Universidad Juárez Autónoma de Tabasco

Tesis Doctoral

Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos

Que presenta

Noel Zacarias Morales

Para obtener el grado de:

Doctor en Ciencias de la Computación

Directores:

Dr. José Adán Hernández Nolasco

Dr. Pablo Pancardo García

Línea de generación y aplicación del conocimiento:

Computación distribuida inteligente

Institución sede:

Universidad Juárez Autónoma de Tabasco

d ***M***

UJAT
Universidad Juárez Autónoma de Tabasco
"Estudio en la duda. Acción en la fe"

División Académica de Ciencias y Tecnologías de la Información
DACyTI
Ciencia y Tecnología al servicio de la humanidad

Universidad Juárez Autónoma de Tabasco

Tesis Doctoral

Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos

Que presenta

Noel Zacarias Morales

Para obtener el grado de:

Doctor en Ciencias de la Computación

Comité tutoral:

Dr. José Adán Hernández Nolasco

Dr. Pablo Pancardo García

Jurado:

Dr. Miguel Antonio Wister Ovando

Dra. Betania Hernández Ocaña

Dr. Pablo Payró Campos

r. os d r d olas o

r. Pablo Pa ardo ar a

d M

UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO

DIVISIÓN ACADÉMICA DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN

F5: Liberación de dirección de tesis

Cunduacán, Tabasco., a 19 de abril de 2023.

MTE. Oscar Alberto González González
Director de la División Académica de Ciencias y Tecnologías de la Información
Presente

Por medio de la presente nos permitimos comunicarle que después de haber concluido la dirección de la Tesis: **"*Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos*"**, elaborada por el ***C. Noel Zacarias Morales***, del Doctorado en Ciencias de la Computación, consideramos que puede continuar con los trámites para la obtención del grado.

Sin otro particular, aprovechamos la ocasión para enviarle un cordial saludo.

A t e n t a m e n t e

Dr. José Adán Hernández Nolasco **Dr. Pablo Pancardo García**

c.c.p. Dr. Eddy Arquímedes García Alcocer. - Encargado del Despacho de la Coordinación de Posgrado.
Directores de Tesis.
Estudiante

RECIBIDO 20 ABR. 2023 D.A.C.Y.T.I. DIRECCIÓN

UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO

DIVISIÓN ACADÉMICA DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN

F6: Solicitud de jurado

Cunduacán, Tabasco, a 20 de abril de 2023.

MTE. Óscar Alberto González González
Director de la División Académica de Ciencias y Tecnologías de la Información
Presente

Por este medio me permito informarle que la tesis: "***Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos***", ha sido liberada por mis directores: **Dr. José Adán Hernández Nolasco** y **Dr. Pablo Pancardo García**, por lo que me dirijo a usted con la finalidad de solicitarle tenga a bien nombrar al jurado para que evalúe el citado trabajo.

Sin otro particular, aprovecho la ocasión para enviarle un cordial saludo.

Atentamente

Noel Zacarias Morales

Matrícula:	201H18002
Domicilio:	Rio Viejo 1ra Sección
Localidad:	Centro, Tabasco
Teléfono:	933-593-4244
E-mail:	n1.zkmt@gmail.com

RECIBIDO 20 ABR. 2023 D.A.C.T.I. DIRECCIÓN

c.c.p. Dr. Eddy Arquímedes García Alcocer. - Encargado del Despacho de la Coordinación de Posgrado.
Estudiante.

UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO

DIVISIÓN ACADÉMICA DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN

F7: Respuesta de jurado

Cunduacán, Tabasco, a 1 de junio de 2023

MTE. Óscar Alberto González González
Director de la División Académica de Ciencias y Tecnologías de la Información
Presente

En atención a los oficios girados por usted, en los que se nos designa como parte del jurado para efectuar la revisión de la tesis titulada ***"Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos"***, realizada por el ***C. Noel Zacarias Morales***, estudiante del Doctorado en Ciencias de la Computación, nos permitimos informarle que, en virtud de que ha atendido las observaciones realizadas, otorgamos nuestra aprobación para que continúe los trámites para la obtención del grado.

Sin otro particular, aprovechamos la ocasión para enviarle un cordial saludo.

Atentamente integrantes del jurado

Dra. Betania Hernández Ocaña

Dr. Miguel Antonio Wister Ovando

Dr. Pablo Payró Campos

RECIBIDO
01 JUN. 2023
D.A.C.Y.T.I.
DIRECCIÓN

c.c.p. Dr. Eddy Arquímedes García Alcocer. - Encargado del Despacho de la Coordinación de Posgrado.
Estudiante.

Recibí, Noel Zacarias 1/06/2023

Cunduacán, Tabasco a 1 de junio de 2023

Oficio No. 706/DACYTI/CP/2023

Asunto: Autorización de impresión de Tesis

C. Noel Zacarias Morales
Matricula: 201H18002

En virtud de que cumple satisfactoriamente los requisitos establecidos en el Reglamento General de Estudios de Posgrado vigente en la Universidad, informo a Usted que se autoriza la impresión del trabajo recepcional **"Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos"**, para presentar examen y obtener el grado de Doctor en Ciencias de la Computación.

Sin otro particular, aprovecho la ocasión para enviarle un afectuoso saludo.

Atentamente

MTE. Óscar Alberto González González
Director

UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO
DIVISIÓN ACADÉMICA DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN

C.c.p. Dr. Eddy Arquímedes García Alcocer. - Encargado del Despacho de la Coordinación de Posgrado DACYTI
Archivo.
Consecutivo.

MTE´OAGG/EAGA

Carretera Cunduacán-Jalpa Km. 1, Colonia Esmeralda, C.P. 86690.
Cunduacán, Tabasco, México.
Tel: (993) 358 1500 ext. 6727; (914) 336 0616; Fax: (914) 336 0870
E-mail: direccion.dacyti@ujat.mx

UNIVERSIDAD JUÁREZ AUTÓNOMA DE TABASCO

DIVISIÓN ACADÉMICA DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN

Cunduacán, Tabasco, a 1 de junio de 2023

Asunto: Cesión de derechos

MTE. Óscar Alberto González González
Director de la División Académica de Ciencias y Tecnologías de la Información
Presente

El que suscribe la presente, declara que el trabajo de tesis titulado, "**Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos**" es de mi autoría intelectual y por lo tanto cedo todos los derechos sobre este proyecto a la Universidad Juárez Autónoma de Tabasco, a la cual relevamos de cualquier sanción y asumimos responder a cualquier reclamo de derecho de autor ante las autoridades competentes

Atentamente

Dr. José Adán Hernández Nolasco

Dr. Pablo Pancardo García

Noel Zacarias Morales

RECIBIDO 1 JUN. 2023 D.A.C.Y.T.I. DIRECCIÓN

c.c.p. Dr. Eddy Arquímedes Garcia Alcocer. Encargado del despacho de la Coordinación de Posgrado.
Estudiante.

CARTA DE AUTORIZACIÓN

El que suscribe, autoriza por medio del presente escrito a la Universidad Juárez Autónoma de Tabasco para que utilice tanto física como digitalmente la Tesis de grado denominada "**Redes neuronales artificiales con atención auditiva selectiva para la separación y mejora de voz en ambientes complejos**", de la cual soy autor y titular de los Derechos de Autor.

La finalidad del uso por parte de la Universidad Juárez Autónoma de Tabasco de la tesis antes mencionada será única y exclusivamente para difusión, educación y sin fines de lucro; autorización que se hace de manera enunciativa más no limitativa para subirla a la Red Abierta de Bibliotecas Digitales (RABIO) y a cualquier otra Red Académica con las que la Universidad tenga relación Institucional.

Por lo antes mencionado, libero a la Universidad Juárez Autónoma de Tabasco de cualquier reclamación legal que pudiera ejercer respecto al uso y manipulación de la Tesis mencionada y para los fines estipulados en este documento.

Se firma la presente autorización en la Ciudad de Cunduacán, Tabasco a los 2 días del mes de junio del año 2023.

AUTRIZO

NOEL ZACARIAS MORALES

Dedicatoria

Con toda mi gratitud, dedico esta tesis al único y verdadero Ser Supremo y Perfecto, el Creador y Gobernador de todo lo que existe; ya que fue Él quien me otorgó la disciplina, constancia, paciencia, inspiración y conocimiento para llegar hasta este punto de mi formación académica.

Agradecimientos

Deseo expresar mi agradecimiento a las siguientes personas, ya que sin su colaboración no hubiera sido posible la realización de esta tesis:

- Dr. José Adán Hernández Nolasco.
- Dr. Pablo Pancardo García.

Adicionalmente, deseo expresar mi agradecimiento a las siguientes personas, ya que sin su colaboración no me hubiera sido posible llegar hasta este punto en mi formación académica:

- Dr. Arturo Corona Ferreira.
- Dr. Julián Javier Francisco León.
- Dr. Pablo Payró Campos.

Por último, agradezco a los siguientes miembros de mi familia por inspirarme día con día:

- Norma Morales Ramón.
- Maria Yanet Morales Ramón.
- Carlos Alberto Morales Ramón.

Resumen

Poder comunicarse en ambientes con fuentes de sonido de distintos tipos interviniendo simultáneamente es una habilidad normal para los humanos, por ejemplo, al sostener una conversación con otras personas resulta sencillo ignorar todas las demás fuentes de sonido presentes, que no son de interés en ese momento. Desafortunadamente, para los sistemas computacionales de procesamiento de audio, realizar esta tarea requiere de la integración de algoritmos que les brinden la capacidad para separar fuentes de audio diferentes cuando se encuentren mezcladas.

En esta investigación se implementaron diversos modelos de redes neuronales artificiales que pertenecen a tres categorías: perceptrón multicapa, red neuronal convolucional y red neuronal de unidades recurrentes. A estos modelos se les incluyó un mecanismo de atención inspirado en la atención auditiva humana para separar y mejorar una señal de voz, lo que permitió atenuar diversas señales ruido presentes en un ambiente sonoro complejo.

Después de analizar las características de los diversos conjuntos de datos públicos, se seleccionaron cuatro conjuntos de datos; como conjunto de datos de voz se utilizó el TIMIT, y como conjunto de datos de ruido se utilizó NoiseX-92, DEMAND, y PNL-100. Se crearon cinco horas de mezclas de audio en fragmentos de un minuto con relación señal a ruido muestreados uniformemente entre -10 dB y 10 dB para que los diferentes ruidos corrompieran la señal de voz. Posteriormente, se calculó el espectro de la magnitud de la señal mediante la transformada de Fourier de tiempo corto (STFT) con un tamaño de 256 puntos de frecuencias. Finalmente, se establecieron y entrenaron las arquitecturas básicas de los modelos para hacer una comparación equitativa entre los modelos y mostrar el efecto de los parámetros en cada red específica, así como el efecto del módulo de atención en el rendimiento general.

Para evaluar el rendimiento de los modelos, se utilizaron tres métricas de evaluación: la evaluación perceptual de la calidad del habla (PESQ), la inteligibilidad objetiva a corto plazo (STOI) y la relación señal-distorsión invariante (SI-SDR). Las cuales son las métricas estándar más utilizadas para evaluar el rendimiento de propuestas que buscan resolver el problema de la mejora del habla. Los resultados de los experimentos reflejaron que la inclusión de un mecanismo de atención permite que las arquitecturas de los modelos de redes neuronales sean menos complejas, sin sacrificar su eficiencia.

En cuanto a los resultados; al evaluar con la métrica PESQ, los experimentos mostraron que los modelos convolucionales obtuvieron resultados similares; sin embargo, el modelo perceptrón multicapa obtuvo el rendimiento más bajo; aun así, los cuatro modelos mostraron mejoras al incluir el módulo de atención. Respecto a los resultados de la métrica STOI, los resultados muestran que la red neuronal convolucional de una dimensión con el módulo de atención alcanzó el mejor resultado, seguida del modelo basado en la red de unidades recurrentes cerradas con el módulo de atención, el modelo perceptrón multicapa obtuvo el rendimiento más bajo. La métrica SI-SDR también mostró

resultados similares a las métricas PESQ y STOI.

Respecto a la capacidad de generalización de los modelos, desde una perspectiva general, la inclusión de un módulo de atención produjo mejoras en las capacidades de generalización de los modelos de perceptrón multicapa y basados en convolución (con algunas excepciones). Sin embargo, la inclusión del módulo de atención en el modelo de red de unidades recurrentes cerradas produjo resultados contrarios a los esperados en términos de la mayoría de las métricas.

Productos de la Investigación

En el siguiente listado se mencionan cada uno de los artículos científicos derivados de la investigación (se pueden consultar en el Anexo B).

1. Zacarias-Morales, N., Pancardo, P., Hernández-Nolasco, J. A., & Garcia-Constantino, M. (2021). Attention-Inspired Artificial Neural Networks for Speech Processing: A Systematic Review. Symmetry, 13(2), 214. MDPI AG. http://doi.org/10.3390/sym13020214.
2. Zacarias-Morales, N., Hernández-Nolasco, J. A., & Pancardo, P. (2022). Red neuronal convolucional ramificada con atención para la mejora de voz. Research in Computing Science, 151(6), 119-132.
3. Zacarias-Morales, N., Hernández-Nolasco, J. A., & Pancardo, P. (2023). Full Single-Type Deep Learning Models with Multihead Attention for Speech Enhancement. Applied Intelligence. https://doi.org/10.1007/s10489-023-04571-y.
4. Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P., & Olvera-López, J. A. (2023). Redes Neuronales Convolucionales con Atención para Atenuar Ruido en Señales de Audio. Komputer Sapiens, 15(2), 16-20.

En el siguiente listado se mencionan cada una de las ponencias derivadas de la investigación (se pueden consultar en el Anexo C).

1. Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P. (2022). Red Neuronal Convolucional Ramificada con Atención para la Mejora de Voz. XIV Congreso Mexicano de Inteligencia Artificial COMIA 2022.
2. Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P. (2022). Impacto de la Atención en Redes Neuronales Convolucionales y Recurrentes en la Mejora de Voz. Congreso Internacional de Ciencias y Tecnologias de la Informacion CYTI-ConXn 2022.

Contenido

Lista de figuras

Lista de tablas

Capítulo 1

Introducción

Cuando utilizamos tecnologías computacionales para recuperar voz humana en entornos reales, tales como una casa, la oficina o un coche, es común que también haya otros sonidos presentes y no solo la señal de la voz del usuario. Se puede tener el sonido de fondo de los aparatos cercanos como el acondicionador de aire, el claxon de los coches, otros altavoces, etcétera. Estos sonidos reducen la calidad de la señal de voz recuperada, haciendo que sea difícil de separar, complicada de entender y en el peor de los casos, que la señal de voz sea ininteligible.

El interés por esta investigación es principalmente científico, ya que en los últimos años investigadores del aprendizaje automático profundo han recalcado las posibles ventajas de incorporar mecanismos inspirados en las redes neuronales del cerebro humano para mejorar la construcción y resultados de modelos computacionales con inteligencia artificial, en donde se requiera atenuar o eliminar el ruido de una señal de voz, produciendo lo que se conoce como mejora de voz. En este sentido, los esfuerzos previos de investigadores han dado como resultado algoritmos que imitan el proceso de atención humano. Dichos modelos con mecanismos de atención también pueden ser aplicados a áreas como reconocimiento de la voz, reconocimiento de las emociones, identificación del lenguaje, entre otros.

El propósito de esta tesis fue conocer cuáles serían los resultados de incluir un módulo de atención en modelos de redes neuronales artificiales, pero empleando arquitecturas simples que demanden pocos recursos computacionales, si se les compara con modelos complejos. La idea surgió porque de modo común se puede pensar que un modelo complejo ofrece mejores resultados respecto a uno simple. Un aspecto importante a conocer en la evaluación de los resultados es el impacto en el rendimiento, medido a través de métricas de calidad para procesos de separación y mejora de voz.

La metodología empleada consistió en implementar diversos modelos de redes neuronales artificiales catalogadas como: perceptrón multicapa, red neuronal convolucional y red neuronal de unidades recurrentes. Cada uno de los modelos contaba con un mecanismo de atención inspirado en las redes neuronales del cerebro humano para recuperar y mejorar una señal de voz, teniendo como finalidad

atenuar las señales de ruido presentes en un audio construido. El paso posterior fue ejecutar cada uno de los modelos para obtener los resultados y analizarlos con métricas de rendimiento, que a su vez se compararán de modo equitativo con otras propuestas disponibles en la literatura.

Es necesario aclarar que para esta investigación, el término -separación y mejora de voz- se refiere al proceso de recuperar la señal de voz corrompida por señales de ruidos de diversos tipos, de modo que las señales de ruido puedan ser atenuadas o eliminadas conservando la señal de voz lo más comprensible posible; se considera necesario de aclarar, ya que en la literatura se puede encontrar que los términos -separación y mejora de voz- se refieren a procesos diferentes. También resulta imprescindible aclarar que el término -ambiente sonoro complejo- en esta investigación se refiere a los ambientes donde la naturaleza o localización de las fuentes de sonido varía continuamente.

En este capítulo se contextualiza la investigación que se desarrolla en esta tesis. Inicialmente, se describe el contexto y problema de la mejora de voz, luego se menciona la hipótesis, después se hacen las preguntas de investigación, más tarde se listan los objetivos, seguidamente se muestran alcances de la investigación y, por último, las contribuciones.

1.1. Contexto del Problema

El procesamiento de audio (o sonido) es un área de investigación dentro de las ciencias de la computación cuyas aplicaciones tienen alcance directo a nivel social, algunos ejemplos claros son los asistentes virtuales desarrollados por las grandes empresas tecnológicas como Microsoft, Amazon y Apple. Los sistemas computacionales desarrollados para análisis y procesamiento de audio hacen uso de diversos algoritmos que permiten la identificación, reconocimiento y autenticación de voz, lo que les permite en cierta medida una comunicación transparente con los usuarios finales.

Cuando dos personas sostienen una conversación es común que puedan existir otras fuentes de sonido de forma simultánea, sin embargo, cada persona pueda aplicar su habilidad humana para silenciar, o ignorar las fuentes que no son de su interés y con ello mantener una conversación de calidad. Solo en los casos que la conversación se presente en un ambiente restringido (como en una habitación), no será necesario aplicar la habilidad de atención auditiva; pero en la mayoría de los casos, las voces se mezclan con otras fuentes de sonido.

Por lo general, un escenario donde se debe recuperar y comprender una voz particular no representa un problema para los humanos, principalmente debido a que desde el nacimiento continuamente se mejora la capacidad de atención selectiva [52]; que les permite seleccionar de forma natural una fuente de sonido específica, separarla de los demás sonidos y comprenderla; un ejemplo de esto es, cuando se tiene una conversación con una o varias personas y en el lugar hay diversos sonidos ambientales o sonidos producidos por cualquier otra fuente de audio en ese momento.

Desafortunadamente, para los sistemas computacionales de procesamiento de audio, realizar esta

tarea requiere de la integración de algoritmos que les brinden la capacidad para separar fuentes de sonido diferentes cuando se encuentran mezcladas. En la literatura se pueden encontrar diversos algoritmos o modelos que permiten realizar esta separación de audio, los cuales incluyen el análisis de componentes independientes, la factorización de matrices no negativas y tensores, y más recientemente, el uso de técnicas de inteligencia artificial como el aprendizaje profundo y las redes neuronales.

Hasta la fecha estos modelos se han aplicado para separar fuentes de audio en dos escenarios, el primero a partir de mezclas capturadas por uno o más micrófonos en condiciones simuladas, y el segundo en mezclas creadas artificialmente; y los resultados varían dependiendo el escenario para los que han sido diseñados. Algunos de estos modelos se han aplicado a música para separar los instrumentos musicales o recuperar la voz de la persona que está cantando, y otros modelos se han aplicado a separar voces que se encuentran mezcladas artificialmente entre sí; y aunque han logrado resultados destacables en la extracción de voz, los modelos aún carecen de la capacidad para lidiar con todas las diversas fuentes de audio de los ambientes del mundo real que podrían llegar a mezclarse.

Uno de los casos que los investigadores encuentran más desafiante es cuando la mezcla de audio capturada se realiza en un ambiente sonoro complejo (no controlado), es decir, cuando la naturaleza o localización de las fuentes de audio varía continuamente y donde el entorno no sufre cambios premeditados por los investigadores, siendo el mejor ejemplo nuestro día a día.

La presente investigación se enfoca en estos escenarios complejos, ya que mediante la utilización de técnicas de inteligencia artificial busca diseñar y construir un modelo computacional inspirado en la capacidad humana de atención auditiva selectiva para recuperar y mejorar una señal de voz y hacer frente a la gran variabilidad de fuentes de sonido y diferentes valores de relación señal a ruido presentes un ambiente complejo.

1.2. Planteamiento del Problema

El procesamiento de voz es un área de investigación de gran interés, la numerosa cantidad de investigaciones realizadas durante la última década es evidencia de esto [68]. Todos estos trabajos resultan relevantes, ya que continuamente se buscan nuevos métodos, algoritmos o técnicas que permitan la construcción de sistemas de procesamiento de audio robustos, capaces de afrontar los diversos escenarios o características y que requieren que su procesamiento se realice de forma automática.

Uno de estos escenarios es el de la separación y mejora de voz de mezclas de audio capturadas con un único micrófono; y es que éste es uno de los escenarios más desafiantes debido a la limitada cantidad de información que una sola mezcla de audio puede generar [64]. Hasta este momento, las investigaciones para este escenario se han basado en la mejora de la precisión, latencia y costo

computacional de los modelos de separación.

Si bien, hasta la fecha los algoritmos y modelos propuestos han mostrado avances progresivos, aún existen limitaciones para los casos de separación y mejora de voz en mezclas de audio, por ejemplo aquellos casos con ambientes ruidosos; con reflexión acústica (eco de la voz); o los grabados en ambientes sonoros complejos (como cuando existen fuentes que se desplazan o salen del rango del micrófono mientras se captura la mezcla de las señales, o cuando existe la presencia de señales de distinta naturaleza).

El caso de la separación y mejora de voz de las mezclas de audio capturadas en ambientes con ruidos presentes mediante un solo micrófono es un escenario particularmente desafiante, ya que se combinan las restricciones en la cantidad de información disponible y la variabilidad de las fuentes de audio que podrían estar mezcladas.

Dotar a los modelos computacionales de procesamiento de audio con la capacidad humana de selección y atención de una fuente de sonido específica, como la voz, podría permitir que los modelos computacionales sean eficientes a pesar de las características variables de un ambiente sonoro complejo y que puedan ser aplicados a sistemas de procesamiento de audio para, por ejemplo, mejorar la capacidad de atención de asistentes virtuales, para detectar violencia verbal en una ubicación, o para ser utilizado por cualquier otro sistema computacional que requiera una voz sin la presencia de alguna otra fuente.

1.3. Hipótesis

Incorporar un módulo de atención inspirado en la atención auditiva humana en modelos computacionales basados en redes neuronales artificiales permite construir modelos computacionales con arquitecturas sencillas, que consumen menos recursos computacionales y con capacidad de recuperar la voz con inteligibilidad (STOI) de al menos el 85 % en el problema de separación y mejora de voz.

1.4. Preguntas de Investigación

1. ¿Qué componentes debe tener un modelo de red neuronal artificial que permita imitar la capacidad de atención auditiva selectiva?

2. ¿Qué características debe tener un modelo de red neuronal artificial que permita la separación y mejora de voz de una mezcla de audio capturada por un único micrófono?

3. ¿El uso de una red neuronal con atención es suficiente para recuperar la voz con la calidad necesaria para ser utilizada por otros sistemas computacionales de procesamiento de audio?

1.5. Objetivo de la Investigación

General

Crear modelos basados en redes neuronales artificiales imitando la capacidad de -atención auditiva- para la separación y mejora de una señal de voz capturada con un único micrófono en ambientes con ruido.

Específicos

- Analizar la problemática de la separación y mejora de voz en función de las características de los conjuntos de datos públicos y de los algoritmos de aprendizaje profundo disponibles.
- Diseñar modelos computacionales capaces de recuperar voz, imitando la capacidad humana de atención auditiva, utilizando redes neuronales artificiales y teniendo como datos de entrada una mezcla de audio de un solo canal.
- Construir y evaluar los modelos propuestos objetivamente mediante el uso de métricas estandarizadas para la separación y mejora de audio.

1.6. Alcances de la Investigación

Como toda investigación, los alcances tuvieron que ver con los objetivos del estudio, así como con el problema de investigación, por lo que se consideraron los siguientes:

- Conjunto de datos: se estableció que las mezclas de audio incluyeran únicamente una sola voz y una sola señal de ruido, con una relación señal a ruido de entre -10 dB a 10 dB.
- Conjunto de datos: Se utilizaron ruidos comúnmente presentes en entornos interiores (domésticos, oficina, y transporte) y entornos al aire libre (calle y naturaleza).

1.7. Limitaciones de la Investigación

Como se ha planteado en esta propuesta, un escenario de aplicación es la situación donde un par de personas se encuentran platicando y cada una de ellas debe descartar el ruido presente en el ambiente para poder capturar de modo eficiente la voz de su interlocutor. Sin embargo, puede darse el caso de un escenario donde hay más de dos voces presentes con ruidos ambientales, por lo que deben separarse y mejorarse las voces humanas. Esta investigación se encuentra limitada a una sola voz humana y ruidos ambientales comúnmente presentes en entornos interiores y al aire libre.

Los ruidos ambientales considerados en esta investigación son: domésticos (cocina, comedor y cuarto de lavado), oficina (pasillos, sala de reuniones, y oficina), áreas públicas (cafetería, restaurante, y estación de metro), transporte (autobús, automóvil, y metro), áreas al aire libre (campo deportivo,

parque, y río), y calle (plaza pública, terraza de cafetería, y parada de autobús); así como ruidos ambientales de balbuceo, ruidos de una fábrica, ruidos militares (aviones de combate, sala de máquinas, sala de operaciones, tanques, y ametralladora), y ruidos de canales de radio de alta frecuencia; por lo que otro tipo de ruido no han sido considerados.

El rango de decibelios por ruido comprende de los -10 dB a 10 dB, por lo que valores fuera de este rango no fueron estudiados. Sin embargo, el rango considerado comprende los valores comunes de ruido empleado en estudios similares.

1.8. Contribuciones

Si bien en esta investigación se implementaron técnicas y métodos generalmente conocidos en el área de aprendizaje profundo, los resultados de nuestros experimentos reflejaron que la inclusión de un mecanismo de atención permite que las arquitecturas de los modelos de redes neuronales artificiales sean menos complejas sin sacrificar eficiencia, además que al incluir el mecanismo de atención se logró, de cierta forma, dotar a los modelos computacionales de una capacidad similar a la atención auditiva humana con la que se puede atenuar o eliminar señales de sonido irrelevantes o no deseadas.

Adicionalmente, los resultados de este estudio son útiles como referencia para la construcción de soluciones en las que los modelos de separación y mejora de la voz no sean complejos, pero sí eficientes, lo que representa una oportunidad a la hora de implementar nuestra propuesta en dispositivos computacionales con recursos limitados, y utilizarse en situaciones como: asistencia para personas con discapacidad sensibles a los sonidos, mejorar la calidad de las llamadas de voz en lugares con mucho ruido, mejorar la capacidad de reconocimiento de voz de dispositivos con asistentes virtuales, o cualquier otra situación donde las señales de ruido representen restricciones o problemas para que la señal de voz pueda ser procesada con la menor cantidad de ruido presente.

1.9. Organización

Este trabajo se encuentra organizado de la siguiente manera. En el capítulo 2 se proporcionan los fundamentos y definiciones relacionados con la atención auditiva humana, el problema de separación y mejora de voz, así como los algoritmos y métricas del área de aprendizaje profundo utilizados en esta investigación. El capítulo 3 contiene los trabajos e investigaciones de la literatura relacionada con esta investigación. En el capítulo 4 se presenta el conjunto de datos de señales de voz y ruido utilizados en toda la investigación, se detallan las arquitecturas de red neuronal artificial seleccionadas y el módulo de atención que permite establecer una correspondencia a la capacidad de atención auditiva selectiva, y por último se describe la estrategia de entrenamiento de las arquitecturas. Los resultados y discusión de los experimentos realizados se muestran en el capítulo 5. Finalmente, en el capítulo 6 se presentan las conclusiones y pretensiones para trabajos futuros.

Capítulo 2

Marco Teórico

Los ambientes naturales se caracterizan por la abundancia de sonidos que bombardean el sistema auditivo y compiten por nuestra atención. Centrar la atención adecuadamente en la voz que nos interesa en entornos auditivos adversos es una hazaña que puede suponer un gran reto para muchas personas, más aún lo es para los algoritmos computacionales interesados en procesar las señales de voz. En este capítulo se encuentran los conceptos necesarios para comprender esta investigación, entre los que destacan la atención y las redes neuronales artificiales.

2.1. Atención Auditiva Selectiva

Con algunas excepciones, la capacidad del sistema auditivo humano para identificar sonidos en ambientes con diferentes fuentes de audio y de distinta naturaleza, resulta sorprendente, sobre todo, la capacidad de atención auditiva selectiva. En 1953, Colin Cherry [4] mencionó dentro de la literatura: ¿Cómo reconocemos lo que una persona dice cuando hablan otras al mismo tiempo (el "problema del cóctel")? ¿Sobre qué base lógica uno podría diseñar una máquina ("filtro") para llevar a cabo tal operación?

