura de la Figura 2.15, corresponde al estimador de filtros complejos o cRF. El sistema recibe dos entradas, primero se encuentra la señal multi-canal (debido a la existencia de varios micrófonos) y el ángulo de incidencia. A estas entradas se les debe realizar la extracción de las siguientes características.

***Interchannel Phase Difference* (IPD)**

Esta característica representa el desfase existente entre los espectrogramas de los distintos micrófonos y un micrófono de referencia, matemáticamente se muestra en la siguiente ecuación [87]:

$$\phi_{i,tf} = \angle \frac{y_{i,tf}}{y_{0,tf}} \tag{2.15}$$

Donde y_i representa el micrófono i, tf indica el punto del espectrograma para el *frame* t y el *bin* f. Por su parte, el índice 0 indica el micrófono de referencia.

Espectrograma logarítmico de potencia (LPS)

Esto corresponde simplemente a calcular el logaritmo del espectrograma.

***Directional Feature* (DF)**

El DF se calcula a partir del DOA, que en el caso de este trabajo se asume conocido. Esta característica se define como la distancia coseno entre el *steering vector* y espectro complejo de cada canal normalizado con el canal de referencia [88]. La fórmula matemática para encontrar su valor es:

$$\mathrm{DF} = \sum_{i}^{M} \frac{v_n^{i,f} \frac{y_{i,tf}}{y_{0,tf}}}{\left| v_n^{i,f} \frac{y_{i,tf}}{y_{0,tf}} \right|} \tag{2.16}$$

Donde el $v_n^{i,f}$ corresponde al *steering vector* de la ecuación 2.13 para el micrófono i y el hablante n (no relevante en este trabajo, ya que no se separan hablantes). M corresponde al total de micrófonos.

Siguiendo el diagrama de la Figura 2.15, una vez se calculan todos las características, estas deben ser concatenadas y son introducidas en bloques de codificación, que corresponden a una variación de la Conv-TasNet [89].

Conv-TasNet

Este bloque tiene como objetivo calcular las máscaras para distintos hablantes y/o el ruido. A continuación, se muestra el esquema de la Conv-TasNet.

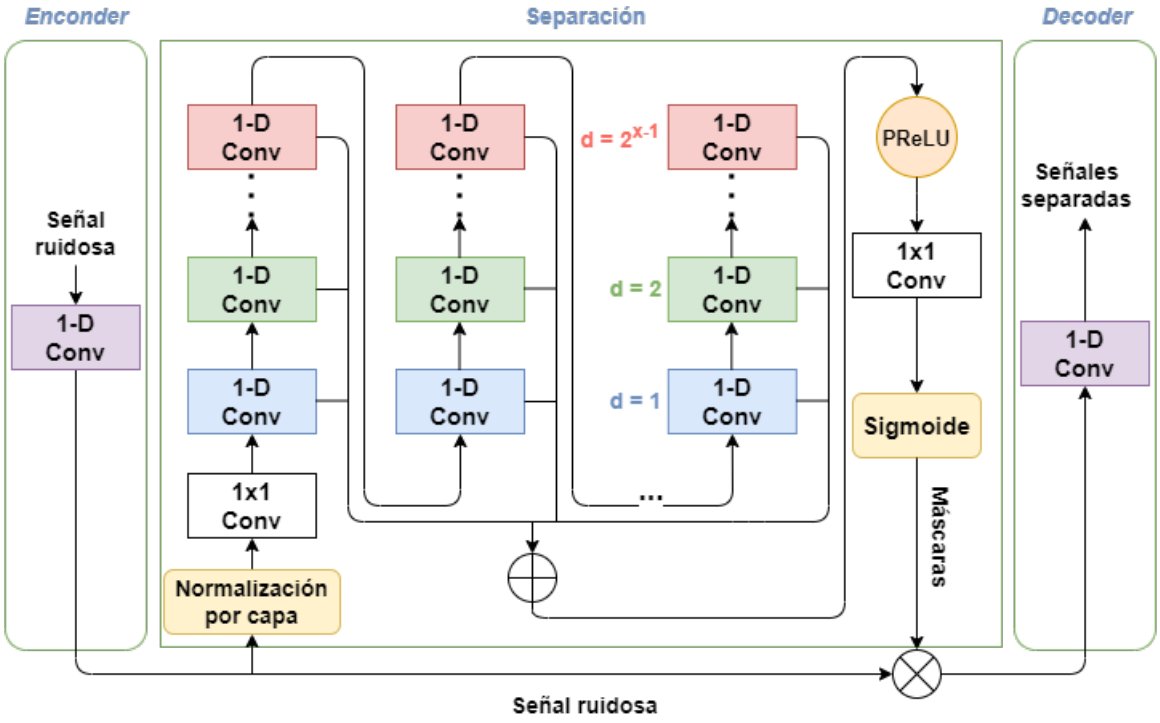


Figura 2.16: Diagrama de bloques de la Conv-TasNet.

Como se aprecia en la Figura 2.16, esta red le aplica una convolución de 1 dimensión a la entrada. El resultado de esta convolución, por un lado entra al bloque de separación, y por otro, se multiplica con la salida de dicho bloque. Cuando entra al bloque de separación, se aplica una normalización por capa y es propagada dentro de una serie de bloques convolucionales de una dimensión (cada salida de los bloques se van sumando para la salida), luego los bloque convolucionales finales presentan una convolución dilatada con distintas tasas, las que aumentan en potencias de 2. La suma de todos estos bloques pasan por una función de activación PReLU, una capa convolucional 1x1 y una sigmoide, lo que da por resultado las máscaras buscadas. La entrada que había sido convolucionada y que no entró al bloque de separación, se multiplica con las máscaras obtenidas, así para encontrar las fuentes separadas.

La Figura 2.17 muestra como se compone cada bloque convolucional, donde destaca la existencia de capas convolucionales 1x1, funciones de activación PReLU y conexiones resi-

duales. La normalización utilizada en corresponde a una normalización global de capa (gLN, por sus siglas en inglés *global layer normalization*)

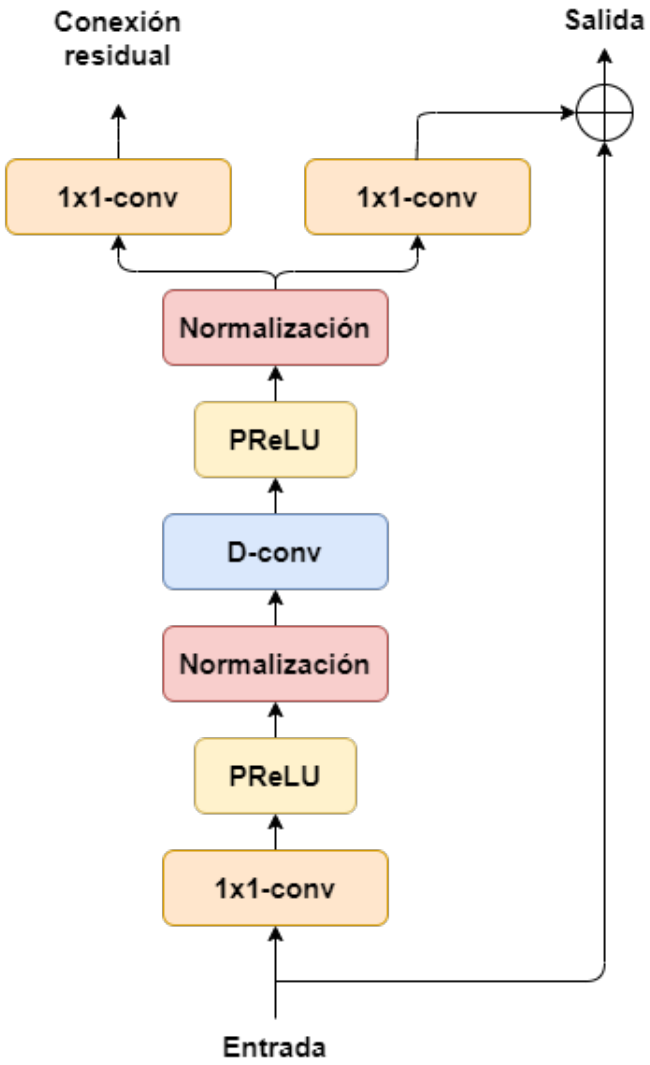


Figura 2.17: Bloque 1-D convolucional de la Conv-TasNet.

Estimación de filtros complejos (CRF)

Este filtro es propuesto en [90], y en la solución del ADL-*beamforming* se le aplica a las máscaras obtenidas de los bloques de *encoding*, con el objetivo de estabilizar y hacer más robustas las matrices de covarianza de la voz y del ruido.

Como se ve en la Figura 2.18, las máscaras tradicionalmente funcionan punto a punto, es decir, un punto de la entrada (espectrograma en este caso) debe ser multiplicada con un punto de la máscara para obtener un punto de la salida. En el caso de cRF, para un punto de la salida se utiliza el punto correspondiente en tiempo y frecuencia, pero además se incluye una vecindad cercana, símil al *kernel* de la convolución.

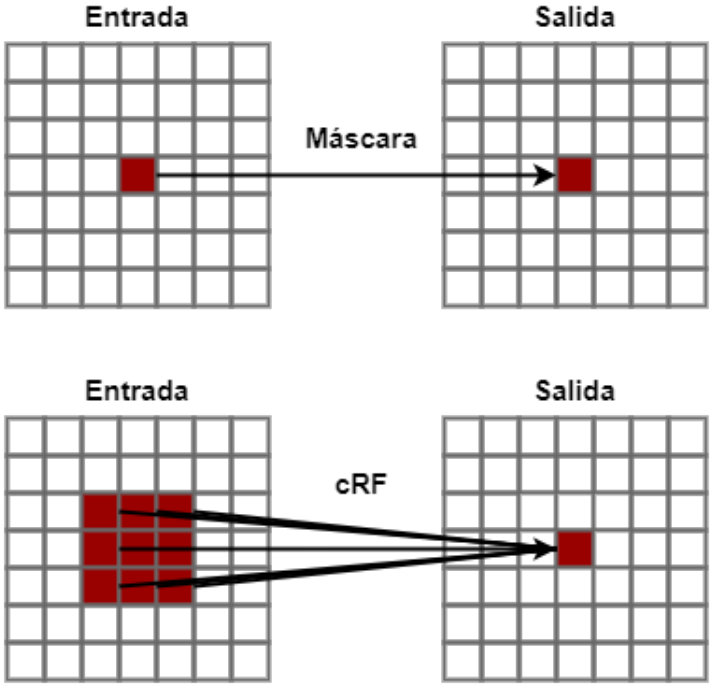


Figura 2.18: Enmascaramiento tradicional y enmascaramiento por filtros.

2.3.4. Modelamiento del canal acústico

Con el fin de reducir la distancia entre resultados simulados y los reales, es necesario tener un entrenamiento robusto que permita a los modelos generalizar, por lo que generar un modelo acústico lo más parecido a un entorno real es esencial. Métodos tradicionales utilizados con este objetivo es tener distintos tipos de ruidos para entrenar, múltiples respuestas impulsivas y/o diferentes hablantes. Además, el utilizar distintos niveles de ruido en términos de SNR es esencial para tener un entrenamiento con múltiples condiciones [91] [92] [86].

Para modelar acústicamente un canal es necesario incluir sus respuestas impulsivas, las que permiten representar la respuesta que tendrá el sistema frente a una entrada. Para este trabajo, el sistema corresponde a la habitación de grabación donde se evaluará el método propuesto.

I. *Room Impulse Response* (RIR)

Una respuesta impulsiva corresponde a la salida de un sistema cuando se le ingresa un impulso o delta de Dirac. En particular las RIRs corresponden a la respuesta que da una habitación frente a los impulsos, lo que permite modelar la sala mediante su función de transferencia.

El tener RIRs es útil para simular datos, ya que si se tiene una señal limpia, al convolucionarla con una RIR, se tendrá el equivalente al haber reproducido la señal en la habitación correspondiente a la RIR, incluyendo su reverberación y atenuaciones. Claramente esta respuesta impulsiva cambiará dependiendo de las dimensiones físicas y los materiales de construcción de la habitación donde hayan sido calculadas o simuladas, además de la posición en la que estén ubicadas las fuentes. En la Figura 2.19, se muestra la respuesta impulsiva de

una habitación en función del tiempo.

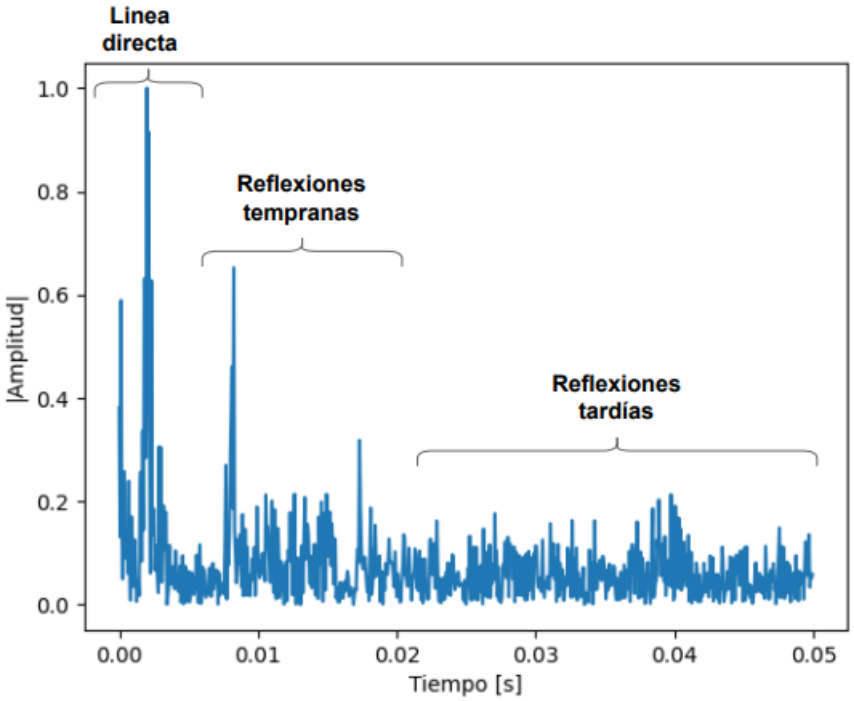


Figura 2.19: Valor absoluto de la respuesta impulsiva de la habitación en función del tiempo.

Como se observa de la Figura anterior, la linea de visión directa corresponde a cuando la magnitud de la respuesta impulsiva es máxima. Luego aparecen las reflexiones tempranas, que equivalen a señales que llegan tras pocas reflexiones al micrófono. Finalmente están las reflexiones tardías, que son las que provocan la reverberación, donde la señal "rebota" muchas veces antes de llegar al objetivo.

En base a su definición, se torna prácticamente imposible grabar una respuesta impulsiva real, ya que reproducir un delta de Dirac (de energía infinita) es imposible. Por lo mismo, existen diferentes métodos para grabar RIRs en condiciones reales.

II. Método de barrido sinusoidal de Farina

Este método permite la obtención de RIRs en entornos débilmente no-lineales, los cuales pueden ser aproximados a sistemas invariantes en el tiempo. Esta algoritmo se basa en la realización de un barrido de frecuencias de forma exponencial. Es aplicable a altavoces y componentes de audios, pero también para modelar acústicamente salas [93].

En la Figura 2.20, se muestra el espectrograma del barrido de frecuencias, donde se aprecia claramente como la frecuencia reproducida crece exponencialmente a lo largo del tiempo, desde los 0 Hz hasta los 8.000 Hz.

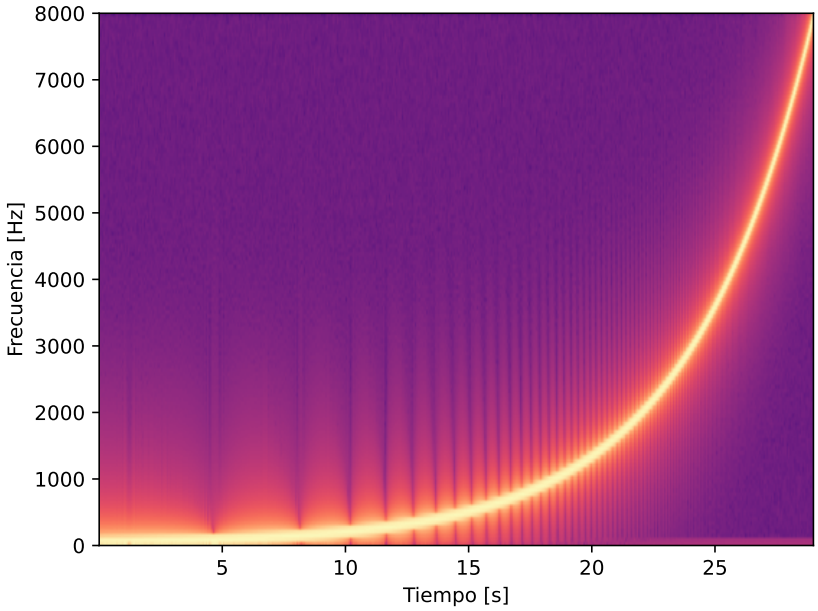


Figura 2.20: Espectrograma del barrido de frecuencias utilizado para grabar RIRs con Farina.

