UNIVERSIDAD AUTÓNOMA DE ZACATECAS

"Francisco García Salinas"

"Esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con fines forenses"

Tesis para obtener el grado de:

Maestro en Ciencias del Procesamiento de la Información

Presenta

Armando Rodarte Rodríguez

Director:

Dr. José Ismael de la Rosa Vargas

Co-Directores:

Dr. Aldonso Becerra Sánchez

Dr. José de Jesús Villa Hernández

Asesores:

Dr. Efrén González Ramírez

Dr. Gamaliel Moreno Chávez

Zacatecas, Zac., 12 de octubre de 2023

Zacatecas, Zac., 26 de octubre de 2023.

C. Armando Rodarte Rodríguez
Estudiante de la MCPI
PRESENTE

Dr. Huizilopoztli Luna García
Responsable de la MCPI

Nos es grato comunicarle que después de haber sometido a revisión académica la propuesta de Tesis titulada **"Esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con fines forenses"**, presentada por el estudiante **Ing. Armando Rodarte Rodríguez** y habiendo efectuado todas las correcciones indicadas por este Comité Tutorial, se **AUTORIZA** el documento de tesis para su impresión.

Sin más por el momento reciban un cordial saludo.

COMITÉ TUTORIAL
PROCESAMIENTO Y ANÁLISIS DE DATOS

Dr. José Ismael de la Rosa Vargas

Dr. Aldonso Becerra Sánchez

Dr. José de Jesús Villa Hernández

Dr. Efrén González Ramírez

Dr. Gamaliel Moreno Chávez

c.c.p. Interesado.

c.c.p. Responsable de la Maestría en Ciencias del Procesamiento de la Información.

Carta de similitud núm. 459/ IyP
Zacatecas, Zacatecas 17/octubre/2023

Dr. Huizilopoztli Luna García
Responsable de la MCPI – UAZ
Presente

Estimado Dr. Huizilopoztli,

Después de saludarlo, sirva el presente oficio para notificar que el documento

" Esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con fines forenses"
Armando Rodarte Rodríguez

Fue analizado con el software Copyleaks, con la intención de detectar similitudes; el resultado en cuestión fue

3.3 % de similitud

De acuerdo a lo anterior, el porcentaje se considera **ACEPTABLE** de acuerdo a los estándares internacionales.

Atentamente
"Somos Arte, Ciencia y Desarrollo Cultural"

Dr. Carlos Francisco Bautista Capetillo
Coordinador de Investigación y Posgrado
Universidad Autónoma de Zacatecas

Zacatecas, Zac., 12 de octubre de 2023

Carta de cesión de derechos

A QUIEN CORRESPONDA

El que suscribe, **C. Armando Rodarte Rodríguez**, alumno del **Programa de Maestría en Ciencias del Procesamiento de la Información**, con número de matrícula **35161767** y afiliado a la Unidad Académica de Ingeniería Eléctrica de la Universidad Autónoma de Zacatecas, manifiesta ser el autor intelectual del presente trabajo de tesis. Bajo la dirección del Dr. José Ismael de la Rosa Vargas, cede los derechos del trabajo titulado **"Esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con fines forenses"** a la Universidad Autónoma de Zacatecas para su difusión con fines académicos e investigativos.

ATENTAMENTE

Armando Rodarte Rodríguez

I.S. Armando Rodarte Rodríguez

Agradecimiento especial

Agradezco profundamente al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el financiamiento otorgado para la obtención de mi grado académico de Maestro en Ciencias del Procesamiento de la Información, a través de la convocatoria de Becas Nacionales (Tradicional) 2021-2023-2.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mis directores de tesis, el **Dr. Aldonso Becerra Sánchez** y el **Dr. Ismael de la Rosa Vargas**, por compartir sus consejos, atenciones, ideas y conocimientos, los cuales fueron fundamentales para llevar a cabo este proyecto de investigación y para mi desarrollo profesional. Asimismo, su paciencia, tiempo, comprensión y apoyo anímico me permitieron afrontar las diversas dificultades.

Gracias a mis **queridos padres** y **amigos**, quienes me brindaron su cariño sincero, sus cuidados y hospitalidad, su apoyo incondicional y sus palabras de aliento que me animaron cada día.

También agradezco a mis asesores el **Dr. Efrén Gonzales Ramírez**, **Dr. José de Jesús Villa Hernández** y **Dr. Gamaliel Moreno Chávez**, quienes dieron su retroalimentación y consejos para mejorar la calidad de este trabajo de investigación.

Reconozco que cualquier logro o cosa que valga la pena en esta vida es el resultado del esfuerzo, contribución y del apoyo colectivo, no del individualismo. Por eso, quiero agradecer de corazón a todas las personas que siempre me han apoyado, pues este pequeño logro también les pertenece.

Dedicatoria

A mis padres, **Mateo Rodarte Trueba** y **Teresita de Jesús Rodríguez Gómez**, quienes han sido mi inspiración y mis superhéroes. Su amor, confianza, comprensión y aliento construyó el camino hacia mis sueños más grandes.

A mis estimados amigos **Abraham Vázquez de la Torre**, **Daniel Román Santos** y **Oscar Ulloa Flores**. Siempre han estado para mí, brindándome refugio, consejo, consuelo, apoyo, buenos momentos, lealtad y honestidad.

También dedico este trabajo al **Dr. Aldonso Becerra Sánchez** y al **Dr. Ismael de la Rosa Vargas**, por su apoyo incondicional y por inspirarme a ser científico.

Dedico este proyecto a mis tiernos amigos peludos de cuatro patas, **Zuky**, **Kamala** y **Gurrumina**. Han sido mis compañeros en los momentos de alegría, tristeza, ansiedad y estrés. Sus ronroneos, ladridos y locuras siempre me han consolado, acompañado en mis aventuras y hecho pasar momentos de diversión.

Finalmente, a mi querido hermano mayor **Ángel Rodarte Rodríguez**.

Resumen

La biometría es una herramienta que permite identificar y autenticar personas por medio de rasgos biológicos que son irrepetibles en cada individuo. Esta herramienta ha permitido el desarrollo de aplicaciones de software y algoritmos inteligentes de procesamiento de voz en áreas como el análisis de información forense. Donde el objetivo de este campo de análisis es realizar la identificación de personas con fines de vigilancia y forenses. Sin embargo, los sistemas de procesamiento de voz aplicados en ambas áreas son poco confiables y precisos para analizar audios de baja calidad y con ruido ambiental. Por lo tanto, es necesario desarrollar nuevos modelos que sean más robustos en el procesamiento de este tipo de información para llevar a cabo tareas de identificación del hablante en escenarios criminales. A partir de la problemática mencionada, el objetivo de esta investigación es desarrollar un esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con propósitos forenses. Proporcionar este tipo de análisis de manera confiable servirá como apoyo adicional para reducir la cantidad de sentencias criminales incorrectas que son emitidas por el criminalista, juez y/o jurado en escenarios forenses. El esquema de red neuronal artificial propuesto utiliza funciones de activación paramétricas y neuronas estándar como unidades de soporte para la optimización de los parámetros entrenables de las funciones de activación. A este modelo de red neuronal se le ha llamado red neuronal artificial con neuronas de soporte. En relación con las funciones de activación paramétricas, se desarrollaron dos funciones con parámetros entrenables, AReLU y MPReLU, que son versiones simplificadas de DPReLU. Adicionalmente, en este estudio se implementaron diferentes configuraciones de redes neuronales artificiales con la finalidad de comparar el rendimiento del esquema propuesto contra la arquitectura de una red neuronal convencional. Para interactuar con el modelo óptimo presentado, se desarrolló la aplicación de escritorio HAAF. Por otra parte, se usó un conjunto de datos con 158 hablantes nativos del idioma español (122 hombres y 36 mujeres) para entrenar y evaluar el rendimiento de los diferentes experimentos. Estas grabaciones incluyen diversos tipos de calidad y ruidos ambientales. La configuración que demostró el mejor rendimiento, en las actividades de identificación de locutores y en audios con ruido ambiental, fue el modelo de red neuronal artificial con neuronas de soporte y el uso de la función de activación MPReLU. Este modelo alcanzó una exactitud del 98.68% y un puntaje F1 del 98.28%. Por último, los resultados obtenidos revelan que las neuronas de soporte son una unidad de procesamiento efectiva para optimizar de manera automática parámetros internos de las redes neuronales artificiales. También, el uso de funciones paramétricas puede ayudar a realizar un modelado más acorde al comportamiento de los datos, añadiendo robustez al modelado de información con ruido.