2.1.1. El Sistema Auditivo Humano

El sistema auditivo procesa el modo en que oímos y comprendemos los sonidos del entorno. Está formado por estructuras periféricas (como el oído interno, medio y externo) y por regiones cerebrales (como la corteza auditiva y la memoria). A continuación se mencionan los elementos considerados clave para esta investigación.

La cóclea es un hueso en forma de espiral ubicado en el oído interno y es fundamental en el sentido de la audición. La forma de espiral de la cóclea permite que diferentes frecuencias estimulen zonas específicas a lo largo de la espiral, lo que convierte las vibraciones de los sonidos en impulsos eléctricos en el conducto coclear mediante la estimulación mecánica de las células ciliadas [49], [38]. Estos

impulsos eléctricos son transportados desde la cóclea hasta el cerebro para su interpretación, es decir, la cóclea transforma las ondas sonoras en impulsos eléctricos que el cerebro puede interpretar como frecuencias individuales de sonido. La capacidad promedio de detección de frecuencias de sonidos en la cóclea humana suele estar entre 20 Hz y 20,000 Hz (ver la figura 2.1).

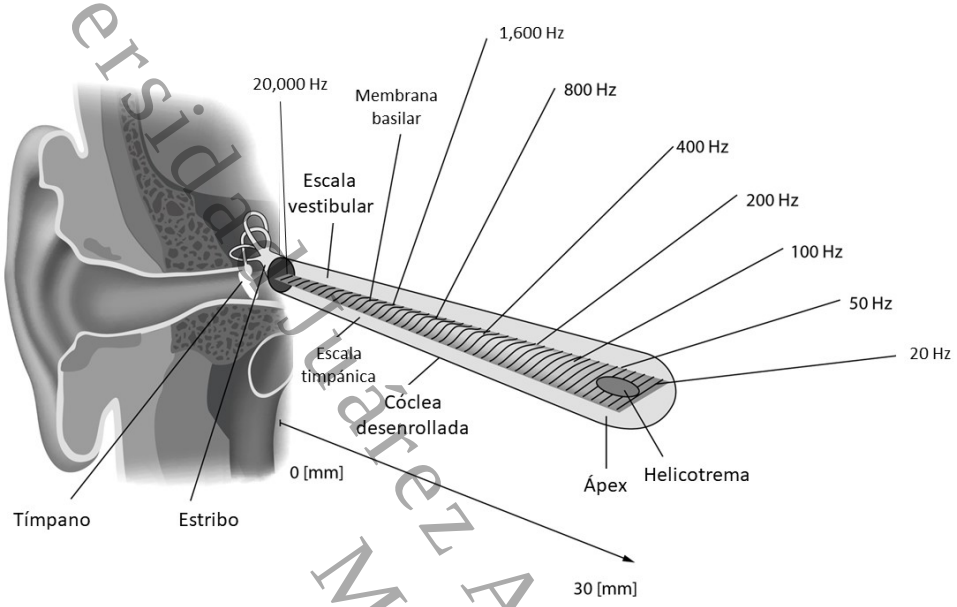

Figura 2.1: Anatomía del oído humano con la cóclea "desenrollada"mostrando la asignación de frecuencias a diferentes regiones de la membrana basilar.

La memoria es la capacidad cognitiva con la que los humanos retenemos y recuperamos nuestras experiencias pasadas para utilizar esa información en nuestro presente. Como proceso, la memoria se refiere a los mecanismos dinámicos asociados con el almacenamiento, la retención y la recuperación de información sobre experiencias pasadas [50]. En 1968, Atkinson and Shiffrin [1] propusieron un modelo que conceptualizaba la memoria en términos de tres almacenes de memoria: (1) Un almacén sensorial, capaz de almacenar cantidades relativamente limitadas de información durante periodos muy breves; (2) un almacén a corto plazo, capaz de almacenar información durante periodos algo más largos pero de capacidad también relativamente limitada; y (3) un almacén a largo plazo, de gran capacidad, capaz de almacenar información durante periodos muy largos, quizás incluso indefinidamente.

En los humanos, la información auditiva entra en nuestro sistema de memoria a través de un almacén denominado ecoico. La memoria auditiva o ecomemoria es un componente de la memoria sensorial responsable del almacenamiento a corto plazo de toda la información auditiva que recibimos del entorno. La memoria auditiva es básicamente un almacén de información que registra y almacena los estímulos sonoros antes de procesarlos [50]. La memoria ecoica también es el depósito inicial de gran parte de la información que acaba entrando en la memoria de corto y largo plazo.

2.1.2. La Atención

Según la psicología cognitiva y la neurociencia, la atención puede ser identificada como una actividad cognitiva que involucra aspectos identificables del comportamiento cognitivo [52], [51]. En la literatura no es posible encontrar una definición precisa y definitiva de lo que es la atención, esto se debe a que la -atención- no es un único concepto, sino un término que engloba diversos fenómenos psicológicos cognitivos, lo que provoca que los investigadores difieran de una definición que englobe los diferentes tipos de atención, como por ejemplo la atención visual, la atención auditiva, y otros tipos de atención sensorial; incluyendo atención consciente o inconsciente.

Una de las definiciones que posiblemente describe mejor el término atención es la de Richard Shiffrín [2], en la cual menciona que la -atención- se ha utilizado como un término para referirse a todos aquellos aspectos de la cognición humana que el individuo puede controlar y a todos los aspectos de la cognición relacionados con las limitaciones de recursos o de capacidad, incluidos los métodos para abordar dichas limitaciones. Por lo que resulta evidente que el término -atención- se emplea para referirse a distintos fenómenos y procesos, y no solo entre los psicólogos o neurocientíficos, sino también en el uso cotidiano que se le da a este vocablo.

La atención no es un proceso único o unidireccional, comúnmente se clasifica en dos funciones básicas distintas: (1) la atención Top-Down y (2) la atención Botton-Up. La atención Top-Down es un proceso selectivo que focaliza los recursos cognitivos en la información sensorial más relevante para mantener un comportamiento dirigido a uno o más objetivos en presencia de múltiples distracciones, es decir, la atención Top-Down implica la asignación voluntaria de recursos cognitivos a un objetivo, mientras que los demás estímulos sensoriales son suprimidos o ignorados; por esto se dice que la atención Top-Down es un proceso guiado por objetivos o expectativas. La atención Botton-Up es un proceso desencadenado a partir de estímulos sensoriales inesperados o sobresalientes, es decir, se refiere al proceso de orientación de la atención guiado puramente por estímulos que son sobresalientes debido a sus propiedades inherentes en relación con el entorno [19].

La atención selectiva se refiere al procesamiento diferencial de fuentes simultáneas de información; y en la naturaleza, estas fuentes pueden ser internas (memoria y conocimiento) así como externas (objetos y eventos ambientales) [18]; y aunque ambos tipos de fuentes son áreas de estudio, es común encontrar en la literatura investigaciones enfocadas solo a las fuentes externas.

Para comprender la atención a nivel cognitivo, se puede considerar el siguiente escenario: un individuo localizado en un ambiente no controlado se encuentra continuamente recibiendo una gran cantidad de estímulos sensoriales, mientras que simultáneamente, y con el objetivo de realizar sus actividades adecuadamente, de alguna manera debe seleccionar algunos estímulos para mejorar su procesamiento mientras ignora otros. Es decir, la atención es un término que se utiliza para referirse a esta capacidad de ignorar o inhibir algunos estímulos (o información) mientras se seleccionan otros con mayor prioridad para ser procesados, con lo cual se logra la capacidad de atención selectiva.

En el caso de la atención auditiva selectiva, se establece entonces que se refiere a la capacidad cognitiva de ignorar o inhibir ciertas fuentes de audio que no se consideren relevantes, mientras que se conservan aquellas consideradas de utilidad o con prioridad para una persona.

En relación con el proceso cognitivo de la atención auditiva en el cerebro humano (ver la figura 2.2), en su trabajo sobre la interacción de la atención focal Top-down y el seguimiento cortical de la voz, [23] menciona que en primer lugar, los estímulos sonoros (la voz) entran en los oídos y su información se procesa a lo largo de la primera etapa de la corteza auditiva (flecha 1). Una vez que esta información llega a las capas corticales, inicia el seguimiento cortical de la voz. A continuación, se inician dos flujos: (2a) un flujo auditivo Botton-Up, y (2b) un flujo de atención Botton-Up, que activan las áreas occipital y frontal del cerebro. Las áreas parieto-occipitales y frontales interactúan en la red de atención dorsal y ventral (flecha 3). Por último, las áreas frontal y parieto-occipital influyen en los procesos auditivos a través del procesamiento Top-Down: el control parieto-occipital Top-Down (flecha 4) precede al control Top-Down frontal (flecha 5).

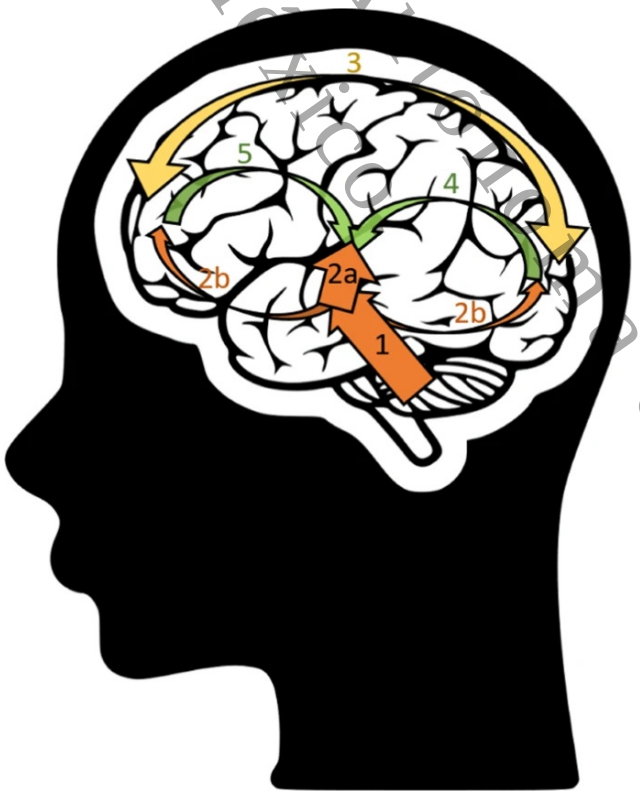

Figura 2.2: Procesos Botton-Up (en naranja) y Top-Down (en verde) implicados en el seguimiento de la voz [23].

2.2. Separación y Mejora de Voz

La separación de fuentes y la mejora de la voz son algunas de las tecnologías más estudiadas en el procesamiento de señales de audio, dentro de sus principales aplicaciones se encuentra el procesamiento de la voz, la música, y el ruido ambiental [61]. Comúnmente, la mejora de la voz se refiere al problema de segregar la voz y el ruido de fondo, mientras que la separación de fuentes se refiere al problema de extraer una o más fuentes específicas de audio mientras que se suprimen las demás fuentes que interfieren, así como el ruido presente; con esto se busca recuperar una fuente de audio específica lo más parecida a la fuente de audio original.

En los casos en los que la fuente de audio a recuperar es voz humana se busca que la señal de audio recuperada sea lo más similar posible a la "voz ideal", la cual es la voz original sin presencia de ningún tipo de ruido o interferencia de otras fuentes; en algunas investigaciones es posible encontrar la frase "voz ideal" o "voz de estudio" para referirse a la mejor versión posible de una voz.

La recuperación de la voz puede ser vista como el objetivo común en las investigaciones relacionadas con la separación de fuentes de audio, siendo el caso de los ambientes complejos el que presenta más retos para los investigadores. Los ambientes complejos (o no controlados) pueden ser considerados como aquellos escenarios donde las fuentes, los tipos de fuentes, y sus posiciones varían continuamente; es decir, una captura de audio en un ambiente complejo contendrá una mezcla de diversas fuentes de audio, como voces de personas, ruidos ambientales, en algunos casos reflexión acústica de las voces o de otras fuentes de audio (eco); además de ser posible que las diversas fuentes de audio estén presentes simultáneamente en la mezcla.

Una de las clasificaciones más comunes en la separación de audio está relacionada con la cantidad de micrófonos con los que se captura la mezcla de señales de audio, pudiendo ser multicanal o de un solo canal. La separación de fuente multicanal se refiere a los casos donde la separación se realiza con la mezcla de audios capturada con más de un micrófono, dicha captura contendrá mezclas diferentes dependiendo la localización, ángulo y características de los micrófonos. La separación de fuente de un solo canal hace uso de una única mezcla capturada con un único micrófono para poder realizar el proceso de separación de fuentes, siendo este caso el que tiene mayor impacto real, ya que una gran cantidad de dispositivos solo integran un único micrófono como sensor de sonido, entre ellos cámaras de vigilancia, radios, teléfonos celulares y computadoras.

2.2.1. Formulación del Problema

Para el problema de mejora de voz, la relación entre la voz y el ruido en el dominio del tiempo puede escribirse como (2.1):

$$y(t) = x(t) + n(t). \tag{2.1}$$

Donde *x(t)* es la señal de voz limpia y *n(t)* es el ruido añadido, dando como resultado que *y(t)* sea la señal de voz con ruido. Ahora bien, sea t el índice de tiempo, la señal puede representarse como $y = [y(1), ..., y(T)]$, donde T es la longitud del fragmento de audio. Al aplicar la transformada de Fourier de tiempo corto (STFT), podemos representar la señal acústica de (2.1) en el dominio de tiempo-frecuencia (TF) como (2.2):

$$Y(k,l) = X(k,l) + N(k,l). \tag{2.2}$$

Donde k es el índice de la banda de frecuencias, l denota el índice de la trama temporal, *Y(k,l)*, *X(k,l)*, y *N(k,l)* son los coeficientes STFT de la señal de voz ruidosa, la señal objetivo y la señal de ruido, respectivamente. Las definiciones anteriores son válidas únicamente para un micrófono de un solo canal. En este caso, la tarea de mejora de voz tiene como objetivo recuperar la señal de voz objetivo x de la señal de voz ruidosa y [67].

Cuando la solución se basa en el mapeo del espectrograma de magnitud, el objetivo de entrenamiento del modelo computacional es mapear una función no lineal F desde la señal de voz con ruido *y(t)* a una señal de voz limpia mejorada *x(t)*, como se escribe en (2.3):

$$y(t) \rightarrow^{F} x(t). \tag{2.3}$$

Debido a que existen problemas de variación rápida cuando se usa la señal de voz sin procesar (en el dominio del tiempo), el método basado en el mapeo se aplica habitualmente al espectrograma de magnitud de la señal de voz (dominio de la frecuencia), que se crea aplicando la transformada de Fourier de tiempo corto en una ventana temporal de un banco de filtros. Posteriormente, se realiza la operación inversa de la transformada de Fourier de tiempo corto para reconstruir el espectrograma de vuelta a la señal en el dominio del tiempo utilizando la información de fase de la señal de voz original con ruido.

2.2.2. Evaluación

El proceso de construcción de una solución para el problema de mejora de voz requiere indiscutiblemente del uso de métricas que permitan cuantificar los datos obtenidos con la finalidad de realizar una comparación objetiva. Para el caso de la separación y mejora de fuentes de audio, las métricas deben reflejar características como la precisión o la pérdida de calidad de la separación de una fuente de audio específica, y de esta manera saber qué tan bien las técnicas de separación de fuentes son adecuadas para una aplicación dada [7].

Para evaluar los resultados de esta investigación se utilizaron las siguientes métricas de evaluación: la evaluación perceptiva de la calidad de la voz (PESQ, por sus siglas en inglés) [41]; la inteligibilidad objetiva a corto plazo (STOI, por sus siglas en inglés) [17]; y la relación señal-distorsión invariable

en escala (SI-SDR, por sus siglas en inglés) [43], que son las métricas estándares más utilizadas para evaluar el desempeño de las propuestas para el problema de mejora de voz.

Evaluación Perceptiva de la Calidad de la Voz (PESQ)

La métrica de calidad de audio PESQ utilizada en la literatura tiene en cuenta características como nitidez del audio, volumen de la llamada, ruido de fondo, latencia variable o retardo en el audio, recorte e interferencias de audio. La prueba compara una señal de voz generada con la señal de voz original para crear un indicador totalmente imparcial y objetivo del audio real que se escucha. Esto es más preciso que otros métodos de medición de la calidad de audio que a menudo se basan en predicciones de la calidad de audio, ya que la intención de PESQ es reproducir la percepción del oído humano. PESQ devuelve una puntuación de -0.5 a 4.5, siendo las puntuaciones más altas las que indican una mejor calidad.

En la evaluación perceptiva de la calidad del habla, el modelo inicialmente alinea ambas señales (señal de voz generada con la señal de voz original) a un nivel de escucha estándar y las depura con un filtro de entrada. Seguidamente, se realiza una alineación temporal entre las dos señales, para ello calcula puntos de retardos definidos por un punto de inicio y un punto final. Una vez alineadas, se realiza una transformación auditiva mediante un modelo perceptual que mapea las señales en diferentes representaciones acústicas que intentan imitar al sistema auditivo humano, y así la calidad de la señal distorsionada es calificada con base en estas representaciones. Posteriormente, se extraen dos parámetros fundamentales basados en la diferencia entre las transformaciones de las señales, y que son agregados en frecuencia y tiempo. Finalmente, el modelo cognitivo devuelve una distancia entre la señal original y la señal degradada. El valor resultante sigue una escala comprendida entre -0.5 y 4.5, donde a mayor valor, mayor similitud habrá entre la señal de voz generada y la señal de voz original. Un valor de 5 significa que la señal de voz generada no tiene distorsión, mientras que un valor de -0.5 significa que existe una degradación severa [10].

Inteligibilidad Objetiva a Corto Plazo (STOI)

El enfoque tradicional de la evaluación objetiva de la inteligibilidad consiste en medir características específicas de la señal de voz que se sabe que son relevantes para la inteligibilidad. Por ejemplo, cuando la señal de voz se degrada por ruido aditivo, es posible predecir su inteligibilidad analizando la intensidad relativa de la voz y del ruido en diferentes bandas de frecuencia y a lo largo del tiempo. Las medidas objetivas que evalúan otras degradaciones de la señal, como la reverberación, la codificación de la voz o el enmascaramiento de tiempo-frecuencia, utilizan intensidades de envolvente por banda o representaciones espectro temporales como características, y las comparan con las características extraídas de la señal original no distorsionada [56].

La métrica de inteligibilidad objetiva en tiempo corto se basa en un coeficiente de correlación entre las señales temporales de referencia y procesadas en segmentos de tiempo corto superpuestos. En

primer lugar, las señales se descomponen mediante un banco de filtros de 1/3 de octava, se segmentan en ventanas de tiempo corto, se normalizan, se recortan y, a continuación, se comparan mediante un coeficiente de correlación. El paso de normalización compensa, por ejemplo, los diferentes niveles de reproducción, que no tienen un gran efecto negativo en la inteligibilidad. El recorte, por su parte, establece un límite superior para la degradación de una unidad de tiempo-frecuencia del habla. El recorte se utiliza para evitar cambios en la predicción de inteligibilidad una vez que el habla ya se ha considerado ininteligible. Los coeficientes de correlación resultantes corresponden a medidas de inteligibilidad intermedias a corto plazo para cada uno de los segmentos, que luego se promedian en un valor escalar correspondiente a la inteligibilidad del habla predicha para la señal procesada [53]. La métrica STOI se propuso originalmente para evaluar la inteligibilidad del habla ruidosa ponderada en tiempo-frecuencia y del habla mejorada. Los valores de STOI suelen oscilar entre 0 y 1 (por lo general se convierte como un porcentaje de inteligibilidad).

Relación Señal-Distorsión Invariable en Escala (SI-SDR)

La métrica relación señal-distorsión (del inglés signal-to-distortion ratio) es una medida de la calidad de la separación de fuentes. La métrica SDR se utiliza para evaluar la calidad de la separación de señales de audio después de que se ha sometido a un proceso de mejora. La métrica SDR mide la relación entre la señal original y la señal distorsionada después del procesamiento. Una relación SDR más alta indica una mejor calidad de separación.

En los años transcurridos desde que se propuso originalmente, se han planteado varios problemas. Aunque algunos de los problemas son específicos de la implementación, uno que persiste es que el SDR es fácil de engañar. La forma en que el SDR calcula sus componentes puede causar problemas cuando las puntuaciones se inflan artificialmente. La relación señal-distorsión invariable en escala (SI-SDR) surgió para remediar esto eliminando la dependencia de SDR de la escala de amplitud de la señal.

2.3. Inteligencia Artificial y Redes Neuronales Artificiales

La inteligencia artificial es un campo de las ciencias de la computación e ingeniería en la cual se estudia la comprensión computacional de lo que es comúnmente llamado -comportamiento inteligente-, así como la creación de algoritmos, modelos o dispositivos que exhiban ese comportamiento; siendo su meta lograr un comportamiento inteligente a un nivel similar al de los humanos [48]. Algunos de las subáreas de la inteligencia artificial son el aprendizaje automático y el aprendizaje profundo.

Ahora bien, el aprendizaje automático (del inglés "Machine Learning") es una de las subáreas de la inteligencia artificial que comprende los algoritmos y técnicas que permiten a los sistemas computacionales adquirir conocimiento propio a partir del procesamiento de datos. Es necesario hacer notar que, aunque existen una gran variedad de algoritmos de aprendizaje automático, en muchas

ocasiones los datos suelen ser más importantes que la selección del algoritmo [3], esto se debe a que el uso de datos insuficientes, de calidad pobre, incorrectos, irrelevantes, duplicados o faltantes, generan errores en los resultados.

En el aprendizaje automático, las relaciones entre los datos están formadas por el sistema de aprendizaje (los datos se ingresan y los resultados de generados a partir de esos datos); esto se denomina como -sistema de entrenamiento-. El sistema de aprendizaje automático relaciona los datos con los resultados y genera reglas que se convierten en parte del sistema, de esta manera cuando se introducen nuevos datos al sistema pueden surgir nuevos resultados que no formaban parte del conjunto de datos de entrenamiento [34].

En el aprendizaje automático, los datos reciben el nombre de -conjuntos de datos- (del inglés Data-Sets), y respecto a su uso, el aprendizaje automático se puede dividir en tres tipos: el aprendizaje supervisado, el aprendizaje no supervisado y el aprendizaje semi-supervisado. En el aprendizaje supervisado los datos del conjunto de datos tienen etiquetas que identifican su contenido; en el aprendizaje no supervisado los datos no tienen etiquetas que indiquen o identifiquen su contenido; el aprendizaje semi-supervisado es una combinación del aprendizaje supervisado y no supervisado, donde algunos datos del conjunto de datos están etiquetados y otros no lo están.

Adicionalmente, a la clasificación dada por las características de los datos, los algoritmos de aprendizaje automático pueden ser divididos en tres tipos principales: regresión, clasificación y agrupamiento. La regresión es una técnica de aprendizaje supervisado utilizada para predecir cantidades numéricas, este tipo de algoritmo incluye la regresión lineal y regresión lineal generalizada (también llamada análisis multivariable); la clasificación también es una técnica de aprendizaje supervisado, pero es utilizada para predecir cantidades categóricas, algunos de los algoritmos de clasificación utilizados en el aprendizaje automático son los árboles de decisión, los bosques aleatorios, k vecinos más cercanos, y máquinas de soporte vectorial; la agrupación es una técnica de aprendizaje no supervisado utilizada para agrupar datos similares, algunos de los algoritmos de agrupamiento son K-medias, desplazamiento medio y el agrupamiento jerárquico.

Por último, un importante subcampo del aprendizaje automático es el aprendizaje profundo (del inglés Deep Learning), el cual se enfoca en el estudio de las redes neuronales artificiales y los algoritmos de aprendizaje automático que contienen al menos dos capas ocultas.

Redes neuronales artificiales

Una red neuronal artificial (ANN, por sus siglas en inglés) o -red neuronal- como es común encontrarla en la literatura, está inspirada en el funcionamiento de las neuronas del cerebro humano. Dentro del cerebro humano, cada neurona recibe estímulos y decide si debe ser activada o no; una neurona activada enviará una señal eléctrica a otras neuronas conectadas; y entonces, si se dispone de una gran red de neuronas interconectadas, es posible aprender a reaccionar a diferentes entradas ajustando la

forma en que se conectan y cuán sensibles son a los estímulos [27].

Los modelos de redes neuronales artificiales mantienen el mismo principio de funcionamiento del cerebro humano, pero más enfocado a resolver problemas utilizando datos. Un componente clave de una red neuronal es la -neurona-, también llamada -nodo- por algunos autores, un nodo consiste en una o más entradas (X_i), sus pesos (W_l) y (b_l), una función de entrada (Z_l), y una función de activación (A_l) y una salida (Y), como en 2.4.

$$Y = A_l(\sum_{i=1}^{l} W_i x_i + b) \tag{2.4}$$

La función de entrada toma la suma de los pesos de todas las entradas, y la función de activación utiliza el resultado para determinar si el nodo debe ser activado o no. Los pesos se ajustan durante el proceso de aprendizaje para amplificarlos o reducirlos de acuerdo con los datos de entrada [27]. La figura 2.3 ilustra la estructura de un nodo o neurona:

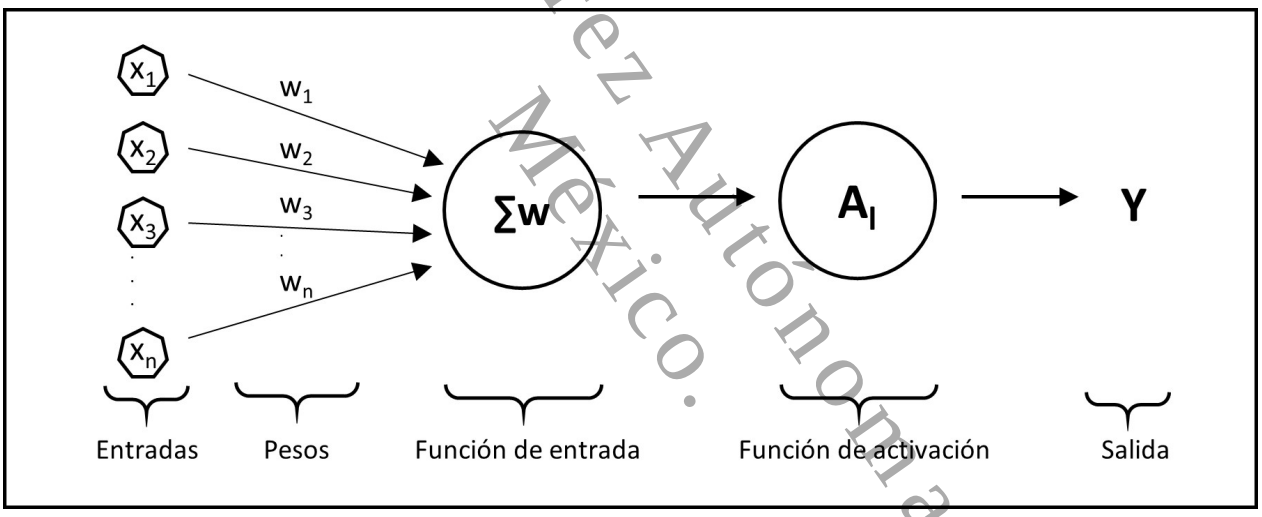

Figura 2.3: Estructura de un nodo o neurona.

Como base, una red neuronal de una capa es la estructura más simple que se puede construir (ver figura 2.4a) y su característica principal es que las neuronas que pertenecen a la misma capa no se pueden comunicar; seguidamente se encuentra la red neuronal multicapa (ver figura 2.4b), donde la primera capa es llamada -capa de entrada-, la última capa es llamada -capa de salida- y las capas intermedias son llamadas -capas ocultas-.

El diseño y creación de las redes neuronales profundas (redes con varias capas ocultas) involucra el uso de los hiperparámetros, que son características cuyos valores son establecidos e inicializados previo al proceso de entrenamiento; como por ejemplo la cantidad de capas de la red neuronal o la cantidad de neuronas de cada una de las capas. Algunos de los hiperparámetros en los modelos de redes neuronales profundas son los siguientes:

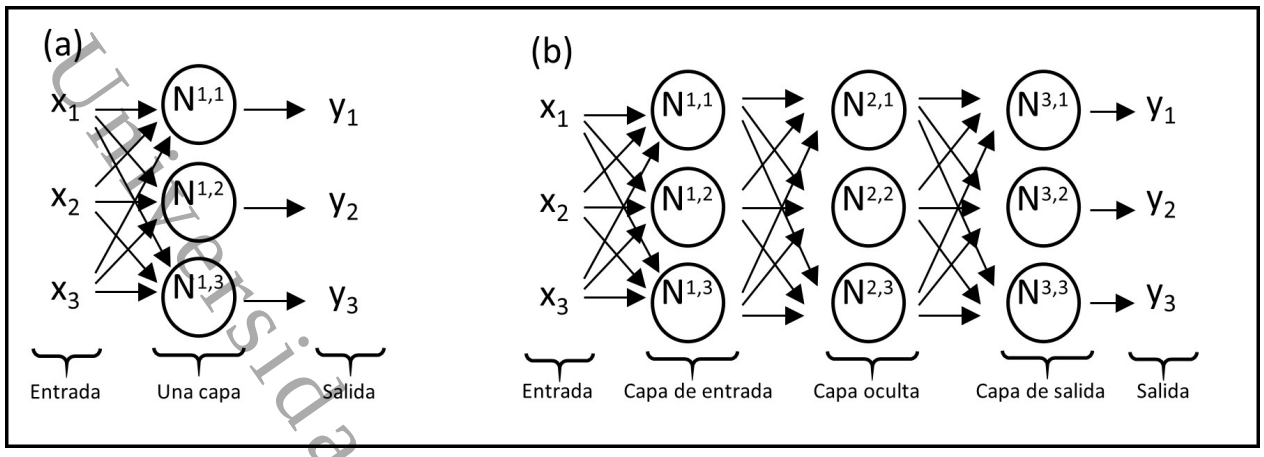

Figura 2.4: Estructura de una red neuronal artificial: (a) red neuronal de una sola capa, (b) red neuronal multicapa; los subíndices de cada neurona indican el número de capa y numero de neurona de esa capa respectivamente.