2.3.5. Reconocimiento automático de voz

El reconocimiento automático de voz o ASR (por sus siglas en inglés, *Automatic Speech Recognition*) es un sistema que a partir de una señal de voz, retorna un texto correspondiente a la transcripción de la señal de entrada. Es fundamental en aplicaciones que utilicen como entrada una señal de voz, ya que debe ser capaz de detectar cuando inicia y termina la información importante, evitando añadir datos que causen ruido a los sistemas. El funcionamiento de un ASR se basa en 4 etapas [94]:

- Extracción de características: Este ítem tiene una relevancia muy importante, ya que el reconocimiento de voz depende en gran parte de la calidad de las características obtenidas, ejemplos de estas son el LPC, MFCC, ΔMFCC, etc.
- Modelo acústico: Esta corresponde a la etapa principal del ASR, aquí se detecta el fonema hablado gracias a las características extraídas anteriormente. La técnica dominante para modelar esta etapa es por medio de modelos ocultos de Markov (HMM).
- Modelo de lenguaje: El modelo lingüístico es el componente que se encarga de evaluar la búsqueda de la palabra correcta a partir de la secuencia de palabras anteriores, esto se hace encontrando la palabra que maximice las probabilidades dada la secuencia anterior.
- Reconocimiento: Una vez ya se le han aplicado todos los pasos anteriores a la señal entrante, se puede retornar la frase predicha. La medición de la efectividad de esta predicción se hace por medio del *Word Error Rate*

I. *Word Error Rate* (WER)

El *Word error rate* o WER por sus siglas, corresponde a la medida que evalúa el desempeño del reconocedor de voz. En lineas generales, corresponde al número de errores sobre el total de palabras. Matemáticamente se tiene:

$$\text{WER} = \frac{\text{Sustituciones} + \text{Inserciones} + \text{Supresiones}}{\text{Total de palabras}} \tag{2.17}$$

Las sustituciones corresponden a cuando una palabra es reemplazada por otra, una inserción es cuando se agrega una palabra que no había sido dicha en la señal original, y por último, una supresión es cuando se borra de la transcripción una palabra que si estaba en la señal de voz.

Capítulo 3

Estimación de dificultad respiratoria en HRI

En el presente capítulo se describirá el sistema propuesto para estimar la dificultad respiratoria en un entorno de interacción Humano-Robot. Este sistema se compone de dos bloques principales, el primero consta del *speech enhancement*, seguido por el bloque de estimación de dificultad respiratoria en sí. El objetivo del sistema es que sea capaz de recibir una señal de voz distorsionada por diferentes factores como el ruido o la reverberación, y mejorar su calidad a través de técnicas de filtrado espacial en el primer bloque, para posteriormente enviar esta señal mejorada al bloque de estimación de dificultad respiratoria, que devolverá al usuario su nivel de disnea.

En el capítulo, se explicará en primer lugar la implementación del entorno HRI, incluyendo la recopilación de la base de datos y cómo se realizó la regrabación en una plataforma robótica bajo diferentes condiciones de ruido y movimiento. En la segunda parte del capítulo se detalla la configuración y el diseño del sistema de estimación de disnea, específicamente los subbloques de *speech enhancement* y estimación de dificultad respiratoria.

3.1. Implementación de entorno HRI

En esta sección se profundizará en la obtención de la base de datos telefónica, incluyendo la cantidad de personas que la conforman, los sonidos que se deben reproducir y el tipo de afección y nivel de disnea que presentan. También se describirá la plataforma robótica en la que se regrabaron los audios, así como los distintos entornos de movimiento y ruido utilizados.

3.1.1. Recolección de base de datos

La base de datos está compuesta por pacientes con enfermedades respiratorias (EPOC, fibrosis pulmonar, COVID-19) reclutados en el Hospital Clínico de la Universidad de Chile (HCUCH) y voluntarios sanos de la Facultad de Ciencias Físicas y Matemáticas (FCFM) de la misma universidad.

El estudio fue aprobado por los comités de ética científica del HCUCH y de la FCFM. Quienes fueron incluidos en la base de datos debieron dar su consentimiento informado para participar en el estudio. Posterior a aceptar el consentimiento, fueron entrevistados por un neumólogo del HCUCH, quién evaluó el grado de disnea mediante la escala mMRC (*Gold Standard*). La puntuación mMRC de cada participante se utilizó como la referencia para

entrenar el sistema de *machine learning*.

A los individuos se le solicitaba que respirarán profundamente y realizarán 3 tipos de vocalizaciones controladas sin pausas hasta quedar sin aire. La primera vocalización es la secuencia de fonemas españoles /a/ y /e/, denotada como /ae-ae/; la siguiente es la secuencia de sílaba española /sa/, indicada como /sa-sa/; y la última se inspiró en el Test de Roth [32] donde se pedía a los sujetos que contaran en español del uno al 30 tan rápido como pudieran, denotada como conteo. De la vocalización /ae-ae/ se obtiene información muy similar a la de una vocal sostenida (secuencia continua) pero evitando el problema de la atenuación causada por el esquema de supresión de ruido de los teléfonos inteligentes, ya que la señal de voz correspondiente es menos estacionaria que la de una sola vocal sostenida como /a/. La secuencia /sa-sa/ tampoco es estacionaria (evitando supresión telefónica), esta debe repetirse lo más rápido posible, generando una tasa de exhalación del volumen de aire mayor que en el caso de /ae-ae/, ya que las cuerdas vocales están distendidas cuando se produce el fonema sordo /s/, en contraste con los fonemas sonoros como en /ae-ae/. Como la frecuencia de corte del canal telefónico de 4 kHz reduce drásticamente la amplitud de la muestra de /s/, la utilización de la vocal /a/ en la secuencia /sa-sa/ permite detectar la señal de forma óptima. La señal resultante del conteo del uno a 30 es altamente no estacionaria y permite una mejor representación del comportamiento espontáneo del usuario, como pausas, cambios de entonación, velocidad de habla, etc., mientras realiza la vocalización. Obsérvese que estas vocalizaciones controladas evitan situaciones o comportamientos forzados, como la tos.

La base de datos esta compuesta por 100 participantes, de los cuales 34 correspondían a individuos sanos y 66 eran pacientes con afecciones respiratorias (39 EPOC, 22 Fibrosis Pulmonar y 5 secuelas de COVID-19). A los participantes sanos se les asignó un mMRC igual a cero. Los pacientes fueron evaluados clínicamente con respecto a la puntuación mMRC resultando 18 con mMRC igual a 1, 27 con mMRC igual a 2, 19 con mMRC igual a 3 y 2 con mMRC igual a 4. Estas puntuaciones se obtuvieron mediante evaluación clínica (*gold standard*), y fueron empleadas como referencia para entrenar los modelos basados en *deep learning*. El número de pacientes con mMRC igual 4 era demasiado bajo (tan solo 2), lo que dió lugar a una clase sub-representada. En consecuencia, estos sujetos se incorporaron al subconjunto de individuos con mMRC 3, dando lugar a cuatro clases con la puntuación mMRC comprendida entre 0 y 3, donde el nivel 3 corresponde al estado de disnea más grave en nuestro caso.

Para obtener las grabaciones de las fonetizaciones, se contactó a las personas por teléfono con un sistema IVR. Se pidió a las personas que repitieran cada vocalización dos veces siguiendo el procedimiento antes mencionado. Los audios obtenidos se almacenaron en formato WAV con una frecuencia de muestreo de 8 kHz y se les asignó un ID aleatorio para proteger la identidad de los participantes. Como la base de datos está compuesta por 100 personas, cada fonetización tiene 200 audios (dos repeticiones por individuo) y el conjunto de datos total alcanza las 600 vocalizaciones. Tras grabar los audios, se entrenó un sistema de reconocimiento automático del habla (ASR) para aislar las vocalizaciones objetivo del ruido de fondo o del audio no deseado.

3.1.2. Plataforma robótica

Para realizar la grabación de la base de datos en HRI, se implementó un *testbed* similar al utilizado en [86], donde se utilizó el Robot Personal 2 o PR2. En la cabeza del PR2 se instaló un Microsoft Kinect 360, que posee en un arreglo lineal de 4 micrófonos y tres cámaras (infrarroja, profundidad, RGB). Al disponer de un arreglo de micrófonos, el audio grabado resulta de 4 canales, lo que permite la realización de *beamforming* para la señal grabada.

Las 600 señales de voz se grabaron en una sala con un volumen de 104 m^3 y un tiempo de reverberación medido de 0,5 segundos, la cual tiene la geometría que se muestra en la Figura 3.1. Dentro de esta sala se montó la plataforma, que se compone de una fuente de voz y una fuente de ruido, separadas por 45° al medir desde la ubicación P1 (a dos metros de la fuente de voz), que es donde se encontraba el robot al momento de grabar los audios.

Se grabaron dos tipos de escenarios, un caso estático, en el que el robot se encontraba en P1 con la cabeza fija, y el caso dinámico, en el que PR2 se encontraba en P1 girando la cabeza. Estos escenarios permiten acercarse lo más posible a un ambiente HRI tradicional, donde además de ser grabado en condiciones reales, también cuenta con la presencia de ruido, reverberación y movimiento.

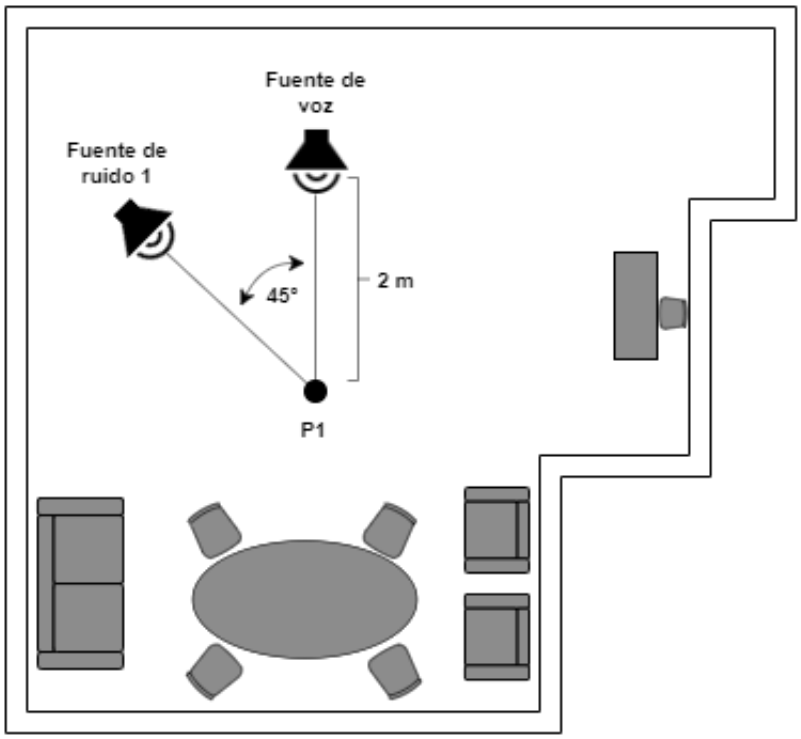


Figura 3.1: Plano de planta de habitación donde se instaló la plataforma robótica.

3.1.3. Base de datos en escenarios HRI

Se grabaron tres bases de datos diferentes en la plataforma robótica descrita anteriormente, en los que hay dos casos estáticos y uno dinámico. Las distintas grabaciones HRI son

resumidas en la Tabla 3.1. Para los casos estáticos (denotados como *static*), el robot está inmóvil en P1 y con la cabeza fija; en *static* 1, el robot está a 0° respecto de la fuente de voz y en *static* 2 a 45°, es decir, mirando directamente a la fuente de ruido.

En la condición dinámica (denominada *dynamic* 1), el robot también se encuentra fijo en P1, pero mueve su cabeza con una velocidad angular constante de 0,42 rad/s, en un rango de movimiento de -50° a 50° (Figura 3.2), por lo que el ángulo de incidencia del habla y del ruido cambian constantemente.

Se adoptó una SNR de 10 dB para la grabación HRI de la base de datos. Para estimar la potencia de las señales del habla, se colocó al PR2 a dos metros de la fuente del habla (P1) con la cabeza a 0° (mirando al frente) y se grabaron las 600 señales de la base de datos concatenadas. Por otro lado, para estimar la potencia del ruido, el robot se colocó en la misma posición, pero ahora se reprodujo solo un minuto de ruido de restaurante Aurora procedente de la fuente de ruido. Con ambas señales, se calculó la energía de las mismas y se ajustó iterativamente el volumen de los altavoces hasta obtener la SNR deseada de 10 dB.

Tabla 3.1: Bases de datos de evaluación.

Condición HRI	**Velocidad Angular [rad/s]**	**Ángulo cabeza**
static 1	0	0°
static 2	0	45°
dynamic 1	0.42	-

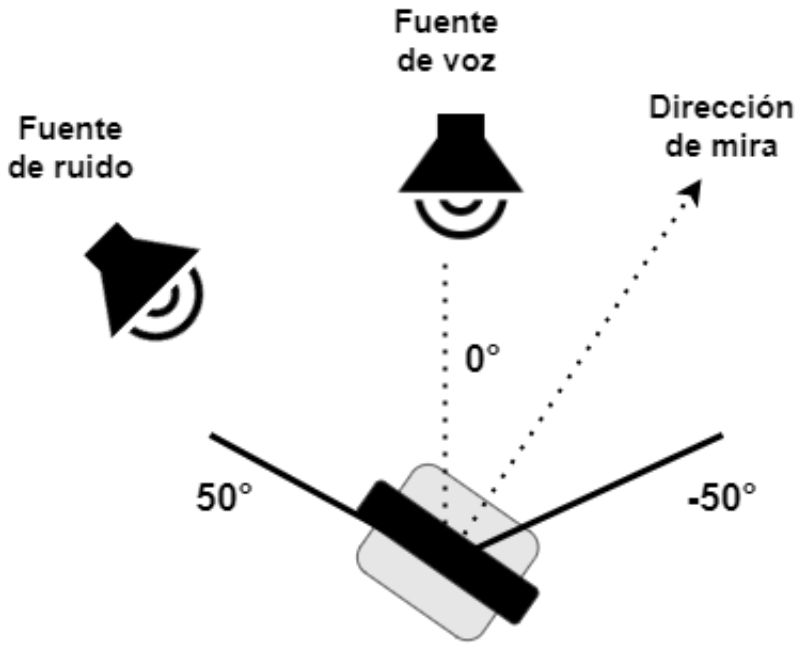


Figura 3.2: Rango de movimiento de la cabeza de PR2.

3.2. Sistema de estimación de la dificultad respiratoria en HRI

El diagrama de bloques del sistema propuesto se muestra en la Figura 3.3. Para utilizar los algoritmos de *speech enhancement* implementados, dicho módulo recibe tres entradas: la señal emanada por la fuente objetivo (voz), la señal de la fuente de ruido y el DOA. De esta forma es posible realizar el *speech enhancement* y limpiar la señal objetivo, la cual a su vez es utilizada como entrada para el sistema de estimación de la dificultad respiratoria. Finalmente, se obtiene la puntuación mMRC para el usuario.

Es importante marcar que el sistema de dificultad respiratoria se entrenó inicialmente con una base de datos telefónica, la cual tiene una frecuencia de muestreo de 8 kHz. Por el lado contrario, la Microsoft Kinect graba en 16 kHz, por lo que los distintos algoritmos de *speech enhancement* se aplicaron en dicha frecuencia de muestreo. Para solucionar esta inconsistencia, los audios previos a entrar a la red de estimación de disnea, fueron sub-muestreados para bajarlos a una frecuencia de 8 kHz.

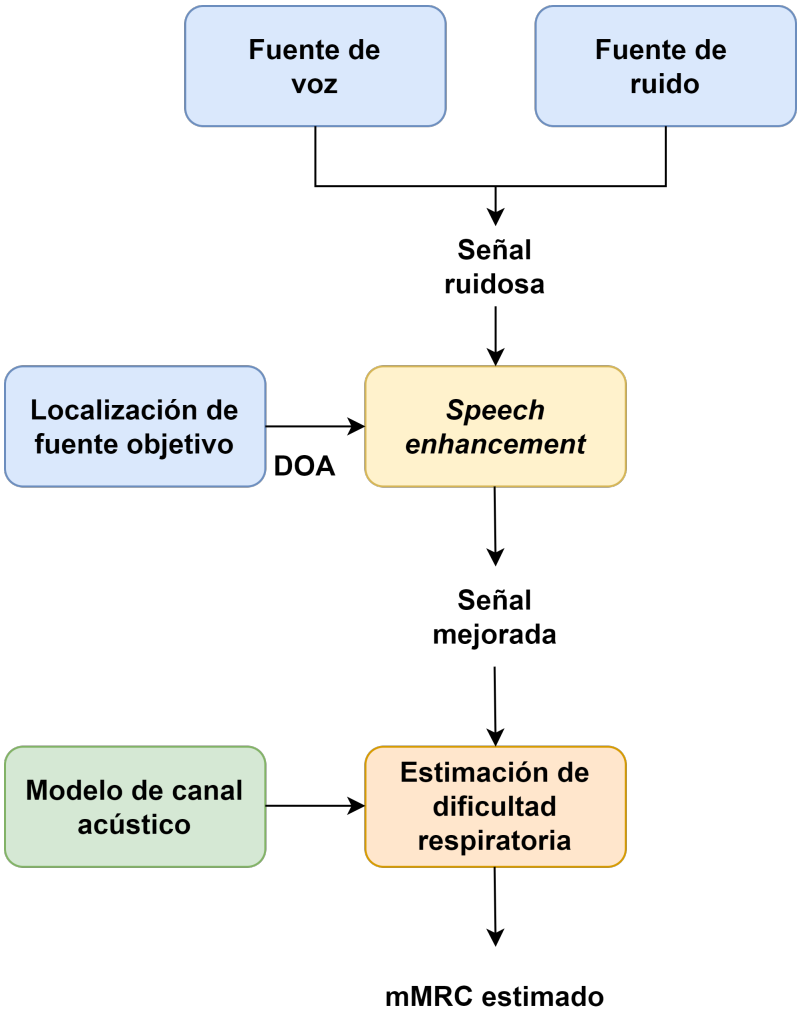


Figura 3.3: Sistema de estimación de dificultad respiratoria en HRI.

A continuación se detallarán los distintos bloques del sistema propuesto.

3.2.1. Localización de fuente objetivo

El PR2 guarda automáticamente el ángulo azimutal de su cabeza, lo que proporciona el ángulo de incidencia de la fuente de voz (DOA) durante la grabación. Este valor es considerado como dato para la realización de *speech enhancement* en este trabajo.