Palabras clave: identificación de locutor, redes neuronales artificiales, neuronas de soporte, funciones de activación paramétricas, procesamiento de voz.

Abstract

Biometrics is a tool that allows to identify and authenticate people through biological characteristics that are unique to each individual. This tool has allowed the development of software applications and intelligent voice processing algorithms in fields such as forensic information analysis. Where the goal of this field of analysis is to identify people for surveillance and forensic purposes. However, the voice processing systems applied in both areas are not very reliable or accurate for analyzing low-quality audio and with environmental noise. Therefore, it is necessary to develop new models more robust in processing this type of information to carry out speaker identification tasks in criminal scenarios. Based on the problem mentioned above, the objective of this research is to develop an artificial neural network scheme for speaker identification tasks in noisy environments and for forensic purposes. Providing this type of analysis reliably will serve as additional support to reduce the issuance of incorrect criminal sentences issued by the criminalist, judge, and/or jury in forensic scenarios. The proposed artificial neural network scheme uses parametric activation functions and standard neurons as support units for the optimization of trainable parameters in the parametric activation functions. This proposed neural network model has been called an artificial neural network with support neurons. In relation to parametric activation functions, two parametric functions were developed: AReLU and MPReLU, which are simplified versions of DPReLU. In addition, in this study, different configurations of artificial neural networks were implemented in order to compare the performance of the proposed scheme against the architecture of a conventional neural network. To interact with the proposed optimal model, the HAAF desktop application was developed. On the other hand, a dataset with 158 native Spanish speakers (122 men and 36 women) was used to train and evaluate the performance of the different conducted experiments. These recordings include various types of quality and environmental noise. The configuration that showed the best performance, in speaker identification tasks and in audio with ambient noise, was the artificial neural network model with support neurons and the use of the MPReLU activation function. This model achieved an accuracy of 98.68% and an F1 score of 98.28%. Finally, the results obtained reveal that support neurons are an effective processing unit for automatically optimizing internal parameters of artificial neural networks. Also, the use of parametric functions can help to perform modeling that is more in line with the behavior of the data, and it improves the robustness of the information modeling with noise.

Keywords: speaker identification, artificial neural networks, support neurons, parametric activation functions, voice processing.

Índice General

Índice de Figuras

Índice de Tablas

Capítulo 1. Introducción

Este primer capítulo introduce conceptos esenciales sobre procesamiento de voz e inteligencia artificial. Asimismo, se presenta el contexto general de esta investigación, que incluye la descripción de la motivación, la problemática y el propósito de la investigación. Finalmente, a partir del contexto general, se establecen los antecedentes, las preguntas de investigación, los objetivos, la justificación, la hipótesis y el alcance del proyecto.

1.1. Antecedentes

En los últimos años ha existido un incremento en el interés por las áreas de la seguridad y la vigilancia, lo que ha provocado el desarrollo de nuevas tecnologías (sensores, apps y algoritmos) de autenticación y espionaje basadas en herramientas como la biometría. Estas técnicas biométricas tienen un gran impacto en diversas disciplinas, ya que permite realizar actividades de identificación y autenticación de personas de forma fiable y rápida [1], [2], [3]. En otras palabras, mediante rasgos biológicos que son irrepetibles, es posible diferenciar de manera precisa y automatizada a un individuo de otro. Por ejemplo, la autentificación e identificación de personas, en escenarios de seguridad o vigilancia, se realiza a través del análisis y reconocimiento automatizado de características biológicas únicas como el rosto, la retina y el iris, la huella dactilar y la voz [1]. Los sensores y los algoritmos pueden realizar un proceso de lectura para capturar las características biométricas mencionadas, que se dividen en dos categorías: físicas y de comportamiento [2], [3].

Como se mencionó, la voz es un rasgo biológico único que es muy útil porque brinda un alto nivel de seguridad, precisión y confiabilidad en actividades de identificación y autenticación. La vibración de las cuerdas vocales y las variaciones de presión de aire producidas por el tracto vocal producen estas ondas acústicas (voz) [1], [4]. Las diferentes formas y tamaños de los órganos de fonación, como la glotis y la laringe, determinan las características particulares de las señales de voz en cada individuo [1], [4]. Estos sonidos son producidos con la intención de interactuar y comunicar información como conceptos, necesidades emocionales, indicaciones, ideas, opiniones, etc. Además, estas señales acústicas transmiten una variedad de características, información y rasgos de la persona como su estado de salud física y mental, la edad, el género, su estado emocional y grupo étnico [1], [4], [5].

El procesamiento de voz es la disciplina que estudia las señales acústicas y las técnicas necesarias para su análisis, procesamiento y modelado. Es un campo diverso de investigación con muchas aplicaciones en áreas como la robótica, interacción humano-maquina, la educación y la medicina [6]–[9]. También, esta disciplina engloba distintas subdisciplinas de investigación, como el reconocimiento de voz, la verificación e identificación del locutor, identificación de emociones y reconocimiento de eventos [6], [8], [10]. El reconocimiento de voz consiste en identificar cuáles palabras fueron pronunciadas, mientras que la identificación de locutor es el proceso de determinar la identidad de quién está hablando [1], [10]. Ambos procesos realizan una fase de extracción de información (características acústicas) de la identidad de la persona y, conservan el contenido relevante y discriminante de la señal acústica para su posterior modelado [1], [11]. En este documento, usaremos en algunos casos el término "identificación de voz" para referirnos al proceso de identificar la persona que está hablando, y el término "reconocimiento de voz" para referirnos a la tarea de identificar las palabras pronunciadas.

Por otra parte, las principales fases involucradas en la identificación de locutores son el pre-procesamiento, la extracción de características y el modelado [1], [12], [13]. En cuanto al reconocimiento de voz, las fases comprenden la extracción de características, la creación del modelo acústico, la clasificación de patrones, la generación del modelo de lenguaje, la creación del diccionario de pronunciación y la decodificación [12], [13]. Para el modelado de la señal de voz en ambas tareas, se utilizan métodos de reconocimiento de patrones, procesamiento de señales, Inteligencia Artificial (IA) y métodos estocásticos. Algunos clasificadores comúnmente usados para modelar las señales de voz son Modelos Ocultos de Markov (MOM), Naive Bayes, GMM (o modelos de mezcla Gaussiana), Redes Neuronales Artificiales (RNA), SVM (o máquina de soporte vectorial), K-vecinos más cercanos (KNN, por sus siglas en inglés), entre otras técnicas [11], [12], [14]–[16]. Como técnicas de extracción de características acústicas, se suelen usar técnicas tales como los coeficientes cepstrales de frecuencia Mel (MFCC, por sus siglas en inglés), LPC (Linear Predictive Coding) y LPCC (o coeficientes cepstrales de predicción lineal). [1], [12], [17]–[19].