- Cantidad de capas ocultas.
- Cantidad de neuronas de cada capa.
- Pesos de inicialización.
- La función de activación.
- La función de costo.
- Un optimizador.
- Una tasa de aprendizaje.

El número de capas, neuronas y los pesos de inicialización son requeridos para la configuración inicial de una red neuronal; la función de activación es requerida para detener o proceder con la propagación; la función de costo, el optimizador y la tasa de aprendizaje son requeridos para realizar la propagación hacia atrás durante las tareas de aprendizaje supervisado (en este paso se calcula un conjunto de números que se utilizan para actualizar los valores de los pesos en la red neuronal para mejorar su precisión); por último, existe otro hiperparámetro que es útil si se requiere reducir el sobre entrenamiento (también llamado sobre ajuste) de un modelo, llamado tasa de dropout. De manera general, se podría considerar que la función de costo es el hiperparámetro más complejo de todos.

2.3.1. Aprendizaje en las Redes Neuronales Artificiales

Propagación Hacia Adelante

La propagación hacia adelante (del inglés Forward Propagation) se puede comprender como el con-

junto de procesos matemáticos, desde que los datos son introducidos en la red neuronal artificial, hasta que la red neuronal da genera un resultado. Las operaciones se realizan de izquierda a derecha, es decir, desde la capa de entrada, hacia la primera capa oculta, después a la segunda capa oculta, y así sucesivamente hasta que por último hacia la última capa que genera el resultado de salida. Esto es la propagación hacia adelante (ver figura 2.5).

Propagación Hacia Atrás

La propagación hacia atrás (del inglés Backward Propagation) sucede únicamente durante el entrenamiento de la red neuronal artificial. La propagación hacia atrás es el proceso que hace que la red neuronal aprenda de los datos de entrenamiento, es decir, sepa hacer cada vez mejores predicciones. La propagación hacia atrás se realiza con cualquier algoritmo de optimización, por ejemplo el descenso de gradiente estocástico y ADAM.

Después de que se realiza la propagación hacia adelante sucede lo siguiente, la red neuronal genera un resultado. A partir de ese resultado, se aplica una función de error; esta función calcula la cantidad de error (diferencia) que hay entre el resultado predicho y el resultado real. Una vez se calcula el error, este se propaga hacia atrás con el objetivo de mejorar los pesos de la red neuronal artificial y que la cantidad de error sea cada vez menor; es decir, sucede lo contrario que en la propagación hacia adelante, en la propagación hacia atrás se calcula el error al final y se propaga el error calculado desde la última capa hasta las primeras de derecha a izquierda (ver figura 2.5).

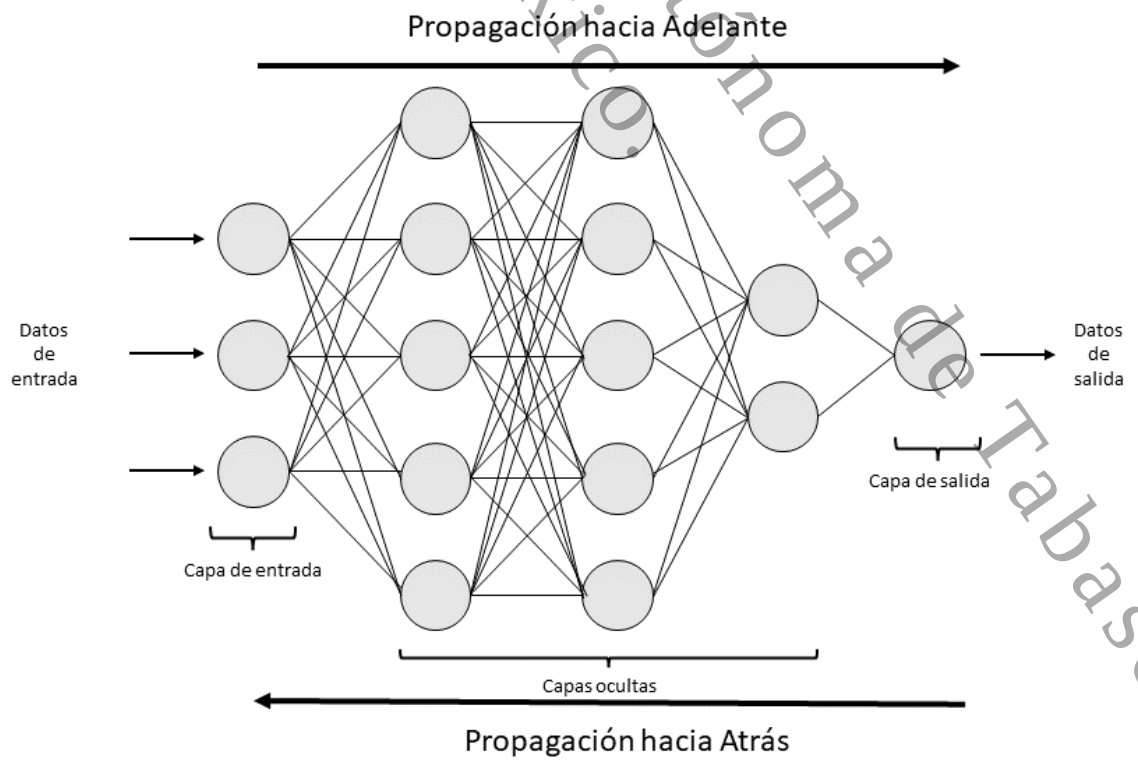

Figura 2.5: Representación gráfica de la propagación hacia adelante y la propagación hacia atrás.

Descenso de Gradiente

El descenso de gradiente es un algoritmo de optimización que se utiliza para entrenar redes neuronales artificiales. El objetivo del descenso de gradiente es minimizar la función de pérdida o error de la red neuronal. La idea detrás del descenso de gradiente es ajustar los pesos (W) de la red neuronal para minimizar el error de salida. El descenso de gradiente se basa en el cálculo del gradiente de la función de pérdida con respecto a los pesos de la red. El gradiente indica la dirección en la que se debe ajustar cada peso para minimizar la función de pérdida (ver figura 2.6).

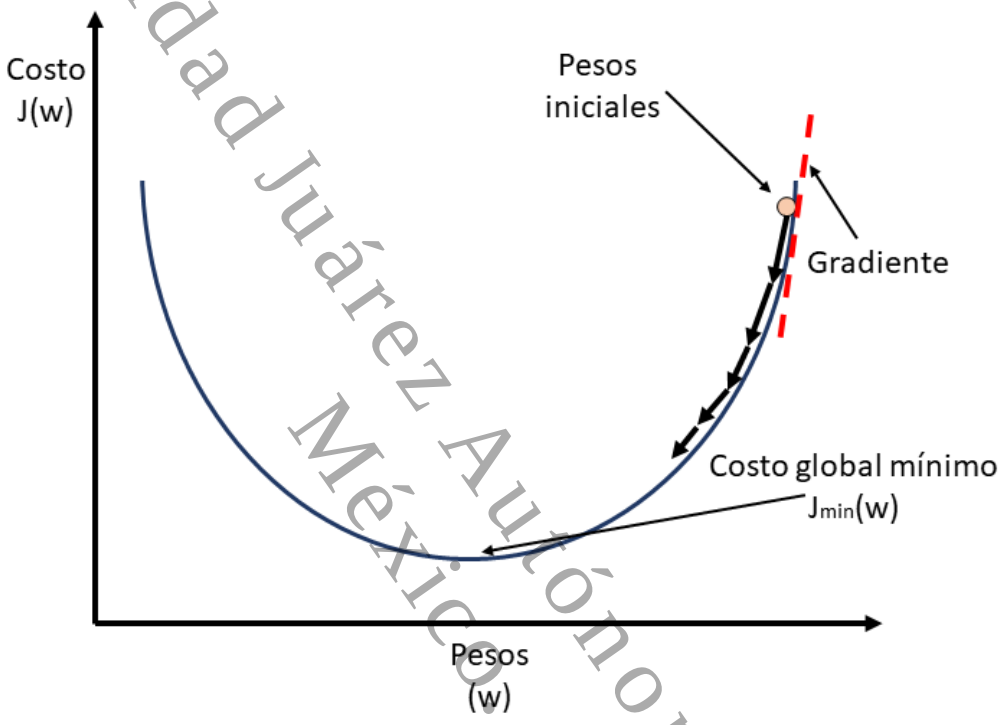

Figura 2.6: Representación gráfica del descenso de gradiente.

2.3.2. Perceptrón Multicapa

El perceptrón multicapa consiste en un sistema de neuronas simples interconectadas, o nodos, que es un modelo que representa un mapeo no lineal entre un vector de entrada y un vector de salida. Los nodos están conectados por pesos y señales de salida, que son una función de la suma de las entradas al nodo modificada por una simple función de transferencia no lineal, o de activación. Es la superposición de muchas funciones de transferencia no lineales simples, lo que permite al perceptrón multicapa aproximar funciones extremadamente no lineales. Si la función de transferencia fuera lineal, el perceptrón multicapa solo podría modelar funciones lineales.

Debido a su fácil cálculo de la derivada, una función de transferencia comúnmente utilizada es la función logística. La salida de un nodo es escalada por el peso de conexión y alimentada para ser una entrada a los nodos en la siguiente capa de la red. Esto implica una dirección de procesamiento

de la información, de ahí que el perceptrón multicapa se conozca como una red neuronal con flujo de datos hacia adelante.

La arquitectura de un perceptrón multicapa es variable, pero en general constará de varias capas de neuronas (ver figura 2.4b). La capa de entrada no juega ningún papel computacional, sino que simplemente sirve para pasar el vector de entrada a la red. Los términos vectores de entrada y salida se refieren a las entradas y salidas del perceptrón multicapa y pueden representarse como vectores simples. Un perceptrón multicapa puede tener una o varias capas ocultas y, finalmente, una capa de salida. Los perceptrones multicapa se describen como totalmente conectados, con cada nodo conectado a cada nodo de la capa siguiente y anterior [11].

2.3.3. Red Neuronal Convolucional

Las redes neuronales convolucionales (CNN, por sus siglas en inglés) son redes neuronales artificiales en las que el flujo de información tiene lugar en una sola dirección, de sus entradas hasta sus salidas. Las redes neuronales convolucionales también fueron inspiradas biológicamente, su arquitectura fue motivada por la corteza visual del cerebro, que consta de capas alternas de células simples y complejas [31]. Las arquitecturas convolucionales pueden ser encontradas en muchas variaciones; sin embargo, en general, constan de capas convolucionales y de agrupación, que se agrupan en módulos (ver figura 2.7). Los módulos a menudo se apilan uno encima del otro para formar modelos convolucionales más profundos [40]

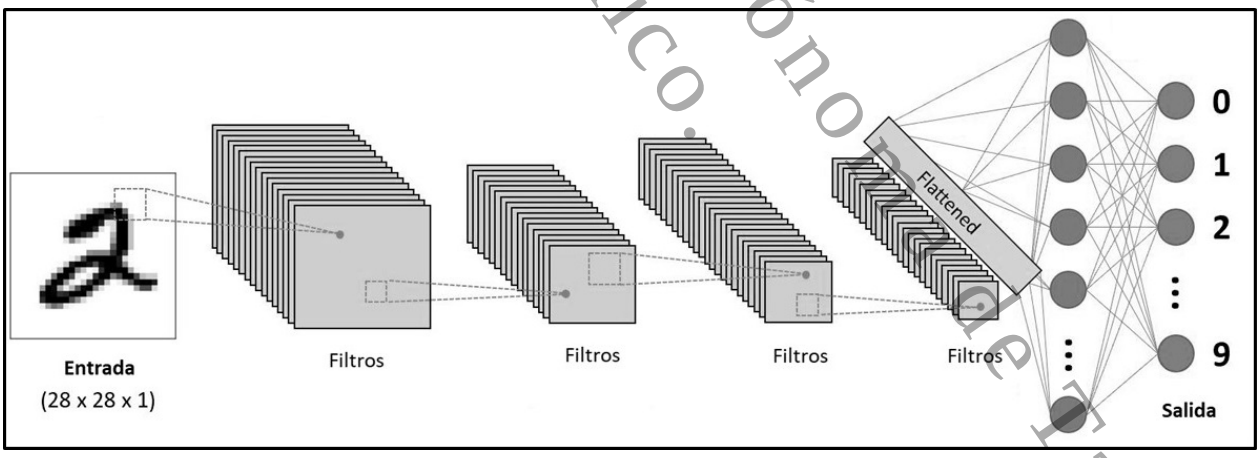

Figura 2.7: Representación gráfica de una red neuronal convolucional.

2.3.4. Red Neuronal Recurrente

Las redes neuronales recurrentes (RNN, por sus siglas en inglés) son ideales para procesar tareas que involucran entradas secuenciales, por ejemplo, tareas de procesamiento de lenguaje natural (texto). La característica típica de la arquitectura RNN es su conexión cíclica, que permite que la red neuronal recurrente posea la capacidad de actualizar su estado actual basándose en estados pasados y datos de

entrada actuales [66]. Desafortunadamente, existe un problema en esta arquitectura de red neuronal: el almacenaje de la información pasada durante mucho tiempo, es decir, dependencias a largo plazo (ver figura 2.8a).

Las redes neuronales de memoria de corto-largo plazo (LSTM-NN, por sus siglas en inglés) son un tipo especial de redes neuronales recurrentes que surgieron para superar el problema de las redes neuronales recurrentes con memoria explícita, ya que utiliza nodos o unidades especiales ocultas para recordar los parámetros en forma de entrada durante mucho tiempo. En la literatura también es posible encontrar un tipo especial de este tipo de red neuronal llamada red neuronal bidireccional de memoria de corto-largo plazo. Este tipo de arquitectura puede entrenarse en ambas direcciones de simultáneamente, con capas ocultas separadas (es decir, capas hacia adelante y capas hacia atrás) [66] (ver figura 2.8b).

Introducida por [5], la red de unidades recurrentes cerradas (GRU, por sus siglas en inglés) tiene como objetivo resolver el problema de fuga del gradiente que se produce en una red neuronal recurrente estándar. La red de unidades recurrentes cerradas también puede considerarse una variante de la red neuronal de memoria de corto-largo plazo, porque ambas están diseñadas de forma similar y, en algunos casos, producen resultados igualmente excelentes, aunque con una menor cantidad de parámetros entrenables (ver figura 2.8c).

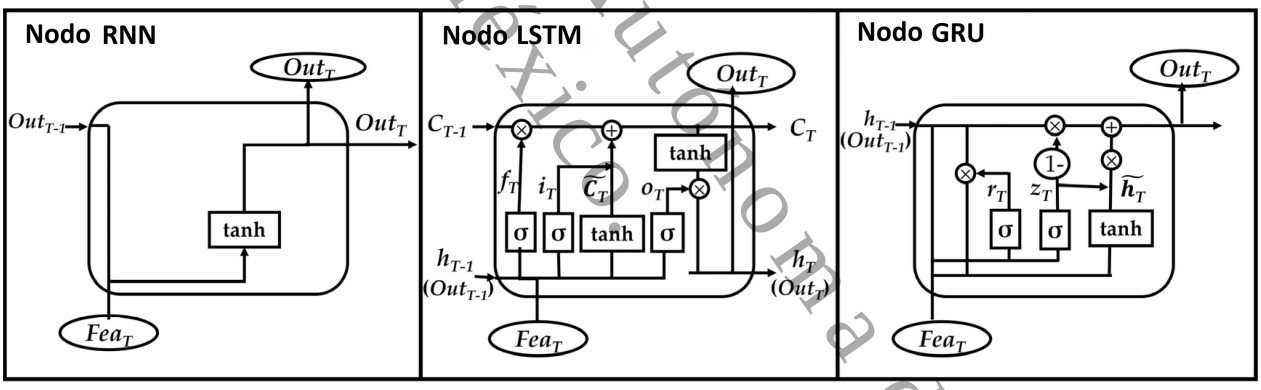

Figura 2.8: Estructura interna de los nodos de la red neuronal recurrente (a), red neural de memoria de corto-largo plazo (b) y la red de unidades recurrentes cerradas (c) [25].

2.3.5. Técnicas para Arquitecturas de Redes Neuronales Artificiales

Conexiones residuales

En las redes neuronales tradicionales, los datos fluyen a través de cada capa de forma secuencial: los datos de salida de una capa son los datos de entrada para la capa siguiente. Las conexiones residuales proporcionan rutas adicionales para que los datos lleguen a otras partes de la red neuronal omitiendo algunas capas.

Considere la siguiente secuencia de capas: capa i a capa $i + n$; sea F la función representada por estas capas, y definamos a x como la entrada para la capa i. En la configuración tradicional de las redes neuronales artificiales (secuencial), x simplemente pasa a través de estas capas una por una, y el resultado de la capa $i + n$ es $F(x)$. Una conexión residual que omite estas capas normalmente funciona como (2.5):

$$x- > F(x) + x. \tag{2.5}$$

$F(x)$ es el resultado de la secuencia de capas desde i hasta $i + n$. La conexión residual agrega x a esta salida, que luego se pasa a la siguiente capa (ver figura 2.9). Comúnmente, toda la arquitectura o fragmento de arquitectura que toma una entrada x y produce una salida $F(x) + x$ generalmente se denomina bloque residual [16]. Algunas veces, un bloque residual también puede incluir una función de activación como ReLU aplicada a $F(x) + x$, o una capa de normalización.

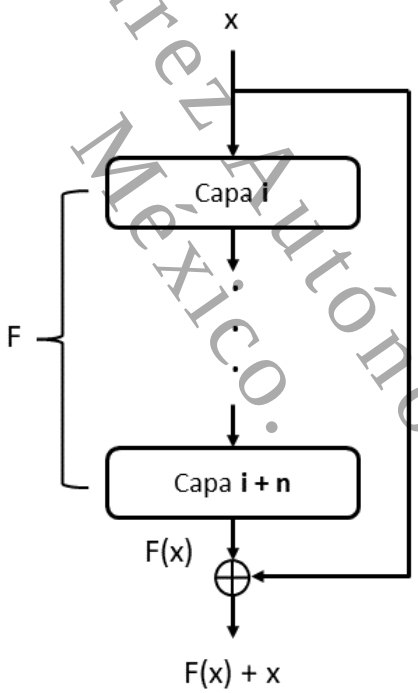

Figura 2.9: Representación gráfica de una conexión residual entre capas de una red neuronal artificial.

Para las redes neuronales artificiales comunes (secuenciales), el entrenamiento de una red neuronal profunda suele ser muy difícil, debido a problemas como la explosión y desvanecimiento de gradientes. Se conoce que el proceso de entrenamiento de una red neuronal artificial con conexiones residuales converge mucho más fácilmente con respecto a cuándo no se incluyen conexiones residuales, incluso si la red tiene cientos de capas.

Ramificaciones

Las ramificaciones en las arquitecturas de redes neuronales son secuencias de capas que temporalmente no tienen acceso a la información "paralela", que sigue pasando a capas posteriores (ver figura 2.9). En este tipo de arquitectura, cada una de las ramas procesa información de forma similar o diferente a las demás ramas, de forma que la información de salida de cada rama pueda ser utilizada por capas posteriores de la red neuronal artificial.

Las arquitecturas ramificadas tienen la ventaja de procesar paralelamente información, y que cada ramificación puede tener arquitecturas deferentes a las otras ramas, como por ejemplo, cada rama podría estar compuesta con diferentes tipos de redes neuronales artificiales (convolucionales, recurrentes, perceptron multicapa, etc.).

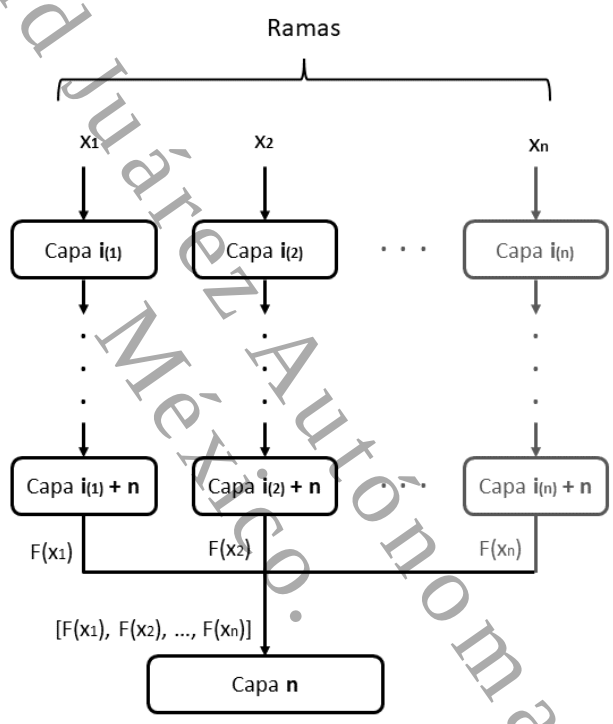

Figura 2.10: Representación gráfica de las ramificaciones en una red neuronal artificial.

2.3.6. Mecanismos de Atención

Los mecanismos de atención empleados en el aprendizaje profundo tuvieron su origen como una mejora a la arquitectura codificador-decodificador que se utilizaban en el procesamiento del lenguaje natural. Este mecanismo y sus variantes fueron aplicados posteriormente a otras áreas, como la visión por computadora y el procesamiento de voz. A continuación se menciona el funcionamiento general de los mecanismos de atención utilizados en los modelos secuencia-a-secuencia, los mecanismos de auto-atención y los mecanismos de atención de múltiples cabezas, aunque se debe considerar que existen muchas variantes de estos mecanismos en la literatura.

Mecanismos de atención en modelos secuencia-a-secuencia

Un modelo secuencia a secuencia es un modelo que toma una secuencia de elementos (palabras, letras, características de una imagen, etc.) y genera otra secuencia de elementos. Este tipo de modelo utiliza la arquitectura codificador-decodificador, donde el codificador (por lo general una red neuronal recurrente de tipo LSTM) es el encargado de procesar los datos de entrada y codificarlos en un vector de contexto (el último estado oculto de la LSTM). Este vector se espera que sea una recopilación o resumen de los datos de entrada, ya que este vector es el estado oculto inicial del decodificador (los estados intermedios codificador se descartan); en otras palabras, el codificador lee los datos de entrada y trata de darles sentido antes de resumirlos. El decodificador (formado por unidades recurrentes o LSTM) toma el vector de contexto y produce los datos de salida en orden secuencial.

Los mecanismos de atención que tuvieron su origen en las arquitecturas codificador-decodificador utilizando LSTM permiten resaltar dinámicamente las características relevantes de los datos de entrada. La idea central detrás del mecanismo de atención es no descartar los estados intermedios del codificador, sino utilizarlos para construir los vectores de contexto requeridos por el decodificador para generar los datos de salida, calculando una distribución de pesos en la secuencia de entrada, y asignando valores más altos a los elementos más relevantes, y pesos más bajos a los elementos menos relevantes [9].

En 2015, [26] introdujo la atención local. En la atención local, primero elige una posición en los datos secuenciales de origen. Esta posición determinará una ventana de datos a las que el modelo buscará prestar atención. Calcular la atención local durante el entrenamiento es más complicado y requiere técnicas como el aprendizaje por refuerzo para entrenar. La idea de la atención global (o atención suave) es utilizar todos los estados ocultos del codificador al calcular cada vector de contexto. El inconveniente de un modelo de atención global es que tiene que atender a todos los datos secuenciales del lado de la fuente para cada dato objetivo, lo que es computacionalmente costoso.

Computacionalmente, existe una forma interesante de comprender la atención. La atención se refiere a permitir el acceso del modelo a la memoria; y en el caso de los mecanismos de atención en las redes neuronales artificiales, la memoria no es más que todos los estados ocultos del codificador. El modelo elige que recuperar de la memoria.

Mecanismos de auto-atención

En 2018 se publicó el trabajo [58], en el que se proponen las arquitecturas basadas en transformadores (Transformers, por su traducción en inglés). Los transformadores se utilizan principalmente para modelar tareas de procesamiento de lenguaje natural y evitan el uso de las redes neuronales de tipo recurrente; en su lugar, confían totalmente en los mecanismos de auto-atención (Self-Attention, por su traducción en inglés) para obtener representaciones globales entre las entradas y las salidas.

Para obtener estas representaciones, cada entrada se multiplica con un conjunto de pesos para las claves K, un conjunto de pesos para las consultas Q y un conjunto de pesos para los valores V (ver

la figura 2.11). El módulo de auto-atención toma n entradas y devuelve n salidas, y permite que las entradas interactúen y descubran a que le deben poner más atención. Los resultados son los cálculos de estas interacciones junto con los valores de los puntajes de atención. A menudo, este módulo de auto-atención también es llamado atención del producto de puntos escalados (Scaled Dot-Product Attention, por su traducción en inglés).

En el mismo trabajo también presenta un mecanismo adicional denominado atención de múltiples encabezados (Multi-Head Attention, por su traducción en inglés). La atención de múltiples encabezados es el uso simultáneo de múltiples mecanismos de auto-atención en paralelo para calcular y seleccionar información múltiple de la información de entrada. Cada mecanismo de auto-atención se enfoca en diferentes partes de la información de entrada y luego los concatena para formar los datos de salida del mecanismo de atención (ver figura 2.11).

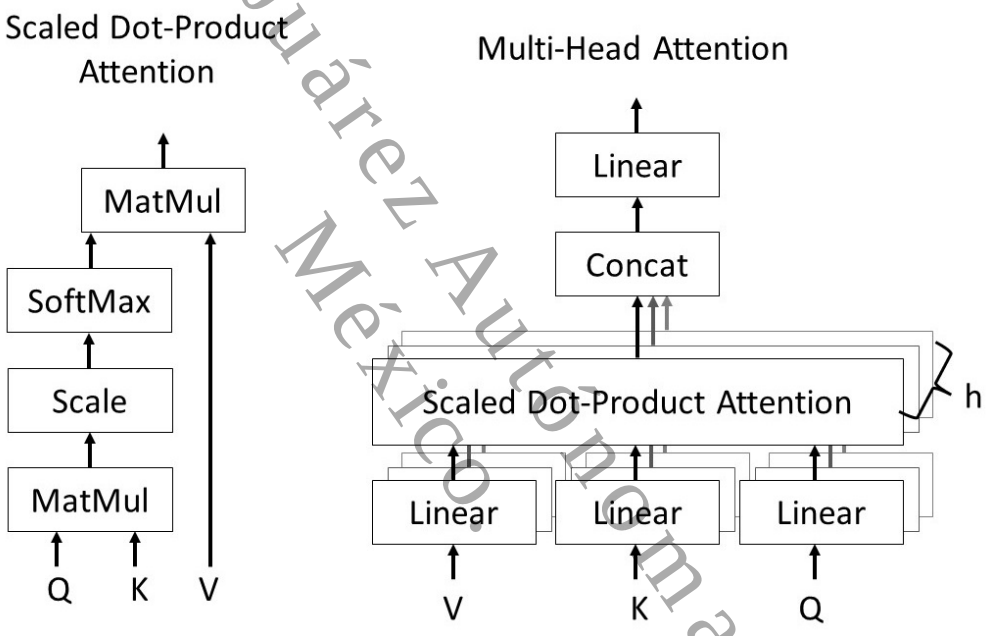

Figura 2.11: (Izquierda) Atención de producto-punto escalado. (Derecha) La atención de múltiples encabezados consiste en varias capas de atención que funcionan en paralelo [58].

Las salidas de atención independientes se concatenan y se transforman linealmente en la dimensión esperada. Intuitivamente, los encabezados de atención múltiples permiten atender a partes de la secuencia de forma diferente y se puede expresar como (2.6):

$$MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [head_1, ..., head_h]W_0 \tag{2.6}$$

Donde (2.7):

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{2.7}$$

Y donde W son todas las matrices de parámetros entrenables. La atención de múltiples encabezados es un módulo que utiliza la atención de producto punto escalado, que es un mecanismo de atención en el que los productos de puntos se escalan de forma $\sqrt{d_k}$. Formalmente, tenemos una consulta $\boldsymbol{Q}$, una clave $\mathbf{K}$ y un valor $\mathbf{V}$, y calculamos la atención como (2.8):

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V} \tag{2.8}$$

La figura 2.12 detalla el flujo de los datos dentro de la atención de múltiples encabezados desde los datos de entrada X hasta los datos de salida M.