Claramente en otras aplicaciones no se puede tener el DOA a priori, por lo mismo el reconocimiento visual es ampliamente utilizado en el estado del arte para la estimación del ángulo de incidencia, o también suelen estimarse calculando las correlaciones cruzadas en una ventana de análisis [95].

3.2.2. *Speech enhancement*

El segundo bloque del sistema propuesto consiste en el mejoramiento de la señal de voz. En este caso, el objetivo es mejorar la calidad de la señal de voz objetivo, reduciendo la intensidad de las señales interferentes o el ruido externo. En este trabajo se evalúa el rendimiento del clasificador de estimación de dificultad respiratoria para tres tipos de *Speech enhancement*.

- *Delay-and-sum* (D&S)
- *Minimum Variance Distortionless Response* (MVDR)
- cRF

Además, se evalúa el rendimiento del sistema en el caso ruidoso, donde se toma el canal 1 del audio y este es introducido al módulo de estimación de disnea.

3.2.3. Módulo de estimación de la dificultad respiratoria basado en *Deep Learning*

El sistema propuesto pretende caracterizar el comportamiento de los usuarios al realizar vocalizaciones controladas para clasificar su nivel de disnea en la escala mMRC [96]. Como ya se ha mencionado, las vocalizaciones controladas se eligieron para proporcionar cierto grado de complementariedad entre ellas y contrarrestar el esquema de supresión de ruido de los teléfonos. Las fonetizaciones seleccionadas permiten representar el comportamiento fonético articulatorio espontáneo del usuario, como las pausas involuntarias, la variación del tono, el cambio en la velocidad del habla, tos o la respiración. Para ello, se definen características dependientes e independientes del tiempo y se extraen por separado de las señales de voz.

Las características dependientes del tiempo se calculan *frame* a *frame* e intentan captar la dinámica de las señales al representar el comportamiento espontáneo de los usuarios, tales como pausas, cambios de velocidad del habla, tos o respiración no voluntaria durante la elocución. Para las fonetizaciones /ae-ae/ y /sa-sa/, se usan filtros de Mel estimados a partir del espectro de potencia logarítmica FFT, en el caso del conteo, se utiliza el espectro de potencia logarítmica FFT. Por otro lado, las características independientes del tiempo tienen como objetivo representar globalmente las vocalizaciones, proporcionando información como la longitud de la fonetización y la variación y pendiente de la curva de la frecuencia fundamental.

Las características fueron escogidas y diseñadas cuidadosamente, donde una de las contribuciones del sistema es el hecho de que no se requiere forzar situaciones o comportamientos, como la tos no espontánea. Por el contrario, la propuesta se basa en fonetizaciones que pueden ser fácilmente replicables. Dado que las características dependientes e independientes del tiempo caracterizan el comportamiento de los usuarios con representaciones complementarias, su combinación debería dar lugar a un clasificador final más preciso y robusto. La figura 3.4 muestra el diagrama de bloques del sistema propuesto. Cada tipo de vocalización proporciona una softmax de cuatro dimensiones que representa la probabilidad de cada puntuación mMRC. Estas tres softmax obtenidas de cada fonetización se combinan con las siguientes cinco reglas: mínimo, máximo, media, mediana y producto. Para cada combinación se tiene una nueva softmax, por lo que para obtener la predicción final, estas son promediadas y la puntuación mMRC estimada corresponde a la probabilidad más alta.

La figura 3.5 muestra cómo se obtienen las softmaxs de cada vocalización. Hay dos clasificadores por tipo de fonetización, uno que recibe las características dependientes del tiempo y otro para los características independientes del tiempo. Los individuos repiten dos veces cada tipo de vocalización. Una vez extraídas las características dependientes e independientes del tiempo, se propagan a través del módulo de aprendizaje automático correspondiente, que genera softmax para cada repetición y tipo de característica. Para el clasificador dependiente del tiempo se emplea una arquitectura basada en CNN o LSTM y para el clasificador independiente del tiempo se hace uso de un esquema MLP. Las softmaxs independientes y dependientes del tiempo resultantes de cada repetición se combinan por separado utilizando el mismo esquema descrito anteriormente (Figura 3.4) con cinco reglas de combinación para obtener una única softmax por tipo de característica. A continuación, se combinan las softmax dependientes e independientes del tiempo mediante un promedio simple para obtener la softmax final de la vocalización. Este proceso se repite para cada fonetización para obtener la puntuación mMRC estimada, como se muestra en la Figura 3.4.

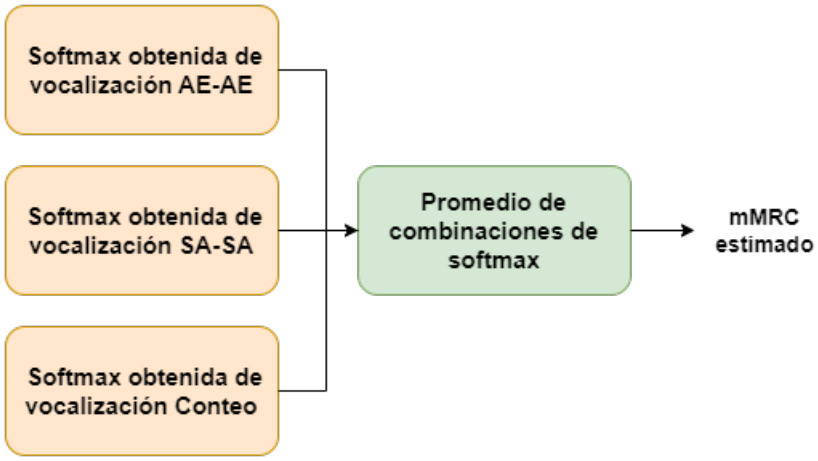


Figura 3.4: Sistema de estimación de dificultad respiratoria propuesto.

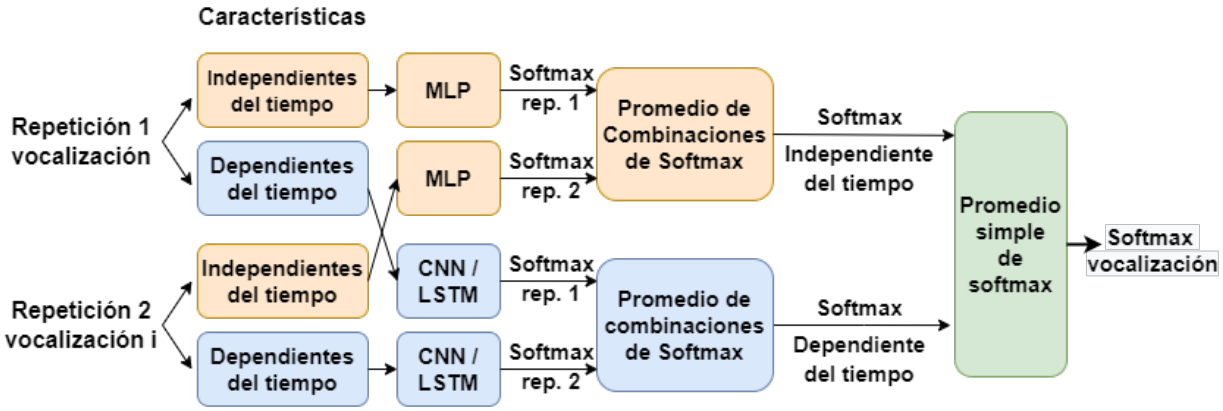


Figura 3.5: Predicción de vocalización i.

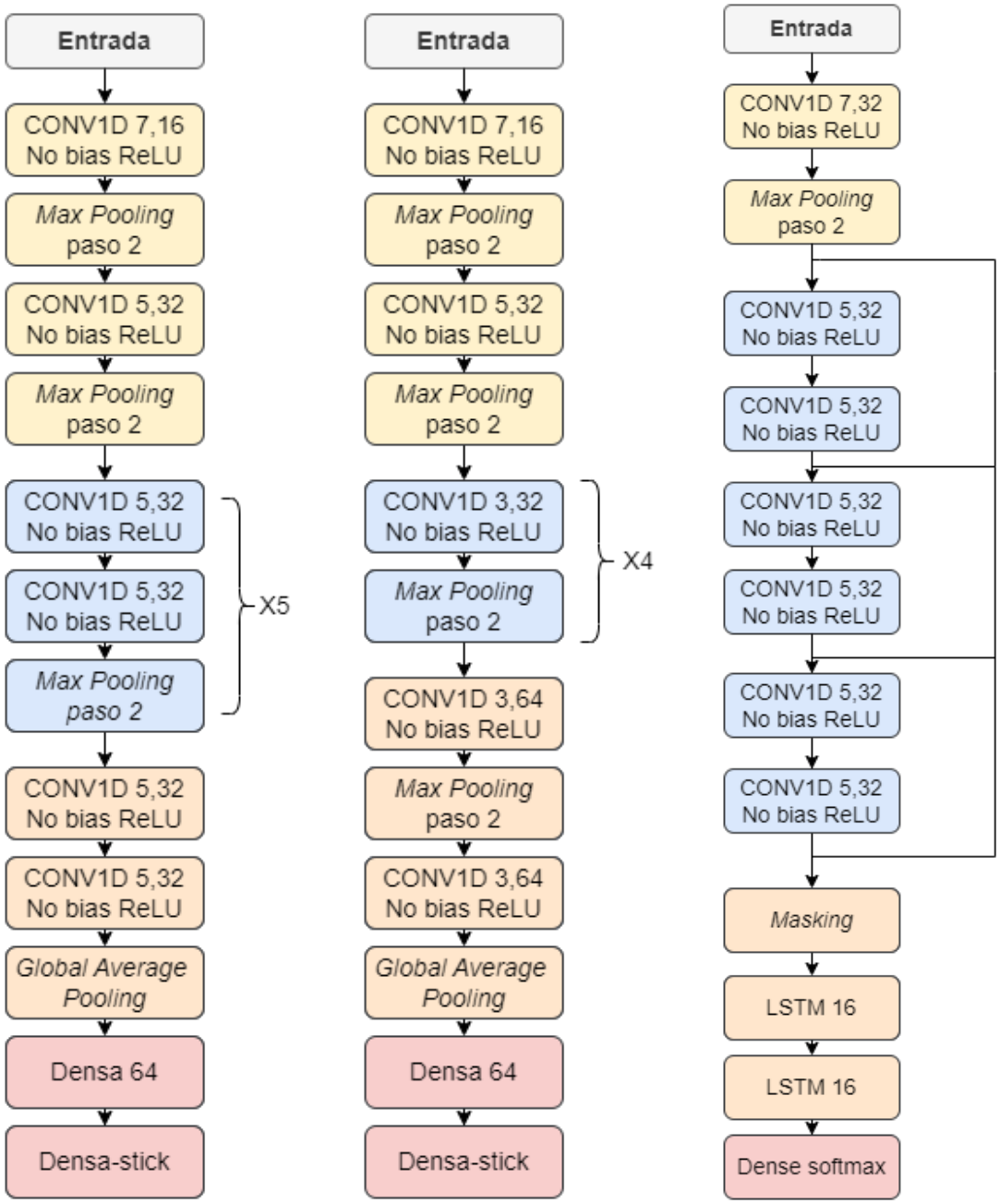
I. MLP para características independientes del tiempo

Dos de las características independientes del tiempo se calculan con la frecuencia fundamental, F0, estimada *frame* a *frame* con Praat [97]. Para representar el comportamiento de los sujetos con respecto a la curva F0 [34], se extraen las siguientes características dentro de cada vocalización: la pendiente normalizada por la media y la desviación estándar. El tercer parámetro corresponde a la duración de la fonetización en segundos. Posteriormente, se aplica MVN (normalización por media y varianza) a cada parámetro, calculando la media y la varianza de cada parámetro en toda la base de datos de entrenamiento. Como ya se ha mencionado, se entrenó un MLP para características independientes del tiempo por cada tipo de fonetización, es decir, /ae-ae/, /sa-sa/ y conteo (véase la Figura 3.5). Se empleó el optimizador ADAM y la entropía cruzada como función de pérdida. Las capas ocultas emplearon la función de activación no lineal ReLU. La capa de salida tenía cuatro neuronas con activación SoftMax. En el caso de /ae-ae/, la red tiene dos capas ocultas de 40 neuronas cada una, con una tasa de aprendizaje de 0.01. La MLP correspondiente para /sa-sa/ posee dos capas ocultas de 20 nodos y una tasa de aprendizaje de 0.01. Por último, la vocalización de conteo utiliza una red de cinco capas ocultas de 10 nodos y tasa de aprendizaje de 0.001.

II. Arquitecturas de redes neuronales para características dependientes del tiempo

Las características dependientes del tiempo se basan en el logaritmo del espectro de la señal y se optimizaron para cada tipo de fonetización. Lo primero es calcular la FFT de 512 muestras en ventanas de 50 ms con un solapamiento del 50 %, en las que se obtienen 257 *bins* de frecuencias. A continuación, se le calculan 14 filtros de Mel al espectro en el caso de las fonetizaciones /ae-ae/ y /sa-sa/, obteniendo 14 *bins/frames*. En el caso de la vocalización del conteo, no se emplearon filtros Mel, sino que se seleccionó el 75 % de los *bins* de frecuencias más bajas del espectro y a estos se les calculó la primera derivada o deltas, dando como resultado una dimensionalidad de 257x0.75x2 = 386 *bins/frames*. El MVN se aplica a las trayectorias temporales de las características dependientes del tiempo, y las medias y varianzas se calculan sobre toda la base de datos de entrenamiento. Esta red necesita que todos los audios posean la misma longitud, por lo que se realiza un relleno con ceros acorde al audio más largo de los datos de entrenamiento para la fonetización correspondiente. La arquitectura de los modelos para las características dependientes del

tiempo y su respectiva optimización de hiperparámetros condujeron a: tasa de aprendizaje igual a 0,0005 para /ae-ae/ y /sa-sa/, y de 0,0001 para el conteo; optimizador ADAM; entropía cruzada como función de pérdida. Las arquitecturas de aprendizaje profundo resultantes se muestran en la Figura 3.6.a (/ae-ae/), Figura 3.6.b (/sa-sa/) y en la Figura 3.6.c (conteo).



(a) Red para /ae-ae/. (b) Red para /sa-sa/. (c) Red para conteo.

Figura 3.6: Arquitecturas de redes neuronales para modelos basados en características dependientes del tiempo.

III. Entrenamiento *K-fold*

Para optimizar la base de datos disponible, se realizó una validación cruzada de nueve particiones, de la que se extrajeron 12 usuarios de la primera partición y 11 de las siguientes para evaluar el rendimiento en cada una. Es importante mencionar que este esquema de división de datos garantiza que un hablante determinado no pueda tener vocalizaciones en los subconjuntos de entrenamiento, validación o prueba simultáneamente.

Es necesario hacer la distinción de entrenamiento del clasificador de estimación de disnea y del *speech enhancement* cRF.

III.I. Entrenamiento módulo de estimación de dificultad respiratoria

En este entrenamiento cada partición se compone de subconjuntos de entrenamiento, validación-1 y validación-2, correspondientes al 70 %, 15 % y 15 % de los individuos de la partición respectivamente.

Los clasificadores se entrenaron ocho veces con cada partición para tener en cuenta la variabilidad debida a la inicialización de los pesos. El subconjunto de entrenamiento se utilizó para estimar los pesos de la red, y los datos de validación-1 se emplearon para detener las iteraciones y evitar el sobre-ajuste (con una parada temprana de 20 épocas). Para cada partición, se eligió el clasificador de red neuronal óptimo entre los ocho repeticiones que se habían entrenado, escogiendo el de mayor precisión al evaluar sobre los subconjuntos de validación-1 y validación-2. Este último no formó parte del procedimiento de entrenamiento, así la red neuronal entrenada elegida es también la que tiene la mejor capacidad de generalización. A continuación, los datos de prueba (ya sean reales o simulados), que nunca fueron vistos por la red, se propagaron para obtener las puntuaciones y métricas mMRC de la partición correspondiente. Estos pasos se repitieron para todas las particiones con el fin de obtener las puntuaciones y métricas de los 100 individuos. Por último, se repitió todo el procedimiento cinco veces para obtener estadísticas más fiables.

III.II. Entrenamiento cRF

El entrenamiento de esta red también se basó en *k-fold*, ya que al tratarse de un sistema integrado de *speech enhancement* más evaluación de disnea, no sería correcto que la red de mejoramiento de voz haya entrenado con un dato que luego va a ser utilizado como prueba para el módulo de dificultad respiratoria. Para hacer el modelo más generalizable, se generaron las mismas particiones en ambos módulos. De esta forma, al evaluar a una persona, sus audios serán un dato totalmente nuevo para el sistema en su conjunto.

Los datos restantes de cada partición se dividieron en 85 % y 15 % para datos de entrenamiento y validación, respectivamente. La red al ser más estable, solo se entrenó una vez por partición, donde la base de validación se encargó de evitar el sobre-ajuste de parámetros.

IV. Modelo del canal acústico para entrenamiento de sistema de dificultad respiratoria

La base de datos telefónica se utilizó para generar datos de entrenamiento simulados. El esquema de simulación es muy similar al implementado en [86] y se encuentra resumido en la tabla 3.2. Los 600 audios que componen la base de datos, fueron convolucionados con 33 respuestas al impulso reales, las que se grabaron en condiciones estáticas a dos metros de la

fuente del habla, pero para diferentes DOAs. Además, se añadió ruido aditivo con una SNR comprendida entre 5 dB y 15 dB. Este ruido es una mezcla entre el ruido real del robot PR2 y diferentes ruidos de Aurora, los que se sumaron con un SNR entre -5 dB y 5 dB.

Tabla 3.2: Esquema de simulación de datos de entrenamiento.