Con relación a lo anterior, la fase de procesamiento modifica la señal de voz haciéndola adecuada para su análisis y modelado. Por esta razón, el pre-procesamiento es un paso crucial y crítico, ya que un pre-procesamiento inadecuado disminuirá el rendimiento de clasificación [1], [13]. El procesamiento de señales acústicas muchas veces implica un subproceso de extracción de características acústicas. Este proceso implica dividir la señal en muestras o fragmentos, y luego extraer una secuencia de características para cada muestra. El objetivo de la extracción de características es obtener información útil y eliminar información redundante e irrelevante. De esta manera, se obtiene información representativa y discriminativa [1], [13]. Las muestras o tramas acústicas obtenidas en la fase de procesamiento son la fuente de información para la fase final (modelado). Finalmente, la idea del modelado es aprender a distinguir las muestras de voz de entre los diferentes locutores o fonemas aplicando técnicas de clasificación [20], [21].

El reconocimiento del locutor es una disciplina que permite identificar a las personas por su voz. Este tiene diversas aplicaciones, como la autenticación de

personas, la asistencia de voz por inteligencia artificial, la vigilancia y el análisis forense. El análisis forense de audio consiste en adquirir, analizar y evaluar grabaciones de audio usadas como evidencia criminal, con la intención de llevar a cabo tareas como la verificación de integridad, la mejora del audio forense, la identificación del sospechoso y el reconocimiento de las palabras pronunciadas. [22]. Este tipo de análisis se utiliza comúnmente para determinar si una expresión ha sido pronunciada por un sospechoso en particular y para determinar la participación de una persona en un hecho criminal [1].

Revisando la literatura científica, se puede ver que se han llevado a cabo diversas investigaciones relacionadas con la identificación del hablante en escenarios forenses y en otros contextos aplicados. Por ejemplo, Manurung et al. [23] diseñaron un prototipo de sistema para reconocimiento de locutor (sospechoso) con fines forenses. El proceso de reconocimiento del orador consistió en extraer muestras de voz representadas por características acústicas, estas muestras fueron extraídas de las grabaciones de audio utilizadas como evidencia criminal. Adicionalmente, en esta primera fase se implementó el método de MFCC para obtener la información relevante de las señales de voz. Como paso final, las muestras se clasificaron utilizando el modelo de red neuronal de cuantificación vectorial de aprendizaje (JST-LVQ, por sus siglas en inglés), el cual logró una exactitud de clasificación del 73.33%.

En el estudio realizado por S. Singh y Singh [24], se examinaron diferentes técnicas de agrupamiento (claustering) y de algoritmos genéticos para encontrar similitudes entre las pronunciaciones en la actividad de identificación de hablantes. Se implementaron los modelos K-medoid, Fuzzy C-means, Gustafson y Kessel, GMM y Gath-Geva clustering con el objetivo principal de agrupar las características más representativas generadas con el método de MFCC. Este trabajo se aplicó en el reconocimiento forense de hablantes, sin importar el idioma.

En Ye y Yang [25] exploraron diferentes esquemas de redes neuronales profundas (DNN, por sus siglas en inglés) para la identificación de hablantes. El modelo propuesto, denominado "GRU profundo", logró la mayor exactitud de reconocimiento con un 98.96%. El clasificador utiliza una CNN (o red neuronal convolucional) $2D$ combinada con una red neuronal recurrente (RNN, por sus siglas en inglés) de varias capas GRU (o unidades recurrentes con compuertas). Durante el procesamiento de los datos, se extrajo el espectrograma de cada grabación y se utilizó para entrenar los modelos profundos. También, mediante capas convolucionales, se extrajeron características relevantes de cada espectrograma. Los diferentes experimentos propuestos se llevaron a cabo utilizando el conjunto de datos Aishell-1.

Por otro lado, Liu et al. [26] desarrollaron un método híbrido que combinó CNN y GMM para el reconocimiento de hablantes en expresiones cortas y basado en el análisis de espectrogramas. Este nuevo enfoque permite representar mejor a cada hablante y logra una distinción más efectiva entre ellos, lo que resultó en una reducción significativa de la tasa de error de reconocimiento del 4.9% al 2.5%. En este estudio, se implementó un corpus de datos compuesto por 50 personas. Como parte del procesamiento, se extrajo el espectrograma (imagen) de cada registro para su posterior análisis. Los investigadores Simić et al. [27] utilizaron redes neuronales convolucionales restringidas para el reconocimiento de hablantes en diferentes

estados emocionales. Se utilizó un conjunto cerrado de hablantes (corpus de datos SEAC) que contiene cinco emociones diferentes registradas en el idioma serbio. El modelo propuesto logró una precisión promedio del 99.24% para el habla neutra, un 79.84% para la tristeza y alrededor del 85% para el resto de las emociones.

En relación al reconocimiento de voz, se encuentra el trabajo propuesto por los investigadores Pop y Burileanu [28]. En este estudio, se propone un método para mejorar la calidad del habla mediante el análisis de características acústicas, usando los MFCCs. Los investigadores implementaron una RNA para evaluar su nuevo método, la cual clasificó grabaciones de audio usadas como evidencia criminal. Se utilizaron dos conjuntos de datos, SSC-eval y NSDTSEA, donde obtuvieron tasas de error de palabra (WER, por sus siglas en inglés) del 20.34% y 8.21%, respectivamente. Finalmente, Ravanelli y Bengio [29] llevaron a cabo un reconocimiento de locutores utilizando datos sin procesar con el modelo SincNet. Este enfoque se basa en funciones sinc parametrizadas e implementa filtros de pasa banda. Por último, se observó que este modelo converge más rápido que una CNN estándar.

Como se puede ver, las actividades de reconocimiento de voz e identificación de locutor se han realizado comúnmente de forma independiente. Sin embargo, en trabajos recientes, los investigadores se han centrado en diseñar modelos que puedan realizar ambas tareas de forma simultánea. Esto se hace con la intención de simular la capacidad del cerebro humano para realizar tareas múltiples [30], [31]. La idea es unificar diferentes tareas de procesamiento de voz en un solo modelo, por ejemplo, unir las actividades de identificación de locutor, reconocimiento de voz, reconocimiento de idioma e identificación de emociones. Esto puede mejorar el rendimiento de clasificación, el tiempo entrenamiento y ejecución, reducir la complejidad de los modelos y disminuir el costo computacional. En la literatura, se pueden encontrar diferentes tipos de arquitecturas de esquemas multitarea para realizar de manera conjunta distintas tareas de procesamiento de voz [30], [32].

Para mejorar el desempeño en los diferentes modelos empleados en el procesamiento de voz, algunos investigadores se han centrado en diseñar nuevas funciones de costo como Becerra et al. [33], crear nuevos esquemas de redes neuronales [34], proponer algoritmos híbridos y diseñar modelos multitarea [30]. También los investigadores se han centrado en simular más fielmente las estructuras neurobiológicas. Esto ha llevado al desarrollo de nuevas áreas de investigación como la neuroevolución y la computación neuromórfica [35], [36].