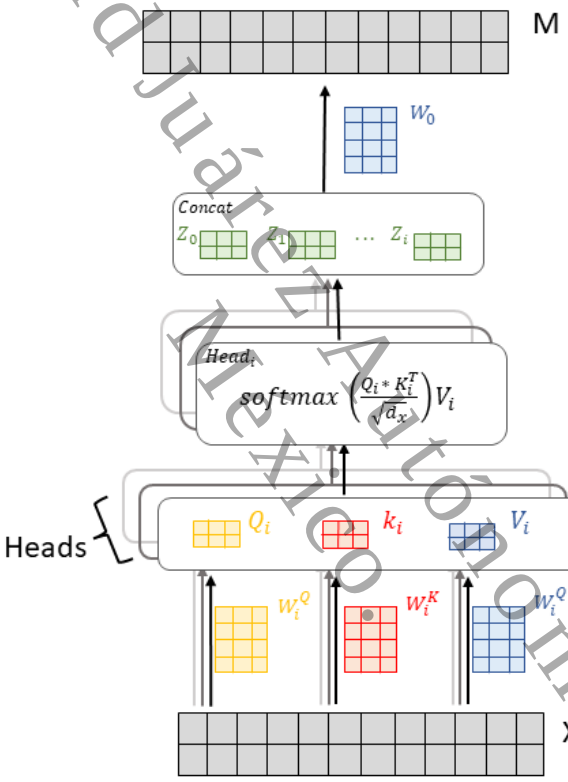

Figura 2.12: Representación gráfica del flujo de datos en la atención de múltiples encabezados (Multi-Head Attention).

2.4. Procesamiento de Señales de Sonido

Todos los equipos, dispositivos y sistemas trabajan con señales de información, es decir, con magnitudes variables que transmiten la información necesaria para el sistema. En el caso de los sistemas de procesamiento de sonido, la información es la forma de onda del sonido. La señal original es el sonido mismo, tal como llega al elemento al micrófono. Los micrófonos convierten la señal sonora capturada en una señal eléctrica.

Idealmente, la señal eléctrica capturada por el micrófono deberá tener exactamente la misma forma de onda que la señal de sonido, con un mero cambio de unidades: la señal de sonido es una presión sonora, mientras que la señal eléctrica es una tensión o voltaje. Por eso se dice que la señal eléctrica es una representación analógica de la señal de sonido [30].

La figura 2.13 muestra una señal de sonido y su analogía eléctrica, y aunque las amplitudes se muestran diferentes, no implica ninguna relación de tamaño entre las señales, simplemente porque no es posible comparar una presión con una tensión eléctrica.

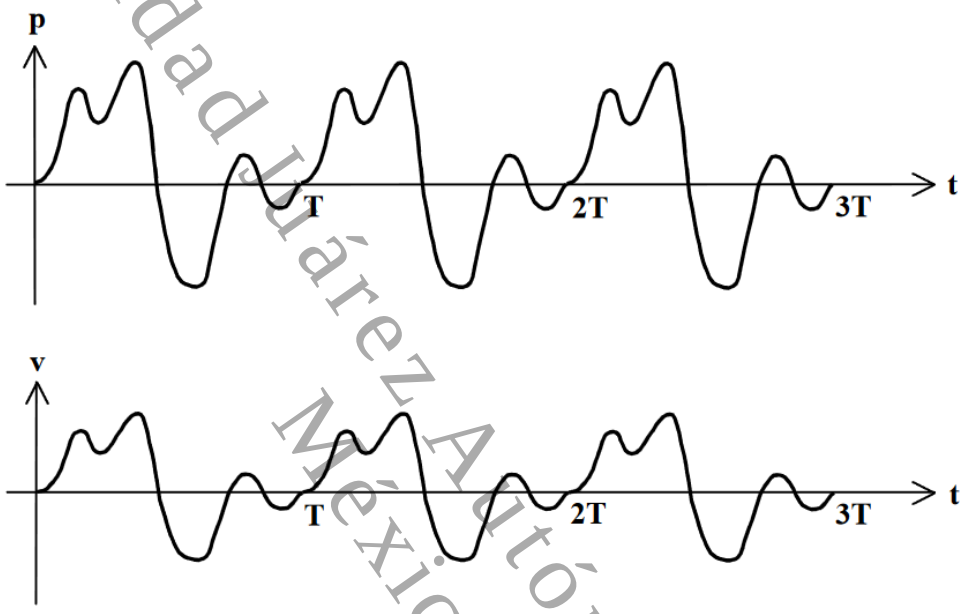

Figura 2.13: Forma de onda de una señal sonora (presión) y su análoga eléctrica (tensión) [30].

Las formas de onda de dos señales análogas, como el sonido y la tensión, producidas por el micrófono deberían coincidir idealmente. Sin embargo, en casos reales, esto es solo aproximado, básicamente debido a dos fenómenos: las distorsiones (deformaciones de la onda) y el ruido. Las distorsiones son las alteraciones en la forma de onda, y el ruido es una señal adicional indeseada que se agrega a la señal de interés. El micrófono ideal no es aquel que carece de distorsión y ruido por completo, ya que esto es una imposibilidad física, sino aquel capaz de reducir los dos fenómenos a un nivel imperceptible para el oído humano.

En el procesamiento de señales de sonido, normalmente se habla de señales en el dominio del tiempo y señales en el dominio de la frecuencia (ver figura 2.14), estas son dos formas básicas de representar a las señales de sonido.

Dominio del Tiempo

En el dominio digital, no se puede tratar con señales continuas. Por lo tanto, la señal continua analógica debe discretizarse tanto en amplitud como en tiempo, donde cuantificación se refiere a

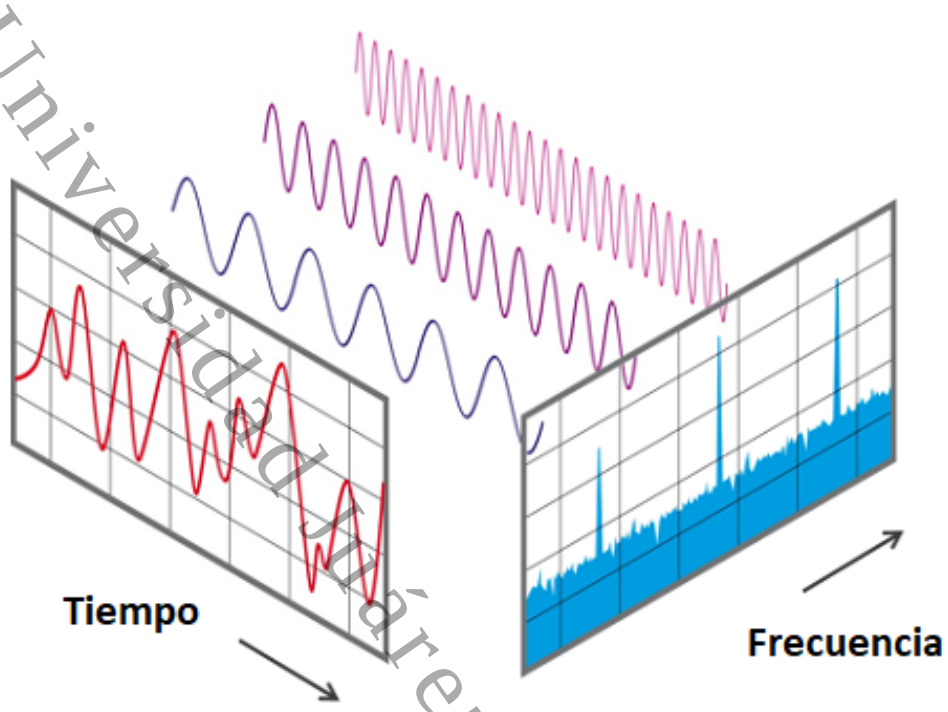

Figura 2.14: Representación gráfica de una señal en el dominio del tiempo y la frecuencia.

la discretización de amplitudes y muestreo se refiere a la discretización en tiempo [22]. La señal resultante es una serie de valores de amplitud cuantificados.

La discretización en el tiempo se realiza mediante el muestreo de la señal en momentos específicos. La distancia Ts en segundos entre los puntos de muestreo es equidistante en el contexto de las señales de audio y puede derivarse directamente de la frecuencia de muestreo Fs mediante (2.9), como se muestra en la figura 2.15.

$$Ts = \frac{1}{Fs} \tag{2.9}$$

Para discretizar las amplitudes de la señal, esta se cuantifica. Cuantificar significa que cada amplitud de la señal se redondea a un conjunto predefinido de valores de amplitud permitidos. Normalmente, se supone que el rango de amplitud de entrada entre la amplitud máxima y la mínima es simétrico alrededor de 0 y se divide en M pasos. Si M es una potencia de base 2, la amplitud de cada muestra cuantificada puede representarse fácilmente con un código binario de longitud de palabra w como 2.10:

$$w = \log_2(M) \tag{2.10}$$

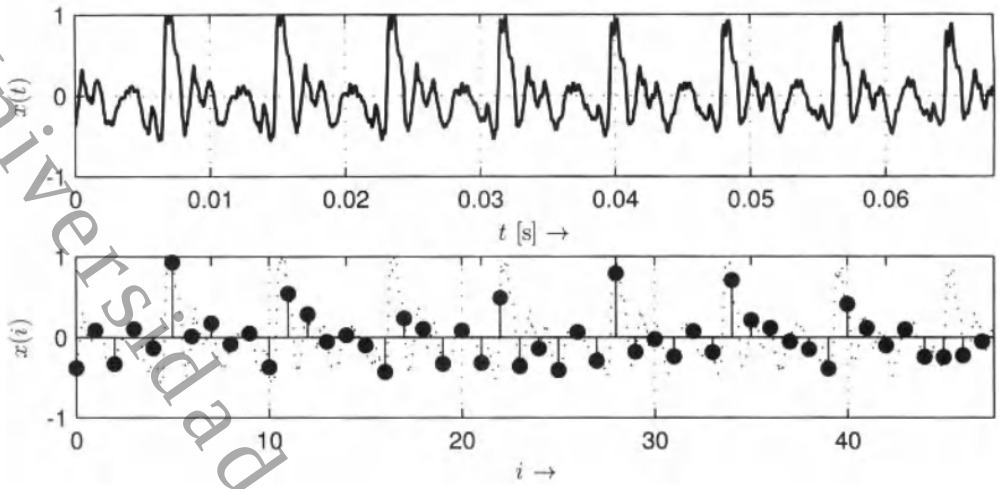

Figura 2.15: Proceso de muestreo mediante el trazado de una señal continua (arriba) y una representación muestreada de la misma (abajo) muestreada a una frecuencia de muestreo de 700 Hz [22].

Como M es un número par, es imposible representar la amplitud cero como un paso de cuantización y mantener un número simétrico de pasos de cuantización por encima y por debajo de cero. El enfoque habitual es tener un paso menos para valores positivos. Las longitudes de palabra w típicas para las señales de audio son de 16, 24 y 32 bits.

La amplitud de las señales de audio se mide en decibelios SPL (del inglés Sound Pressure Level) que representan la relación entre dos señales y se basa en un logaritmo de base 10 del cociente entre dos amplitudes sonoras o presiones. En el área del audio digital, la amplitud es medida en decibelios bajo escala completa (del inglés Decibels Full Scale, dBFS). La amplitud de la señal se refiere al valor máximo que la amplitud de la señal puede alcanzar.

Dominio de la Frecuencia

El dominio de la frecuencia en las señales de audio se refiere a la representación frecuencial de una señal de audio. Una representación en el dominio frecuencial (también llamado representación espectral) muestra el contenido frecuencial de una señal de sonido. Los componentes de frecuencias individuales del espectro son denominados armónicos. Las frecuencias armónicas son valores enteros simples de la frecuencia fundamental. Cualquier frecuencia puede denominarse parcial, sea o no múltiplo de la frecuencia fundamental.

La frecuencia en las señales de audio se mide en ciclos por segundo o hercios (Hz). Los humanos tienen un rango en su audición entre 20 Hz y 20,000 Hz. La amplitud o intensidad e la señal de sonido se refiere a la fuerza de la onda del sonido, que el oído del ser humano interpreta como volumen.

El dominio de la frecuencia está relacionado con las series de Fourier, las cuales permiten descomponer una señal periódica en un número finito o infinito de frecuencias (ver figura 2.16).

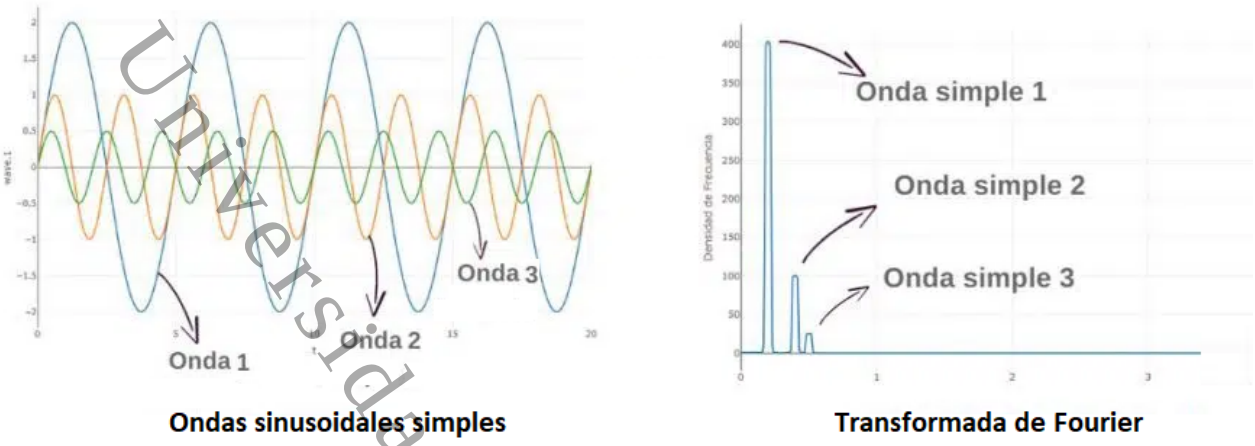

Figura 2.16: Representación gráfica de descomponer una señal de ondas simples en el tiempo, a sus frecuencias utilizando la Transformada de Fourier.

La transformada de Fourier de tiempo corto (del inglés Short-time Fourier transform, STFT) se utiliza para determinar el contenido en frecuencia y de fase en ventanas limitadas de una señal de sonido, así como los cambios que ocurren en el tiempo. La transformada de Fourier de tiempo corto es el método de análisis de frecuencia de tiempo más utilizado y expresa las características de una señal en un momento determinado a través de una señal en la ventana de tiempo (ver figura 2.17).

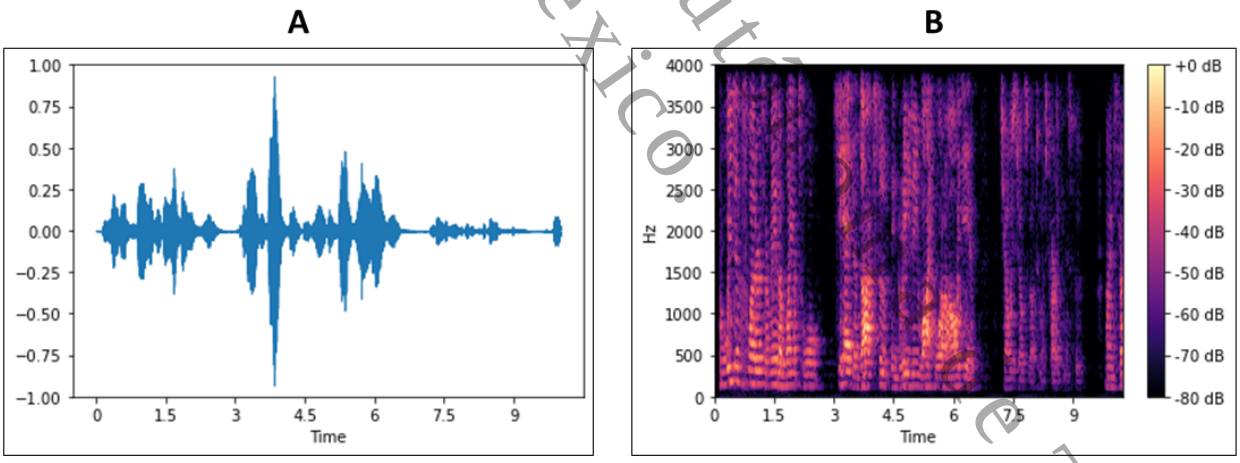

Figura 2.17: Representación gráfica de aplicar la transformada de Fourier de tiempo corto a una señal de audio. Señal de audio en el dominio del tiempo (A). Señal de audio en el dominio de la frecuencia al aplicar la transformada de Fourier de tiempo corto a 256 puntos de frecuencia (B).

2.4.1. Técnicas de Normalización y Escalado

La normalización de señales es un proceso que se realiza para ajustar los valores de una señal de audio para que se ajuste a un estándar específico. En general, los algoritmos de aprendizaje se benefician de la normalización del conjunto de datos. Si en el conjunto hay algunos valores atípicos, son más

apropiados los escaladores o transformadores robustos. Dos de las técnicas aplicadas para procesar las señales de audio son el escalado estándar y escalado dentro de un rango.

Escalado Estándar

La estandarización de los conjuntos de datos es requisito para la mayoría de los algoritmos de aprendizaje automático y profundo; ya que estos algoritmos podrían comportarse erróneamente si las características no se parecen más o menos a datos estándar con distribución normal: Gaussianos con media cero y varianza unitaria. En la práctica, a menudo se ignora la forma de la distribución y simplemente se transforman los datos para centrarlos, eliminando el valor medio de cada característica y, a continuación, se escalan dividiendo las características no constantes por su desviación estándar.

Los elementos utilizados en la función objetivo del algoritmo de aprendizaje (por ejemplo, los regularizadores l1 y l2 de un modelo lineal) pueden asumir que todas las características están centradas alrededor de 0 o tienen el mismo orden de varianza. Si una característica tiene una varianza de un orden de magnitud mayor que otras, puede dominar la función objetivo y hacer que el estimador no pueda aprender de otras características correctamente como se espera.

El escalado estándar elimina la media y escala los datos a la varianza unitaria. El escalado reduce el rango de los valores de las características, como se muestra en la figura 2.18 (B) (la figura 2.18 (A) muestra los datos originales sin escalar). Sin embargo, los valores atípicos influyen en el cálculo de la media empírica y la desviación típica.

Escalado Dentro de un Rango

Una normalización alternativa consiste en escalar los rasgos para que se sitúen entre un valor mínimo y un valor máximo determinados, a menudo entre cero y uno, o para que el valor absoluto máximo de cada rasgo se escale a la unidad de tamaño (ver figura 2.18 (C)). La motivación para utilizar este escalado incluye la robustez frente a desviaciones estándar muy pequeñas de las características y la preservación de las entradas cero en datos dispersos.

En el siguiente capítulo se mencionan los trabajos relacionados considerados más relevantes para esta investigación.

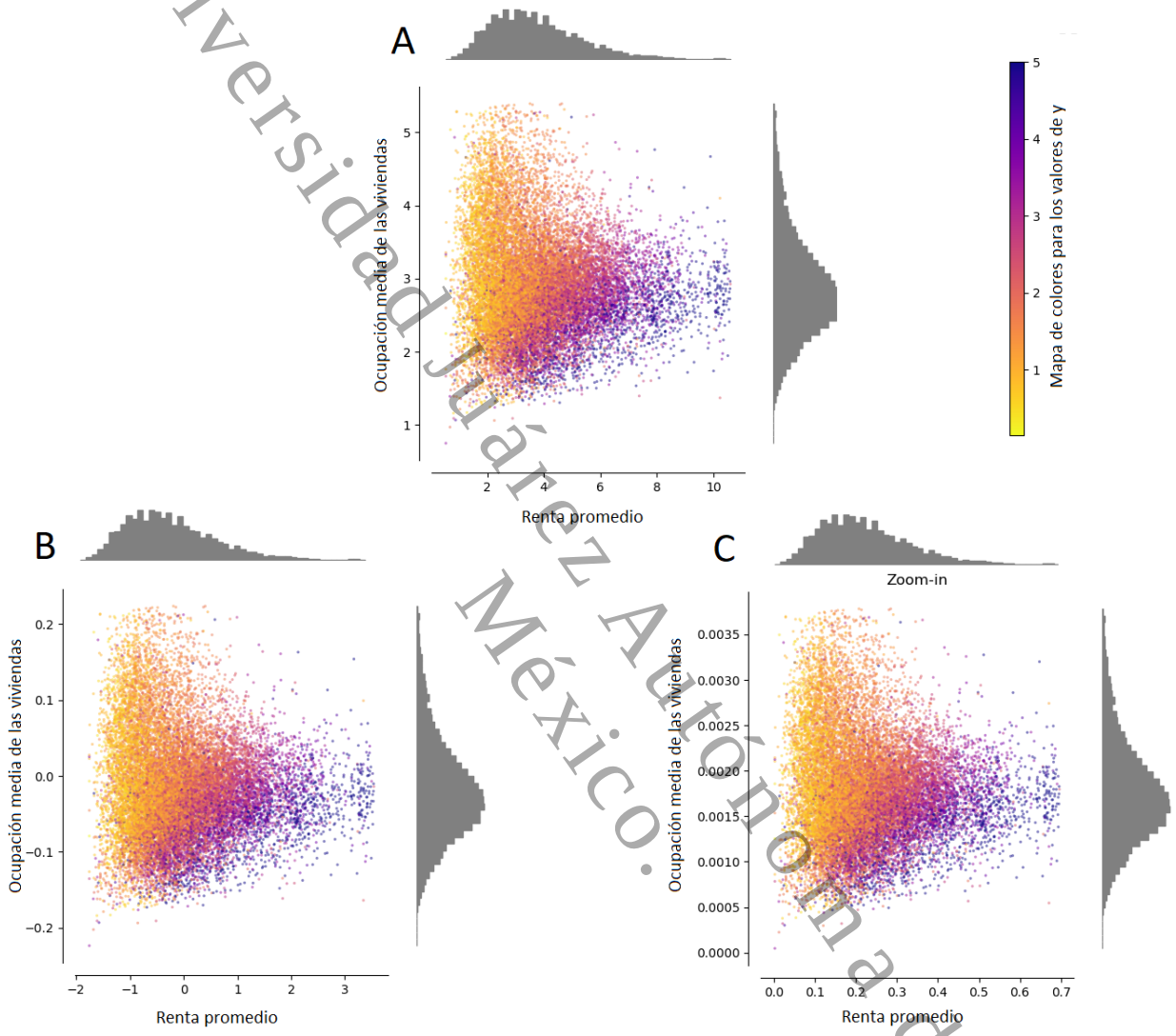

Figura 2.18: Datos del conjunto de datos sobre vivienda en California [20]. Datos originales sin escalar donde la mayoría de los datos se concentran en un intervalo concreto, 0 a 10 para la renta promedio y 0 a 6 para la ocupación media de las viviendas (A). Datos escalados donde la mayoría de los datos se sitúan en el intervalo -2 a 4 para la renta promedio, mientras que los datos se comprimen en el intervalo más pequeño -0.2 a 0.2 para la ocupación media de las viviendas (B). Datos escalados en un rango específico, donde se escala la característica de la renta promedio en el intervalo 0 a 1, y 0 a 0.005 para la ocupación media de las viviendas (C).

Capítulo 3

Literatura relacionada

La mejora de la voz es una tarea de procesamiento de señales que consiste en mejorar la calidad de las señales de voz captadas en condiciones con ruido o degradadas. El objetivo de la mejora de la voz es hacer que las señales de voz sean más claras, inteligibles y agradables de escuchar, lo que puede utilizarse para diversas aplicaciones como el reconocimiento de voz, las teleconferencias y los audífonos.

En el trabajo de Ashutosh Pandey y DeLiang Wang [36], los autores propusieron una red convolucional densa (DCN) con auto-atención para la mejora dela voz en el dominio temporal. Su DCN es un modelo basada en la arquitectura codificador - decodificador con conexiones residuales. Cada capa del codificador y del decodificador comprendió un bloque denso y un módulo de atención. Los bloques densos y los módulos de atención ayudaron en la extracción de características utilizando una combinación de reutilización de características, aumento de la profundidad de la red y agregación máxima de contextos. Además, su trabajo reveló problemas desconocidos relacionados con una función de pérdida basada en la magnitud espectral de la voz. Para solucionar esos problemas, propusieron una nueva función de pérdida basada en las magnitudes de la voz mejorada y el ruido previsto. Aunque la función de pérdida propuesta se basó únicamente en magnitudes, una restricción impuesta por la predicción del ruido pudo garantizar que la pérdida mejorara tanto la magnitud como la fase. Sus resultados experimentales demostraron que su DCN entrenada con la función de pérdida propuesta supera otros enfoques para la mejora de la señal de voz.

Por otro lado, algunas investigaciones han optado por explorar los algoritmos de mejora de la voz basados en el filtro aumentado de Kalman (AKF, por sus siglas en inglés) que utilizan una red convolucional temporal (TCN, por sus siglas en inglés) para estimar el coeficiente de predicción lineal (LPC, por sus siglas en inglés) de la voz limpia y el ruido, así como las redes de atención de múltiples encabezados (MHANet) que han demostrado la capacidad de modelar de forma más eficiente las dependencias a largo plazo de la voz con ruido que las TCN. Motivados por esto, Sujan Kumar Roy, Aaron Nicolson y Kuldip K. Paliwal exploraron el uso de la red MHANet para la esti-

mación del LPC [45]. Su objetivo fue producir parámetros LPC de la voz limpia y ruido con el menor posible. Mediante este enfoque también pretendieron producir una voz mejorada de mayor calidad e inteligibilidad que otros algoritmos para la mejora de voz basados en filtros de Kalman o filtros aumentados de Kalman. Para ello, investigaron el uso de las MHANet en el marco de DeepLPC (DeepLPC es un marco de aprendizaje profundo para estimar conjuntamente los espectros de potencia LPC de la voz limpia y del ruido). Los investigadores seleccionaron DeepLPC porque mostró un sesgo significativamente menor que otros marcos, al evitar el uso de filtros de blanqueamiento y post-procesamiento. En comparación con otros métodos de aprendizaje profundo, la combinación DeepLPC-MHANet produjo estimaciones de LPC de voz limpias con la menor cantidad de sesgo. La combinación DeepLPC-MHANet-AKF también produjo puntuaciones objetivas más altas que los métodos con los que se compararon (con una mejora de 0.24 para PESQ, 3.70 % para STOI, 1.03 dB para SegSNR y 1.04 dB para SI-SDR). Al producir estimaciones de LPC con la menor cantidad de sesgo, DeepLPC-MHANet permitió al filtro aumentado de Kalman producir voz mejorada con mayor calidad e inteligibilidad que cualquier método anterior basado en filtros de Kalman o filtros aumentados de Kalman.

En la literatura existen otros enfoques para solucionar el problema de la mejora de voz; como por ejemplo; modelar las dependencias a largo plazo de la voz con ruido, ya que es fundamental para el rendimiento de un sistema de mejora de voz. Los enfoques actuales de aprendizaje profundo para la mejora de la voz emplean diversos tipos de redes neuronales artificiales, como la red neuronal recurrente (RNN) o la red convolucional (CNN). Sin embargo, tanto las RNN como las CNN presentan algunas deficiencias a la hora de modelar dependencias a largo plazo. Un nuevo enfoque fue incorporar la atención de múltiples encabezados, ya que es capaz de modelar con mayor eficacia las dependencias a largo plazo; además que se puede emplear el enmascaramiento para garantizar que el mecanismo de atención de múltiples encabezados siga siendo causal, un atributo fundamental para el procesamiento en tiempo real.

Motivados por lo mencionado anteriormente, Aaron Nicolson y Kuldip K. Paliwal investigaron el uso de una red neuronal profunda (DNN, por sus siglas en inglés) que utilizó atención de múltiples encabezados enmascarada para la mejora causal de la voz [32]. Las condiciones utilizadas por los investigadores para evaluar la DNN propuesta incluyeron fuentes de ruido no estacionarias del mundo real a múltiples niveles de SNR. Después de una extensa investigación experimental, los investigadores fueron capaces de demostrar que la red neuronal profunda propuesta pudo producir voz mejorada con mayor calidad e inteligibilidad que las redes neuronales recurrentes y las redes convolucionales temporales.

En otra investigación, Weiwei Yu, Jian Zho, HuaBin Wang y Liang Tao exploraron en [65] un enfoque más ambicioso, en su investigación proponen un modelo de mejora de la voz basado en la computación cognitiva denominado SETransformer que pudo mejorar la calidad del habla en entornos ruidosos desconocidos. El SETransformer propuesto por los investigadores, aprovechó las ventajas de las redes

de memoria de corto y largo plazo (LSTM, por sus siglas en inglés) y del mecanismo de atención de múltiples encabezados, ambos inspirados en el principio de percepción auditiva del ser humano. En concreto, el SETransformer pudo lograr la capacidad de caracterizar la estructura local implicada en el espectro del habla y tuvo una complejidad de cálculo más baja gracias a su habilidad de paralelización. Los resultados experimentales de los investigadores demostraron que, en comparación con la arquitectura Transformer estándar y el modelo de red LSTM, el modelo SETransformer pudo lograr sistemáticamente un mejor rendimiento de eliminación de ruido en términos de calidad del habla (PESQ) e inteligibilidad del habla (STOI) en condiciones de ruido no percibido.