Simulación de datos	
RIRs	33 RIRs obtenidas en P1 para diferentes DOAs
Ruido	Añadido a la señal en un rango de 5 dB a 15 dB
Tipo de ruido	Suma de ruidos de robot y Aurora en rango de -5 dB a 5 dB

V. Métricas de evaluación de clasificación

Para evaluar los modelos de clasificación y en particular el sistema de dificultad respiratoria, se utilizarán diversas métricas que permiten analizar el rendimiento del clasificador.

Precisión

La precisión o *accuracy* determina el porcentaje de aciertos de las predicciones del clasificador con relación a las etiquetas reales de los datos.

Curva ROC

Representa la relación entre la tasa de verdaderos positivos frente a la tasa de falsos positivos para distintos umbrales de decisión. Estas tasas se pueden calcular como:

- Tasa de Verdaderos Positivos:

$$\text{TVP} = \frac{VP}{VP + FN}$$

- Tasa de Falsos Positivos:

$$\text{TFP} = \frac{FP}{FP + VN}$$

De donde VN corresponde a los verdaderos negativos, VP a los verdaderos positivos, FP a los falsos positivos y FN los falsos negativos (FN). En la Figura 3.7, se muestra la curva de un clasificador ideal (azul) y de un clasificador aleatorio (rojo) para una tarea binaria.

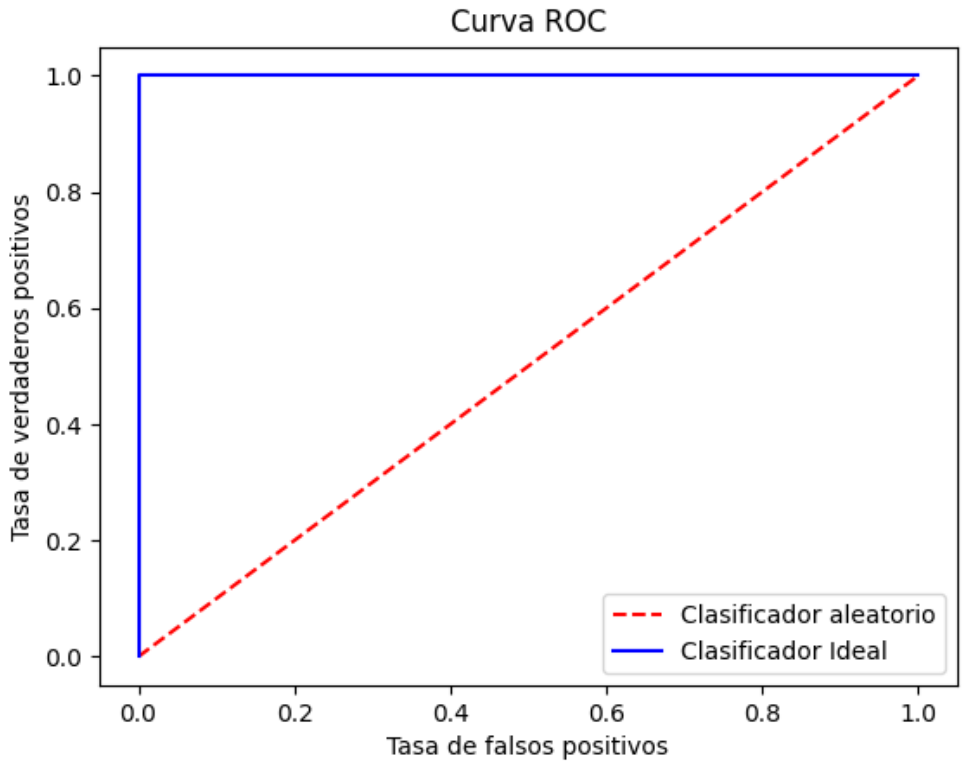


Figura 3.7: Curva de ROC.

AUC

El área bajo la curva ROC o AUC (AUC, Figura 3.7), es una métrica robusta y bastante utilizada para medir el rendimiento de un clasificador binario, donde 0,5 equivale a un clasificador aleatorio y 1 a un clasificador ideal.

VI. Métricas de evaluación del *speech enhancement*

Las métricas utilizadas en este trabajo evaluar el rendimiento del *speech enhancement* son el SNR y Si-SNR.

Relación señal a ruido (SNR)

El SNR (por sus siglas en inglés, *Signal to Noise Ratio*), es una medida que relaciona la potencia de la señal por sobre la potencia del ruido. Este ratio generalmente es medido en decibeles (dB) y se expresa matemáticamente como:

$$\text{SNR} = 10 \cdot \text{Log}_{10}\left(\frac{P_{señal}}{P_{ruido}}\right) \tag{3.1}$$

Donde $P_{señal}$ y P_{ruido} corresponden a la potencia de la señal y del ruido respectivamente. La potencia para de una señal discreta puede ser calculada como:

$$P_{signal} = \frac{\sum_{i=0}^{N} x_n^2}{N} \tag{3.2}$$

Para N igual al largo de la secuencia y x_n como una muestra n de la señal. En el análisis

de señales y en particular en el *speech enhancement*, el objetivo es encontrar métodos que aumenten el SNR, ya sea aumentando la ganancia en la señal de interés o reduciendo la potencia del ruido.

En este trabajo, para estimar el SNR se toma una ventana de 0,3 segundos al inicio y al final de la señal, donde dichos segmentos son considerados como solo ruido. Suponiendo estacionariedad para el ruido, se puede aproximar a que dicha señal representa fiablemente al ruido en su conjunto, con lo que se calcula la potencia de esos segmentos y se tiene una potencia del ruido estimada. La señal restante entre esas zonas, son consideradas como señal + ruido. Al calcular dicha potencia y restarle la potencia del ruido, se puede estimar la potencia de la señal. Con esto se tiene una potencia estimada para la señal y para el ruido, siendo directo el cálculo del SNR con la fórmula 3.1

Si-SNR

La otra métrica utilizada para para evaluar el sistema es el *Signal invariant signal to noise ratio* (Si-SNR). La gran ventaja de esta métrica es que es invariante a factores de escala para las distintas señales, por lo que permite un mayor grado de robustez al momento de evaluar la relación entre la señal y el ruido. Esta métrica es utilizada como función de pérdida para el entrenamiento del cRF implementado para el *speech enhancement*. La fórmula que define el Si-SNR es la siguiente [98].

$$\text{Si-SNR} = 10\text{Log}_{10}\frac{||y_{target}||^2}{||\text{noise}||^2} \tag{3.3}$$

Donde y_{target} corresponde a:

$$y_{target} = \frac{<\hat{y}, y> \cdot y}{||y||^2} \tag{3.4}$$

Con $\hat{y}$ e y representando la señal estimada y de referencia respectivamente. El operador <> corresponde a la multiplicación seguida por suma.

Capítulo 4

Resultados y análisis

En este capítulo se presentarán los resultados y el análisis correspondiente de los diversos experimentos llevados a cabo en este trabajo. En primer lugar, se explicará cómo se ajustaron los hiperparámetros, las arquitecturas y los tipos de entrenamiento de las redes de estimación de dificultad respiratoria. A continuación, se detallarán los resultados por vocalización y complementariedad de características en el caso telefónico, y posteriormente se mostrará cómo esta red responde ante la propagación de bases de datos reales HRI.

Por último, se presentarán los entrenamientos alineados, ya sea utilizando bases de datos simuladas o reales. Esto consiste en entrenar con audios simulados utilizando algún algoritmo de mejora de voz y luego evaluar en bases de datos reales o simuladas, pero utilizando el mismo algoritmo de mejora de voz.

4.1. Optimización de características, arquitecturas, hiperparámetros y entrenamiento

Las características independientes del tiempo que se consideraron inicialmente fueron: duración; promedio del tono; pendiente del tono; desviación estándar del tono; *jitter*, interrupciones de la voz; *shimmer*; y, centro de energía por *frames*. A continuación, se eligieron las características que proporcionaban la mayor discriminación entre individuos con dificultad respiratoria (es decir, puntuación mMRC de referencia igual a 1, 2 o 3) e individuos sin disnea (es decir, puntuación mMRC de referencia igual a 0) obteniendo finalmente: la pendiente del tono normalizada por el promedio de este: la desviación estándar de la curva F0; y la duración de la vocalización en segundos. El MLP usado para entrenar las de características independientes del tiempo fue ajustado con respecto a: las tasas de aprendizaje, entre 0,1, 0,01, 0,001 o 0,0001; el número de neuronas por capa, con valores entre 10, 20, 30, 40, 50 o 60; y, el número de capas ocultas de 1, 2, 3, 4 o 5.

En el caso de las características dependientes del tiempo, se probaron las siguientes configuraciones: número de muestras FFT entre 128, 256, 512 y 1024; longitud de la ventana para 128, 256 y 512 muestras; distintas porciones del espectrograma entre 25 %, 50 %, 75 % y 100 % de los *bins* FFT; espectro logarítmico FFT frente y MFCCs; y la utilización o no de deltas y delta-delta. El solapamiento de la ventana se hizo igual al 50 % y el número de filtros Mel fue de 14. En cuanto a las redes neuronales para las características dependientes del tiempo, se llevó a cabo una optimización más exhaustiva: para las redes convolucionales CNN 1-D se varió el tamaño del *kernel* (3, 5 o 7), número de filtros (16, 32, 64 o 128) y número

de capas convolucionales (3, 6, 10 ,10 o 14); bloques de *max pooling*; conexiones residuales; y, LSTM o BiLSTM. Como se puede ver en la Figura 3.6.a y 3.6.b, se definieron dos tipos bloques, los que se diferencian por tener una o dos capas convolucionales seguidas por un *Max Pooling*, y el número de este tipo de bloques también fue ajustado. Además, se optimizó la capa totalmente conectada de la salida final, ajustando: el número de capas (1, 2 y 3); y, el número de neuronas por capa, es decir, 16, 32, 64, 128 o 256. La salida del bloque totalmente conectado se compone de cuatro nodos softmax correspondientes a las cuatro puntuaciones o clases mMRC.

4.2. Estimación de dificultad respiratoria sobre red telefónica

En esta sección se analizará el desempeño de las distintas vocalizaciones que componen la base de datos telefónica generada (/ae-ae/, /sa-sa/ y conteo), tanto de forma individual como la complementariedad al fusionarlas entre sí. También se tendrá en consideración la complementariedad presente entre los modelos dependientes e independientes del tiempo para cada vocalización.

La Figura 4.1 muestra la precisión de los modelos para los distintos clasificadores de las vocalizaciones, las tres barras de cada vocalización indican los clasificadores dependientas del tiempo, independientes del tiempo y la combinación de ambos, respectivamente (ver leyenda del gráfico). Lo más directo de notar es la complementariedad existente entre las características dependientes e independientes del tiempo, donde para todas las fonetizaciones se obtiene una mejora frente a los clasificadores dependientes o independientes de forma individual, a excepción de la combinación /ae-ae/ $\oplus$ conteo ($\oplus$ denota la fusión entre elocuciones). Si se calcula un promedio entre características dependientes e independientes del tiempo, el modelo de características combinadas tiene una mejora de 2 %, 10 %, 7 %, 10 %, -3 %, 14 % y 10 % para la vocalización /ae-ae/,/sa-sa/, conteo, /ae-ae/ $\oplus$ /sa-sa/, /ae-ae/ $\oplus$ conteo, /sa-sa/ $\oplus$ conteo y /ae-ae/ $\oplus$ /sa-sa $\oplus$ conteo respectivamente, donde solo /ae-ae/ $\oplus$ conteo presenta una caída, pero es más una excepción que la regla.

Ahora bien, si el análisis se centra en la complementariedad de fonetizaciones, solo se considerará el caso combinado (ya que se corroboró la complementariedad entre características). Uno de los resultados que más resalta corresponde al caso del clasificador /sa-sa/, el cual de forma individual presenta un gran desempeño, alcanzando una precisión cercana al 51 %, al igual que el caso combinado de las tres fonetizaciones. De igual forma, cuando esta fonetización se combina con el conteo, el cual tiene una precisión bastante menor (40 %), aumentando su rendimiento un 4 % su valor, obteniendo el resultado más alto de todas las combinaciones. Es importante tener en cuenta, que el utilizar las tres vocalizaciones controladas se obtiene un mejor desempeño que prácticamente todas las vocalizaciones individuales (a excepción de /sasa/ $\oplus$ conteo), donde se tiene una mejora de 25 %, 0 %, 26 %, 4 %, 21 % y -4 % para la vocalización /ae-ae/,/sa-sa/, conteo, /ae-ae/ $\oplus$ /sa-sa/, /ae-ae/ $\oplus$ conteo y /sa-sa/ $\oplus$ conteo, respectivamente.

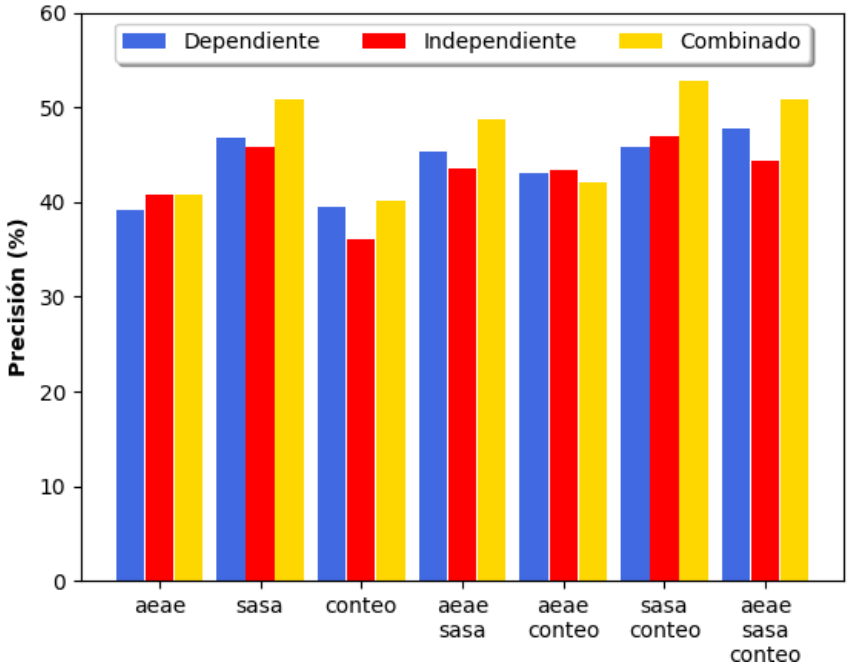


Figura 4.1: Precisión de características dependientes del tiempo, independientes del tiempo y su combinación para las distintas vocalizaciones y fusión de estas en base telefónica..

Por otro lado, la Figura 4.2 muestra el área bajo la curva ROC para las distintas vocalizaciones y combinaciones de características dependientes e independientes del tiempo. Si realizamos un análisis de complementariedad, la combinación de características otorga mejores resultados en la gran mayoría de los casos, a excepción del /ae-ae/ y de /ae-ae/ $\oplus$ conteo, donde existe una caída con respecto a las características independientes del tiempo. De igual forma, si se calcula un promedio entre dependientes e independientes, se obtiene que la combinación entrega una mejora de 3 %, 8 %, 4 %, 5 %, 5 %, 6 % y 5 % para la vocalización /ae-ae/,/sa-sa/, conteo, /ae-ae/ $\oplus$ /sa-sa/, /ae-ae/ $\oplus$ conteo, /sa-sa/ $\oplus$ conteo y /ae-ae/ $\oplus$ /sa-sa $\oplus$ conteo, respectivamente.

Si en el mismo gráfico, se analiza la complementariedad entre fonetizaciones controladas (para el caso de las características combinadas), es claro notar que el peor rendimiento de los clasificadores individuales corresponde para el caso del conteo. A pesar de esto, el modelo que presenta un mejor rendimiento en AUC corresponde al clasificador que utiliza las tres vocalizaciones controladas, las que poseen una mejora de 10 %, 3 %, 26 %, 1 %, 5 % y 3 % en relación a /ae-ae/,/sa-sa/, conteo, /ae-ae/ $\oplus$ /sa-sa/, /ae-ae/ $\oplus$ conteo y /sa-sa/ $\oplus$ conteo.

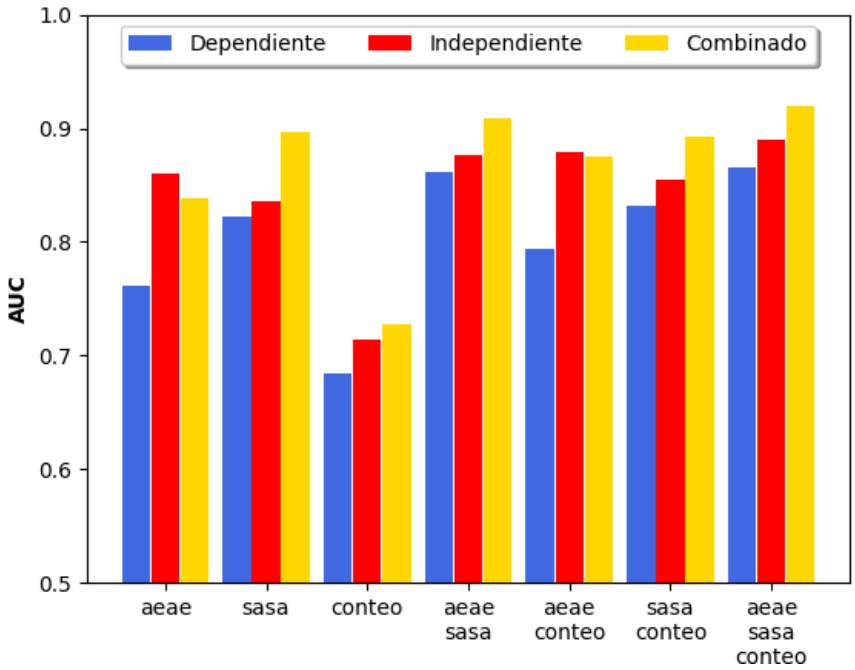


Figura 4.2: AUC de características dependientes del tiempo, independientes del tiempo y su combinación para las distintas vocalizaciones y fusión de estas en base telefónica.