Con respecto a las nuevas arquitecturas de RNA, una nueva variante bioinspirada son las Redes Neuronales Artificiales-Glía (RNA-Glía). Estas redes se inspiran en cómo las neuronas y las células gliales procesan la información en el cerebro, interactuando entre sí [37]. Las células gliales desempeñan un papel de apoyo para las neuronas; existen cuatro tipos: astrocitos, oligodendrocitos, microglía y Schwann. Los astrocitos son las células gliales más abundantes en el sistema nervioso y trabajan de manera conjunta con las neuronas para procesar la información. La interacción entre astrocitos y neuronas se lleva a cabo mediante una comunicación bidireccional llamada sinapsis tripartita. Las RNA-Glía han intentado emular esta interacción entre astrocitos y neuronas [38], [39]. Las redes artificiales-Glía están diseñadas para regular el potencial de acción de las RNA estándar, lo que les permite ser más robustas y

precisas que los enfoques tradicionales de aprendizaje automático [30].

La motivación del presente proyecto de investigación se origina a partir de tres problemáticas. En primer lugar, los modelos actuales de identificación de locutor no son precisos o confiables, específicamente en términos de exactitud de clasificación. En segundo lugar, estos clasificadores no se pueden validar para ser aplicados a escenarios reales y críticos porque tienen una baja presión. Es decir, que no son lo suficientemente confiables como para ser utilizados en contextos profesionales y de la vida cotidiana [33]. La tercera problemática surge de la dificultad que existe en examinar evidencia criminal (grabaciones de voz) de mala calidad. Estos problemas de análisis pueden conducir a análisis incorrectos y errores en las tareas de identificación forense del hablante o en la distinción de la voz del sospechoso en contextos de vigilancia. Como resultado, el juez, el jurado y el criminalista pueden tomar decisiones poco imparciales e injustas [40]. Por lo cual se requiere de un modelo de reconocimiento de locutor que reconozca con alta eficacia qué expresiones fueron dichas por un sospechoso en particular, o quién pronunció cierta expresión a partir de grabaciones de audio (evidencia criminal) con ruido.

Este trabajo se enfoca en el reconocimiento del hablante con propósitos forenses (vigilancia y seguridad). Su objetivo principal es presentar un nuevo esquema de red neuronal artificial para actividades de reconocimiento de locutor que pueda ser aplicado en grabaciones de audio con ruido, en español y utilizadas como evidencia criminal. Proveer este tipo de análisis e información como una segunda opinión, ayudará a eliminar la subjetividad y parcialidad en las deliberaciones y decisiones emitidas por los jueces y jurados. También, permitirá reducir el error cometido por las técnicas de procesamiento de voz por medio de un nuevo esquema de RNA. Esto último conduce a la generalización de estos modelos para ser aplicados en múltiples contextos y posibilitará su validación para su uso en escenarios críticos.

La arquitectura de RNA que se propone utiliza funciones de activación paramétricas y una nueva unidad de cómputo, que es una variante de las neuronas artificiales; estas nuevas unidades son llamadas "neuronas de soporte". Las neuronas de soporte están diseñadas únicamente para optimizar los parámetros de las funciones de activación paramétricas. Esta arquitectura se inspira en nuevos enfoques que buscan imitar de manera más precisa las estructuras neurobiológicas. Específicamente, este nuevo esquema se basa en las RNA-Glía, que se inspiran a su vez en las células gliales, concretamente en los astrocitos. El modelo presentado (modelo óptimo) de red neuronal artificial, con función de activación MPReLU y neuronas de soporte, es capaz de reconocer con confiabilidad cuál sospechoso en particular pronunció cierta expresión en grabaciones de audio con diferentes tipos de ruido en el idioma español, con una exactitud del 98.68%.

Se desarrollaron diferentes configuraciones de redes neuronales artificiales con la finalidad de comparar el rendimiento del esquema propuesto contra la arquitectura de una red neuronal convencional. Para entrenar y validar los distintos experimentos se utilizó un corpus que consta de 158 clases (personas); de estas, 122 son hombres y 36 son mujeres. Cada clase fue etiquetada con el nombre de la persona. Las mismas personas se repiten en los subconjuntos de entrenamiento, validación y pruebas. El conjunto de datos consta de $4{,}911$ archivos de audio, de los cuales $3{,}395$ se utilizan

para entrenamiento (70%), 758 para validación y 758 para pruebas. Todos los archivos del conjunto de datos fueron grabados en español nativo con diferentes calidades y tipos de ruido ambiental.

Este trabajo de investigación presenta varias contribuciones. En primer lugar, se presenta un esquema de red neuronal con una nueva unidad de procesamiento. Esto tiene la intención de mejorar el desempeño de las redes neuronales artificiales en tareas de identificación del hablante en audios con ruido ambiental, de baja calidad, en el idioma español y con utilizados con fines forenses. En segundo lugar, se propone un modelo más robusto al considerar una combinación, normalización y selección de características acústicas. También, se propone la aplicación (prototipo) de escritorio HAAF, que permite interactuar con el esquema propuesto de RNA óptimo en escenarios reales. Por último, se desarrollan dos nuevas variaciones de funciones de activación paramétricas basadas en ReLU. Las mejoras y propuestas tienen la intención de añadir robustez al modelado de datos con ruido, optimizar de manera automática los parámetros de las funciones de activación paramétricas, reducir el costo computacional y mejorar el rendimiento en el modelado de señales acústicas.

1.2. Planteamiento del problema de investigación

La principal problemática en el ámbito del análisis forense digital es la tarea de examinar grabaciones de audio de mala calidad que se utilizan como evidencia en casos criminales. Este proceso puede ser propenso a errores, complejo y subjetivo. Por lo tanto, se requieren modelos de reconocimiento de locutor más precisos como apoyo para llevar a cabo este tipo de análisis de manera confiable.

Con base a lo anterior, este proyecto se inspira en dos problemáticas existentes en el análisis forense de evidencia digital. La primera problemática surge de la dificultad que existe en examinar grabaciones de audio de mala calidad o con ruido. Para el criminalista, juez y/o jurado, el proceso de analizar y evaluar este tipo de evidencia criminal puede llegar a ser un proceso de análisis subjetivo debido al criterio propio en la percepción auditiva. En el proceso de análisis forense de audio digital pueden existir dificultades específicas como reconocer con claridad cuáles palabras fueron pronunciadas e identificar con precisión quién (cuál sospechoso) pronunció ciertas palabras. Estos problemas de análisis son causados por varios factores, tales como la mala calidad del micrófono y dispositivo de reproducción, variaciones de voz, distorsión y ruido ambiental. Además, los problemas mencionados anteriormente pueden causar un análisis de la evidencia criminal erróneo, lo que a su vez puede conducir a la toma de decisiones o deliberaciones penales equivocadas por parte de la autoridad pertinente. Por ejemplo, la autoridad puede declarar de manera incorrecta el estado de inocencia o culpabilidad de un posible sospechoso.

Debido a la dificultad de las personas para analizar información de mala calidad, y a pesar de la asombrosa capacidad auditivo-perceptiva del ser humano, es necesario utilizar las herramientas tecnológicas como una ayuda adicional en el análisis forense de evidencia digital. Estas herramientas tecnológicas de uso forense deben manifestar un alto nivel de confiabilidad; sin embargo, los modelos de reconocimiento de locutor

o de identificación de voz para uso forense no presentan tasas de error significativamente pequeñas, lo que da origen a una segunda problemática en el análisis forense de audio digital. Por tanto, es imprescindible que los métodos de procesamiento de voz sigan reduciendo sus tasas de error en diversas tareas de modelado de señales acústicas, con la finalidad principal de que estos clasificadores puedan ser validados y utilizados en escenarios reales. De esta manera, ante un hecho delictivo y a partir de muestras de voz (evidencia criminal) se podrá determinar con mayor precisión si una expresión ha sido dicha por una persona en particular (sospechoso). El resultado será declarar el estado de inocencia o culpabilidad del sospechoso con la menor subjetividad y parcialidad posible.