Otros investigadores como R. Venkatesan y A. Balaji Ganesh presentan en su investigación [60] enfoques novedosos como modelos de atención auditiva, que consisten en la segregación de la fuente de audio binaural y también en la localización completa de una señal de voz de destino en un entorno de varios interlocutores. En este trabajo, las características acústicas conjuntas, como la relación monoaural (de un canal), binaural (de dos canales) y directa a reverberante (DRR, por sus siglas en inglés) se incorporaron con éxito en una red neuronal recurrente profunda (DRNN, por sus siglas en inglés) basada en el modelo discriminativo conjunto para el proceso de segregación de la fuente de voz. Las características monoaurales y binaurales fueron extraídas de mezclas de voz binaurales de dos hablantes utilizando coeficientes medios de envolvente de Hilbert (MHEC, por sus siglas en inglés) y diferencias de tiempo y nivel interaurales, respectivamente. El rendimiento del modelo propuesto por los investigadores para la segregación de voz basada en redes recurrentes profundas fue validada en términos de señal a interferencia, señal a distorsión y señal a artefactos, y fue comparado con arquitecturas existentes, incluidos otros modelos de redes neuronales profundas (DNN). Los investigadores observaron que el sistema propuesto fue más adecuado que la segregación monoaural de la voz, especialmente cuando el objetivo deseado y las fuentes de interferencia se encuentran en posiciones diferentes. El estudio también propuso la localización completa de la fuente de voz segregada, que creó la posibilidad de seleccionar a la persona de interés deseada a partir de una mezcla de voz acústica de entrada en un entorno reverberante.

El problema de la separación de fuentes de audio es un problema cuya solución tiene impacto en diferentes áreas, algunas de ellas se mencionan a continuación.

- Procesamiento de música: uno de los casos más particulares, donde la separación de audio es ampliamente estudiada, es en el análisis musical, ya sea para la extracción individual de los instrumentos musicales en una canción [8] [46] [15], o para la recuperación e identificación de la voz de cantantes [55].
- Procesamiento de voz: lograr separar la voz de una persona cuando esta se encuentra mezclada con las voces de otras personas es una aplicación más de la separación de fuentes de audio [21], así como la separación del ruido de fondo en una grabación [28].
- Métodos de autenticación: similar a las huellas dactilares o el iris, la voz como método de

autenticación es estudiado para el aumento de la seguridad [39].

En la literatura de procesamiento de audio, estos dos términos (separación y mejora) se intercambian a menudo, especialmente cuando se refiere al problema de suprimir el ruido y otras señales de audio para conservar únicamente la voz.

En el siguiente capítulo se encuentra el desarrollo experimental generado a partir de los conceptos descritos en al capítulo 2, así como de la revisión de la literatura.

Capítulo 4

Desarrollo Experimental

En el desarrollo de diversas áreas de la ciencia (como la neurociencia, robótica o inteligencia artificial) es común buscar inspiración en las ciencias biológicas y cognitivas; una gran cantidad de los enfoques inspirados se concentran en el modelado de áreas cerebrales y fenómenos psicológicos como la memoria, la percepción, el lenguaje, la categorización y el aprendizaje utilizando métodos de procesamiento de información neuronal [37]. Las ciencias de la computación no se quedan exentas al momento de buscar inspiración en las ciencias biológicas, un ejemplo claro es el campo de la -neurociencia computacional-, con un enfoque particular en comprender el desarrollo y la función de los sistemas nerviosos en muchas escalas estructurales diferentes, incluidos los niveles biofísicos, de circuito y de sistemas; utilizando métodos como los análisis teóricos, los modelos de neuronas, redes y sistemas cerebrales [6].

Considerando los enfoques actuales aplicados a la ciencia de la computación, como la creación de modelos basados en las redes neuronales, la idea de dotar a los modelos computacionales con la capacidad cognitiva de atención auditiva selectiva tomó sentido a medida que se estudiaron y analizaron las características y funcionalidades propias de la atención auditiva selectiva humana.

4.1. Conjunto de Datos

Para esta investigación fue necesario identificar y obtener conjuntos de datos públicos que incluyeran señales de voz, así como de ruidos en entornos interiores y al aire libre; a partir de estos conjuntos de datos se construyeron los datos de entrenamiento, validación y prueba de los modelos de red neuronal. Un resumen de los conjuntos de datos analizados se encuentra en la tabla 4.1.

Después de analizar las características y disponibilidad de los diversos conjuntos de datos públicos, se seleccionaron cuatro conjuntos de datos; como conjunto de datos de voz se utilizó el TIMIT [12], y como conjunto de datos de ruido se utilizó NoiseX-92 [57], DEMAND [54], y PNL-100 [14].

Tabla 4.1: Revisión de los conjuntos de datos disponibles de voz y ruido.

Conjunto de datos de voz	
Conjunto de datos	Descripción
TIMIT [12]	Grabación de voz en inglés de 630 personas, 10 frases por cada persona, muestreadas a 16 kHz
Voice Bank [59]	Grabación de voz en inglés de 500 personas, 400 frases por cada persona, muestreadas a 48 kHz
LibriSpeech [35]	1000 horas de voz en inglés, muestreada a 16 kHz
WSJCAMO [42]	Grabaciones de voz de 140 personas hablando unas 110 frases en inglés británico de cada persona, muestreadas a 16 kHz
Conjunto de datos de ruido	
NOISEX-92 [57]	Grabación de varios ruidos, entre ellos: balbuceo, fábrica, canal HF, ruido rosa, blanco y militar
Demand [54]	Colección de grabaciones multicanal de ruido acústico en diversos entornos, como: parque oficina, cafetería y calle.
PNL-100 [14]	100 sonidos ambientales como ruidos de máquinas, sonidos de animales, pasos y movimiento de puertas
UrbanSound8K [47]	8732 grabaciones de 10 ruidos urbanos que incluyen: aire acondicionado, bocina de coche, niños jugando, ladrido de perro, taladro, disparo, sirena y música callejera

El TIMIT es un conjunto de datos que contiene grabaciones de enunciados de 630 hablantes que representan 8 divisiones dialectales del inglés americano, cada uno de ellos hablando 10 frases con diferente fonética de hablantes masculinos y femeninos. El material del TIMIT está subdividido originalmente en porciones equilibradas para el entrenamiento y las pruebas (los criterios de subdivisión se describen en [12]). NoiseX-92 es un conjunto de datos compuesto de grabaciones de varios tipos de ruidos acústicos; entre ellos: ruidos de equipos de corte y soldadura eléctrica, ruido blanco, ruidos militares y de vehículos. DEMAND contiene grabaciones de varios tipos de ruidos acústicos en entornos interiores (domésticos, oficina, público y transporte), y entornos al aire libre (calle y naturaleza). Por último, PNL-100 contiene sonidos ambientales como ruidos de máquinas, sonidos de animales, pasos y movimiento de puertas.

4.1.1. Preprocesamiento del Conjunto de Datos

La señal de audio suele ser artificial en el entrenamiento de los modelos de redes neuronales, ya que es necesario mezclar la voz con las señales de ruido. Para emular los entornos naturales de ruido en las señales de voz, es necesario recopilar las señales de ruido de las bases de datos. Estas bases de datos contienen sonidos ambientales de diferentes fuentes y en diferentes áreas, donde el ruido agregado es la suma de sonidos con formas y magnitudes variables.

El uso de sonidos ambientales repercute en la complejidad del problema de la mejora de la voz, ya que es difícil establecer un patrón en la señal de ruido. Los investigadores suelen utilizar valores de relación señal a ruido (SNR, por sus siglas en inglés) que van de +10 dB a -10 dB, donde una señal de audio a +10 dB equivale a un aula escolar (la señal de voz es mayor que la señal de ruido) y -10 dB a una estación de tren (la señal de ruido es mayor que la señal de voz) [29].

Se seleccionó la parte de entrenamiento del conjunto de datos TIMIT y el ruido de los conjuntos de datos NoiseX-92 y DEMAND para el conjunto de entrenamiento y validación. A continuación, se crearon cinco horas de mezclas de audio en clips de un minuto con SNR muestreados uniformemente entre -10 dB y 10 dB (con lo que los diferentes ruidos corrompen la señal de voz). Los clips de voz y de ruido fueron elegidos aleatoriamente para crear las mezclas. A continuación, se muestrearon los audios a 8 kHz para alimentar el modelo con las bandas de frecuencia más relevantes. Después, se calculó el espectro de la magnitud de la señal mediante la transformada de Fourier de tiempo corto (STFT) con un tamaño de 256 puntos de frecuencias, una ventana de longitud de trama de 32 ms (256 muestras) y un solapamiento de 50 % (128 muestras). Por último, se normalizaron los datos de entrenamiento y validación con media cero y varianza unitaria para facilitar el proceso de entrenamiento.

En la Figura 4.1 se encuentra la representación gráfica de una señal de voz y ruido en el dominio del tiempo y en el dominio de la frecuencia. La figura 4.1(A) y 4.1(B) pertenecen a una señal de voz limpia; la figura 4.1(C) y 4.1(D) pertenecen a una señal de ruido; y la figura 4.1(E) y 4.1(F)

pertenecen a las señales de voz y ruido mezcladas a SNR de -10 dB.

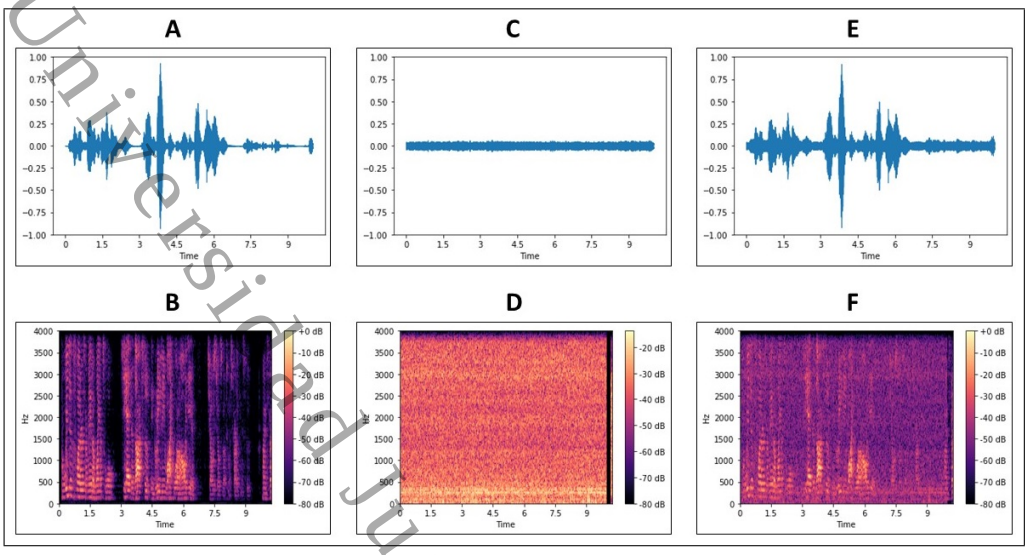

Figura 4.1: Representación gráfica de señales de voz y ruido en el dominio del tiempo y en el dominio de la frecuencia, mezcladas a SNR de -10 dB.

Adicionalmente, en la Figura 4.2 se muestra otra señal de voz y ruido en el dominio del tiempo y en el dominio de la frecuencia. La figura 4.2(A) y 4.2(B) pertenecen a una señal de voz limpia; la figura 4.2(C) y 4.2(D) pertenecen a una señal de ruido; y la figura 4.2(E) y 4.2(F) pertenecen a las señales de voz y ruido mezcladas a SNR de 10 dB.

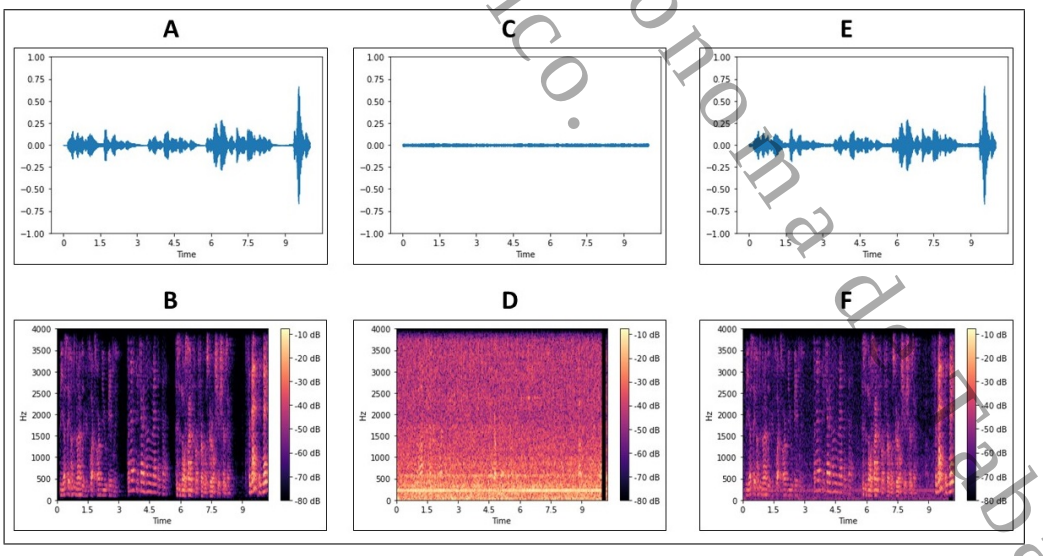

Figura 4.2: Representación gráfica de señales de voz y ruido en el dominio del tiempo y en el dominio de la frecuencia, mezcladas a SNR de 10 dB.

Para evaluar los modelos, se hicieron dos conjuntos de pruebas. En primer lugar, se crearon nuevas mezclas de audio con muestras uniformes de SNR de -10 dB, -5 dB, 0 dB, 5 dB y 10 dB, utilizando la parte de prueba del conjunto de datos TIMIT y el ruido de los conjuntos de datos NoiseX-

92 y DEMAND. Además, repetimos el mismo proceso para el segundo conjunto de pruebas, pero utilizando la parte de pruebas del conjunto de datos TIMIT y el ruido de PNL-100; se utilizaron estas segundas mezclas de audio para evaluar la capacidad de generalización de los modelos entrenados.

Se conservó la fase de la señal solo durante el proceso de predicción del modelo y luego se añadió a la señal limpia estimada, de forma similar a lo realizado en [33] y [24].

4.2. Diseño de las Arquitecturas

En esta investigación, se implementaron diversos modelos de redes neuronales artificiales que pertenecen a tres categorías: perceptrón multicapa, red neuronal convolucional y red neuronal de unidades recurrentes. Se establecieron las arquitecturas de los modelos de forma básica para hacer una comparación equitativa entre los modelos y mostrar el efecto de los parámetros de cada red específica y el efecto del módulo de atención en el rendimiento general. Los códigos de las arquitecturas se encuentran en el Anexo A.

4.2.1. Arquitectura del Módulo de Atención Auditiva

Después de analizar las características del funcionamiento del sistema auditivo humano [50] y de las teorías de la atención selectiva [23], se diseñó y construyó un módulo encargado de emular la capacidad cognitiva de atención auditiva para ayudar a los modelos de redes neuronales artificiales para identificar las características más importantes para mejorar su rendimiento. El módulo de atención se basó en el uso de la atención de múltiples encabezados. Gráficamente, la arquitectura del módulo de atención se puede ver en la figura 4.3, y está compuesto de la siguiente forma:

1. Una capa de atención de múltiples encabezados, con heads = 4 y con dropout = 0.1.
2. Una capa de normalización, la cual suma los datos de salida de la capa de atención con los datos de entrada del módulo de atención, para después aplicar la normalización.
3. Una capa densa con ReLu como función de activación.
4. Una capa de normalización, la cual suma los datos de entrada y salida de la capa densa, para finalmente aplicar de nuevo normalización.

Se diseñó un módulo de atención, sin embargo, se implementaron diferentes modificaciones, las cuales están relacionadas con las características y dimensiones de los datos en el flujo de información en el interior de los modelos de red neuronal artificial. La primer arquitectura del módulo de atención recibe datos de entrada en una dimensión (de tamaño 1x256 y 1x512), y la segunda arquitectura del módulo de atención recibe datos de entrada en dos dimensiones (de tamaño 8x256).

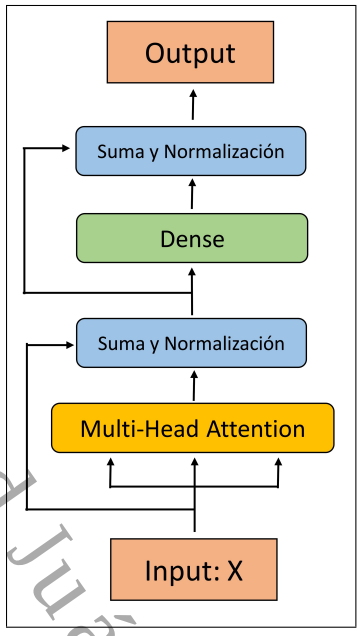

Figura 4.3: Diagrama del módulo de atención.

4.2.2. Arquitectura del Perceptrón Multicapa

La arquitectura del modelo de red neuronal artificial de tipo perceptrón multicapa, figura 4.4, tiene ocho capas densas de 512 unidades con funciones de activación ReLU. Una capa de dropout con una tasa del 5 % sigue a cada una de las siete primeras capas ocultas para evitar el sobreajuste y estabilizar el entrenamiento. La última capa es una capa densa sin función de activación (también conocida como capa densa con activación lineal) con 256 unidades. Los Hiperparámetro de la arquitectura perceptrón multicapa se encuentran en la tabla 4.2.

Tabla 4.2: Hiperparámetros utilizados en la arquitectura perceptrón multicapa.

Tipo	Nodos	Activación	Dropout
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
MLP	512	ReLU	0.05
Módulo Atención	-	-	0.1
MLP	256	Lineal	-

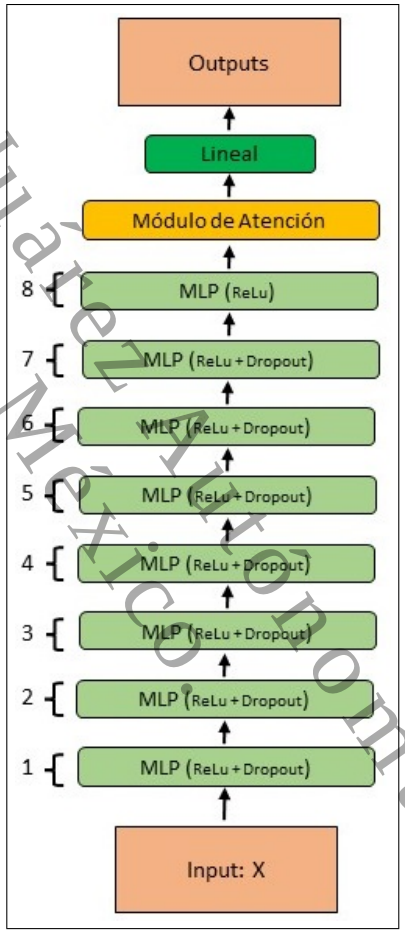

Figura 4.4: Diagrama de la arquitectura perceptrón multicapa.

4.2.3. Arquitectura de la Red Neuronal Convolucional

Se diseñaron e implementaron tres arquitecturas del modelo de red neuronal artificial de tipo convolucional. La primera arquitectura, figura 4.5A, tiene ocho capas convolucionales de una dimensión con activaciones PReLU, seguidas por una capa densa de 256 unidades sin función de activación. Se estableció el número de filtros en las primeras siete capas convolucionales en 64, con un solo filtro en la última capa de convolución. Además, se utilizaron kernels de tamaño 16 y capas de dropout con una tasa del 10 % en las capas uno, tres, cinco y siete. Por último, se utilizó stride = 1 y padding = same. Los hiperparámetros de la arquitectura convolucional de una dimensión se encuentran en la tabla 4.3.

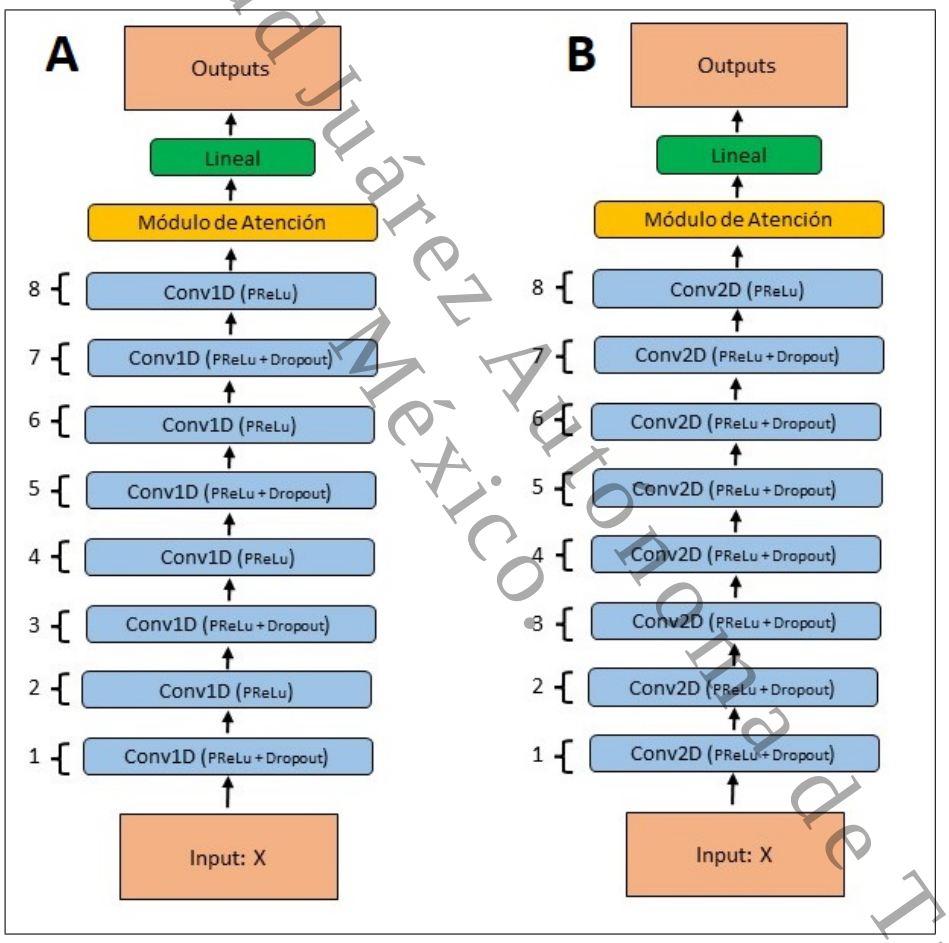

Figura 4.5: Diagrama de la arquitectura convolucional de una dimensión (A), y dos dimensiones (B).

La segunda arquitectura, figura 4.5B, tiene capas convolucionales de dos dimensiones con activaciones PReLU y una capa densa final de 256 unidades sin función de activación. Se estableció el número de filtros en las primeras siete capas convolucionales en 64 (con un solo filtro en la última capa de convolución), pero con kernels de tamaño 4x4 en todas las capas convolucionales. Además, se empleó dropout con una tasa del 10 % en las primeras siete capas convolucionales. Por último, utilizó stride = 1 y padding = same. Los Hiperparámetro de la arquitectura convolucional de dos dimensiones se

Tabla 4.3: Hiperparámetros utilizados en la arquitectura convolucional de una dimensión.

Tipo	Kernel	Filtros	Nodos	Activación	Dropout	Strides	Padding
1D-Conv	16	64	-	PReLU	0.1	1	same
1D-Conv	16	64	-	PReLU	-	1	same
1D-Conv	16	64	-	PReLU	0.1	1	same
1D-Conv	16	64	-	PReLU	-	1	same
1D-Conv	16	64	-	PReLU	0.1	1	same
1D-Conv	16	64	-	PReLU	-	1	same
1D-Conv	16	64	-	PReLU	0.1	1	same
1D-Conv	16	1	-	PReLU	-	1	same
Módulo Atención	-	-	-	-	0.1	-	-
MLP	-	-	256	Lineal	-	-	-

encuentran en la tabla 4.4.

Tabla 4.4: Hiperparámetros utilizados en la arquitectura convolucional de dos dimensiones.

Tipo	Kernel	Filtros	Nodos	Activación	Dropout	Strides	Padding
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	64	-	PReLU	0.1	1	same
2D-Conv	4x4	1	-	PReLU	-	1	same
Módulo Atención	-	-	-	-	0.1	-	-
MLP	-	-	256	Lineal	-	-	-

La tercera arquitectura convolucional implementada fue una arquitectura ramificada. La arquitectura general de la red neuronal convolucional ramificada se puede dividir en cinco componentes. Primeramente, un bloque compuesto de 4 capas convolucionales con 64, 64, 32 y 2 filtros respectivamente, utilizando un tamaño de kernel = 16, stride = 1, padding = same, PReLu como función de activación. Se empleó dropout = 0.1 únicamente en las 3 primeras capas. Posteriormente, se encuentran las dos ramificaciones con configuraciones similares al primer bloque convolucional mencionado; cada una con 64, 64, 32 y 1 filtros respectivamente, usando un tamaño de kernel = 16, stride = 1, padding = same y PReLu como función de activación, y con dropout = 0.1, únicamente en las tres

primeras capas. Seguidamente, se encuentra el módulo de atención, el cual recibe como datos de entrada dos vectores provenientes de las dos ramificaciones, los cuales son concatenados antes de ingresar en el módulo de atención. Por último, se encuentran dos capas densas con 512 nodos cada una que emplean ReLu como función de activación, más una capa final de tipo lineal con 256 nodos. La arquitectura general de la red neuronal convolucional ramificada se muestra en la figura 4.6, y los detalles de los hiperparámetros utilizados se pueden consultar en la tabla 4.5

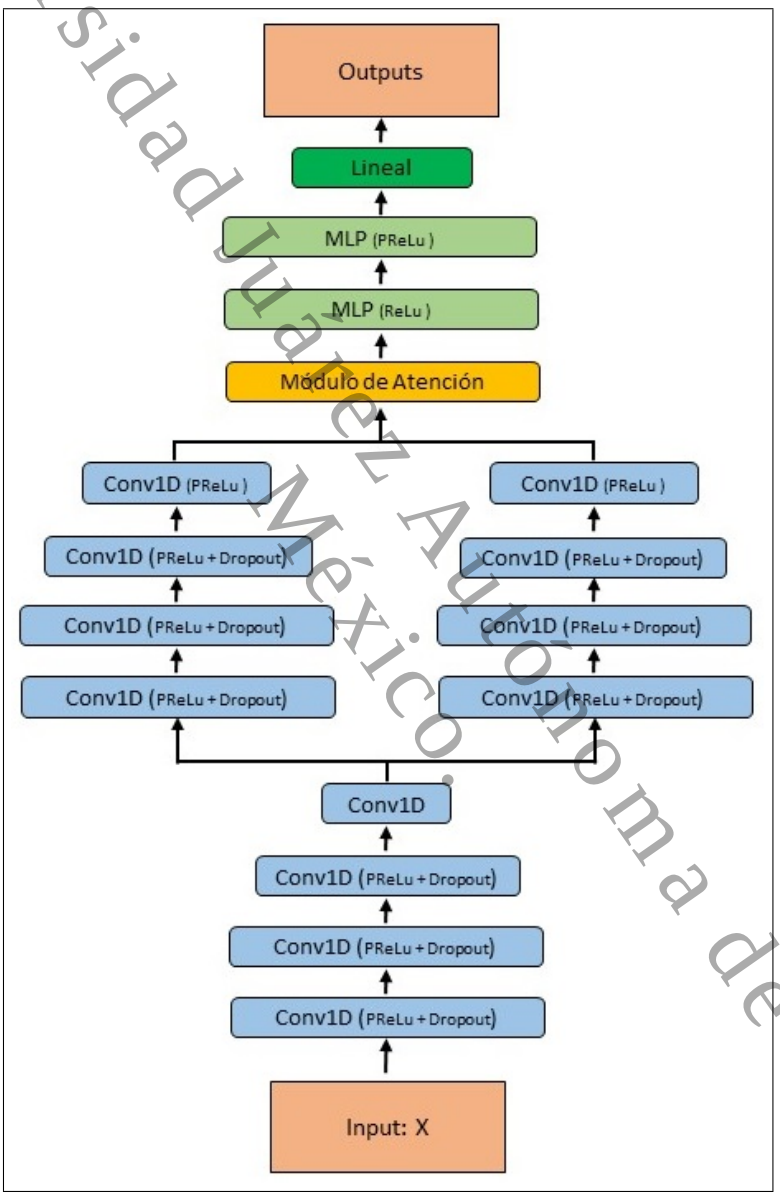

Figura 4.6: Diagrama de la arquitectura convolucional ramificada.

4.2.4. Arquitectura de la Red de Unidades Recurrentes Cerradas

La arquitectura del modelo de red neuronal artificial de tipo unidades recurrentes cerradas tiene cuatro capas conectadas de 256 unidades con activaciones ReLU, y la última capa es densa sin

Tabla 4.5: Hiperparámetros utilizados en la arquitectura convolucional ramificada.