Si bien el conteo presenta un rendimiento menor que las otras elocuciones en torno a su precisión y AUC, esta vocalización si presenta información relevante y complementaria frente a las demás, ya que los mejores resultados en ambas métricas se obtienen cuando se utiliza el conteo. En base a esto, si debe ser considerado como una buena fuente de información, la cual permite la representación del comportamiento espontáneo del usuario al tratarse de un discurso no sostenido (como /ae-ae/ o /sa-sa/).

La Figura 4.3 muestra la matriz de confusión de clases para el sistema entrenado y evaluado con la base de datos telefónica. Como es posible apreciar en ella, se tiene una gran precisión para el mMRC 0, acertando en un 91 % de los casos. Sin embargo, para las clases mayores, no existe una diagonal marcada, lo que indica que el clasificador no es tan preciso en detectar diferencias entre clases enfermas. Por otro lado, es destacable que los errores entre clases no son muy grandes, donde al ver la Figura 4.3, se aprecia que ninguna persona sana fue catalogada con mMRC 3 y que solo el 0,045 % de las personas con mMRC 3 fueron catalogadas como sanas. La precisión binaria para este caso corresponde a un 85 %.

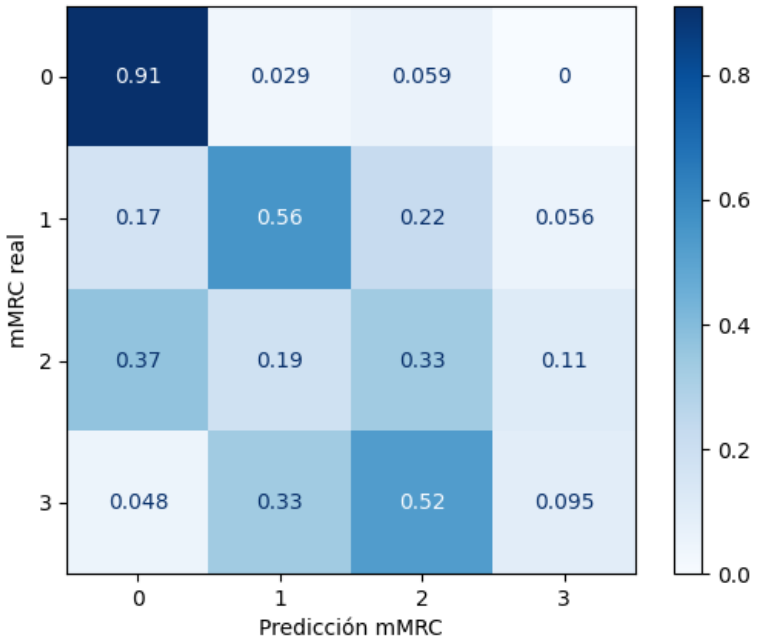


Figura 4.3: Matriz de confusión para sistema entrenado y evaluado con base de datos telefónica.

4.3. Estimación de dificultad respiratoria HRI

En esta sección se muestra el rendimiento de la red de dificultad respiratoria al realizar diferentes tipos de entrenamiento y evaluación, ya sea entrenando con datos telefónicos y evaluando las bases de datos HRI, o entrenando con datos simulados y evaluando bases de datos HRI simuladas o reales. En el caso de tratarse de entrenamiento con bases simuladas, el entrenamiento y evaluación se realiza de forma alineada, es decir, para cada condición de HRI se prueba con su base de datos de evaluación equivalente, ya sea simulada o con la base de datos real de HRI. Los resultados expuestos corresponden a la propagación de todos los subconjuntos de prueba de las 9 particiones (validación cruzada k-*fold*), por lo que al evaluar todas las particiones se recupera la base de datos completa y, por tanto, se obtienen métricas más robustas.

4.3.1. *Speech enhancement*

Para evaluar el rendimiento de los distintos algoritmos de *speech enhancement* utilizados para entrenar el sistema de dificultad respiratoria, se utilizará la métrica de la relación señal a ruido o SNR. En la Figura 4.4, se muestra el SNR para las distintas bases de datos de evaluación (Simulada, estática y dinámica) y los respectivos algoritmos de *speech enhancement* implementados. La tendencia de los distintos algoritmos se repite para las distintas bases de datos, donde el caso ruidoso presenta el SNR más bajo, seguido por *delay-and-sum*, MVDR y cRF. Esto demuestra que el modelamiento acústico del canal fue acertado, ya que los resultados obtenidos para la base de datos simulada son muy similares a los reales. Calculando un promedio sobre las distintas bases de datos, el peor resultado en torno a SNR corresponde

a no hacer nada (caso ruidoso), luego al aplicarle *speech enhancement* se tiene un alza sobre el caso ruidoso de 27 %, 93 % y 140 % para *delay-and-sum*, MVDR y cRF, respectivamente.

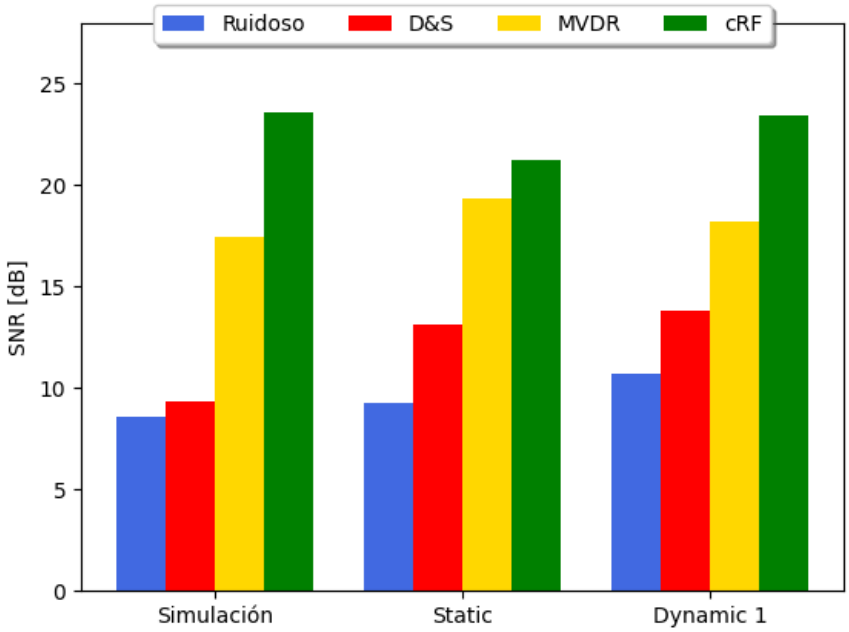
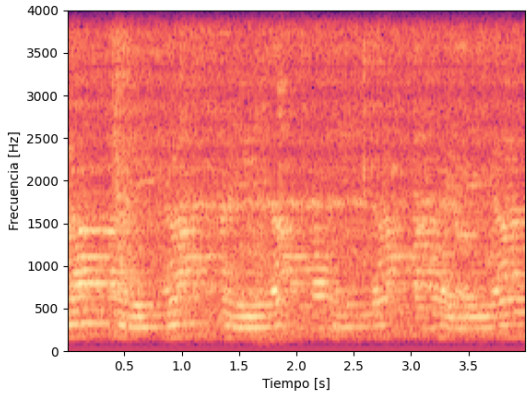
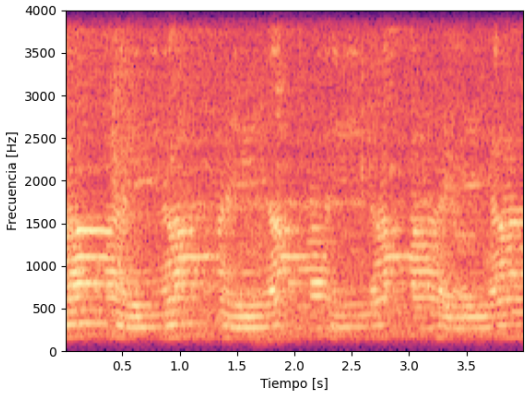


Figura 4.4: SNR para las distintas bases de datos y algoritmos de *speech enhancement*.

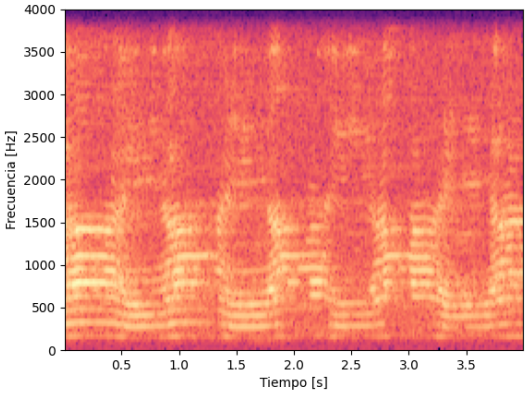
La Figura 4.5 muestra los espectrogramas de los distintos algoritmos de *speech enhancement* implementados, para esto se escogió un audio la fonetización /ae-ae/ de la base de datos *static* 2. De los espectrogramas, se puede notar claramente la periodicidad de la elocución /ae-ae/. Ahora bien, analizando los distintos casos, el ruidoso (Figura 4.5.a) presenta una clara atenuación de la frecuencia fundamental y de los armónicos de la voz, debido a la presencia del ruido que distorsiona la señal. En la Figura 4.5.b, *delay-and-sum* presenta mayor energía en los fonemas, lo que permite observarlos con mayor claridad en las distintas frecuencias. MVDR (Figura 4.5.c) mantiene esta "alza" en la calidad, reduciendo el ruido alrededor de la voz y por tanto potenciando la señal objetivo, donde sobre todo se empiezan a notar los componentes de la voz sobre los 1500 Hz. cRF es el algoritmo de *speech enhancement* que posee mejor relación señal a ruido (recordar gráfico de la Figura 4.4), pero al observar su espectrograma (Figura 4.5.d) se tiene que este algoritmo si bien reduce el ruido en gran cantidad, también atenúa zonas importantes de la voz, como lo son los armónicos. Esto implica que el tener un buen SNR no asegura un buen rendimiento del *speech enhancement* en el caso de la estimación de dificultad respiratoria, ya que el enmascaramiento utilizado por cRF puede quitar información relevante para el sistema en su conjunto.



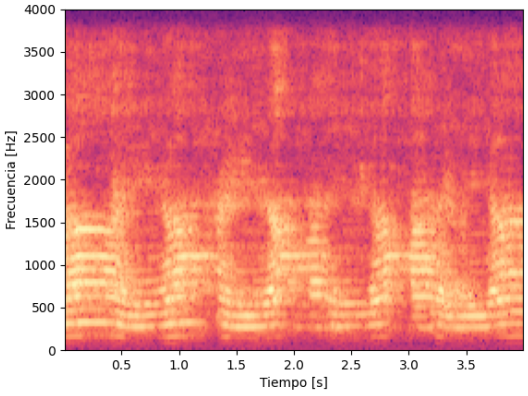
(a) Espectrograma para caso ruidoso



(b) Espectrograma al aplicar *delay-and-sum*.



(c) Espectrograma al aplicar MVDR.



(d) Espectrograma al aplicar cRF.

Figura 4.5: Espectrogramas de vocalización /ae-ae/ para los distintos algoritmos de *speech enhancement*.

4.3.2. Entrenamiento con data telefónica y evaluación en data real

La Tabla 4.1 muestra la precisión, precisión binaria y el AUC del sistema de estimación de dificultad respiratoria al entrenar con datos telefónicos y propagar las diferentes bases de datos reales HRI. Por orden y condensación de resultados, solo se muestra el resultado final del sistema, es decir, sin los resultados desagregados por vocalización y/o tipo de característica (dependiente e independiente del tiempo). Ahora bien, de la Tabla 4.1 es fácil notar una disminución importante en el desempeño del modelo al aplicarlo al conjunto de datos *static* ruidoso sin *speech enhancement*, donde se registró una disminución del 34 %, 12 % y 12 % en la precisión, precisión binaria y AUC, respectivamente, en comparación a los datos telefónicos. Siguiendo la misma línea, se observa una disminución del 28 %, 8 % y 10 % en precisión, precisión binaria y AUC, respectivamente, al aplicar el modelo al conjunto de datos *dynamic* ruidoso, en comparación con los datos telefónicos.

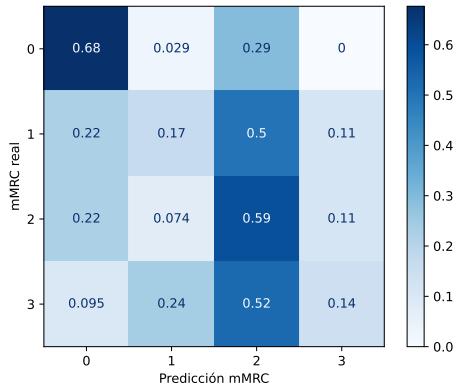
El uso de técnicas como *delay-and-sum* y MVDR permiten reducir la degradación del desempeño del sistema en ambientes ruidosos, mientras que cRF no se muestra consistente en las condiciones estáticas y dinámicas. En la condición estática, MVDR presenta el mejor

rendimiento al lograr una mejora considerable del 24 %, 11 % y 4 % en precisión, precisión binaria y AUC, respectivamente, en comparación con el caso *baseline static* ruidoso. Por otro lado, al centrarse en el caso dinámico, *delay-and-sum* presenta los mejores resultados al lograr un aumento del 8 %, 1 % y 2 % en precisión, precisión binaria y AUC, respectivamente, en comparación con el caso *baseline dynamic* ruidoso. Es importante tener en cuenta que, aunque el aumento en el desempeño en el caso dinámico es menor, la tarea en este caso es más compleja que en el caso estático, por lo que aplicar técnicas de *speech enhancement* y obtener mejores resultados resulta más difícil.

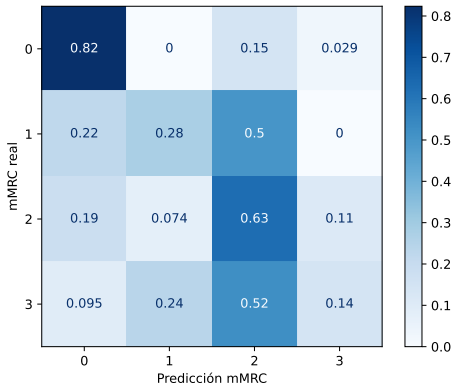
Tabla 4.1: Precisión, precisión binaria y AUC para modelo entrenado con datos telefónicos al evaluar diferentes condiciones reales HRI.

Configuración	**Base de evaluación**	**Precisión (%)**	**Precisión Binaria (%)**	**AUC**
-	Telefónica	51	85	0.92
Static	Ruidosa	38	76	0.82
	D&S	42	82	0.84
	MVDR	47	84	0.86
	cRF	39	76	0.83
Dynamic	Ruidosa	40	79	0.84
	D&S	43	80	0.85
	MVDR	42	79	0.84
	cRF	42	75	0.84

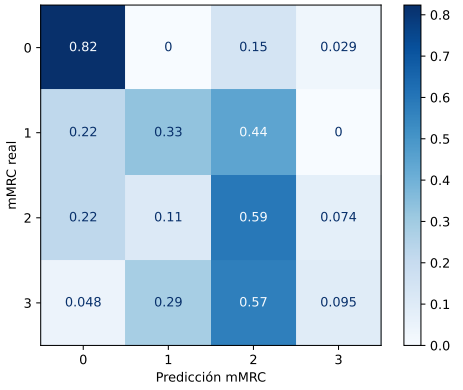
En la Figura 4.6 se muestran las distintas matrices de confusión del sistema para los distintos *speech enhancement*. Como existen diferentes configuraciones de bases de datos, se presentan los resultados promedios de evaluar en la condición *static* y *dynamic*, para un modelo entrenado con base de datos telefónica. Al igual que en la Tabla 4.1, los peores rendimientos vienen dados al evaluar la base de datos ruidosa (Figura 4.6.a), seguido sorprendentemente por cRF (Figura 4.6.d), lo que demuestra que aplicar redes complejas no necesariamente entregará mejores resultados a priori. El *speech enhancement* que produce mejores resultados corresponde a *delay-and-sum* (Figura 4.6.b), el cual provoca un aumento de 21 %, 65 %, 7 % y 0 % para los mMRC 0, 1, 2 y 3 respectivamente, frente al caso ruidoso.



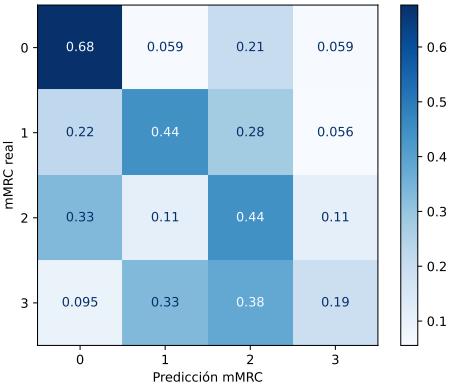
(a) Matriz de confusión para caso ruidoso



(b) Matriz de confusión al aplicar *delay-and-sum*.



(c) Matriz de confusión al aplicar MVDR.



(d) Matriz de confusión al aplicar cRF.

Figura 4.6: Matrices de confusión al evaluar los distintos algoritmos de *speech enhancement* sobre un modelo entrenado con la base de datos telefónica.

4.3.3. Entrenamiento y evaluación en data simulada

En esta subsección, se analiza el rendimiento del sistema al entrenar y evaluar en una condición alineada para los datos simulados. Al observar la precisión y el AUC de los sistemas (Figura 4.7 y Figura 4.8 respectivamente), el modelo de peor rendimiento viene dado por el caso ruidoso, que presenta una caída a nivel de características combinadas respecto al modelo telefónico del 16 % y 6 % en precisión y AUC, respectivamente. Si se toma un promedio del combinado de la Figura 4.7 y 4.8, se obtiene una mejora sostenida frente a no hacer nada, es decir, promedio de *static* y *dynamic* ruidoso (Tabla 4.1) de un 12 % y 6 %. Lo anterior demuestra que el entrenamiento alineado, sea con data ruidosa o con *speech enhancement*, aporta una mejora considerable en la estimación de dificultad respiratoria. Ahora bien, comparando entre los mismos resultados alineados, MVDR es el que presenta la mejora más alta frente al modelo ruidoso, siendo de un 7 % y 3 % en precisión y AUC, respectivamente.