Para resolver ambas problemáticas, se necesita desarrollar un modelo de reconocimiento de locutor para uso forense en escenarios ruidosos. Este modelo serviría como apoyo en el análisis de evidencia criminal y en los esfuerzos de reducir la parcialidad en la toma de decisiones penales.

1.3. Justificación del problema de investigación

A través de un modelo de identificación de locutor, el objetivo de esta investigación es permitir a los criminalistas, jueces y jurados analizar evidencia criminal (audios) con alta confiabilidad y en el menor tiempo posible. Este análisis, en grabaciones de voz con ruido ambiental, funge como una segunda opinión (información de apoyo), la cual proporciona información sobre quién pronunció cada expresión en las grabaciones de audio usadas como evidencia. El desarrollo de sistemas de reconocimiento de locutor que asemejen las capacidades auditivas y perceptivas de los humanos para realizar análisis de información acústica con alta precisión en grabaciones de mala calidad y con ruido ambiental, ayudará a reducir la toma de decisiones parciales e incorrectas emitidas por el criminalista, juez y/o jurado en escenarios forenses. Es decir, es imprescindible facilitar al juez y/o al jurado este tipo de información como una ayuda adicional para que emitan deliberaciones imparciales y correctas.

Mejorar las técnicas de reconocimiento e identificación de voz y obtener tasas de error más pequeñas, hace posible su validación para ser aplicadas en escenarios reales y críticos. También, este proyecto ayudará a los investigadores a seguir trabajando en el desarrollo de nuevos modelos que sean más robustos en el análisis de información con ruido en tareas de identificación del hablante. Por último, la mejora de estos modelos también contribuirá a que puedan ser generalizados y aplicados en múltiples contextos.

1.4. Preguntas de investigación

¿Cómo disminuir la tasa de error de las redes neuronales artificiales en tareas de identificación de locutor en entornos ruidosos y con fines forenses?

¿Calcular la media, mediana, moda, desviación estándar, el máximo y mínimo de los coeficientes de energía obtenidos por cada archivo de audio proporciona información adicional y relevante?

¿La selección, normalización y fusión de características acústicas pueden reducir los problemas de sobre-ajuste y sub-ajuste?

¿Los parámetros de una función de activación paramétrica pueden optimizarse automáticamente a través de unidades de procesamiento especializadas?

¿Cómo optimizar los pesos y sesgos del esquema de red neuronal propuesto mediante las técnicas de retro-propagación y la regla delta?

¿Ajustar las pendientes de la función de activación ReLU permite obtener un modelado más acorde al comportamiento de los datos?

¿Cómo interactuar con el modelo de identificación de locutor propuesto en escenarios reales?

¿El esquema de red neuronal artificial propuesto es superior en tareas de identificación de locutor en escenarios ruidosos en comparación con esquemas de red neuronal existentes?

1.5. Objetivo general

Desarrollar un esquema de red neuronal artificial para tareas de identificación de locutor en entornos ruidosos y con fines forenses.

1.6. Objetivos particulares

- Realizar una fusión, normalización y selección de características acústicas para solucionar problemas de sobre-ajuste y sub-ajuste.
- Construir características estadísticas (media, mediana, moda, desviación estándar, etc.) a partir de los coeficientes de energía.
- Desarrollar una unidad de procesamiento que permita optimizar los parámetros de las funciones de activación paramétricas de las redes neuronales artificiales.
- Diseñar un algoritmo de entrenamiento para la red neuronal artificial propuesta, el cual optimizará los pesos y sesgos de las diferentes unidades empleadas en la red.
- Desarrollar dos nuevas variantes de función de activación paramétrica basadas en la función DPReLU, que permitan ajustar las pendientes de la función ReLU.
- Diseñar una aplicación de escritorio que permita a los usuarios interactuar con el modelo propuesto para identificar a hablantes en audios con ruido ambiental.

- Evaluar y comparar el rendimiento del esquema de red neuronal propuesto con respecto a un esquema de red neuronal convencional.

1.7. Hipótesis

H0: El esquema de red neuronal artificial propuesto reducirá el error en un 0.5% (en términos de exactitud) en comparación con las arquitecturas de RNA existentes en tareas de identificación de locutor en entornos ruidosos.

1.8. Alcance

Esta investigación permitirá a otros científicos desarrollar nuevos esquemas de redes neuronales artificiales que incluyan nuevas funciones de activación y unidades de procesamiento basadas en células cerebrales biológicas. Incorporar nuevas unidades de procesamiento puede ayudar a procesar información con presencia de ruido y a mejorar los procesos internos de las RNA, como la optimización de parámetros específicos.

Las principales contribuciones de este trabajo son:

- Se propone un esquema de red neuronal más robusto para mejorar el desempeño de las RNA en tareas de identificación del hablante en grabaciones de audio en el idioma español, con diferentes tipos de ruido ambiental y utilizadas con fines forenses.
- El esquema de RNA desarrollado integra un nuevo tipo de unidad de procesamiento que permite optimizar de manera automática los parámetros entrenables de las funciones de activación paramétricas. Estas nuevas unidades son una variante de las neuronas artificiales convencionales llamadas neuronas de soporte.
- Se presenta el procesamiento matemático para entrenar las neuronas de soporte mediante los métodos de propagación hacia atrás, gradiente estocástico descendente y la regla delta.
- Se realiza una fusión de características entre MFCCs y variables estadísticas obtenidas a partir del coeficiente de energía. Posteriormente, se lleva a cabo una normalización y selección de características acústicas para generar un modelo de identificación de locutor más robusto.
- Dos nuevas variantes de funciones de activación (AReLU y MPReLU) paramétricas basadas en ReLU y DPReLU fueron desarrolladas para realizar un modelado más acorde con el comportamiento de los datos.

1.9. Estructura de la tesis

El trabajo de investigación presentado se encuentra estructurado en 6 capítulos,

los cuales se organizan de la siguiente manera:

- Capítulo 1: Introducción. Este capítulo describe el contexto general de la investigación y se desglosa en las siguientes subsecciones: antecedentes, planteamiento del problema, justificación, preguntas de investigación, alcance, objetivo general e hipótesis.
- Capítulo 2 y 3: Marco teórico. Ambos capítulos introducen el aspecto teórico necesario para comprender los diversos conceptos y técnicas de procesamiento de voz e inteligencia artificial que sustentan la investigación. Adicionalmente, se presenta una revisión y comparación de los trabajos relacionados.
- Capítulo 4: Propuesta de investigación. En esta sección se describe el modelo propuesto para llevar a cabo la identificación de locutor, la fase recolección y análisis de los datos, el diseño de los experimentos, la metodología de investigación y el marco de trabajo empleado.
- Capítulo 5: Resultados y discusión. El capítulo presenta los resultados de los experimentos desarrollados y descritos en el capítulo 4.
- Capítulo 6: Conclusiones. Este capítulo discute las conclusiones obtenidas del análisis de los resultados expuestos en el capítulo 5, así como el trabajo futuro.

Capítulo 2. Inteligencia Artificial

En este capítulo se introduce el aspecto teórico necesario para comprender los diversos conceptos y técnicas de inteligencia artificial utilizadas para el sustento de la propuesta de investigación. En las diferentes secciones del capítulo se abordan temas como recolección de datos, procesamiento de datos, extracción de características y modelos inteligentes. En conclusión, el capítulo proporciona un resumen de la información técnica y científica recopilada durante la investigación.