Tipo	Filtros	Nodos	Kernel	Activación	Strides	Dropout	Conexión
1D-Conv_01	64	-	16	PReLU	1	0.1	-
1D-Conv_02	64	-	16	PReLU	1	0.1	-
1D-Conv_03	32	-	16	PReLU	1	0.1	-
1D-Conv_04	2	-	16	PReLU	1	-	-
Rama_01							
1D-Conv_05	64	-	16	PReLU	1	0.1	1D-Conv_04
1D-Conv_06	64	-	16	PReLU	1	0.1	-
1D-Conv_07	32	-	16	PReLU	1	0.1	-
1D-Conv_08	1	-	16	PReLU	1	-	-
Rama_02							
1D-Conv_09	64	-	16	PReLU	1	0.1	1D-Conv_04
1D-Conv_10	64	-	16	PReLU	1	0.1	-
1D-Conv_11	32	-	16	PReLU	1	0.1	-
1D-Conv_12	1	-	16	PReLU	1	-	-
Módulo Atención	-	-	-	-	-	0.1	1D-Conv_08 1D-Conv_12
MLP	-	512	-	ReLU	-	-	-
MLP	-	512	-	ReLU	-	-	-
MLP	-	256	-	Lineal	-	-	-

función de activación con 256 unidades. No se utilizó capas de dropout en este modelo. El diagrama se muestra en la Figura 4.7. Los detalles de los hiperparámetros utilizados se pueden consultar en la tabla 4.6.

4.3. Estrategia de Entrenamiento

La estrategia de entrenamiento implementada consistió en mapear el espectrograma de magnitud como objetivo de entrenamiento. Se implementaron todos los modelos de red neuronal artificial utilizando la librería Keras en TensorFlow. La función de pérdida que se utilizó durante el proceso de entrenamiento fue el error cuadratico medio (MSE, por sus siglas en ingles), ya que el objetivo era mejorar todas las métricas de evaluación, no una en específico. Se empleó Adam como optimizador con una tasa de aprendizaje de 0.0001, b1 = 0,9, b2 = 0,999 y epsilon = 1e-08. Se utilizó un tamaño de lote de 64 y el 10 % de los datos de entrenamiento se reservó para el proceso de validación con el fin de supervisar y controlar el rendimiento de la red y evitar el sobreajuste. Además, se seleccionó la precisión como métrica para controlar el entrenamiento.

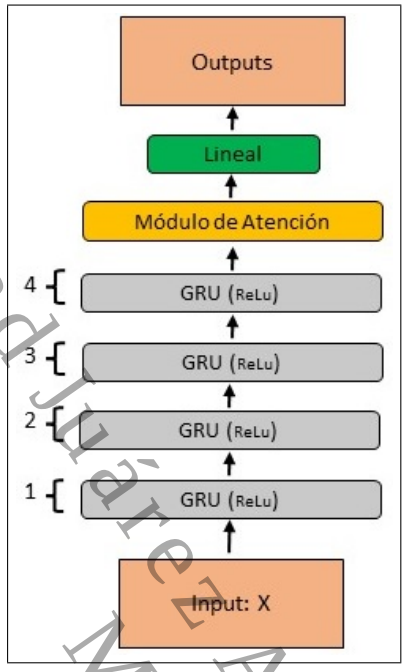

Figura 4.7: Diagrama de la arquitectura de tipo unidades recurrentes cerradas.

Tabla 4.6: Hiperparámetros utilizados en la arquitectura de tipo unidades recurrentes cerradas.

Tipo	Nodos	Activación
GRU	256	RelU
GRU	256	RelU
GRU	256	RelU
GRU	256	RelU
MLP	256	Lineal

La duración de los entrenamientos se fijó en 60 épocas y se aplicaron dos estrategias durante el entrenamiento. La primera fue una estrategia de detención anticipada con la que el entrenamiento se detenía si la métrica monitorizada deja de mejorar después de seis épocas consecutivas. La segunda, una estrategia de reducción de la tasa de aprendizaje, con un factor de 0,8 y una tasa de aprendizaje mínima de 0,000001, aplicada a cada época en la que la métrica monitorizada dejaba de mejorar.

Se mantuvo la misma configuración de entrenamiento para todas las arquitecturas con el fin de realizar una evaluación y comparación equitativas con los resultados. Entrenamos todos los modelos usando una tarjeta gráfica NVIDIA Tesla P100 con 16GB de memoria. Además, realizamos todos los experimentos utilizando un dispositivo computacional convencional con 8 GB de memoria y un procesador AMD Ryzen de 4 núcleos a 2 GHz.

En el siguiente capítulo se encuentran los resultados obtenidos mediante los experimentos.

Capítulo 5

Resultados y Discusión

En esta sección se presentan los resultados. Como se mencionó en el apartado 3.1.1, se realizaron dos experimentos. El primer experimento muestra el rendimiento de los modelos en nuevas mezclas de sonido creadas con la parte de prueba del conjunto de datos de voz TIMIT y el ruido de los conjuntos de datos NoiseX-92 y DEMAND (los mismos ruidos que en el entrenamiento). En el segundo experimento, se comprobó la capacidad de generalización de los modelos en mezclas de sonido creadas con la misma parte de prueba del conjunto de datos TIMIT y el ruido del conjunto de datos PNL-100.

Para evaluar el rendimiento de los modelos, se utilizaron tres métricas de evaluación: la evaluación perceptual de la calidad del habla (PESQ) [41]; la inteligibilidad objetiva a corto plazo (STOI) [17]; y la relación señal-distorsión invariante en escala (SI-SDR) [44], que son las métricas estándar más utilizadas para evaluar el rendimiento de las propuestas para el problema de la mejora del habla.

Los valores de PESQ oscilan entre -0,5 y 4,5; cuanto mayor sea el valor, mejor será la calidad del habla. Los valores de STOI suelen oscilar entre 0 y 1, donde un valor más alto indica una mejor inteligibilidad (el valor de la métrica normalmente es convertido en porcentaje). La relación señal-distorsión invariante (SI-SDR) se utiliza para calcular la cantidad de distorsión introducida por el proceso de separación de la voz. La SDR es una de las métricas estándar de evaluación de la separación del habla que mide la cantidad de distorsión introducida por la señal separada; la SDR es la relación entre la energía de la señal limpia y la energía de la distorsión [13]. Por lo tanto, los valores más altos de SDR indican un mejor rendimiento de la separación de la voz.

5.1. Resultados Individuales

5.1.1. Resultados de la Red Perceptrón Multicapa

Con la inclusión del módulo de atención en la red neuronal MLP, se obtuvieron peores resultados con las métricas STOI y SI-SDR a niveles bajos de SNR (-10 dB a 0 dB), mientras que PESQ mostró que las mejoras fueron mínimas. Sin embargo, a niveles de SNR más altos (5 dB a 10 dB), la inclusión del módulo de atención mostró mejoras en las tres métricas utilizadas, lo que indica que las redes neuronales MLP pueden mejorar al incluir un módulo de atención siempre que el volumen de ruido en las señales de audio se mantenga bajo. La Tabla 5.1 muestra los resultados completos para los niveles de SNR -10 dB, -5 dB, 0 dB, 5 dB y 10 dB.

Tabla 5.1: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo MLP en señales de audio con ruido que formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	68.25 %	68.16 %	2.42	2.43	6.08	6.07
-5	79.79 %	79.77 %	2.70	2.71	10.68	10.61
0	87.94 %	88.08 %	3.03	3.04	14.05	13.94
5	92.21 %	92.61 %	3.37	3.39	19.67	19.95
10	94.88 %	95.45 %	3.63	3.65	18.27	18.48

5.1.2. Resultados de las Redes Neuronales Convolucionales

Cuando se incorporó el módulo de atención en la arquitectura convolucional de una dimensión, las métricas STOI y SI-SDR mostraron mejoras en los niveles de SNR de -5 dB a 10 dB. En cambio, PESQ mostró mejoras solo en los niveles de SNR -10 dB, 0 dB y 5 dB. No obstante, en -5 dB y 10 dB, PESQ muestro que la calidad del habla se mantuvo igual. En el caso del nivel de SNR más bajo (-10 dB), no se encontró ninguna mejora con las métricas STOI y SI-SDR. La tabla 5.2 muestra los resultados completos.

Ahora bien, al utilizar redes convolucionales de dos dimensiones con el módulo de atención, se observaron mejoras en los niveles de SNR de -5 dB a 10 dB con la métrica STOI; PESQ muestra mejoras excepto para SNR a 0 dB, y SI-SDR solo mostró mejoras de 0 dB a 10 dB. La tabla 5.3 muestra los resultados completos.

Finalmente, al utilizar la red convolucional ramificada con el módulo de atención, se observaron mejoras en todos los niveles de SNR (de -10 dB a 10 dB) con las tres métricas STOI, PESQ y SI-SDR; con la excepción de los resultados de la métrica PESQ en SNR de -5, donde el resultado de la métrica no cambió. La tabla 5.4 muestra los resultados completos.

Tabla 5.2: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional de una dimensión en señales de audio con ruido que formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	69.36 %	69.30 %	2.44	2.45	6.28	6.27
-5	81.55 %	81.67 %	2.79	2.79	11.48	11.49
0	89.79 %	90.06 %	3.15	3.16	15.97	16.13
5	94.24 %	95.43 %	3.48	3.55	20.44	24.28
10	97.17 %	97.44 %	3.79	3.79	22.77	23.38

Tabla 5.3: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional de dos dimensiones en señales de audio con ruido que formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	68.80 %	68.75 %	2.45	2.46	6.38	6.23
-5	80.88 %	81.02 %	2.79	2.80	11.36	11.34
0	89.29 %	89.54 %	3.16	3.15	15.52	15.69
5	93.51 %	94.98 %	3.47	3.56	18.17	23.24
10	96.28 %	96.91 %	3.77	3.78	21.23	22.01

Tabla 5.4: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional ramificado en señales de audio con ruido que formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	69.54 %	69.56 %	2.43	2.45	6.21	6.25
-5	81.57 %	81.84 %	2.78	2.78	11.42	11.50
0	89.89 %	90.17 %	3.13	3.15	15.87	16.12
5	94.66 %	95.36 %	3.52	3.56	22.09	24.30
10	97.43 %	97.61 %	3.78	3.80	22.71	23.57

5.1.3. Resultados de la Red de Unidades Recurrentes Cerradas

Por último, al añadir el módulo de atención en el modelo GRU, la tabla 5.5 muestra mejoras en la mayoría de las tres métricas, excepto en STOI a -10 dB o en el caso de PESQ a 5 dB y 10 dB, donde los resultados no cambiaron.

Tabla 5.5: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo GRU en señales de audio con ruido que formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	69.45 %	69.44 %	2.42	2.43	6.16	6.19
-5	81.37 %	81.42 %	2.76	2.77	11.38	11.44
0	89.81 %	89.90 %	3.10	3.12	15.92	16.03
5	95.30 %	95.39 %	3.52	3.52	24.27	24.47
10	97.17 %	97.37 %	3.75	3.75	22.88	23.03

5.2. Resultados de la Capacidad de Generalización de los Modelos

Uno de los problemas de la mejora del habla basada en redes neuronales es tener una red neuronal que funcione bien en el conjunto de datos de entrenamiento, pero que no pueda generalizar y mantener el mismo rendimiento para los datos nuevos. Este problema se conoce como problema de varianza o de sobreajuste. Por lo tanto, probar la capacidad de generalización de los modelos es crucial para hacer una comparación equitativa entre ellos.

Se evaluó la capacidad de generalización de los cuatro modelos implementados, probando el rendimiento de los modelos con nuevos ruidos del conjunto de datos PNL-100. Las tablas 5.6, 5.7, 5.8, 5.9 y 5.10, muestran los resultados completos de generalización para niveles de SNR a -10 dB, -5 dB, 0 dB, 5 dB, y 10 dB para los cuatro modelos con y sin el módulo de atención.

Desde una perspectiva general, la inclusión de un módulo de atención mostró mejoras en la capacidad de generalización de los modelos basados en MLP y convolución, con algunas excepciones, por supuesto. Sin embargo, la inclusión del módulo de atención en el modelo GRU obtuvo resultados contrarios a los esperados en la mayoría de las métricas.

5.3. Resultados Generales Promediados

En las figuras 5.1, 5.2, y 5.3 se muestran los resultados medios de SNR para PESQ, STOI, y SI-SDR, respectivamente. Las figuras (A) muestran los resultados promediados de las mezclas de la parte de prueba del conjunto de datos TIMIT con ruido de NoiseX-92 y DEMAND. Las figuras (B) muestran

Tabla 5.6: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo MLP en señales de audio con ruido que no formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	73.05 %	73.88 %	2.49	2.50	2.75	2.78
-5	86.55 %	87.05 %	3.10	3.11	10.28	10.22
0	88.19 %	88.96 %	3.14	3.17	12.32	12.38
5	92.32 %	93.03 %	3.43	3.46	16.14	16.34
10	94.51 %	95.17 %	3.67	3.70	18.64	18.99

Tabla 5.7: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional de una dimensión en señales de audio con ruido que no formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	76.32 %	76.28 %	2.52	2.55	3.17	3.65
-5	88.72 %	89.60 %	3.18	3.17	10.64	10.73
0	90.84 %	91.63 %	3.20	3.23	12.86	13.25
5	94.66 %	95.55 %	3.49	3.53	17.31	17.91
10	96.65 %	97.60 %	3.75	3.78	21.00	22.21

Tabla 5.8: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional de dos dimensión en señales de audio con ruido que no formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	74.70 %	75.28 %	2.46	2.48	2.79	2.93
-5	87.65 %	89.21 %	3.18	3.17	10.23	10.55
0	89.94 %	90.64 %	3.16	3.18	12.46	12.77
5	93.79 %	94.89 %	3.47	3.49	16.64	17.40
10	95.72 %	97.02 %	3.72	3.76	19.71	21.37

Tabla 5.9: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo convolucional ramificado en señales de audio con ruido que no formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	76.04 %	75.50 %	2.51	2.52	3.12	3.58
-5	89.25 %	89.47 %	3.17	3.15	10.67	10.70
0	90.92 %	91.18 %	3.19	3.19	12.87	13.11
5	95.00 %	95.33 %	3.49	3.49	17.48	17.85
10	97.13 %	97.56 %	3.76	3.75	21.48	22.31

Tabla 5.10: Resultados de las métricas STOI, PESQ y SI-SDR obtenidos por el modelo GRU en señales de audio con ruido que no formó parte del entrenamiento.

	STOI (%)		PESQ		SI-SDR	
SNR	sin at.	con at.	sin at.	con at.	sin at.	con at.
-10	75.87 %	75.62 %	2.49	2.49	2.97	2.87
-5	89.39 %	89.35 %	3.15	3.14	10.83	10.73
0	91.02 %	90.84 %	3.17	3.16	12.84	12.79
5	95.17 %	95.13 %	3.48	3.47	17.64	17.63
10	97.38 %	97.43 %	3.74	3.74	22.08	22.14

de igual manera los resultados promediados de las mezclas de la parte de prueba del conjunto de datos TIMIT, pero con el ruido del conjunto de datos PNL-100.

En cuanto a los resultados promedio de la métrica PESQ, Figura 5.1(A), muestra que los modelos convolucionales obtuvieron resultados similares en mezclas de audio con ruidos utilizados durante el entrenamiento. Por otro lado, el modelo MLP mostró el rendimiento más bajo. Sin embargo, cabe mencionar que los cinco modelos mostraron mejoras al incluir el módulo de atención. No obstante, en los experimentos que mostraron la capacidad de generalización de los modelos, Figura 5.1(B), el modelo convolucional de una dimensión, mostró el mejor rendimiento con y sin el módulo de atención. También es relevante mencionar que el modelo convolucional ramificado obtuvo el segundo mejor resultado en capacidad de generalización, pero no obtuvo resultados mejores al incorporar el módulo de atención.

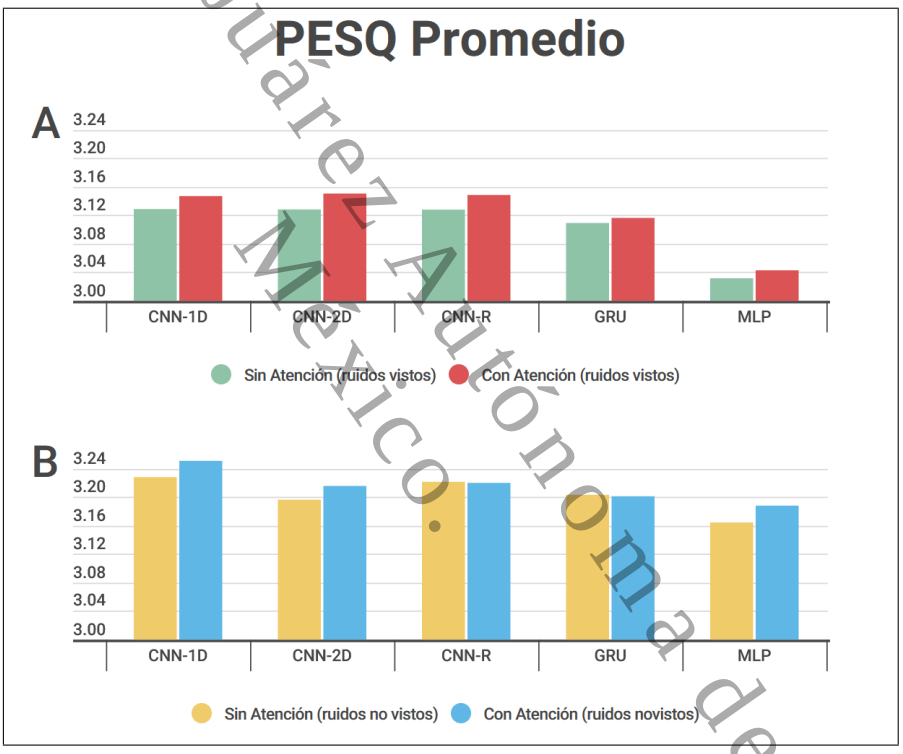

Figura 5.1: Resultados promedio de la métrica PESQ para los modelos con y sin modulo de atención. (A) la parte de prueba del conjunto de datos TIMIT y el ruido de NoiseX-92 y DEMAND, y (B) la parte de prueba del conjunto de datos TIMIT y el ruido de PNL-100.

En el caso de los resultados promedio de la métrica STOI, Figura 5.2(A), la red neuronal convolucional ramificada con el módulo de atención alcanzó el mejor resultado, seguido por el modelo convolucional de una dimensión con el módulo de atención. De forma similar a los resultados de la métrica PESQ, el modelo MLP mostró el menor rendimiento. En los experimentos que muestran la capacidad de generalización de los modelos, Figura 5.2(B), el modelo convolucional de una dimensión con el módulo de atención obtuvo los valores más altos. Por último, de forma similar a los

resultados de PESQ, el modelo basado en GRU obtuvo un rendimiento inferior al incluir el módulo de atención.

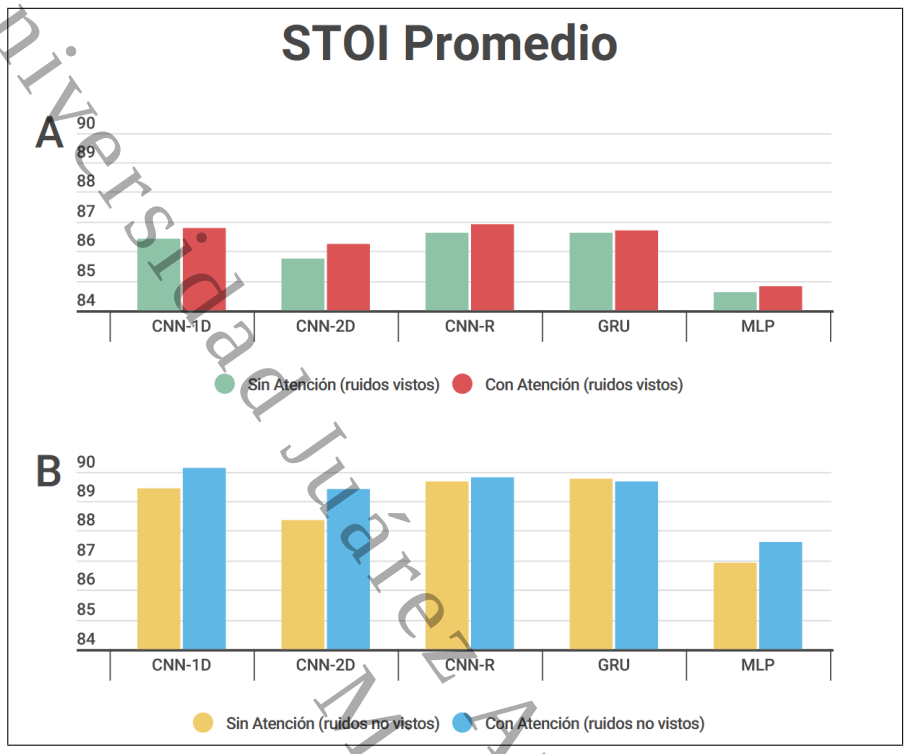

Figura 5.2: Resultados promedio de la métrica STOI para los modelos con y sin modulo de atención. (A) la parte de prueba del conjunto de datos TIMIT y el ruido de NoiseX-92 y DEMAND, y (B) la parte de prueba del conjunto de datos TIMIT y el ruido de PNL-100.

En cuanto a los resultados medios de la métrica SI-SDR, la Figura 5.3(A) muestra el mismo patrón que STOI, donde la red neuronal convolucional ramificada con el módulo de atención logró el mejor resultado, seguido del modelo convolucional de una dimensión con el módulo de atención. Por otro lado, la Figura 5.3(B) muestra que los resultados de la métrica promedio SI-SDR fueron menores en los datos que no formaron parte del entrenamiento. En consecuencia, es posible interpretar el resultado como que la cantidad de distorsión introducida por el proceso de separación en estos datos es mayor.

5.4. Estabilidad del Proceso de Entrenamiento

La Figura 5.4 muestra las curvas de pérdida de datos de entrenamiento y validación de los cinco modelos (diez considerando los modelos con el módulo de atención) para comparar la estabilidad del entrenamiento de los modelos. Las curvas de pérdida de datos muestran el entrenamiento completo; sin embargo, se utilizó una estrategia de detención anticipada, por lo que los modelos convergieron seis épocas antes de la última época mostrada en cada curva. Los modelos convolucionales de una dimensión y ramificado fueron los que convergieron en el menor número de épocas (25). Por otro

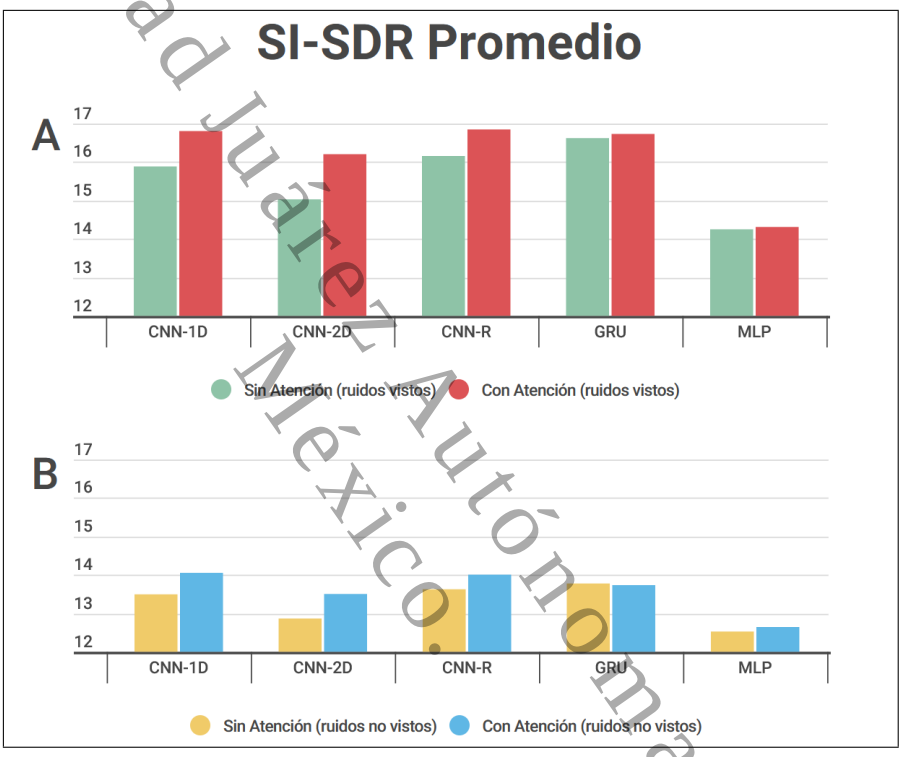

Figura 5.3: Resultados promedios de la métrica SI-SDR para los modelos con y sin módulo de atención. (A) la parte de prueba del conjunto de datos TIMIT y el ruido de NoiseX-92 y DEMAND, y (B) la parte de prueba del conjunto de datos TIMIT y el ruido de PNL-100.

lado, el modelo MLP fue el que más épocas necesitó para entrenarse, 57 en total, convergiendo en la época 51.

5.5. Consumo de Recursos Computacionales

Como se mencionó en el apartado 4.3, todos los modelos de red neuronal artificial fueron entrenados utilizando un equipo de cómputo en la nube, con procesador Intel Xeon de 2 GHz con 1 núcleo y 2 hilos, 12 GB de memoria RAM, 75 GB de almacenamiento, y una tarjeta gráfica NVIDIA Tesla P100 con 16 GB de memoria y 3584 núcleos CUDA.

La tabla 5.11 muestra el consumo de recursos durante el entrenamiento para cada uno de los tipos de modelo que formaron parte de la investigación (CNN 1D & 2D, GRU, y MLP). De modo específico, se aprecian los resultados para los siguientes tres tipos: modelos sin módulo de atención, modelos con módulo de atención y modelos profundos sin módulo de atención. Es importante mencionar que los modelos profundos CNN 1D & 2D y GRU tienen el triple de capas que esos mismos modelos en su versión no profunda; mientras que el modelo profundo MLP tiene el doble de capas que los otros modelos de su mismo tipo. Todos los modelos antes mencionados fueron creados con la finalidad de conocer el impacto del número de capas y parámetros en el consumo de recursos.

Tabla 5.11: Consumo de recursos computacionales durante el entrenamiento.

Modelo	CPU	RAM	GPU	HDD	Minutos (aprox. por época)
CNN 1D	95 % a 100 %	5.4 GB	70 % a 75 %	4 GB	04:46
CNN 1D + Atención	100 %	7.7 GB	65 % a 75 %	4 GB	05:29
CNN 1D Profundo	90 % a 100 %	6.7 GB	80 % a 95 %	4 GB	**22:56**
CNN 2D	75 % a 80 %	8.9 GB	85 % a 95 %	4 GB	03:32
CNN 2D + Atención	75 % a 80 %	9.8 GB	85 % a 95 %	4 GB	03:33
CNN 2D Profundo	70 % a 80 %	6.8 GB	90 % a 100 %	4 GB	**08:41**
GRU	100 %	5 GB	5 % a 10 %	4 GB	04:44
GRU + Atención	100 %	6.9 GB	5 % a 10 %	4 GB	05:18
GRU Profundo	100 %	5.5 GB	3 % a 10 %	4 GB	**13:59**
MLP	100 %	3.7 GB	20 % a 35 %	4 GB	01:02
MLP + Atención	100 %	5.7 GB	15 % a 30 %	4 GB	01:25
MLP Profundo	100 %	4.2 GB	20 % a 50 %	4 GB	**01:49**

Como se puede observar en la tabla 5.11, las versiones profundas de los modelos CNN de 1 y 2 dimensiones, así como el modelo GRU, tienden a aumentar significativamente el tiempo necesario para su entrenamiento por época, lo cual se traduce en una mayor necesidad de recursos computacionales

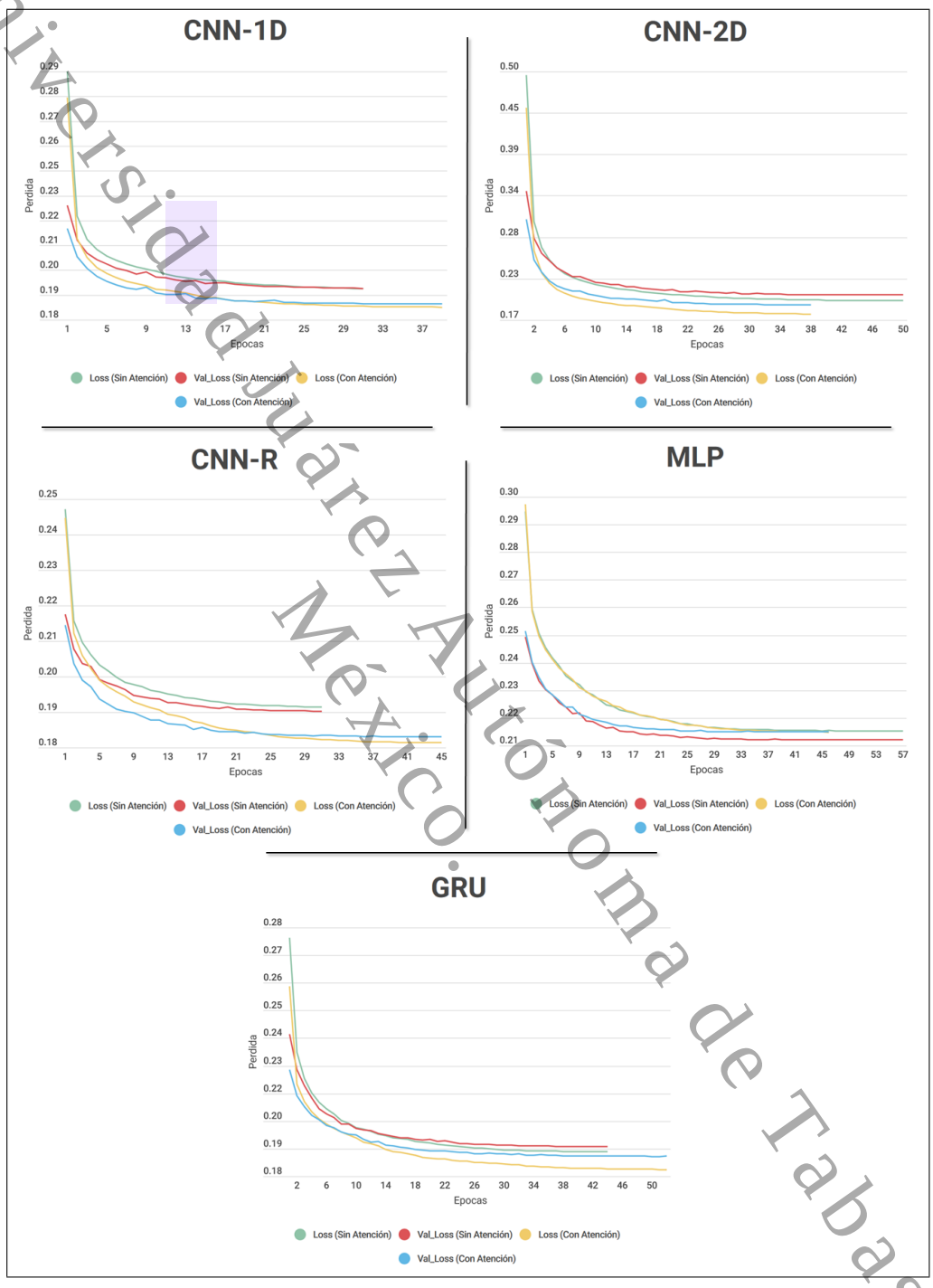

Figura 5.4: Las curvas de pérdida de entrenamiento de los modelos con y sin atención para los datos de entrenamiento y validación.

para entrenar modelos más profundos (con mayor cantidad de capas y parámetros entrenables). Ahora bien, aunque el modelo MLP profundo solo aumenta unos segundos el tiempo necesario para su entrenamiento por época, cabe recordar que el modelo MLP fue el que obtuvo el rendimiento más bajo conforme a las métricas utilizadas.