Si el análisis se centra en la complementariedad entre las características dependientes e independientes del tiempo a nivel de precisión (Figura 4.7), en la mayoría de los casos se

obtiene una mejora al combinar contra los clasificadores individuales, excepto para MVDR y cRF. Mientras que MVDR sigue estando por encima del caso ruidoso, cRF empeora considerablemente y devuelve una precisión muy similar al caso ruidoso. Centrando ahora el análisis en el AUC de los sistemas (Figura 4.8), podemos confirmar la complementariedad existente entre características, donde para todos los modelos (teléfono, limpio, ruidoso y mejorado) el caso combinado está por encima de las características individuales.

Otro punto importante a destacar es la mayor robustez de las características independientes del tiempo en los diferentes sistemas, donde en AUC el pasar del caso telefónico al ruidoso genera un decaimiento del 3 %, que luego con las diferentes mejoras se mantiene cercano a 0,86 (con una baja variabilidad). Esto, ya que al tratarse de una métrica global sobre el audio, tienden a ser más estables. Por el contrario, las características dependientes del tiempo obtienen una caída importante al pasar del sistema telefónico al ruidoso (9 %), pero al realizar mejoras se provoca una mejora del 6 % en promedio (resaltando MVDR con un 8 %). Asimismo, se espera una mayor volatilidad para las características dependientes del tiempo, ya que la entrada del clasificador corresponde al espectrograma o a las MFCCs, por lo que alteraciones o artefactos en la señal temporal causan más impacto en la salida. Para contrarrestar estas variaciones, el uso de la combinación de ambos tipos de características junto con un entrenamiento alineado con la base de evaluación, es decir, ruidoso con ruidoso, *delay-and-sum* con *delay-and-sum*, MVDR con MVDR, etc., hacen que el sistema de estimación respiratoria sea más robusto, con resultados similares entre el modelo ruidoso y los mejorados.

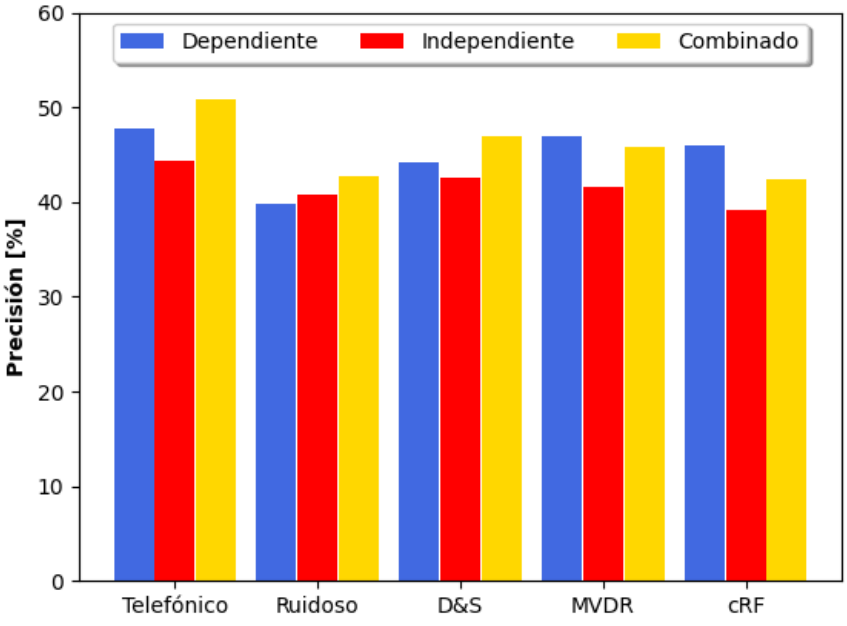


Figura 4.7: Precisión de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base simulada.

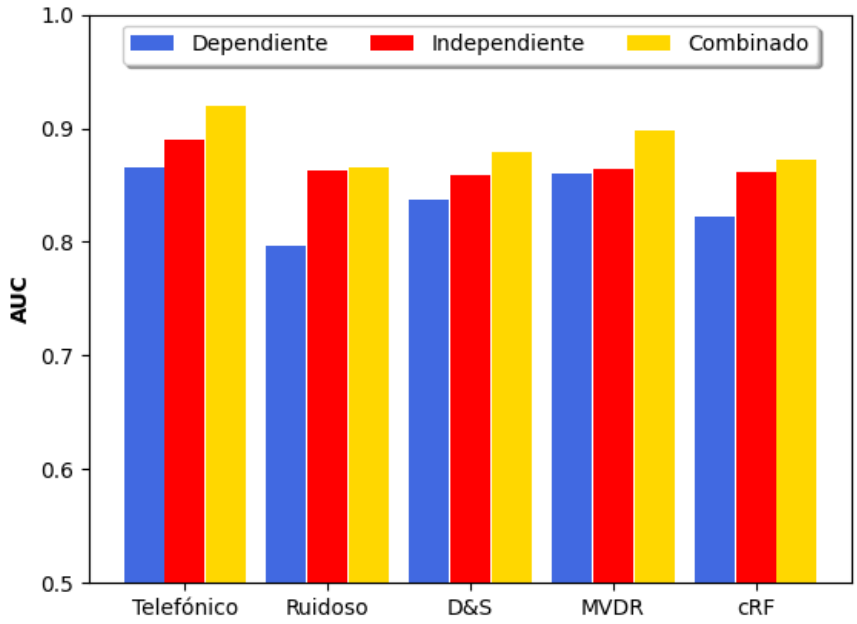


Figura 4.8: AUC de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base simulada

La Figura 4.9 muestra la matriz de confusión del mejor sistema encontrado para la base simulada, es decir, entrenar y evaluar con la base de datos simulada de MVDR. Como se aprecia en esta Figura, existe una mejora considerable en relación a lo encontrado en la Figura 4.6.a, con una importante alza en las clases 0 y 1, mejorando un 25 % y 225 % respectivamente. De igual forma, existe una caída para las clases 3 y 4, ya que si se observa la matriz de confusión, es claro notar que las predicciones para los enfermos se centran en la clase 1 y 2, lo que si bien genera menor error entre clases, provoca que muy pocas personas sean catalogadas con mMRC igual a 3. La precisión binaria en esta condición es de un 85 %, la cual es idéntica a la obtenida en el caso telefónico (Tabla 4.1), lo que indica la robustez de entrenar en condiciones alineadas, es decir, la misma condición de entrenamiento y evaluación.

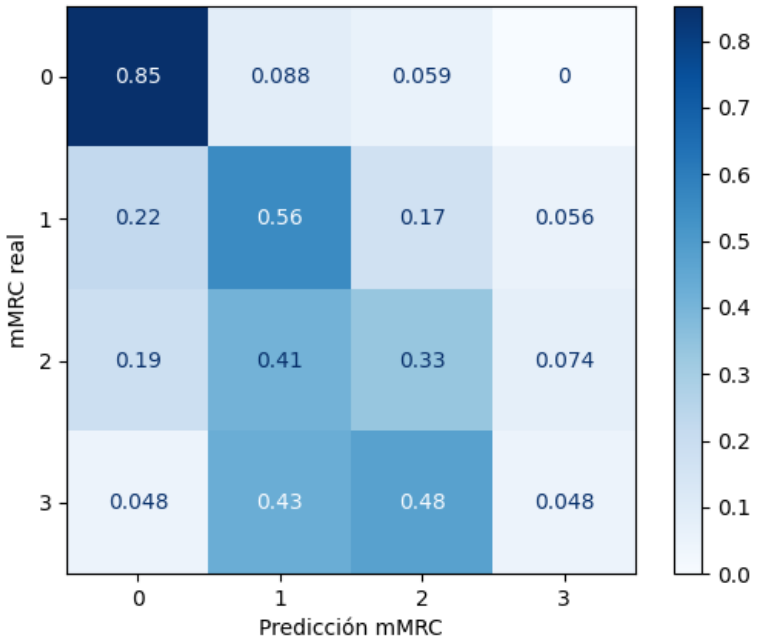


Figura 4.9: Matriz de confusión para sistema entrenado y evaluado con base de datos simulada MVDR.

4.3.4. Entrenamiento y evaluación en data real

Para cada condición de *speech enhancement* de la subsección anterior, en lugar de evaluar con datos simulados, se utilizaron las distintas bases de datos HRI reales (Tabla 3.1) y se aplicó la mejora correspondiente para utilizar los modelos entrenados con data simulada.

I. Condición estática real HRI

En esta sección se ha calculado una media entre las bases de datos *static* 1 y 2 para la compactación de los resultados. A partir de los gráficos de precisión y AUC de los sistemas (Figura 4.10 y Figura 4.11, respectivamente), es posible observar una caída al pasar del sistema telefónico al sistema real ruidoso de la precisión y del AUC de un 11 % y 6 %, respectivamente. Por otro lado, el algoritmo que presenta un mejor rendimiento es MVDR (como en los datos simulados), aunque prácticamente no presenta mejora en precisión (0,2 %), y en AUC aumenta un 2 % su valor al compararlo con caso ruidoso.

Ahora bien, si se compara el resultado de MVDR con el caso base de entrenar con datos telefónicos y evaluar en el caso *static*, se obtiene una mejora de 20 % y 7 % en precisión y AUC respectivamente.

A partir de la Figura 4.10, es posible observar la complementariedad entre las características (excepto de nuevo para cRF), destacando MVDR por la mejora del 13 % cuando se utiliza la combinación de características frente al uso exclusivo de características dependientes del tiempo. Sin embargo, observando el AUC en la Figura 4.11, la complementariedad entre características no es coherente con la Figura anterior, donde la combinación de carac-

terísticas es siempre inferior al rendimiento de las características independientes del tiempo individualmente. Mientras que el AUC no muestra una mejora en la complementariedad, la precisión sí lo hace, por lo que la combinación de ambas características proporciona información complementaria entre los clasificadores, haciendo que el modelo sea más robusto a diferentes condiciones de mejora, por lo que (como en el caso simulado) los resultados finales combinados son muy similares.

También es posible notar que se obtuvieron resultados similares al evaluar la base de datos simulada (Figuras 4.7 y 4.8) y con la real *static* HRI (Figuras 4.10 y 4.11), lo que indica la buena capacidad de generalización del sistema y, por tanto, que la simulación de datos fue efectiva. Si se calcula una media sobre las distintas condiciones de mejora (para estática simulada y real), no hay cambios en la precisión, y en el AUC hay un ligero descenso del 1 % al pasar a datos reales. Del mismo modo, si el análisis es más específico, las características dependientes del tiempo muestran una caída significativa del 6 %, mientras que las independientes del tiempo son notablemente más robustas, mostrando incluso un aumento medio del 2 % en comparación con los datos simulados. Lo que nuevamente confirma la necesidad de combinar los tipos de características para obtener un sistema más estable a las distintas bases de datos y configuraciones.

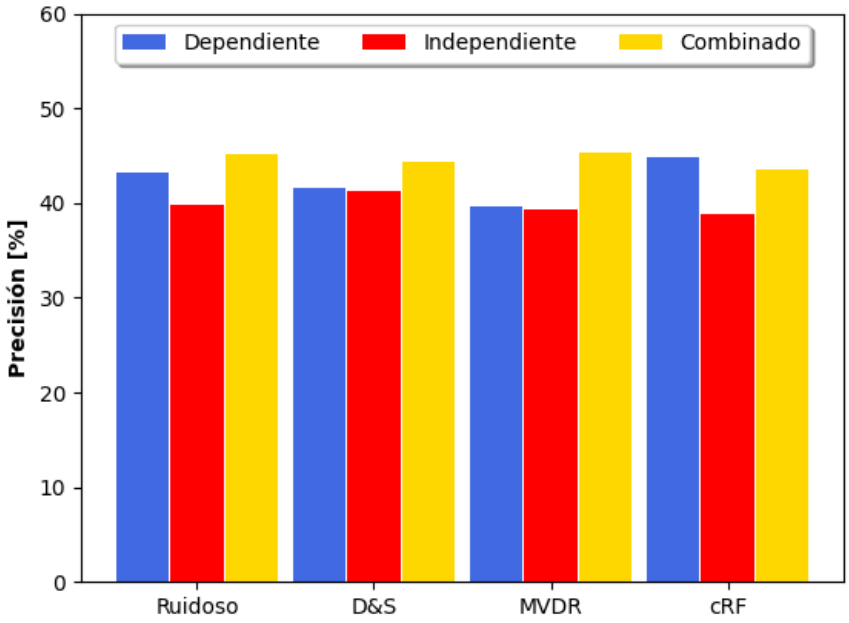


Figura 4.10: Precisión de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base *static*.

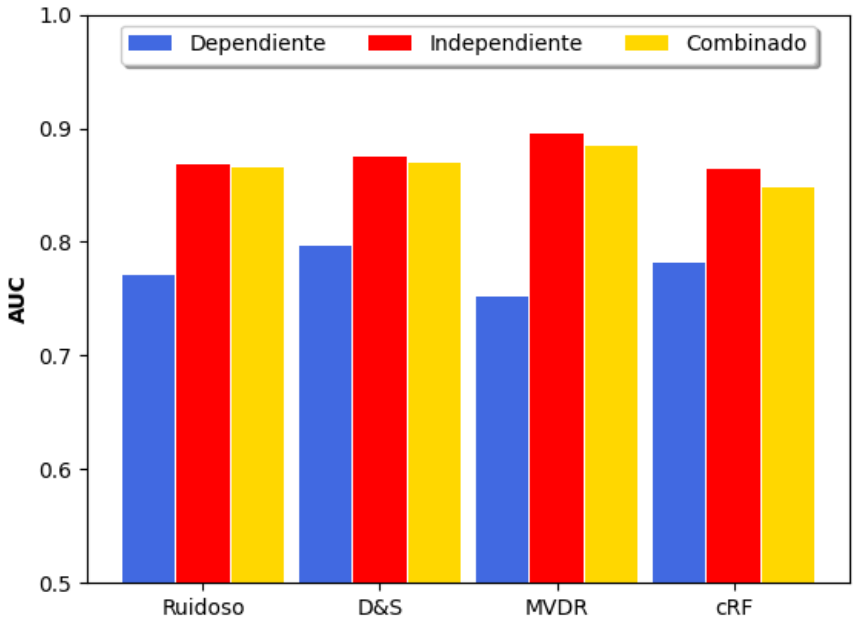


Figura 4.11: AUC de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base *static*.

La Figura 4.12 muestra la matriz de confusión del mejor sistema encontrado para la base estática, es decir, entrenar con la base de datos simulada de MVDR y evaluar con la base estática real mejorada con MVDR. Al igual que en el caso entrenado y evaluado con base simulada (Figura 4.9), la principal mejora se genera es en la clase 0 y un 1, con un alza de 21 % y 129 % respectivamente, con respecto a no aplicar *speech enhancement* (Figura 4.6.a). La precisión binaria en esta condición alcanza un 84 %, la cual es considerablemente alta y solo tiene una caída de 1 % frente a la condición telefónica y a la simulada con MVDR.

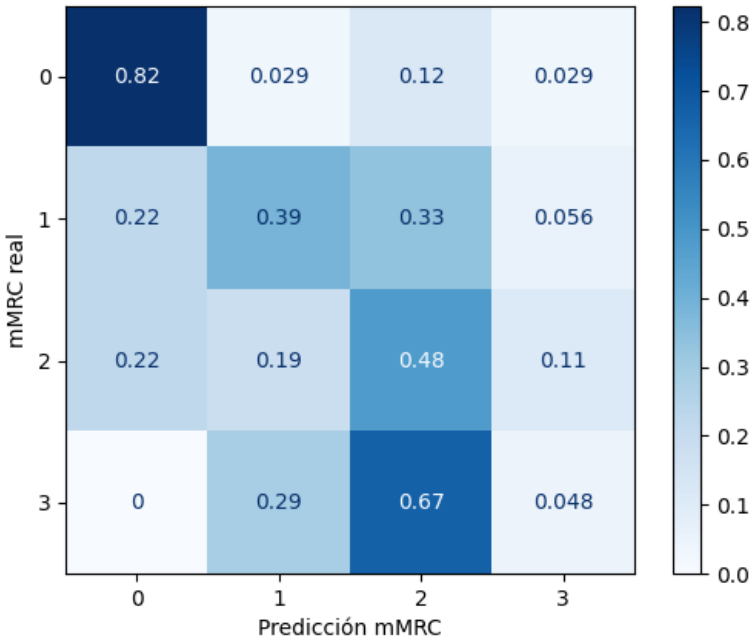


Figura 4.12: Matriz de confusión para sistema entrenado con base de datos simulada MVDR y evaluado en condición *static* con MVDR.

II. Condición dinámica real HRI

A partir de los gráficos de precisión (Figura 4.13) y AUC (Figura 4.14) de los sistemas, es posible observar una caída del 13 % y del 6 % en precisión y AUC, respectivamente, al pasar del sistema telefónico al sistema ruidoso. Por otro lado, la mejora con mejor rendimiento corresponde a *delay-and-sum*, que muestra una ligera mejora del 2 % tanto en precisión como en AUC al compararlo con el sistema ruidoso.
Ahora bien, si se compara el resultado de *delay-and-sum* con el caso base de entrenar con datos telefónicos y evaluar en el caso *dynamic*, se obtiene una mejora de 11 % y 5 % en precisión y AUC respectivamente. Aunque esta mejora es menor en comparación con el sistema en condiciones reales estáticas, en gran parte se debe a la mayor complejidad de la tarea.