2.1. Fundamentos de inteligencia artificial

La inteligencia artificial es una ciencia que crea entidades inteligentes, como algoritmos, sistemas de software, máquinas y robots. Estas entidades tienen la habilidad de comprender y analizar datos, comunicarse, generar información y establecer relaciones entre elementos (reconocer patrones). También pueden emular la manera de cómo piensan los humanos para llevar a cabo tareas de clasificación, predicción y manipulación de objetos; por ejemplo, clasificar imágenes, predecir el clima y servir una taza de café [41]–[43]. La IA es una ciencia multidisciplinar que involucra el conocimiento de distintos campos de estudio como la psicología, neurociencia, biología, electrónica, matemáticas, física y ciencias de la computación. Actualmente, la IA se aplica en muchos escenarios, como la educación, las finanzas, la ciberseguridad, la criminología, el análisis forense, la salud y otros campos [44], [45].

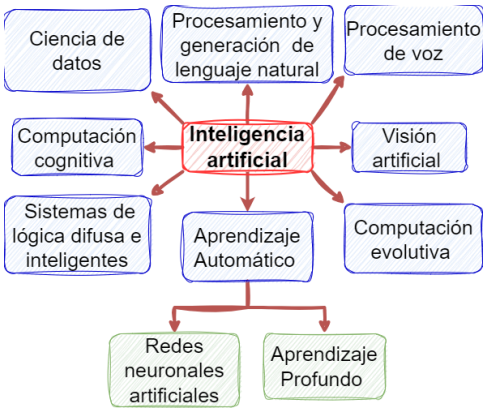

Figura 2.1 Principales campos de la IA.

En la Figura 2.1 se muestran los principales subcampos de la IA. Cada subcampo trabaja con un tipo de enfoque, y podemos identificar dos enfoques principales: 1) basado en la psicología y 2) basado en la biología. Los sistemas basados en la psicología se centran en emular qué hace el cerebro humano, es decir, emulan la manera de cómo piensan y actúan los humanos. Algunos ejemplos son agentes inteligentes y sistemas basados en lógica difusa. Mientras que los sistemas basados en la biología se orientan en cómo se estructura y procesa la información el cerebro humano a través de los procesos y la interacción de células cerebrales. El objetivo de este enfoque es emular la estructura biológica (hardware) y las interacciones de las células del cerebro humano, por ejemplo, las RNA y algoritmos evolutivos [41], [42].

2.2. Ciencia de datos y reconocimiento de patrones

El reconocimiento de patrones y la ciencia de datos son dos herramientas fundamentales en la inteligencia artificial. Ambos campos de estudio permiten analizar, comprender y modelar información. La ciencia de datos encuentra patrones, genera conocimiento y describe comportamientos a partir de los datos. Además de eso, esta disciplina emplea herramientas matemáticas para analizar estadísticamente los datos y hace uso del reconocimiento de patrones para modelar la información [46]. Por otra parte, el reconocimiento de patrones se ocupa de descubrir automáticamente reglas y comportamientos en los datos, es decir, descubrir patrones. Este proceso es realizado mediante el uso de algoritmos, un procesamiento de los datos y la extracción de propiedades significativas que permiten discriminar y presentar la información. La finalidad del reconocimiento de patrones es extraer y modelar información para realizar actividades de clasificación o predicción [47].

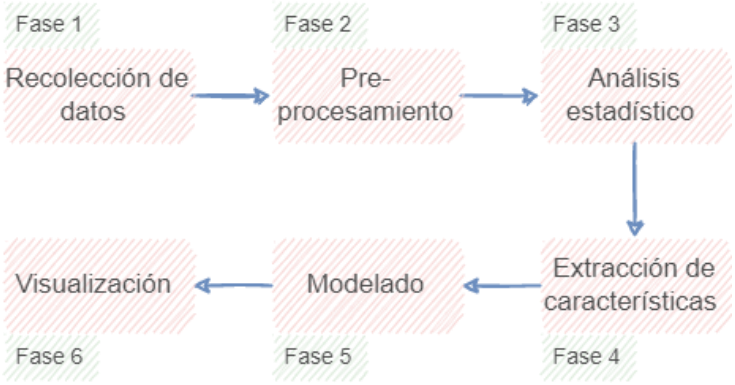

Figura 2.2 Etapas de la ciencia de datos.

La ciencia de datos y el reconocimiento de patrones son dos áreas de estudio que comparten algunas etapas en su ciclo de desarrollo, lo cual permite que se complementen (ver Figuras 2.2 y 2.3). La ciencia de datos recolecta información y después la preprocesa, la analiza estadísticamente, la representa en propiedades, la modela y la visualiza. De manera similar, el reconocimiento de patrones involucra

etapas como la recolección de datos, el pre-procesamiento, la extracción de características y la clasificación [46], [47].

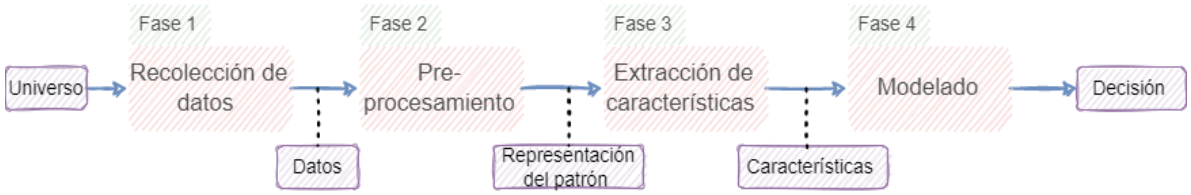

Figura 2.3 Etapas del reconocimiento de patrones.

La Figura 2.4 indica que la ciencia de datos y el reconocimiento de patrones deben pasar por dos etapas generales: entrenamiento y pruebas. También, se puede apreciar que algunas veces se incluye una subproceso adicional, conocido como selección de características [46], [47].

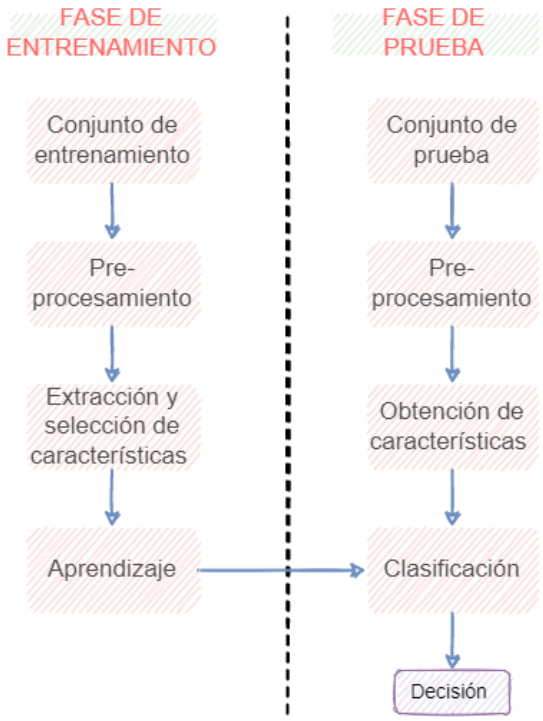

Figura 2.4 Fases de entrenamiento y prueba.