Respecto al proceso de experimentación, todos se realizaron utilizando un dispositivo de cómputo convencional con 8 GB de memoria y un procesador AMD de 4 núcleos a 2 GHz. La tabla 5.12 muestra el tiempo de ejecución para las predicciones de los modelos de red neuronal con atención. El tiempo de ejecución para las predicciones es el tiempo promedio de 10 ejecuciones para procesar segmentos de audio de 60 segundos.

Tabla 5.12: Consumo de recursos computacionales durante las predicciones.

Modelo	Parámetros entrenables	Tamaño (en MB)	Tiempo (en segundos)
CNN 1D	576,449	6.72	47.21
CNN 1D + Atención	647,629	7.59	47.30
CNN 1D Profunda	5,012,225	**57.7**	**447.88**
CNN 2D	1,381,057	15.9	41.23
CNN 2D + Atención	1,480,993	17.1	41.78
CNN 2D Profunda	5,906,689	**67.8**	**295.29**
GRU	1,644,800	18.8	1.60
GRU + Atención	1,715,980	19.7	1.70
GRU Profunda	5,261,568	**60.3**	**5.02**
MLP	1,904,640	21.8	1.94
MLP + Atención	1,975,820	22.7	2.07
MLP Profunda	10,760,448	**123**	**13.42**

De forma similar a los resultados de consumo de recursos computacionales durante el entrenamiento, al promediar el tiempo necesario para ejecutar los modelos CNN de 1 y 2 dimensiones, GRU y MLP; se identificó que las versiones profundas de los modelos requieren mucho más tiempo para procesar las señales de audio comparadas con las versiones pequeñas con y sin atención exploradas en esta investigación. También se puede apreciar que los modelos más profundos generan más parámetros entrenables, lo que a la vez genera que los modelos de red neuronal crezcan en tamaño (MB).

5.6. Discusión

Los resultados de los diferentes experimentos realizados durante este proyecto demostraron que la inclusión de un módulo de atención basado en la atención de múltiples encabezados y aplicados a

diferentes tipos de redes neuronales artificiales mejora el rendimiento de los modelos en el problema de la mejora de la voz. Además, se pudo constatar que la mejora es más significativa en las redes convolucionales que en los modelos MLP o recurrentes como GRU, incluso cuando se evalúa el modelo con datos nuevos. En las redes neuronales GRU, se observó una mejora en el modelo al incorporar el módulo de atención basado en la atención de múltiples encabezados en los experimentos que utilizaron mezclas con el mismo ruido que se utilizó durante el entrenamiento. Sin embargo, se obtuvieron peores resultados al probar la capacidad de generalización del modelo GRU con el módulo de atención en datos nuevos. Los modelos MLP muestran una mejora al incluir el módulo de atención; aun así, su rendimiento es inferior al de las redes convolucionales, que pueden tener entre un tercio y un cuarto de los parámetros entrenables que los MLP.

Los resultados de los experimentos reflejaron que la inclusión de un mecanismo de atención permite que las arquitecturas de los modelos de redes neuronales sean menos complejas sin sacrificar su eficiencia. Además, los resultados de esta investigación pretenden ayudar a otros investigadores a reducir el tiempo y los recursos comúnmente invertidos durante el proceso iterativo de prueba y error para entrenar nuevos modelos de redes neuronales MLP, CNN o GRU que incorporan un mecanismo basado en la atención.

Resulta necesario mencionar que se consideró esencial que los modelos evaluados fueran modelos de redes neuronales con arquitecturas sencillas. De este modo, se concluye que es posible crear modelos con arquitecturas sencillas (y con relativamente pocos parámetros entrenables) que obtengan resultados competitivos y consuman pocos recursos computacionales en su entrenamiento e implementación.

En la literatura, algunos trabajos han explorado las aplicaciones de la atención (basada en la atención de múltiples encabezados) al problema de la mejora de voz [36], [45], [32], [65]; estos trabajos demostraron que la inclusión de la atención mejora los resultados de los modelos de redes neuronales artificiales. Sin embargo, es necesario mencionar que estos trabajos propusieron arquitecturas complejas como redes neuronales híbridas con muchos parámetros entrenables, conexiones residuales o ramificaciones y utilizaron diferentes conjuntos de datos con diferentes técnicas de procesamiento.

La tabla 5.13 muestra los resultados promedio en términos de dos métricas objetivas que se usan comúnmente en la mejora del habla. La tabla compara los valores obtenidos por nuestros cuatro modelos de redes neuronales con los de los trabajos de [36], [45], [32] y [65].

Es necesario aclarar que los valores obtenidos por [36] son los SNR promedio de -5 dB a 5 dB; los valores obtenidos por [32], [45] y [65] son los SNR promedio de -5 dB a 15 dB. Algunas de las propuestas por otros autores utilizaron diferentes conjuntos de voz y ruido. Nuestros resultados en la Tabla 5.13 son las SNR promedio de -5 dB a 10 dB obtenidas por los modelos en el conjunto de prueba en un contexto con ruido visto durante el entrenamiento.

Tabla 5.13: Comparaciones promedio con las métricas PESQ y STOI entre los diferentes modelos y otras propuestas en la literatura de la mejora de voz.

Modelos	PESQ promedio	STOI promedio
MLP + Atención	3.20	88.98 %
CNN-1D + Atención	3.32	91.15 %
CNN-2D + Atención	3.32	90.61 %
CNN-R + Atención	3.32	91.24 %
GRU + Atención	3.29	91.02 %
DCN-SM [36]	2.83	90.40 %
DeepLPC-MHANet [45]	2.59	88.41 %
MHANet [32]	2.88	93.60 %
SE-T [65]	2.62	93.00 %

Hay algunas consideraciones de los experimentos que podrían haber influido en los resultados obtenidos.

- En primer lugar, aunque las señales de voz se corrompieron utilizando solo una señal de ruido a la vez, el ruido del conjunto de datos DEMAND (utilizado durante el entrenamiento) contiene grabaciones de ruidos acústicos de entornos interiores y exteriores en los que están presentes simultáneamente diferentes fuentes de ruido (de distinta naturaleza). El uso del conjunto de datos DEMAND introduce ruidos complejos de entornos naturales reales, que son muy difíciles de procesar para los modelos de redes neuronales.
- En segundo lugar, únicamente se utilizó el conjunto de datos de voz TIMIT, sin embargo, aunque podría considerarse pequeño en comparación con otros conjuntos de datos de voz, TIMIT contiene diversos tipos de voz con diversos acentos en la voz humana.
- En tercer lugar, se decidió aplicar únicamente la transformada de Fourier de tiempo corto (STFT, por sus siglas en inglés) con 256 puntos de frecuencia y normalización como preprocesamiento de las señales de audio, sin aplicar ningún filtro de frecuencia adicional ni ningún otro procesamiento, para no añadir carga computacional adicional y probar la verdadera capacidad de los diferentes tipos de redes neuronales.
- Por último, utilizamos una relación señal a ruido de -10dB a 10dB, ya que es un rango en el que habitualmente se realizan estudios de investigación sobre el problema de mejora de voz, puesto que se intentó cubrir una importante variedad de escenarios de ruido. Sin embargo, no se han tenido en cuenta los escenarios con niveles de ruido muy elevados, como por ejemplo las condiciones de ruido en la cabina de los aviones durante el despegue. Esto representó una limitación para este estudio.

64

Los resultados de esta investigación pueden servir como referencia en la construcción de soluciones en las que los modelos de red neuronal artificial aplicados a la mejora de voz no sean complejos pero a la vez eficientes, representando una oportunidad a la hora de implementar propuestas con características similares las nuestras en dispositivos computacionales con recursos limitados.

Los artículos publicados con los resultados mostrados en la investigación se encuentran en el Anexo B, y las ponencias derivadas en el Anexo C. En el siguiente capítulo se encuentran las conclusiones obtenidas mediante los experimentos realizados.

Capítulo 6

Conclusiones y Trabajos Futuros

Las conclusiones obtenidas en esta tesis son las siguientes:

Un modelo de red neuronal artificial que busque imitar la capacidad de atención auditiva selectiva puede ser construido tomando como elemento esencial un módulo de atención que tenga por función ponderar los diversos parámetros del modelo. Dicha ponderación contribuirá a que la red neuronal obtenga resultados eficientes a pesar de tener una cantidad de parámetros considerada baja (no más de dos millones). Los modelos propuestos fueron capaces de ofrecer muy buenos resultados para la separación y mejora de voz en mezclas de audio generadas en ambientes con ruidos presentes y mediante un solo micrófono, por tanto, se combinan las restricciones en cuanto a la cantidad de información disponible y la variabilidad de las fuentes de audio que podrían estar mezcladas.

Los resultados obtenidos con los experimentos nos permiten afirmar que un modelo de red neuronal que incluye un módulo de atención puede construirse sobre una arquitectura con pocos atributos de clasificación (simple), y tener como resultado un modelo que consume menos recursos computacionales, pero que, de acuerdo a las métricas de rendimiento, ofrece valores competitivos respecto a propuestas con arquitecturas compuestas de millones de parámetros (complejas), disponibles en la literatura. En virtud de lo anterior, se puede contar con una propuesta con la calidad necesaria para la separación y mejora de voz que puede ser utilizada por otros sistemas computacionales.

Las conclusiones antes expresadas se encuentran sustentadas en las métricas de evaluación del rendimiento aplicadas a los modelos experimentados. Dichas métricas fueron PESQ, STOI y SI-SDR, empleadas de modo común para el problema de la separación y mejora del habla.

La métrica PESQ comparó la señal de voz original contra la señal de voz generada a partir del proceso de separación y mejora. Como se ha mencionado, un valor de PESQ más grande implica una relación más estrecha entre la voz generada y la original. Los valores de PESQ van de -0.5 a 5, donde 5 representa una voz generada sin distorsiones, respecto a la original.

Una vez que se realizó el proceso de separación y mejora de voz, la señal resultante fue evaluada en pequeños segmentos temporales y luego promediado todos los valores de calidad obtenidos, dando como resultado el STOI, que representa el porcentaje de inteligibilidad.

La métrica SI-SDR nos indicó la calidad de la separación de dos fuentes de sonido. Para nuestro caso, la primera fuente la representó la señal de voz con ruido y la segunda fuente fue la voz obtenida una vez que se realizó el proceso de separación y mejora. Un valor SI-SDR más grande representa una mejor calidad de separación.

Los trabajos futuros de esta investigación podrían explorar lo siguiente:

- El diseño de una comparación utilizando estrategias de mapeo y enmascaramiento [62], ya que se puede definir como un problema de regresión si el objetivo es mapear una representación de tiempo-frecuencia del habla limpia directamente; o como un problema de clasificación, si el objetivo es producir una matriz (conocida como máscara) que clasifique cada porción de la señal (como habla o ruido), para filtrar el habla ruidosa con esta máscara y generar la señal de voz limpia mejorada [63].
- Aunque al aplicar un módulo de atención a los modelos, los resultados de eficiencia mejoraron en algunas métricas obtenidas, se pudo observar que el incremento en la eficiencia fue poco, por tanto, consideramos conveniente que un trabajo futuro sea colocar el módulo de atención en distintas ubicaciones dentro de la arquitectura de la red neuronal artificial, por ejemplo, al inicio o en el medio de las arquitecturas, y evaluar el desempeño de los modelos a partir de estos cambios.
- Los valores utilizados por los modelos son determinísticos, sin embargo, muchas situaciones de la vida real son imprecisas y con vaguedad, por tanto, tratar de capturar esa vaguedad en los modelos puede ofrecer resultados que sean mejores. Un trabajo futuro es incluir algoritmos computacionales de lógica difusa al módulo de atención para tratar problemas donde exista incertidumbre en los datos de entrada o en la predicción y clasificación de los datos.
- En el aprendizaje profundo, la eficiencia de los resultados puede estar muy ligada a conjuntos de datos muy grandes, ya que precisamente esta tecnología fue concebida para funcionar sobre conjuntos de datos muy grandes. Dado lo anterior, un trabajo futuro es explorar el desempeño del módulo de atención al utilizar conjuntos de datos de voz y ruido mucho más grandes, con decenas o cientos de horas de mezclas de audio.
- En las investigaciones de separación y mejora de voz se busca que los conjuntos de datos utilizados sean lo más representativo posible de los entornos reales, y dado que precisamente esta investigación busca un fin social al contribuir a la solución de problemáticas en escenarios reales, un trabajo a futuro es experimentar utilizando más ruidos simultáneamente presentes a diferentes SNR para generar una mejor similitud de un entorno natural en el que están

presentes simultáneamente diferentes fuentes de ruido de distinta naturaleza.

Referencias

[1] R.C. Atkinson and R.M. Shiffrin. Human Memory: A Proposed System and its Control Processes. In Kenneth W. Spence and Janet Taylor Spence, editors, *Psychology of Learning and Motivation*, volume 2, pages 89–195. Academic Press, January 1968.

[2] Richard C. Atkinson, Richard J. Herrnstein, Gardner Lindzey, and R. Duncan Luce, editors. *Stevens'handbook of experimental psychology: Perception and motivation; Learning and cognition*. John Wiley & Sons, Oxford, England, 1988. Pages: xxviii, 1932.

[3] O. Campesato. *Artificial Intelligence, Machine Learning, and Deep Learning*. Mercury Learning & Information, 2020.

[4] E. Colin Cherry. Some Experiments on the Recognition of Speech, with One and with Two Ears. *The Journal of the Acoustical Society of America*, 25(5):975–979, September 1953. Publisher: Acoustical Society of America.

[5] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, October 2014. arXiv:1409.1259 [cs, stat].

[6] Patricia S Churchland and Terrence J Sejnowski. *The computational brain*. MIT press, 2016.

[7] D. Ward, H. Wierstorf, R. D. Mason, E. M. Grais, and M. D. Plumbley. BSS Eval or Peass? Predicting the Perception of Singing-Voice Separation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 596–600, April 2018. Journal Abbreviation: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

[8] Jean-Louis Durrieu and Jean-Philippe Thiran. Musical Audio Source Separation Based on User-Selected F0 Track. In Fabian Theis, Andrzej Cichocki, Arie Yeredor, and Michael Zibulevsky, editors, *Latent Variable Analysis and Signal Separation*, pages 438–445, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[9] Andrea Galassi, Marco Lippi, and Paolo Torroni. Attention in Natural Language Processing. *IEEE transactions on neural networks and learning systems*, PP, sep 2020. Place: United States.

[10] Laura Gamin Perez. *Arquitecturas de Redes Neuronales Profundas para mejora de voz: De los perceptrones multicapa a las redes completamente convolucionales (FCNs)*. PhD thesis, Universidad Autónoma de Madrid, Madrid, 2019.

[11] M. W Gardner and S. R Dorling. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric Environment*, 32(14):2627–2636, August 1998.

[12] J. Garofolo, Lori Lamel, W. Fisher, Jonathan Fiscus, D. Pallett, N. Dahlgren, and V. Zue. Timit acoustic-phonetic continuous speech corpus. *Linguistic Data Consortium*, November 1992.

[13] Mandar Gogate, Kia Dashtipour, Peter Bell, and Amir Hussain. Deep neural network driven binaural audio visual speech separation. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, July 2020. ISSN: 2161-4407.

[14] Guoning Hu and DeLiang Wang. A tandem algorithm for pitch estimation and voiced speech segregation. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):2067–2079, November 2010. Conference Name: IEEE Transactions on Audio, Speech, and Language Processing.

[15] J. Durrieu, A. Ozerov, C. Févotte, G. Richard, and B. David. Main instrument separation from stereophonic audio signals using a source/filter model. In *2009 17th European Signal Processing Conference*, pages 15–19, August 2009. Journal Abbreviation: 2009 17th European Signal Processing Conference.

[16] Stanisław Jastrzębski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual Connections Encourage Iterative Inference, March 2018. arXiv:1710.04773 [cs].

[17] Jesper Jensen, Cees H. Taal, Jesper Jensen, and Cees H. Taal. An algorithm for predicting the intelligibility of speech masked by modulated noise maskers. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 24(11):2009–2022, 2016.

[18] William A Johnston and Veronica J Dark. Selective attention. *Annual review of psychology*, 37(1):43–75, 1986. Publisher: Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA.

[19] Fumi Katsuki and Christos Constantinidis. Bottom-Up and Top-Down Attention: Different Processes and Overlapping Neural Systems. *The Neuroscientist*, 20(5):509–521, December 2013. Publisher: SAGE Publications Inc STM.

[20] R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics and Probability Letters*, 33(3):291–297, May 1997.

[21] Trausti Kristjansson, John Hershey, Peder Olsen, Steven Rennie, and R.A. Gopinath. *Super-human multi-talker speech recognition: The IBM 2006 speech separation challenge system*, volume 1. Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, January 2006.

[22] Alexander Lerch. *Audio content analysis: an introduction*. Wiley, Hoboken, N.J, 2012.

[23] D. Lesenfants and T. Francart. The interplay of top-down focal attention and the cortical tracking of speech. *Scientific Reports*, 10(1):6922, April 2020.

[24] Lujun Li, Zhenxing Lu, Tobias Watzel, Ludwig Kürzinger, and Gerhard Rigoll. Light-weight self-attention augmented generative adversarial networks for speech enhancement. *Electronics*, 10(13), 2021.

[25] Xuechen Li, Xinfang Ma, Fengchao Xiao, Fei Wang, and Shicheng Zhang. Application of Gated Recurrent Unit (GRU) Neural Network for Smart Batch Production Prediction. *Energies*, 13(22):6121, January 2020. Number: 22 Publisher: Multidisciplinary Digital Publishing Institute.

[26] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. *CoRR*, abs/1508.04025, 2015.

[27] Shing Lyu. Artificial Intelligence and Machine Learning. In *Practical Rust Projects: Building Game, Physical Computing, and Machine Learning Applications*. Apress, Berkeley, CA, February 2020.

[28] M. N. Schmidt, J. Larsen, and F. Hsiao. Wind Noise Reduction using Non-Negative Sparse Coding. In *2007 IEEE Workshop on Machine Learning for Signal Processing*, pages 431–436, August 2007. Journal Abbreviation: 2007 IEEE Workshop on Machine Learning for Signal Processing.

[29] Ian McLoughlin. *Applied Speech and Audio Processing: With Matlab Examples*. Cambridge University Press, Cambridge, 2009.

[30] Federico Miyara. *Acústica y sistemas de sonido*. UNR Editora, Rosario, 4a ed edition, 2010.

[31] N. K. Chauhan and K. Singh. A Review on Conventional Machine Learning vs Deep Learning. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, pages 347–352, September 2018. Journal Abbreviation: 2018 International Conference on Computing, Power and Communication Technologies (GUCON).

[32] Aaron Nicolson and Kuldip K. Paliwal. Masked multi-head self-attention for causal speech enhancement. *Speech Communication*, 125:80–96, December 2020.

[33] Soha A. Nossier, Julie Wall, Mansour Moniri, Cornelius Glackin, and Nigel Cannings. An experimental analysis of deep learning architectures for supervised speech enhancement. *Electronics*, 10(1), 2021.

[34] Michael Paluszek and Stephanie Thomas. *Practical MATLAB Deep Learning: A Project-Based Approach*. Apress, 2020.

[35] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, April 2015. ISSN: 2379-190X.

[36] Ashutosh Pandey and DeLiang Wang. Dense CNN With Self-Attention for Time-Domain Speech Enhancement. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 29:1270–1279, January 2021.

[37] Rolf Pfeifer, Fumiya Iida, and Max Lungarella. Cognition from the bottom up: on biological inspiration, body morphology, and soft materials. *Trends in Cognitive Sciences*, 18(8):404–413, August 2014.

[38] James O. Pickles. Chapter 1 - Auditory pathways: anatomy and physiology. In Michael J. Aminoff, François Boller, and Dick F. Swaab, editors, *Handbook of Clinical Neurology*, volume 129, pages 3–25. Elsevier, January 2015.

[39] R. G. M. M. Jayamaha, M. R. R. Senadheera, T. N. C. Gamage, K. D. P. B. Weerasekara, G. A. Dissanayaka, and G. N. Kodagoda. VoizLock - Human Voice Authentication System using Hidden Markov Model. In *2008 4th International Conference on Information and Automation for Sustainability*, pages 330–335, December 2008. Journal Abbreviation: 2008 4th International Conference on Information and Automation for Sustainability.

[40] Waseem Rawat and Zenghui Wang. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation*, 29(9):2352–2449, September 2017.

[41] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, 2:749–752 vol.2, 2001.

[42] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals. WSJCAMO: a British English speech corpus for large vocabulary continuous speech recognition. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 81–84 vol.1, May 1995. ISSN: 1520-6149.

[43] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey. Sdr - half-baked or well done? *Computer and information sciences*, 2018.

[44] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey. SDR – Half-baked or Well Done? In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630, May 2019. ISSN: 2379-190X.

[45] Sujan Kumar Roy, Aaron Nicolson, and Kuldip K. Paliwal. DeepLPC-MHANet: Multi-Head Self-Attention for Augmented Kalman Filter-Based Speech Enhancement. *IEEE Access*, 9:70516–70530, 2021.

[46] S. Ewert, B. Pardo, M. Muller, and M. D. Plumbley. Score-Informed Source Separation for Musical Audio Recordings: An overview. *IEEE Signal Processing Magazine*, 31(3):116–124, May 2014.

[47] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A Dataset and Taxonomy for Urban Sound Research. In *Proceedings of the 22nd ACM international conference on Multimedia*, MM '14, pages 1041–1044, New York, NY, USA, November 2014. Association for Computing Machinery.

[48] Stuart C. Shapiro. Artificial Intelligence (AI). In *Encyclopedia of Computer Science*. John Wiley and Sons Ltd., GBR, 2003.

[49] D.U. Silverthorn. *Fisiologia humana: Un enfoque integrado*. Médica Panamericana, Argentina, 4ta edition, 2009.

[50] Robert J. Sternberg, Karin Sternberg, and Jeff Mio. *Cognitive Psychology*. Wadsworth, Cengage Learning, USA, 6ta edition, 2012.

[51] E.A. Styles. Attention, perception and memory: An integrated introduction. *Attention, Perception and Memory: An Integrated Introduction*, pages 1–368, January 2005.

[52] Elizabeth A Styles. *Psicología de la atención*. Editorial Centro de Estudios Ramón Areces, 2010.

[53] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, September 2011.

[54] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. The diverse environments multi-channel acoustic noise database (demand): A database of multichannel environmental noise recordings. In *Proceedings of Meetings on Acoustics ICA2013*, volume 19, page 035081. Acoustical Society of America, 2013.

[55] Wei-Ho Tsai, Hsin-min Wang, and Dwight Rodgers. *Automatic singer identification of popular music recordings via estimation and modeling of solo vocal signal.* Eighth European Conference on Speech Communication and Technology, January 2003.

[56] Raphael Ullmann, Ramya Rasipuram, Mathew Magimai-Doss, and Hervé Bourlard. Objective intelligibility assessment of text-to-speech systems through utterance verification. In *Interspeech 2015*, pages 3501–3505. ISCA, September 2015.

[57] Andrew Varga and Herman JM Steeneken. Assessment for automatic speech recognition: Ii. noisex-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech communication*, 12(3):247–251, 1993.

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. event-place: Long Beach, California, USA.

[59] Christophe Veaux, Junichi Yamagishi, and Simon King. The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. In *2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, pages 1–4, November 2013.

[60] R. Venkatesan and A. Balaji Ganesh. Deep Recurrent Neural Networks Based Binaural Speech Segregation for the Selection of Closest Target of Interest. *Multimedia Tools Applications*, 77(15):20129–20156, aug 2018. Place: USA Publisher: Kluwer Academic Publishers.

[61] Emmanuel Vincent, Tuomas Virtanen, and Sharon Gannot. *Audio source separation and speech enhancement*. John Wiley & Sons, 2018.

[62] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 26(10):1702–1726, oct 2018.

[63] Yuxuan Wang, Arun Narayanan, and DeLiang Wang. On training targets for supervised speech separation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 22(12):1849–1858, dec 2014.

[64] Y. Luo, Z. Chen, and N. Mesgarani. Speaker-Independent Speech Separation With Deep Attractor Network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(4):787–796, April 2018.

[65] Weiwei Yu, Jian Zhou, HuaBin Wang, and Liang Tao. SETransformer: Speech Enhancement Transformer. *Cognitive Computation*, 14(3):1152–1158, May 2022.

[66] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural Computation*, 31(7):1235–1270, 2019.

[67] Asri Yuliani, M. Faizal Amri, Endang Suryawati, Ade Ramdan, and Hilman Pardede. Speech enhancement using deep learning methods: A review. *Jurnal Elektronika dan Telekomunikasi*, 21(1):19–26, 2021.

[68] Noel Zacarias-Morales, Pablo Pancardo, José A. Hernández-Nolasco, and Matias Garcia-Constantino. Attention-Inspired Artificial Neural Networks for Speech Processing: A Systematic Review. *Symmetry*, 13(2), 2021.