Analizando ahora la complementariedad a nivel de precisión (Figura 4.13), todos los modelos presentan una mejora al combinar las características dependientes e independientes del tiempo, donde el mejor resultado se obtiene (sorprendentemente) para cRF, siendo un 5 % superior al caso ruidoso. A nivel de AUC (Figura 4.14), el comportamiento es muy similar al caso estático (Figura 4.11), donde aunque existe complementariedad entre las características, no es capaz de superar a la independiente del tiempo de forma individual. De nuevo, se puede observar la solvencia de las características independientes del tiempo para el AUC, donde el rendimiento es muy similar para los diferentes sistemas. Las características dependientes del tiempo, como ya se ha comentado, presentan mayor variabilidad en los diferentes sistemas, donde por ejemplo al pasar del caso ruidoso al mejorado, el AUC mejora de media un 4 %, excluyendo MVDR por ser la única que empeora respecto al ruidoso. Esto puede deberse a la dificultad de estimar e interpolar el ruido (necesario para hallar las matrices de covarianza)

para el caso de las fonetizaciones controladas.

Otro punto a destacar, corresponde a la robustez del modelo de estimación de la dificultad respiratoria en condiciones estáticas y dinámicas. Y es que si se toma la media de las diferencias porcentuales al pasar del caso estático al dinámico, en promedio para los distintos sistemas, tenemos una mejora del 0,44 % en la precisión y una caída del AUC del 0,2 %. Estas variaciones son prácticamente marginales, por lo que se podría postular que la red de estimación de dificultad respiratoria tiene una respuesta coherente para las condiciones estáticas y dinámicas HRI.

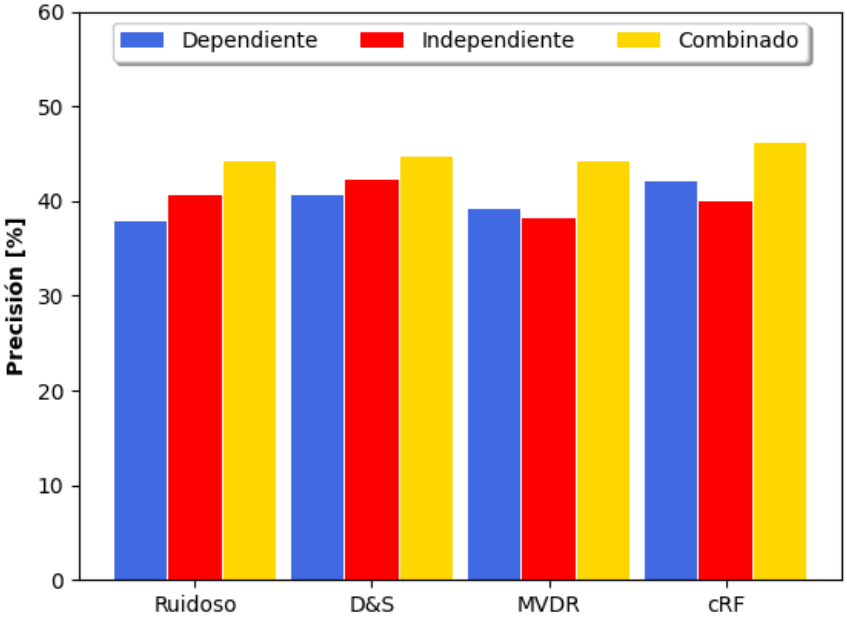


Figura 4.13: Precisión de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base *dynamic* 1.

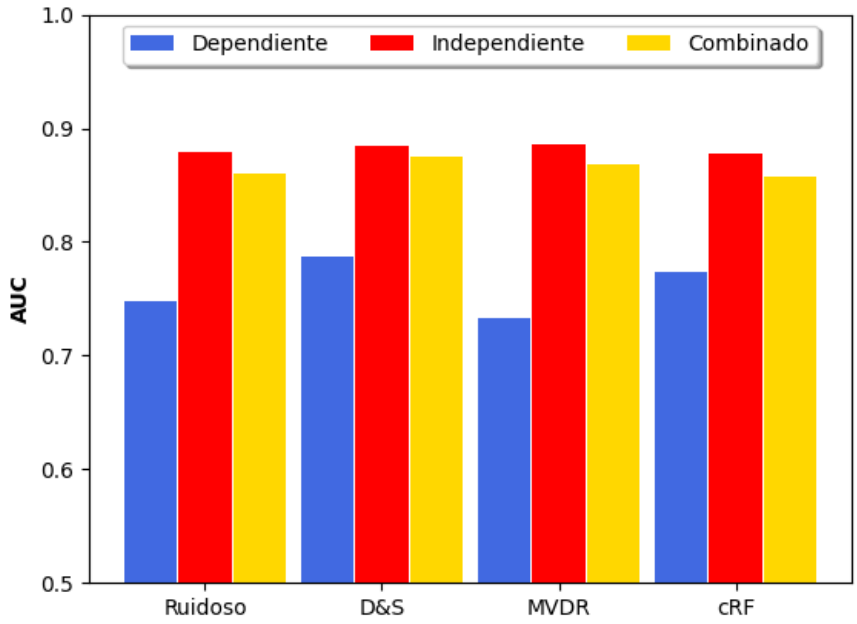


Figura 4.14: AUC de características dependientes del tiempo, independientes del tiempo y su combinación para los distintos *speech enhancement* en base *dynamic* 1.

La Figura 4.15 muestra la matriz de confusión del mejor sistema encontrado para la base dinámica real, es decir, entrenar con la base de datos simulada de *delay-and-sum* y evaluar con la base dinámica real mejorada con *delay-and-sum*. Al igual que en el caso simulado (Figura 4.9) y en el caso con base real estática (Figura 4.12), la mejora viene dada en la clase 0 y 1 principalmente, con un alza de 21 % y 159 % respectivamente. La precisión binaria en esta condición alcanza un 83 %, la cual es considerablemente alta y solo tiene una caída de 2 %, 2 % y 1 % frente a la condición telefónica, simulada y real estática respectivamente.

Es importante volver a destacar la robustez de resultados existente entre las distintas condiciones de bases de datos, lo que reafirma la consistencia de resultados de la red de dificultad respiratoria y la buena simulación del canal acústico.

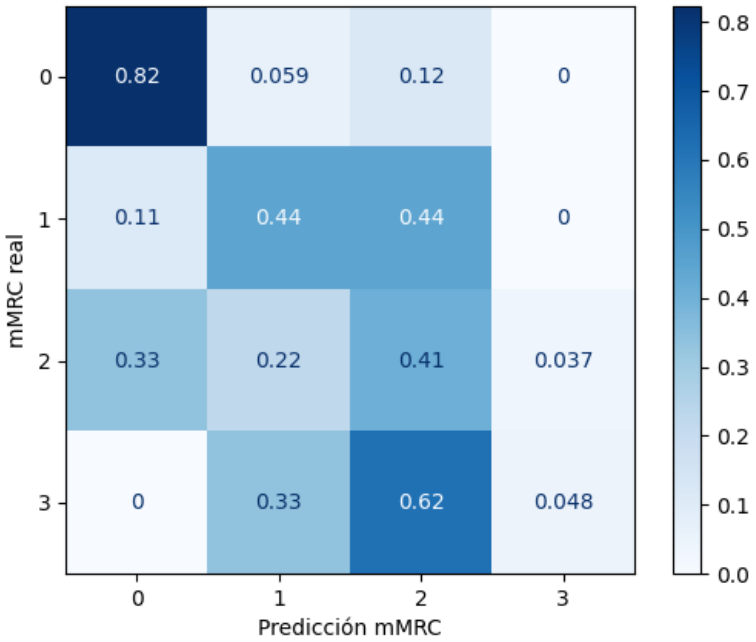


Figura 4.15: Matriz de confusión para sistema entrenado con base de datos simulada *delay-and-sum* y evaluado en condición *dynamic* 1 con *delay-and-sum*.

Capítulo 5

Conclusiones y trabajo futuro

En este trabajo se propone un sistema integral para la estimación de dificultad respiratoria en un entorno de interacción humano-robot. Este método se compone de dos etapas fundamentalmente, el *speech enhancement* y la estimación de disnea propiamente tal. El usuario debe realizar una serie de elocuciones controladas, las que son distorsionadas por la presencia de ruido y reverberación, problemas propios de una ambiente HRI. Para solucionar ese problema, la etapa de *speech enhancement* se encarga de limpiar las distintas señales y posteriormente introducirlas en el modelo de estimación de disnea, el cual obtiene características dependientes e independientes del tiempo a partir de las elocuciones, con lo cual al combinarlas permite estimar de forma robusta el grado de severidad de disnea de las personas.

En este trabajo se demostró la complementariedad existente entre las características independientes y dependientes del tiempo, las cuales a partir de la misma información (vocalización) estiman la severidad de la disnea, pero con un foco distinto. Por un lado, las características independientes del tiempo son métricas globales que se encargan de representar con un solo valor la totalidad del audio, y por otro, las dependientes del tiempo son mucho más detalladas, donde para cada *frame* del audio se estima un parámetro, lo que si bien aporta información valiosa, también las hace más susceptible al entorno. Además, se logró demostrar la complementariedad existente entre las distintas vocalizaciones (/ae-ae/, /sa-sa/ y conteo), las que fueron diseñadas cuidadosamente con el fin de obtener información relevante que permitiese representar el grado de disnea. La elocución /ae-ae/ entrega información similar a una vocal sostenida, pero evitando la supresión de voz intrínseca en los teléfonos inteligentes. La fonetización /sa-sa/ permite obtener información acerca de la cantidad de aire expelido por el usuario, lo que puede ser captado por la duración y los silencios entre las repeticiones. Por último, el conteo se encarga de representar el comportamiento espontáneo de los usuarios que buscan llegar a una meta (contar hasta el 30), lo que genera cambios en el tono, pronunciación, quiebres de voz, tos, etc. Si bien esta información es muy importante, es la más difícil de extraer, por lo que en general esta fonetización presentó el peor rendimiento en base a precisión y AUC, pero lo importante es que esta es altamente complementaria con las demás, con lo cual se aporta al objetivo principal y es útil para el sistema.

Ahora bien, centrándose en lo obtenido para la dificultad respiratoria en HRI, el sistema se entrenó con datos simulados derivados del modelado del canal acústico, lo que permitió un nivel de robustez para distintas bases de datos importante. El entrenar con distintas respuestas impulsivas reales (diferentes DOAs) y distintos ruidos, permite al modelo generalizar

de mejor forma, lo que se comprobó al evaluar el sistema en una habitación real con un ruido distinto, donde los resultados no presentaron una caída respecto a los datos simulados. El entrenamiento alineado entre las bases de datos de entrenamiento y evaluación, mostró sostenidamente mejores resultados que entrenar con base telefónica, inclusive utilizando *speech enhancement*. De hecho, cuando se evalúan en promedio los distintos *speech enhancement* entrenados de forma alineada, se obtiene una mejora de 20 % y 7 % en precisión y AUC para el caso estático, mientras que para el caso dinámico es de un 7 % y 5 %, al compararlos con el caso entrenado con base de datos telefónica y sin aplicar filtrado espacial. Estos resultados respaldan la hipótesis propuesta, donde por medio de la implementación de un modelo acústico y técnicas de reducción de ruido, se puede reducir la degradación de resultados del sistema en condiciones reales HRI.

Si bien en el caso telefónico la combinación de características siempre aportó en precisión y AUC, en el caso HRI no fue así, en gran parte por la complejidad del problema que se aborda. De igual forma, al combinar las características si hubo mejora en precisión, pero una leve disminución en AUC. Lo que indica que el fusionar características, aporta robustez al sistema y en lineas generales un rendimiento aceptable. Tal como se esperaba, las características dependientes del tiempo fueron considerablemente más susceptibles que las independientes del tiempo al ruido y la reverberación, pero al combinarlas entre sí, este efecto de variabilidad se redujo. Los resultados HRI alineados mostraron que en promedio, prácticamente no hay diferencias para la precisión y el AUC entre el caso estático y el dinámico real. Además, si se comparan los resultados de los modelos simulados con un promedio de las bases reales, la precisión se mantiene y el AUC sólo disminuye un 1 %. Lo que demuestra la robustez del sistema propuesto para las distintas configuraciones HRI reales. El haber obtenido precisiones y AUC similares entre las distintas condiciones de grabación de la base de datos, es decir, en condiciones estáticas y con movimiento, cumplen a cabalidad los objetivos propuestos en este trabajo de tesis, ya que se permite reducir la degradación del sistema HRI en su conjunto.

Como trabajo futuro queda pendiente la implementación de nuevos modelos de *speech enhancement* basados en inteligencia artificial, ya que es el motor que esta llevando el estado del arte y el optimizar estas redes de buena forma pueden llevar a mejores resultados de estimación de dificultad respiratoria en HRI. Este trabajo pretende sentar un precedente importante, debido a que no existen trabajos que combinen la dificultad respiratoria con la interacción humano-robot. De esta forma, se espera que los estudios puedan ir avanzando para que en un futuro cercano, esta tecnología sea implementada y utilizada en la vida cotidiana de los centros asistenciales.

Bibliografía

[1] Jahanmahin, R., Masoud, S., Rickli, J., y Djuric, A., "Human-robot interactions in manufacturing: A survey of human behavior modeling," Robotics and Computer-Integrated Manufacturing, vol. 78, pp. 102404–102413, 2022, doi:https://doi.org/10.1016/j.rcim.2022.102404.

[2] Ingrand, F. y Ghallab, M., "Deliberation for autonomous robots: A survey," Artificial Intelligence, vol. 247, pp. 10–44, 2017, doi:https://doi.org/10.1016/j.artint.2014.11.003.

[3] Breazeal, C., Dautenhahn, K., y Kanda, T., Social Robotics, pp. 1935–1972. Cham: Springer International Publishing, 2016, doi:10.1007/978-3-319-32552-1_72.

[4] Rossi, S., Ferland, F., y Tapus, A., "User profiling and behavioral adaptation for hri: A survey," Pattern Recognition Letters, vol. 99, pp. 3–12, 2017, doi:https://doi.org/10.1016/j.patrec.2017.06.002.

[5] Dunn, J., Runge, R., y Snyder, M., "Wearables and the medical revolution," Personalized medicine, vol. 15, no. 5, pp. 429–448, 2018.

[6] Mahloko, L. y Adebesin, F., "A systematic literature review of the factors that influence the accuracy of consumer wearable health device data," en Responsible Design, Implementation and Use of Information and Communication Technology (Hattingh, M., Matthee, M., Smuts, H., Pappas, I., Dwivedi, Y. K., y Mäntymäki, M., eds.), (Cham), pp. 96–107, Springer International Publishing, 2020.

[7] Smuck, M., Odonkor, C. A., Wilt, J. K., Schmidt, N., y Swiernik, M. A., "The emerging clinical role of wearables: factors for successful implementation in healthcare," NPJ Digital Medicine, vol. 4, no. 1, pp. 1–8, 2021.

[8] Cole, J., "Prosody in context: A review," Language, Cognition and Neuroscience, vol. 30, no. 1-2, pp. 1–31, 2015.

[9] Lella, K. K. y PJA, A., "A literature review on covid-19 disease diagnosis from respiratory sound data," arXiv preprint arXiv:2112.07670, 2021.

[10] World Health Organization, "Chronic respiratory diseases," 2022, https://www.who.int/news-room/fact-sheets/detail/chronic-obstructive-pulmonary-disease-(copd) (visitado el 16 de Febrero del 2022).

[11] Pramono, R. X. A., Low-complexity algorithms to enable long-term symptoms monitoring in chronic respiratory diseases. PhD thesis, Imperial College London, 2020.

[12] Alkhodari, M. y Khandoker, A. H., "Detection of covid-19 in smartphone-based breathing recordings: A pre-screening deep learning tool," PloS one, vol. 17, no. 1, pp. e0262448–e0262473, 2022.

[13] Lella, K. K. y Pja, A., "Automatic diagnosis of covid-19 disease using deep convolutional

neural network with multi-feature channel from respiratory sound data: cough, voice, and breath," Alexandria Engineering Journal, vol. 61, no. 2, pp. 1319–1334, 2022.

[14] Kranthi Kumar, L. y Alphonse, P., "Covid-19 disease diagnosis with light-weight cnn using modified mfcc and enhanced gfcc from human respiratory sounds," The European Physical Journal Special Topics, pp. 1–18, 2022.

[15] Mazumder, A. N., Ren, H., Rashid, H.-A., Hosseini, M., Chandrareddy, V., Homayoun, H., y Mohsenin, T., "Automatic detection of respiratory symptoms using a low-power multi-input cnn processor," IEEE Design & Test, vol. 39, no. 3, pp. 82–90, 2021.

[16] Olaronke, I., Ojerinde, O., y Ikono, R., "State of the art: A study of human-robot interaction in healthcare," International Journal of Information Engineering and Electronic Business, vol. 3, pp. 43–55, 2017, doi:10.5815/ijieeb.2017.03.06.

[17] Kyrarini, M., Lygerakis, F., Rajavenkatanarayanan, A., Sevastopoulos, C., Nambiappan, H., Chaitanya, K., Babu, A., Mathew, J., y Makedon, F., "A survey of robots in healthcare," Am. J. Public Health, vol. 106, pp. 1855–1857, 2016.

[18] Kolpashchikov, D., Gerget, O., y Meshcheryakov, R., "Robotics in healthcare," en Handbook of Artificial Intelligence in Healthcare, pp. 281–306, Springer, 2022.

[19] Fiorini, L., Coviello, L., Sorrentino, A., Sancarlo, D., Ciccone, F., D'Onofrio, G., Mancioppi, G., Rovini, E., y Cavallo, F., "User profiling to enhance clinical assessment and human–robot interaction: A feasibility study," International Journal of Social Robotics, pp. 1–16, 2022, doi:10.1007/s12369-022-00901-1.