2.3. Recolección de datos

La fase de recolección de datos es un proceso ordenado y estructurado, la cual permite recopilar información necesaria sobre algún fenómeno físico que se desea medir durante la investigación. En este proceso, se pueden recolectar datos

cuantitativos y cualitativos. Tanto los datos cuantitativos como los cualitativos se pueden obtener mediante métodos de recolección de datos numéricos y no numéricos [48], [49]. En ocasiones, no se recopilan datos nuevos, sino que se selecciona un conjunto de datos ya existente.

La Tabla 2.1 [49] muestra un resumen de métodos para recopilar datos cuantitativos y cualitativos. Se pueden obtener datos cuantitativos a través de métodos como sensores, encuestas, pruebas y ecuaciones. Por otra parte, los métodos de recolección de datos cualitativos permiten la recolección mediante las observaciones, el análisis de documentos y cuestionarios. Es importante recalcar que para medir algunos fenómenos físicos, a veces es necesario utilizar métodos de recolección de datos tanto numéricos como no numéricos [49], [50]. Por último, en la Figura 2.5 [49] se muestra el proceso general de recolección de datos, el cual es definido por Hernández Sampieri.

Tabla 2.1 Técnicas de recolección de datos.

Instrumento	Variables cuantitativas	Variables cualitativas
Entrevistas/cuestionaros	Sí	Sí
Escalas de actitudes	Sí	No
Grupos de enfoque	No	Sí
Equipos, aparatos y sistemas basados en escalas universales	Sí	No
Formulas y ecuaciones	Sí	No

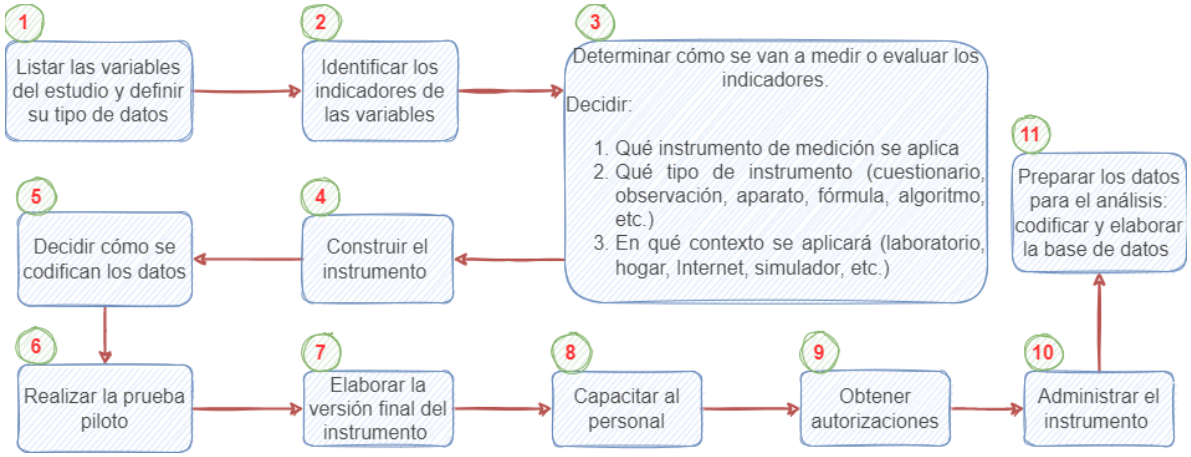

Figura 2.5 Proceso de recolección de datos.

2.4. Pre-procesamiento

El pre-procesamiento es una etapa crucial en la ciencia de datos, en el reconocimiento de patrones y el aprendizaje automático (machine learning). Esta fase es necesaria para mejorar el rendimiento de los modelos, en otras palabras, hacer clasificaciones y predicciones más acertadas. El pre-procesamiento implica diversas

fases iniciales, como la limpieza de datos, la integración de datos, la corrección de datos y la transformación de datos. Estas fases tienen la intención de adecuar la información para realizar otras fases posteriores de pre-procesamiento, como la normalización, la extracción de características y la reducción de dimensiones de características. En algunos casos, también se puede requerir una fase adicional de selección de características para mantener solo la información más representativa. El pre-procesamiento de datos tiene varios objetivos, como preparar los datos para su modelación, eliminar el ruido, corregir inconsistencias, eliminar información redundante e irrelevante, ayudar a identificar patrones en los datos o encontrar relaciones entre las variables, además de reducir la dimensión de los datos y encontrar el conjunto de características más informativo [46], [51]–[53].

La Figura 2.6 muestra el marco de trabajo para el pre-procesamiento de la información. Las fases para el procesamiento de los datos pueden realizarse de forma independiente o en paralelo, estrictamente no se realizan de manera secuencial, y no todas las fases son ejecutadas, ya que una fase puede o no requerir de las otras. Por ejemplo, para eliminar el ruido de los datos pueden ser necesarias las transformaciones de datos, la normalización y la extracción de propiedades, pero no la reducción de variables [52], [53].

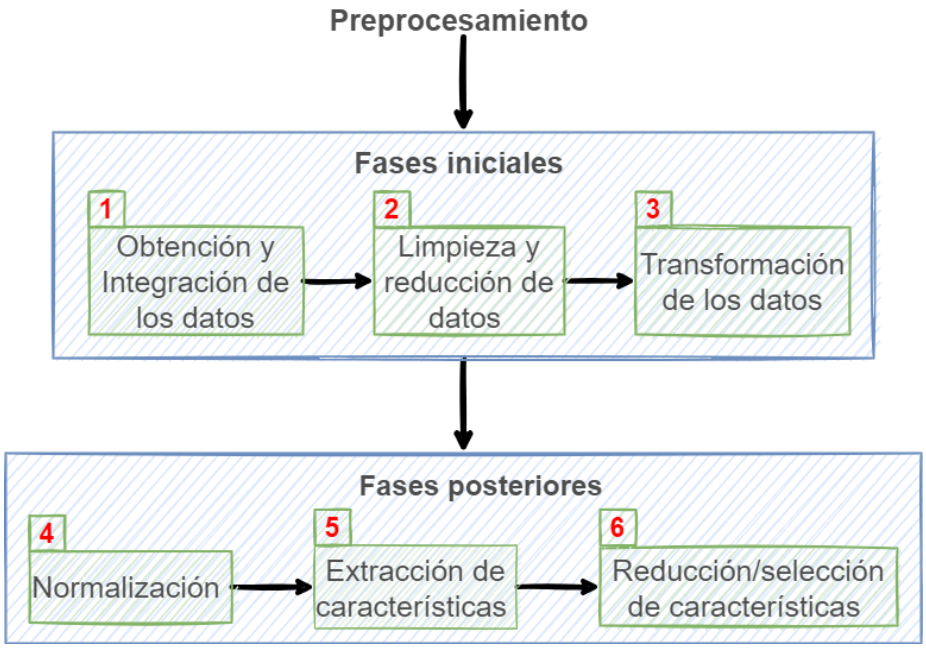

Figura 2.6 Fases del pre-procesamiento de la información.

El procesamiento de los datos (ver Figura 2.6) inicia con la integración de datos, esto consiste en recopilar y combinar información proveniente de diversas fuentes. Posteriormente, en la fase de limpieza, se eliminan o corrigen todas las inconsistencias, como registros innecesarios, eliminar valores duplicados y corruptos. La preparación de la información pasa a una tercera fase llamada transformación de

los datos, que involucra diversos pasos, tales como el muestreo de datos, la creación de nuevas características, cambios de formato, completar valores faltantes, ordenar variables, reescalar magnitudes y la creación de variables ficticias. El pre-procesamiento de los datos continúa con la normalización de las variables para eliminar valores atípicos generados por información incorrecta y la variación de escalas entre las variables. Luego, se lleva a cabo la extracción de características para extraer la información representativa y discriminativa. Finalmente, se realiza una reducción y/o selección de propiedades para obtener un grupo de variables más simplificado e informativo; esto se logra al agregar y eliminar características redundantes o agrupándolas [46], [51]–[53].