Anexo A

Códigos

```python
import numpy as np
import soundfile as sf
import matplotlib.pyplot as plt
import librosa
import librosa.display
from scipy import signal

import tensorflow as tf
from tensorflow.keras.layers import *
from tensorflow.keras.models import *
from tensorflow.keras.utils import *
from tensorflow.keras.callbacks import *
from tensorflow.keras.optimizers import *

from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler
```

Figura A.1: Código de las librerías utilizadas.

```python
class DotProductAttentionManual(tf.keras.layers.Layer):
    def __init__(self, use_scale=True, **kwargs):
        super(DotProductAttentionManual, self).__init__(**kwargs)
        self.use_scale = use_scale

    def build(self, input_shape):
        query_shape = input_shape[0]
        if self.use_scale:
            dim_k = tf.cast(query_shape[-1], tf.float32)
            self.scale = 1 / tf.sqrt(dim_k)
        else:
            self.scale = None

    def call(self, input):
        query, key, value = input
        score = tf.matmul(query, key, transpose_b=True)
        if self.scale is not None:
            score *= self.scale

        softmax_QK = tf.nn.softmax(score)
        out = tf.matmul(softmax_QK, value)

        return out
```

Figura A.2: Código de la atención de productos de puntos escalados.

```python
class MultiHeadAttentionManual(tf.keras.layers.Layer):
    def __init__(self, h=8, **kwargs):
        super(MultiHeadAttentionManual, self).__init__(**kwargs)
        self.h = h

    def build(self, input_shape):
        query_shape, key_shape, value_shape = input_shape
        d_model = query_shape[-1]

        units = d_model // self.h

        self.layersQ = []
        for _ in range(self.h):
            layer = Dense(units, activation=None, use_bias=False)
            layer.build(query_shape)
            self.layersQ.append(layer)

        self.layersK = []
        for _ in range(self.h):
            layer = Dense(units, activation=None, use_bias=False)
            layer.build(key_shape)
            self.layersK.append(layer)

        self.layersV = []
        for _ in range(self.h):
            layer = Dense(units, activation=None, use_bias=False)
            layer.build(value_shape)
            self.layersV.append(layer)

        self.attention = DotProductAttentionManual(True)

        self.out = Dense(d_model, activation=None, use_bias=False)
        self.out.build((query_shape[0], query_shape[1], self.h * units))

    def call(self, input):
        query, key, value = input

        q = [layer(query) for layer in self.layersQ]
        k = [layer(key) for layer in self.layersK]
        v = [layer(value) for layer in self.layersV]


        head = [self.attention([q[i], k[i], v[i]]) for i in range(self.h)]

        out = self.out(tf.concat(head, -1))
        return out
```

Figura A.3: Código de la atención de múltiples encabezados.

```python
class AttentionModule(tf.keras.layers.Layer):
    def __init__(self, h=8, dr=0.10, **kwargs):
        super(AttentionModule, self).__init__(**kwargs)

        self.num_heads = h
        self.dropRate = dr

    def build(self, input_shape):
        d_model = input_shape[-1]

        self.MHA = MultiHeadAttentionManual(h=self.num_heads)
        self.layernorm1 = LayerNormalization(epsilon=1e-6)
        self.layernorm2 = LayerNormalization(epsilon=1e-6)
        self.mlp1 = Dense(d_model, activation=None)
        self.relu = ReLU()
        self.drop1 = Dropout(rate=self.dropRate)

    def call(self, inputs):
        attn_output = self.MHA([inputs, inputs, inputs])
        drop1_output = self.drop1(attn_output)

        norm1_output = self.layernorm1(inputs + drop1_output)

        feedfoward_output = self.mlp1(norm1_output)
        feedfoward_output = self.relu(feedfoward_output)

        output = self.layernorm2(norm1_output + feedfoward_output)

        return output
```

Figura A.4: Código del módulo de atención.

```python
def DNN_Network(shape_1, shape_2):
    input = Input(shape=shape_1, name="base_input")
    x = Dense(512, activation='relu', name="Dense_01")(input)
    x = Dropout(0.05, name='DropOut_01')(x)
    x = Dense(512, activation='relu', name="Dense_02")(x)
    x = Dropout(0.05, name='DropOut_02')(x)
    x = Dense(512, activation='relu', name="Dense_03")(x)
    x = Dropout(0.05, name='DropOut_03')(x)
    x = Dense(512, activation='relu', name="Dense_04")(x)
    x = Dropout(0.05, name='DropOut_04')(x)
    x = Dense(512, activation='relu', name="Dense_05")(x)
    x = Dropout(0.05, name='DropOut_05')(x)
    x = Dense(512, activation='relu', name="Dense_06")(x)
    x = Dropout(0.05, name='DropOut_06')(x)
    x = Dense(512, activation='relu', name="Dense_07")(x)
    x = Dropout(0.05, name='DropOut_07')(x)
    x = Dense(256, activation='relu', name="Dense_08")(x)

    x = AttentionModule(h=4, dr=0.10)(x)
    output = Dense(shape_2, name="Lineal_final")(x)

    return Model(inputs=input, outputs=output)
```

Figura A.5: Código de la arquitectura perceptrón multicapa.

```python
def CNN_1D_Network(shape_1, shape_2):
    input = Input(shape=shape_1, name="base_input")

    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_01")(input)
    x = PReLU(name="PReLU_01")(x)
    x = Dropout(0.1, name='DropOut_01')(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_02")(x)
    x = PReLU(name="PReLU_02")(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_03")(x)
    x = PReLU(name="PReLU_03")(x)
    x = Dropout(0.1, name='DropOut_02')(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_04")(x)
    x = PReLU(name="PReLU_04")(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_05")(x)
    x = PReLU(name="PReLU_05")(x)
    x = Dropout(0.1, name='DropOut_03')(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_06")(x)
    x = PReLU(name="PReLU_06")(x)
    x = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_07")(x)
    x = PReLU(name="PReLU_07")(x)
    x = Dropout(0.1, name='DropOut_04')(x)
    x = Conv1D(1, 16, strides=1, padding='same', name="Conv1D_08")(x)
    x = PReLU(name="PReLU_08")(x)

    x = AttentionModule(h=4, dr=0.10)(x)
    output = Dense(shape_2, name="Lineal_final")(x)

    return Model(inputs=input, outputs=output)
```

Figura A.6: Código de la arquitectura convolucional de una dimensión.

```python
def CNN_2D_Network(shape_1, shape_2):
    input = Input(shape=shape_1, name="base_input")

    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_01")(input)
    x = PReLU(name="PReLU_01")(x)
    x = Dropout(0.1, name='DropOut_01')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_02")(x)
    x = PReLU(name="PReLU_02")(x)
    x = Dropout(0.1, name='DropOut_02')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_03")(x)
    x = PReLU(name="PReLU_03")(x)
    x = Dropout(0.1, name='DropOut_03')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_04")(x)
    x = PReLU(name="PReLU_04")(x)
    x = Dropout(0.1, name='DropOut_04')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_05")(x)
    x = PReLU(name="PReLU_05")(x)
    x = Dropout(0.1, name='DropOut_05')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_06")(x)
    x = PReLU(name="PReLU_06")(x)
    x = Dropout(0.1, name='DropOut_06')(x)
    x = Conv2D(64, 4, strides=1, padding='same', name="Conv2D_07")(x)
    x = PReLU(name="PReLU_07")(x)
    x = Dropout(0.1, name='DropOut_07')(x)
    x = Conv2D(1, 4, strides=1, padding='same', name="Conv2D_08")(x)
    x = PReLU(name="PReLU_08")(x)

    x = AttentionModule(h=4, dr=0.10)(x)
    output = Dense(shape_2, name="Lineal_final")(x)

    return Model(inputs=input, outputs=output)
```

Figura A.7: Código de la arquitectura convolucional de dos dimensiones.

```python
def Branch_CNN(shape_1, shape_2):
    input = Input(shape=shape_1, name="base_input")

    x_01 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_01")(input)
    x_01 = PReLU(name="PReLU_01")(x_01)
    x_01 = Dropout(0.1, name='DropOut_01')(x_01)
    x_01 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_02")(x_01)
    x_01 = PReLU(name="PReLU_02")(x_01)
    x_01 = Dropout(0.1, name='DropOut_02')(x_01)
    x_01 = Conv1D(32, 16, strides=1, padding='same', name="Conv1D_03")(x_01)
    x_01 = PReLU(name="PReLU_03")(x_01)
    x_01 = Dropout(0.1, name='DropOut_03')(x_01)
    x_01 = Conv1D(2, 16, strides=1, padding='same', name="Conv1D_04")(x_01)
    x_01 = PReLU(name="PReLU_04")(x_01)

    x_02 = Reshape((256, 1), input_shape=(256,), name="Reshape_01")(x_01[:,:,0])
    x_02 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_05")(x_02)
    x_02 = PReLU(name="PReLU_05")(x_02)
    x_02 = Dropout(0.1, name='DropOut_04')(x_02)
    x_02 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_06")(x_02)
    x_02 = PReLU(name="PReLU_06")(x_02)
    x_02 = Dropout(0.1, name='DropOut_05')(x_02)
    x_02 = Conv1D(32, 16, strides=1, padding='same', name="Conv1D_07")(x_02)
    x_02 = PReLU(name="PReLU_07")(x_02)
    x_02 = Dropout(0.1, name='DropOut_06')(x_02)
    x_02 = Conv1D(1, 16, strides=1, padding='same', name="Conv1D_08")(x_02)
    x_02 = PReLU(name="PReLU_08")(x_02)

    x_03 = Reshape((256, 1), input_shape=(256,), name="Reshape_02")(x_01[:,:,1])
    x_03 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_09")(x_03)
    x_03 = PReLU(name="PReLU_09")(x_03)
    x_03 = Dropout(0.1, name='DropOut_07')(x_03)
    x_03 = Conv1D(64, 16, strides=1, padding='same', name="Conv1D_10")(x_03)
    x_03 = PReLU(name="PReLU_10")(x_03)
    x_03 = Dropout(0.1, name='DropOut_08')(x_03)
    x_03 = Conv1D(32, 16, strides=1, padding='same', name="Conv1D_11")(x_03)
    x_03 = PReLU(name="PReLU_11")(x_03)
    x_03 = Dropout(0.1, name='DropOut_09')(x_03)
    x_03 = Conv1D(1, 16, strides=1, padding='same', name="Conv1D_12")(x_03)
    x_03 = PReLU(name="PReLU_12")(x_03)

    concat_01 = Concatenate(axis=-1, name="Concat_01")([x_03, x_02])

    x_04 = Flatten(name="Flatten_03")(concat_01)
    x_04 = AttentionModule(h=4, dr=0.10)(x_04)
    x_04 = Dense(512, activation='relu', name="Dense_01")(x_04)
    x_04 = Dense(512, activation='relu', name="Dense_02")(x_04)

    output = Dense(shape_2, name="Lineal_final")(x_04)

    return Model(inputs=input, outputs=output)
```

Figura A.8: Código de la arquitectura convolucional ramificada.

```python
def GRU_Network(shape_1, shape_2):
    input = Input(shape=shape_1, name="base_input")
    x = GRU(256, activation='relu',
            recurrent_activation='sigmoid',
            return_sequences=True, name="gru_01")(input)
    x = GRU(256, activation='relu',
            recurrent_activation='sigmoid',
            return_sequences=True, name="gru_02")(x)
    x = GRU(256, activation='relu',
            recurrent_activation='sigmoid',
            return_sequences=True, name="gru_03")(x)
    x = GRU(256, activation='relu',
            recurrent_activation='sigmoid',
            return_sequences=True, name="gru_04")(x)

    x = AttentionModule(h=4, dr=0.10)(x)
    output = Dense(shape_2, name="Lineal_final")(x)

    return Model(inputs=input, outputs=output)
```

Figura A.9: Código de la arquitectura de tipo unidades recurrentes cerradas.

```
Model.compile(optimizer=Adam(learning_rate=0.0001,
                             beta_1=0.9,
                             beta_2=0.999,
                             epsilon=1e-08),
              loss='MeanSquaredError',
              metrics=['accuracy'])
```

Figura A.10: Código para compilar los modelos de red neuronal artificial.

```
save_model_path = 'work/Model-{epoch:03d}-{accuracy:.4f}-{val_accuracy:.4f}.h5'
save_csv_path = 'work/Model_csv_logger_data.csv'

checkpoint_callback = ModelCheckpoint(filepath=save_model_path,
                                      monitor='val_accuracy',
                                      verbose=1,
                                      save_best_only=True,
                                      mode='auto',
                                      save_freq='epoch')

earlystop_callback = EarlyStopping(monitor='val_accuracy',
                                   min_delta=0,
                                   patience=6,
                                   verbose=1,
                                   mode='auto',
                                   baseline=0.50,
                                   restore_best_weights=True)

csv_logger_callback = CSVLogger(save_csv_path, append=True)

ReduceLRoP_callback = ReduceLROnPlateau(monitor='val_accuracy',
                                        factor=0.8,
                                        patience=1,
                                        verbose=1,
                                        mode='auto',
                                        min_lr=0.000001)
class Add_LR_CSVLogger(Callback):
    def on_epoch_end(self, lr, logs):
        logs['learning rate'] = lr
```

Figura A.11: Código de las -callback- utilizadas en los entrenamientos de todos los modelos de red neuronal artificial.

```
history = Model.fit(x_train, y_train,
                    batch_size=64,
                    epochs=60,
                    validation_split=0.1,
                    callbacks=[checkpoint_callback,
                               ReduceLRoP_callback,
                               Add_LR_CSVLogger(),
                               csv_logger_callback,
                               earlystop_callback])
```

Figura A.12: Código para entrenar todos los modelos de red neuronal artificial.

Anexo B

Artículos Derivados de la Investigación

1) Zacarias-Morales, N., Pancardo, P., Hernández-Nolasco, J. A., & Garcia-Constantino, M. (2021). Attention-Inspired Artificial Neural Networks for Speech Processing: A Systematic Review. Symmetry, 13(2), 214. MDPI AG. http://doi.org/10.3390/sym13020214.

2) Zacarias-Morales, N., Hernández-Nolasco, J. A., & Pancardo, P. (2022). Red neuronal convolucional ramificada con atención para la mejora de voz. Research in Computing Science, 151(6), 119-132.

3) Zacarias-Morales, N., Hernández-Nolasco, J. A., & Pancardo, P. (2023). Full Single-Type Deep Learning Models with Multihead Attention for Speech Enhancement. Applied Intelligence. https://doi.org/10.1007/s10489-023-04571-y

4) Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P., & Olvera-López, J. A. (2023). Redes Neuronales Convolucionales con Atención para Atenuar Ruido en Señales de Audio. Komputer Sapiens, 15(2), 16-20.

symmetry

MDPI

Systematic Review

Attention-Inspired Artificial Neural Networks for Speech Processing: A Systematic Review

Noel Zacarias-Morales [1], Pablo Pancardo [1,*], José Adán Hernández-Nolasco [1] and Matias Garcia-Constantino [2]

[1] Academic Division of Sciences and Information Technology, Juarez Autonomous University of Tabasco, Tabasco 86690, Mexico; 201H18002@alumno.ujat.mx (N.Z.-M.); adan.hernandez@ujat.mx (J.A.H.-N.)
[2] School of Computing, Ulster University, Jordanstown BT37 0QB, UK; m.garcia-constantino@ulster.ac.uk
* Correspondence: pablo.pancardo@ujat.mx

Abstract: Artificial Neural Networks (ANNs) were created inspired by the neural networks in the human brain and have been widely applied in speech processing. The application areas of ANN include: Speech recognition, speech emotion recognition, language identification, speech enhancement, and speech separation, amongst others. Likewise, given that speech processing performed by humans involves complex cognitive processes known as auditory attention, there has been a growing amount of papers proposing ANNs supported by deep learning algorithms in conjunction with some mechanism to achieve symmetry with the human attention process. However, while these ANN approaches include attention, there is no categorization of attention integrated into the deep learning algorithms and their relation with human auditory attention. Therefore, we consider it necessary to have a review of the different ANN approaches inspired in attention to show both academic and industry experts the available models for a wide variety of applications. Based on the PRISMA methodology, we present a systematic review of the literature published since 2000, in which deep learning algorithms are applied to diverse problems related to speech processing. In this paper 133 research works are selected and the following aspects are described: (i) Most relevant features, (ii) ways in which attention has been implemented, (iii) their hypothetical relationship with human attention, and (iv) the evaluation metrics used. Additionally, the four publications most related with human attention were analyzed and their strengths and weaknesses were determined.

Keywords: artificial neural networks; deep learning; attention; speech; systematic review

Citation: Zacarias-Morales, N.; Pancardo, P.; Hernández-Nolasco, J.A.; Garcia-Constantino, M. Attention-Inspired Artificial Neural Networks for Speech Processing: A Systematic Review. *Symmetry* **2021**, *13*, 214. https://doi.org/10.3390/sym13020214

Received: 30 December 2020
Accepted: 22 January 2021
Published: 28 January 2021

1. Introduction

The analysis and processing of signals generated by the human speech consists in identifying and quantifying some physical features from the signals in such a way that they can be used for different speech related applications like identification, recognition and authentication. In that sense, Artificial Neural Networks (ANNs) have been a valuable computational tool because of their effectiveness in speech processing. Using deep learning algorithms, ANNs try to mimic the behaviour of the human brain to perform the functionalities involved in speech processing and, to improve the results, some algorithms implement some type of attention.

Given the above, it is of interest to know the diverse research works published between 2000 and 2020 that use ANNs and that implement attention for speech processing. While there are some systematic reviews related to speech processing using Artificial Intelligence techniques, to our best knowledge are no systematic reviews focused on attention such as the one presented in this paper.

Therefore, the literature search for this review was conducted on the ACM Digital Library, IEEE Explorer, Science Direct, Springer Link, and Web of Science databases to identify studies in the field of speech processing that reported the use of ANNs with some type of attention included in the title and/or abstract. We present a comprehensive

ISSN 1870-4069

Red neuronal convolucional ramificada con atención para la mejora de voz

Noel Zacarias-Morales, José Adán Hernández-Nolasco, Pablo Pancardo

Universidad Juárez Autónoma de Tabasco,
División Académica de Ciencias y Tecnologías de la Información,
México

{adan.hernandez,pablo.pancardo}@ujat.mx
201H18002@alumno.ujat.mx

Resumen. La mejora de la voz es un proceso que implica eliminar o atenuar el ruido presente en una señal de voz. En este sentido, muchos autores han empleado las redes neuronales para extraer la voz de manera inteligible y de calidad. A diferencia de los artículos que se revisaron, este trabajo propone una red convolucional ramificada con atención de múltiples encabezados que hace posible reducir el número de parámetros entrenables, así como incrementar la precisión en la extracción de la voz. Los resultados obtenidos demostraron que la incorporación del mecanismo de atención basado en múltiples encabezados mejoró en términos generales la capacidad del modelo convolucional ramificado, conforme a los valores de las métricas de calidad PESQ, STOI y SI-SDR. Los valores logrados confirman que incorporar un mecanismo de atención permite la mejora de los modelos de redes neuronales convolucionales ramificadas para extraer voz inteligible y de calidad.

Palabras clave: Red neuronal, convolución, atención, voz, ruido.

Branched Convolutional Neural Network with Attention for Voice Enhancement

Abstract. Speech enhancement is a process that involves removing or attenuating noise present in a speech signal. In this sense, many authors have used neural networks to extract the voice in an intelligible and quality way. Unlike the articles that were reviewed, this work proposes a branched convolutional network with attention to multiple headers that makes it possible to reduce the number of trainable parameters, as well as increase the precision in voice extraction. The results obtained showed that the incorporation of the attention mechanism based on multiple headers generally improved the capacity of the branched convolutional model, according to the values of the PESQ, STOI and SI-SDR quality

Applied Intelligence
https://doi.org/10.1007/s10489-023-04571-y

Full single-type deep learning models with multihead attention for speech enhancement

Noel Zacarias-Morales[1] · José Adán Hernández-Nolasco[1] · Pablo Pancardo[1]

Accepted: 11 March 2023

Abstract

Artificial neural network (ANN) models with attention mechanisms for eliminating noise in audio signals, called speech enhancement models, have proven effective. However, their architectures become complex, deep, and demanding in terms of computational resources when trying to achieve higher levels of efficiency. Given this situation, we selected and evaluated simple and less resource-demanding models and utilized the same training parameters and performance metrics to conduct a fair comparison among the four selected models. Our purpose was to demonstrate that simple neural network models with multihead attention are efficient when implemented on computational devices with conventional resources since they provide results that are competitive with those of hybrid, complex and resource-demanding models. We experimentally evaluated the efficiency of multilayer perceptron (MLP), one-dimensional and two-dimensional convolutional neural network (CNN), and gated recurrent unit (GRU) deep learning models with and without multiheaded attention. We also analyzed the generalization capability of each model. The results showed that although these architectures were composed of only one type of ANN, multihead attention increased the efficiency of the speech enhancement process, yielding results that were competitive with those of complex models. Therefore, this study is helpful as a reference for building simple and efficient single-type ANN models with attention.

Keywords Artificial neural network · Attention · Deep learning models · Speech enhancement

1 Introduction

Attention involves selectively focusing on specific elements while ignoring less relevant or important elements according to a goal. The notion of selective attention in humans is related to senses such as sight, hearing and smell. Auditory attention is a selection process or set of processes that focuses sensory and cognitive resources on the most relevant events in the sound landscape [5]. Therefore, attention has allowed humans to focus their interest on only a fraction of the available information, making optimal use of resources to better achieve their goals [8].

Attention is a mechanism that is being increasingly used in a wide range of deep learning architectures. The mechanism itself is available in various formats [3]. The use of attention mechanisms has become more relevant due to the publication of the proposed transformer architecture based mainly on an attention mechanism, operating even with recurrence and convolutions [26]. The authors highlighted its high degree of parallelization, thus increasing its computational speed.

An increasing number of architectures are using attention mechanisms and are being applied to natural language problems [3] and different types of data, especially sequential data. Examples include the improvement of biomedical image segmentation [24], speech recognition [2, 33], and speech enhancement [11, 32].

Among all the different attention mechanisms in the literature [1, 15], we selected multihead attention because it is a mechanism that can be efficiently parallelized and

José Adán Hernández-Nolasco and Pablo Pancardo are contributed equally to this work.

✉ José Adán Hernández-Nolasco
adan.hernandez@ujat.mx

Noel Zacarias-Morales
201h18002@alumno.ujat.mx

Pablo Pancardo
pablo.pancardo@ujat.mx

[1] Academic Division of Sciences and Information Technology, Juarez Autonomous University of Tabasco, Cunduacan, 86690, Tabasco, Mexico

Published online: 15 April 2023

ARTÍCULO ACEPTADO

Redes Neuronales Convolucionales con Atención para Atenuar Ruido en Señales de Audio

Noel Zacarias-Morales, José Adán Hernández-Nolasco, Pablo Pancardo y J. Arturo Olvera-López

Introducción

Los humanos tenemos la capacidad de entender los mensajes cuando estamos platicando con otra persona a pesar de que se encuentren presentes varias fuentes de sonido adicionales (motores funcionando, música, aves, etcétera), las cuales representan ruido para la comunicación. Por lo general, en un escenario donde debemos poner atención a una voz en particular, mitigar el ruido no representa un problema para nosotros. Esto es debido a que desde nuestro nacimiento continuamente mejoramos nuestra capacidad de atención selectiva [1]; esta capacidad nos permite seleccionar de forma natural una fuente de sonido específica (como una voz) y atenuar los ruidos que puedan estar presentes para comprender lo que nos están diciendo.

Desafortunadamente para los sistemas computacionales de procesamiento de audio, realizar esta tarea requiere de la integración de algoritmos que les brinden la capacidad para atenuar o eliminar diversas fuentes de audio que representan ruidos en la señal de voz. Uno de los casos que los investigadores encuentran más desafiante es cuando el audio a analizar es capturado en un ambiente complejo, es decir, cuando el tipo o ubicación de sonidos varía continuamente. Conversaciones reales en nuestro día a día representan el mejor ejemplo.

Con la finalidad de encontrar una solución a la problemática es común que diversas áreas de la ciencia (neurociencia, robótica o inteligencia artificial) busquen inspiración en las ciencias biológicas y cognitivas; una gran cantidad de los enfoques inspirados se concentran en el modelado de áreas cerebrales y fenómenos cognitivos como la memoria, la percepción, el lenguaje, la categorización y el aprendizaje utilizando métodos de procesamiento de información [2].

Dotar a los modelos computacionales de procesamiento de audio con la capacidad humana de selección y atención de una fuente de sonido específica, como la voz, podría permitir que los modelos computacionales puedan lidiar con las características variables de un ambiente complejo y que puedan ser aplicados a sistemas de procesamiento de audio para mejorar la capacidad de atención de asistentes virtuales, para detectar violencia verbal en una ubicación, o por cualquier otro sistema computacional que requiera utilizar la voz de una persona pero con la menor cantidad de ruidos, por decir algunos ejemplos (ver figura 1).

Recientemente, se ha avanzado en la resolución de problemas de atenuación o eliminación de señales de ruido en mezclas de audio con voz en escenarios cada vez más difíciles, gracias a los métodos de aprendizaje profundo (las redes neuronales artificiales) [3]. Un tipo de red neuronal artificial muy utilizado en este problema es la red neuronal convolucional (CNN, por sus siglas en inglés), que tienen la capacidad de capturar patrones en las señales de audio. Se ha reportado que las redes neuronales convolucionales son más eficaces que las redes neuronales artificiales de tipo perceptrón multicapa [4] y más eficientes que las redes neuronales artificiales recurrentes [5] cuando se aplican al procesamiento de señales de audio.

Redes neuronales convolucionales

El diseño de las redes neuronales artificiales se hizo tomando como referencia el comportamiento y estructura de las neuronas en el cerebro humano, de modo que se construye una red de nodos interconectados y organizados en capas.

Los modelos computacionales con capacidad de atención podrían procesar mejor los ambientes sonoros con características más complejas.

Las redes neuronales convolucionales son una subcategoría de redes neuronales artificiales y son llamadas así porque aplican una operación matemática, llamada convolución, que transforma a las capas en convolucionales. El objetivo de una capa convolucional en una red neuronal artificial es extraer características particulares a partir de los datos de entrada en forma de matrices o vectores, comprimiéndolos para reducir su tamaño inicial. Los datos de entrada proporcionados pasan a través de un conjunto de filtros, creando nuevos datos llamados características extraídas. Finalmente, los nuevos datos se concatenan en una matriz, o vector de características, que serían los datos de entrada para la siguiente capa de la red neuronal.

ISSN 2007-0691

Anexo C

Ponencias Derivadas de la Investigación

Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P. (2022). Red Neuronal Convolucional Ramificada con Atención para la Mejora de Voz. XIV Congreso Mexicano de Inteligencia Artificial COMIA 2022.

Zacarias-Morales, N., Hernández-Nolasco, J. A., Pancardo, P. (2022). Impacto de la Atención en Redes Neuronales Convolucionales y Recurrentes en la Mejora de Voz. Congreso Internacional de Ciencias y Tecnologias de la Informacion CYTI-ConXn 2022.

Noel Zacarias Morales.pdf

Universidad Juárez Autónoma de Tabasco

Detalles del documento

Identificador de la entrega
trn:oid:::3117:582713510

Fecha de entrega
24 abr 2026, 1:56 p.m. GMT-6

Fecha de descarga
24 abr 2026, 3:29 p.m. GMT-6

Nombre del archivo
Noel Zacarias Morales.pdf

Tamaño del archivo
11.9 MB

115 páginas

23.030 palabras

156.079 caracteres

18% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Texto citado
- Coincidencias menores (menos de 10 palabras)
- Abstract

Fuentes principales

18% Fuentes de Internet
3% Publicaciones
0% Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alerta de integridad para revisión

Caracteres reemplazados
136 caracteres sospechosos en N.º de páginas
Las letras son intercambiadas por caracteres similares de otro alfabeto.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Fuentes principales

18% Fuentes de Internet
3% Publicaciones
0% Trabajos entregados (trabajos del estudiante)

Fuentes principales

Las fuentes con el mayor número de coincidencias dentro de la entrega. Las fuentes superpuestas no se mostrarán.

1 Internet
rcs.cic.ipn.mx 5%

2 Internet
smia.mx 4%

3 Internet
idoc.pub <1%

4 Internet
dspace.utalca.cl <1%

5 Internet
calebrascon.info <1%

6 Internet
www.coursehero.com <1%

7 Publicación
Noel Zacarias-Morales, José Adán Hernández-Nolasco, Pablo Pancardo. "Full singl... <1%

8 Internet
hdl.handle.net <1%

9 Internet
archivos.ujat.mx <1%

10 Trabajos entregados
Universidad Juárez Autónoma de Tabasco on 2025-09-11 <1%

11 Internet
coggle.it <1%

12 Internet
runebook.dev <1%

13 Internet
nisamarymati.wordpress.com <1%

14 Internet
ri.uaemex.mx <1%

15 Internet
www.colibri.udelar.edu.uy <1%

16 Internet
export.arxiv.org <1%

17 Internet
www.fcfm.buap.mx <1%

18 Publicación
Rafii, Zafar. "Source Separation by Repetition.", Proquest, 2014. <1%

19 Internet
e-archivo.uc3m.es <1%

20 Internet
etheses.whiterose.ac.uk <1%

21 Internet
sedici.unlp.edu.ar <1%

22 Internet
www.saber.ula.ve <1%

23 Internet
researchr.org <1%

24 Internet
www.buenastareas.com <1%

25 Internet
people.idiap.ch <1%

26 Internet
repositorio.pucese.edu.ec <1%

27 Internet
ddd.uab.cat <1%

28 Internet
www.cognifit.com <1%

29 Internet
eprints.ucm.es <1%

30 Internet
qu4nt.github.io <1%

31 Internet
uvadoc.uva.es <1%

32 Internet
www.crimsonbird.org <1%

33 Internet
repositorioinstitucional.buap.mx <1%

34 Internet
bibliotecadigital.udea.edu.co <1%

35 Internet
www.clubensayos.com <1%

36 Internet
docplayer.es <1%

37 Internet
riunet.upv.es <1%

38 Internet
repositorio.uam.es <1%

39 Internet
research-repository.griffith.edu.au <1%

40 Internet
arxiv.org <1%

41 Internet
bdigital.unal.edu.co <1%

42 Internet
manglar.uninorte.edu.co <1%

43 Internet
vbn.aau.dk <1%

44 Internet
web10.unl.edu.ar:8080 <1%

45 Internet
fr.scribd.com <1%

46 Internet
henrywells.blogspot.com <1%

47 Internet
repositorio.chapingo.edu.mx <1%

48 Internet
revistas.anahuac.mx <1%

49 Internet
tr-ex.me <1%

50 Internet
www.scss.tcd.ie <1%

51 Internet
www3.microstrategy.com <1%

52 Internet
ricaxcan.uaz.edu.mx <1%

53 Internet
www.camu.com.pe <1%

54 Internet
www.nti-audio.com <1%

55 Internet
www.nxtbook.com <1%

56 Publicación
"Modelamiento de predictor de fuga de clientes, con la utilización del valor de vid... <1%

57 Publicación
Calli Olvea, Javier. "Deep learning para la detección e identificación de estudiante... <1%

58 Publicación
Sosa Maydana, Carlos Boris. "Modelo supervisado para la clasificación de manusc... <1%

59 Internet
link.springer.com <1%

60 Internet
search.bvsalud.org <1%

61 Internet
www.hear-it.org <1%

62 Internet
repositorio.usm.cl <1%

63 Internet
repositorioacademico.upc.edu.pe <1%

64 Internet
revhipertension.com <1%

65 Internet
www.archivos.ujat.mx <1%

66 Internet
www.xevt.com <1%