[20] Manning, H. L. y Schwartzstein, R. M., "Pathophysiology of dyspnea," New England Journal of Medicine, vol. 333, no. 23, pp. 1547–1553, 1995.

[21] Launois, C., Barbe, C., Bertin, E., Nardi, J., Perotin, J.-M., Dury, S., Lebargy, F., y Deslee, G., "The modified medical research council scale for the assessment of dyspnea in daily living in obesity: a pilot study," BMC pulmonary medicine, vol. 12, no. 1, pp. 1–7, 2012.

[22] Willer, K., Fingerle, A. A., Noichl, W., De Marco, F., Frank, M., Urban, T., Schick, R., Gustschin, A., Gleich, B., Herzen, J., *et al.*, "X-ray dark-field chest imaging for detection and quantification of emphysema in patients with chronic obstructive pulmonary disease: a diagnostic accuracy study," The Lancet Digital Health, vol. 3, no. 11, pp. e733–e744, 2021.

[23] Barreiro, T. y Perillo, I., "An approach to interpreting spirometry," American family physician, vol. 69, no. 5, pp. 1107–1114, 2004.

[24] Huang, Y., Meng, S., Zhang, Y., Wu, S., Zhang, Y., Zhang, Y., Ye, Y., Wei, Q., Zhao, N., Jiang, J., *et al.*, "The respiratory sound features of covid-19 patients fill gaps between clinical data and screening methods," medRxiv, pp. 2020–2032, 2020, doi:10.1101/2020.04.07.20051060.

[25] Shoeibi, A., Khodatars, M., Alizadehsani, R., Ghassemi, N., Jafari, M., Moridian, P., Khadem, A., Sadeghi, D., Hussain, S., Zare, A., *et al.*, "Automated detection and forecasting of covid-19 using deep learning techniques: A review," arXiv preprint arXiv:2007.10785, 2020.

[26] Mohammad-Rahimi, H., Nadimi, M., Ghalyanchi-Langeroudi, A., Taheri, M., y Ghafouri-Fard, S., "Application of machine learning in diagnosis of covid-19 through

x-ray and ct images: a scoping review," Frontiers in cardiovascular medicine, vol. 8, pp. 638011–638036, 2021.

[27] Serrurier, A., Neuschaefer-Rube, C., y Röhrig, R., "Past and trends in cough sound acquisition, automatic detection and automatic classification: A comparative review," Sensors, vol. 22, no. 8, pp. 2896–2926, 2022.

[28] Suppakitjanusant, P., Sungkanuparph, S., Wongsinin, T., Virapongsiri, S., Kasemkosin, N., Chailurkit, L., y Ongphiphadhanakul, B., "Identifying individuals with recent covid-19 through voice classification using deep learning," Scientific Reports, vol. 11, no. 1, pp. 1–7, 2021.

[29] Smith, J. O., Mathematics of the discrete Fourier transform (DFT): with audio applications. Julius Smith, 2008.

[30] Khdour, T., Muaidi, P., Ahmad, A., Alqrainy, S., y Alkoffash, M., "Arabic audio news retrieval system using dependent speaker mode, mel frequency cepstral coefficient and dynamic time warping techniques," Research Journal of Applied Sciences, Engineering and Technology, vol. 7, pp. 5082–5097, 2014, doi:10.19026/rjaset.7.903.

[31] Qawaqneh, Z., Mallouh, A. A., y Barkana, B. D., "Deep neural network framework and transformed mfccs for speaker's age and gender classification," Knowledge-Based Systems, vol. 115, pp. 5–14, 2017.

[32] Verde, L., De Pietro, G., Ghoneim, A., Alrashoud, M., Al-Mutib, K. N., y Sannino, G., "Exploring the use of artificial intelligence techniques to detect the presence of coronavirus covid-19 through speech and voice analysis," Ieee Access, vol. 9, pp. 65750–65757, 2021.

[33] Teixeira, J. P., Oliveira, C., y Lopes, C., "Vocal acoustic analysis–jitter, shimmer and hnr parameters," Procedia Technology, vol. 9, pp. 1112–1122, 2013.

[34] Farrús, M., Codina-Filbà, J., Reixach, E., Andrés, E., Sans, M., Garcia, N., y Vilaseca, J., "Speech-based support system to supervise chronic obstructive pulmonary disease patient status," Applied Sciences, vol. 11, no. 17, pp. 7999–8010, 2021.

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

[36] Muguli, A., Pinto, L., Sharma, N., Krishnan, P., Ghosh, P. K., Kumar, R., Bhat, S., Chetupalli, S. R., Ganapathy, S., Ramoji, S., *et al.*, "Dicova challenge: Dataset, task, and baseline system for covid-19 diagnosis using acoustics," arXiv preprint arXiv:2103.09148, 2021.

[37] Orlandic, L., Teijeiro, T., y Atienza, D., "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.

[38] Saleheen, N., Ahmed, T., Rahman, M. M., Nemati, E., Nathan, V., Vatanparvar, K., Blackstock, E., y Kuang, J., "Lung function estimation from a monosyllabic voice segment captured using smartphones," en 22nd International Conference on Human-Computer Interaction with Mobile Devices and Services, pp. 1–11, 2020.

[39] Udugama, B., Kadhiresan, P., Kozlowski, H. N., Malekjahani, A., Osborne, M., Li, V. Y., Chen, H., Mubareka, S., Gubbay, J. B., y Chan, W. C., "Diagnosing covid-19: the disease

and tools for detection," ACS nano, vol. 14, no. 4, pp. 3822–3835, 2020.

[40] Ritwik, K. V. S., Kalluri, S. B., y Vijayasenan, D., "Covid-19 patient detection from telephone quality speech data," arXiv preprint arXiv:2011.04299, 2020.

[41] Verde, L., De Pietro, G., Ghoneim, A., Alrashoud, M., Al-Mutib, K. N., y Sannino, G., "Exploring the use of artificial intelligence techniques to detect the presence of coronavirus covid-19 through speech and voice analysis," IEEE Access, vol. 9, pp. 65750–65757, 2021.

[42] Rashid, M., Alman, K. A., Hasan, K., Hansen, J. H., y Hasan, T., "Respiratory distress detection from telephone speech using acoustic and prosodic features," arXiv preprint arXiv:2011.09270, 2020.

[43] Tang, S., Hu, X., Atlas, L., Khanzada, A., y Pilanci, M., "Hierarchical multi-modal transformer for automatic detection of covid-19," en Proceedings of the 2022 5th International Conference on Signal Processing and Machine Learning, pp. 197–202, 2022.

[44] Erdoğan, Y. E. y Narin, A., "Covid-19 detection with traditional and deep features on cough acoustic signals," Computers in Biology and Medicine, vol. 136, pp. 104765–104775, 2021.

[45] Fakhry, A., Jiang, X., Xiao, J., Chaudhari, G., Han, A., y Khanzada, A., "Virufy: A multi-branch deep learning network for automated detection of covid-19," arXiv preprint arXiv:2103.01806, 2021.

[46] Solera-Ureña, R., Botelho, C., Teixeira, F., Rolland, T., Abad, A., y Trancoso, I., "Transfer learning-based cough representations for automatic detection of covid-19.," en Interspeech, pp. 436–440, 2021.

[47] Ponomarchuk, A., Burenko, I., Malkin, E., Nazarov, I., Kokh, V., Avetisian, M., y Zhukov, L., "Project achoo: a practical model and application for covid-19 detection from recordings of breath, voice, and cough," IEEE Journal of Selected Topics in Signal Processing, vol. 16, no. 2, pp. 175–187, 2022.

[48] El Naqa, I. y Murphy, M. J., What Is Machine Learning?, pp. 3–11. Cham: Springer International Publishing, 2015, doi:10.1007/978-3-319-18305-3_1.

[49] LeCun, Y., Bengio, Y., y Hinton, G., "Deep learning," nature, vol. 521, no. 7553, pp. 436–444, 2015.

[50] Kanal, L. N., "Perceptron," en Encyclopedia of Computer Science, pp. 1383–1385, 2003.

[51] Gardner, M. W. y Dorling, S., "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," Atmospheric environment, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[52] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., *et al.*, "Recent advances in convolutional neural networks," Pattern recognition, vol. 77, pp. 354–377, 2018.

[53] Li, Z., Liu, F., Yang, W., Peng, S., y Zhou, J., "A survey of convolutional neural networks: Analysis, applications, and prospects," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999–7019, 2022, doi:10.1109/TNNLS.2021.3084827.

[54] Yu, F. y Koltun, V., "Multi-scale context aggregation by dilated convolutions," arXiv

preprint arXiv:1511.07122, 2015.

[55] Gholamalinezhad, H. y Khosravi, H., "Pooling methods in deep neural networks, a review," arXiv preprint arXiv:2009.07485, 2020.

[56] Wu, D., Wang, Y., Xia, S.-T., Bailey, J., y Ma, X., "Skip connections matter: On the transferability of adversarial examples generated with resnets," arXiv preprint arXiv:2002.05990, 2020.

[57] He, K., Zhang, X., Ren, S., y Sun, J., "Deep residual learning for image recognition," en Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[58] Salehinejad, H., Sankar, S., Barfett, J., Colak, E., y Valaee, S., "Recent advances in recurrent neural networks," arXiv preprint arXiv:1801.01078, 2017.

[59] Feng, W., Guan, N., Li, Y., Zhang, X., y Luo, Z., "Audio visual speech recognition with multimodal recurrent neural networks," pp. 681–688, 2017, doi:10.1109/IJCNN.2017.7965918.

[60] Lipton, Z. C., Berkowitz, J., y Elkan, C., "A critical review of recurrent neural networks for sequence learning," arXiv preprint arXiv:1506.00019, 2015.

[61] Hochreiter, S. y Schmidhuber, J., "Long short-term memory," Neural computation, vol. 9, pp. 1735–80, 1997, doi:10.1162/neco.1997.9.8.1735.

[62] Sun, Q., Jankovic, M., Bally, L., y Mougiakakou, S., "Predicting blood glucose with an lstm and bi-lstm based deep neural network," pp. 1–5, 2018, doi:10.1109/NEUREL.2018.8586990.

[63] Yu, Y., Si, X., Hu, C., y Zhang, J., "A review of recurrent neural networks: Lstm cells and network architectures," Neural computation, vol. 31, no. 7, pp. 1235–1270, 2019.

[64] Anguita, D., Ghelardoni, L., Ghio, A., Oneto, L., y Ridella, S., "The 'k'in k-fold cross validation," en 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 441–446, i6doc. com publ, 2012.

[65] Sheridan, T. B., "Human–robot interaction: status and challenges," Human factors, vol. 58, no. 4, pp. 525–532, 2016.

[66] Saleem, N. y Khattak, M. I., "A review of supervised learning algorithms for single channel speech enhancement," International Journal of Speech Technology, vol. 22, no. 4, pp. 1051–1075, 2019.

[67] Van Veen, B. D. y Buckley, K. M., "Beamforming: A versatile approach to spatial filtering," IEEE assp magazine, vol. 5, no. 2, pp. 4–24, 1988.

[68] Díaz, A., Mahu, R., Novoa, J., Wuth, J., Datta, J., y Yoma, N. B., "Assessing the effect of visual servoing on the performance of linear microphone arrays in moving human-robot interaction scenarios," Computer Speech & Language, vol. 65, pp. 101136–101155, 2021.

[69] Segers, L., Vandendriessche, J., Vandervelden, T., Lapauw, B. J., da Silva, B., Braeken, A., y Touhafi, A., "Cabe: A cloud-based acoustic beamforming emulator for fpga-based sound source localization," Sensors, vol. 19, no. 18, pp. 1–37, 2019.

[70] Xiao, Y., Yin, J., Qi, H., Yin, H., y Hua, G., "Mvdr algorithm based on estimated diagonal loading for beamforming," Mathematical Problems in Engineering, vol. 2017, 2017, doi:https://doi.org/10.1155/2017/7904356.

[71] Zhang, Z., Xu, Y., Yu, M., Zhang, S.-X., Chen, L., y Yu, D., "Adl-mvdr: All deep learning mvdr beamformer for target speech separation," en ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6089–6093, IEEE, 2021.

[72] Higuchi, T., Kinoshita, K., Ito, N., Karita, S., y Nakatani, T., "Frame-by-frame closed-form update for mask-based adaptive mvdr beamforming," en 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 531–535, IEEE, 2018.

[73] Pfeifenberger, L. y Pernkopf, F., "Blind speech separation and dereverberation using neural beamforming," Speech Communication, vol. 140, pp. 29–41, 2022.

[74] Liu, Y., Ganguly, A., Kamath, K., y Kristjansson, T., "Neural network based time-frequency masking and steering vector estimation for two-channel mvdr beamforming," en 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6717–6721, IEEE, 2018.

[75] Xiao, X., Zhao, S., Jones, D. L., Chng, E. S., y Li, H., "On time-frequency mask estimation for mvdr beamforming with application in robust speech recognition," en 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3246–3250, IEEE, 2017.

[76] Zhang, Z., He, B., y Zhang, Z., "X-tasnet: Robust and accurate time-domain speaker extraction network," arXiv preprint arXiv:2010.12766, 2020.

[77] Hao, Y., Xu, J., Shi, J., Zhang, P., Qin, L., y Xu, B., "A unified framework for low-latency speaker extraction in cocktail party environments.," en INTERSPEECH, pp. 1431–1435, 2020.

[78] Ochiai, T., Delcroix, M., Ikeshita, R., Kinoshita, K., Nakatani, T., y Araki, S., "Beam-tasnet: Time-domain audio separation network meets frequency-domain beamformer," en ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6384–6388, IEEE, 2020.

[79] Aroudi, A. y Braun, S., "Dbnet: Doa-driven beamforming network for end-to-end reverberant sound source separation," en ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 211–215, IEEE, 2021.

[80] Ren, X., Zhang, X., Chen, L., Zheng, X., Zhang, C., Guo, L., y Yu, B., "A causal u-net based neural beamforming network for real-time multi-channel speech enhancement.," en Interspeech, pp. 1832–1836, 2021.

[81] Tawara, N., Kobayashi, T., y Ogawa, T., "Multi-channel speech enhancement using time-domain convolutional denoising autoencoder.," en INTERSPEECH, pp. 86–90, 2019.

[82] Pandey, A., Xu, B., Kumar, A., Donley, J., Calamia, P., y Wang, D., "Multichannel speech enhancement without beamforming," en ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6502–6506, IEEE, 2022.

[83] Tzirakis, P., Kumar, A., y Donley, J., "Multi-channel speech enhancement using graph neural networks," en ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3415–3419, IEEE, 2021.

[84] Liu, C.-L., Fu, S.-W., Li, Y.-J., Huang, J.-W., Wang, H.-M., y Tsao, Y., "Multichan-

nel speech enhancement by raw waveform-mapping using fully convolutional networks," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 1888–1900, 2020.

[85] Zhang, W., Shi, J., Li, C., Watanabe, S., y Qian, Y., "Closing the gap between time-domain multi-channel speech enhancement on real and simulation conditions," en 2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp. 146–150, IEEE, 2021.

[86] Novoa, J., Mahu, R., Wuth, J., Escudero, J. P., Fredes, J., y Yoma, N. B., "Automatic speech recognition for indoor hri scenarios," ACM Transactions on Human-Robot Interaction (THRI), vol. 10, no. 2, pp. 1–30, 2021.

[87] Zhang, Z., Xu, Y., Yu, M., Zhang, S.-X., Chen, L., Williamson, D. S., y Yu, D., "Multi-channel multi-frame adl-mvdr for target speech separation," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 3526–3540, 2021.

[88] Chen, Z., Xiao, X., Yoshioka, T., Erdogan, H., Li, J., y Gong, Y., "Multi-channel overlapped speech recognition with location guided speech extraction network," en 2018 IEEE Spoken Language Technology Workshop (SLT), pp. 558–565, 2018, doi: 10.1109/SLT.2018.8639593.

[89] Luo, Y. y Mesgarani, N., "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," IEEE/ACM transactions on audio, speech, and language processing, vol. 27, no. 8, pp. 1256–1266, 2019.

[90] Mack, W. y Habets, E. A., "Deep filtering: Signal extraction and reconstruction using complex time-frequency filters," IEEE Signal Processing Letters, vol. 27, pp. 61–65, 2019.

[91] Vincent, E., Watanabe, S., Nugraha, A. A., Barker, J., y Marxer, R., "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," Computer Speech & Language, vol. 46, pp. 535–557, 2017.

[92] Tan, K., Xu, Y., Zhang, S.-X., Yu, M., y Yu, D., "Audio-visual speech separation and dereverberation with a two-stage multimodal network," IEEE Journal of Selected Topics in Signal Processing, vol. 14, no. 3, pp. 542–553, 2020.

[93] Farina, A., "Simultaneous measurement of impulse response and distortion with a swept-sine technique," en Audio Engineering Society Convention 108, Audio Engineering Society, 2000.

[94] Ghai, W. y Singh, N., "Literature review on automatic speech recognition," International Journal of Computer Applications, vol. 41, no. 8, pp. 42–50, 2012.

[95] Anguera, X., Wooters, C., y Hernando, J., "Acoustic beamforming for speaker diarization of meetings," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 7, pp. 2011–2022, 2007.

[96] Alvarado, E., Grágeda, N., Luzanto, A., Mahu, R., Wuth, J., Mendoza, L., y Yoma, N. B., "Dyspnea severity assessment based on vocalization behavior with deep learning on the telephone," Sensors, vol. 23, no. 5, pp. 2441–2459, 2023.

[97] Boersma, P., "Praat, a system for doing phonetics by computer," Glot. Int., vol. 5, no. 9, pp. 341–345, 2001.

[98] Bahmaninezhad, F., Wu, J., Gu, R., Zhang, S.-X., Xu, Y., Yu, M., y Yu, D., "A comprehensive study of speech separation: spectrogram vs waveform separation," arXiv preprint arXiv:1905.07497, 2019.