2.5. Extracción/selección de características

La extracción de características consiste en obtener información significativa, discriminativa y representativa de un objeto o señal. Esta nueva información debe ser capaz de representar patrones para realizar actividades de clasificación o predicción. La finalidad principal de la extracción/selección de propiedades es reducir la dimensionalidad de los datos, permitir la discriminación y eliminar información redundante e irrelevante. En el caso de la reducción de la dimensionalidad, el proceso inicia a partir de una colección de objetos representados por vectores de datos discretos de n variables. Después, estos vectores son mapeados a vectores más pequeños, con el objetivo de reducir la cantidad de datos y representar las características o propiedades más importantes del objeto o señal [54]–[58].

Tabla 2.2 Métodos de extracción/reducción de características.

Método	Descripción
Análisis de componentes principales (PCA).	Es un método utilizado para extraer características, visualizar datos y reducir la dimensionalidad. Para hacer esto, PCA realiza una proyección lineal u ortogonal donde se minimiza la distancia media cuadrática entre los datos (distancias) [59].
Análisis de Discriminantes Lineales (LDA).	Para la reducción de dimensionalidad, LDA usa técnicas de transformación lineal supervisadas. La idea de LDA es calcular la separabilidad entre clases mediante la varianza de cada clase. Después, se crea el espacio de menor dimensión que maximice la separabilidad entre clases y minimice la varianza de cada clase [60].
Análisis de Componentes Independientes (ICA).	Para que los componentes sean estadísticamente independientes, ICA busca una representación lineal de los datos no Gaussianos. En lugar de minimizar el error cuadrático medio como es el caso de PCA, en ICA se busca una representación lineal de los datos no Gaussianos. En otras palabras, ICA basa su búsqueda de proyección en un dimensión donde la dependencia entre características sea mínima, es decir, minimiza esa dependencia [61], [62].

Existen muchos métodos de extracción y reducción de características. Algunos de estos métodos son específicos para el tipo de datos a procesar, como en el procesamiento de imágenes y voz. La Tabla 2.2 presenta los métodos más comunes de extracción y reducción de propiedades.

Algunas veces la extracción de características también se refiere al proceso de reducción de dimensionalidad. En este caso, la idea es que un corpus de variables sea reducido a un conjunto de propiedades más pequeño, conservando la información más importante. Es decir, se crean nuevas variables combinando atributos y conservando la información más relevante. Este proceso puede ocurrir a partir de un conjunto de datos sin procesar o de un corpus de datos que se ha reducido previamente mediante una selección de características [54]–[58], [61].

2.5.1. Selección de características

La selección de características es el proceso de elegir automáticamente las variables más informativas mediante un criterio de selección o discriminación. El resultado es un subconjunto de atributos que describen mejor la información original o el fenómeno físico. El objetivo de la selección de propiedades significativas es eliminar el ruido, las variables redundantes e irrelevantes del corpus de datos. En otras palabras, se desecha toda la información que no ayuda o dificulta el modelado de la información. Otro de los objetivos es disminuir el costo computacional, mantener o incrementar la capacidad de generalización y, en algunos casos, reducir el sobr-eajuste del clasificador [63]–[65].

Tabla 2.3 Métodos de selección de características.

Método	Tipo de método
Selección secuencial hacia adelante (SFS, por sus siglas en inglés).	Método de envoltura.
Selección secuencial hacia atrás (SBS, por sus siglas en inglés).	Método de envoltura.
Eliminación de características recursivas (RFE, por sus siglas en inglés).	Método de envoltura.
Chi-cuadrado.	Método de filtro.
Correlación de Pearson.	Método de filtro.
LDA.	Método de filtro.
Fisher score.	Método de filtro.
Gain ratio univariate.	Método de filtro.
Bosques aleatorios.	Método integrado e hibrido.
Arboles de decisión.	Método integrado e hibrido.
Regresión logística.	Método integrado e hibrido.
Algoritmos genéticos.	Método integrado e hibrido.

Por otra parte, el criterio de selección de atributos debe medir la relevancia o influencia de las características en las clases o variables dependientes. La Tabla 2.3 muestra los métodos de selección de características más comunes [63]–[65].

Los métodos de selección de características se clasifican comúnmente en 3 tipos: i) métodos integrados e híbridos (embedded and hybrid), ii) de filtro y iii) wrappers (o envoltura). Los algoritmos wrapper llevan a cabo la selección de características basándose en la búsqueda de combinaciones de variables. Dicho de otro modo, esta metodología genera varias combinaciones de variables o subconjuntos, los cuales son posibles soluciones subóptimas. Posteriormente, cada subconjunto de atributos se evalúa utilizando un modelo y se compara el rendimiento de los subconjuntos para elegir la combinación que ofrezca los mejores resultados en términos de desempeño en tareas de clasificación o predicción [63]–[66].

Las técnicas de filtrado seleccionan las características y filtran cada propiedad bajo un criterio estadístico, donde el desempeño o importancia de cada variable se calcula mediante una métrica o medida estadística. Este proceso es independiente de un modelo de clasificación o predicción. Por otro lado, los métodos integrados son algoritmos de modelado capaces de evaluar la importancia de cada variable. En este tipo de técnicas, es necesario utilizar un modelo de clasificación o regresión que contenga esta función integrada para realizar la selección de características. Algunos métodos de aprendizaje automático integran un componente que realiza una selección de atributos mediante una ponderación de variables basada en la regularización [63]–[66].

Finalmente, los métodos híbridos combinan los procesos de filtro y envoltura. Primero, se utiliza una técnica de envoltura para reducir las dimensiones del corpus de propiedades original y generar varias posibles soluciones o subconjuntos. Después, se aplica un algoritmo de filtro para encontrar el subconjunto de características más representativo [63]–[66].

2.6. Análisis estadístico

En esta fase se utilizan herramientas matemáticas para entender el comportamiento de los datos, resumir las propiedades de las variables y encontrar relaciones entre ellas. El análisis de datos depende del tipo de datos: cuantitativos o categóricos. Además, esta fase se realiza después de la recolección y el pre-procesamiento de los datos.

En esta investigación solo se trabaja con variables cuantitativas o numéricas. Por tanto, en la Figura 2.7 [49] se describe paso a paso el proceso básico para el análisis de datos cuantitativos. Es importante señalar que este método fue propuesto por Hernández Sampieri. El proceso comienza eligiendo la aplicación de software o lenguaje de programación que permita efectuar la manipulación y el análisis estadístico de los datos. En el segundo paso se verifica que no existan errores en los datos capturados. Después de depurar los errores se realiza el análisis estadístico. Una vez ejecutado este paso, se procede a probar las hipótesis y estimar parámetros mediante estadística inferencial. Finalmente, en los pasos 5 y 6 se asegura que las mediciones son precisas y se visualizan los resultados del análisis estadístico.

Las variables nominales, ordinales, de intervalo y de razón son las cuatro categorías existentes de variables cuantitativas. En la ciencia de datos y en cualquier

investigación con un enfoque cuantitativo, el análisis estadístico básico se utiliza comúnmente para identificar tendencias, resumir la información, encontrar relaciones y describir el comportamiento de atributos numéricos. La distribución de frecuencias o distribución de probabilidad, las medidas de dispersión y la tendencia central se determinan en este análisis descriptivo de valores por variable [49].

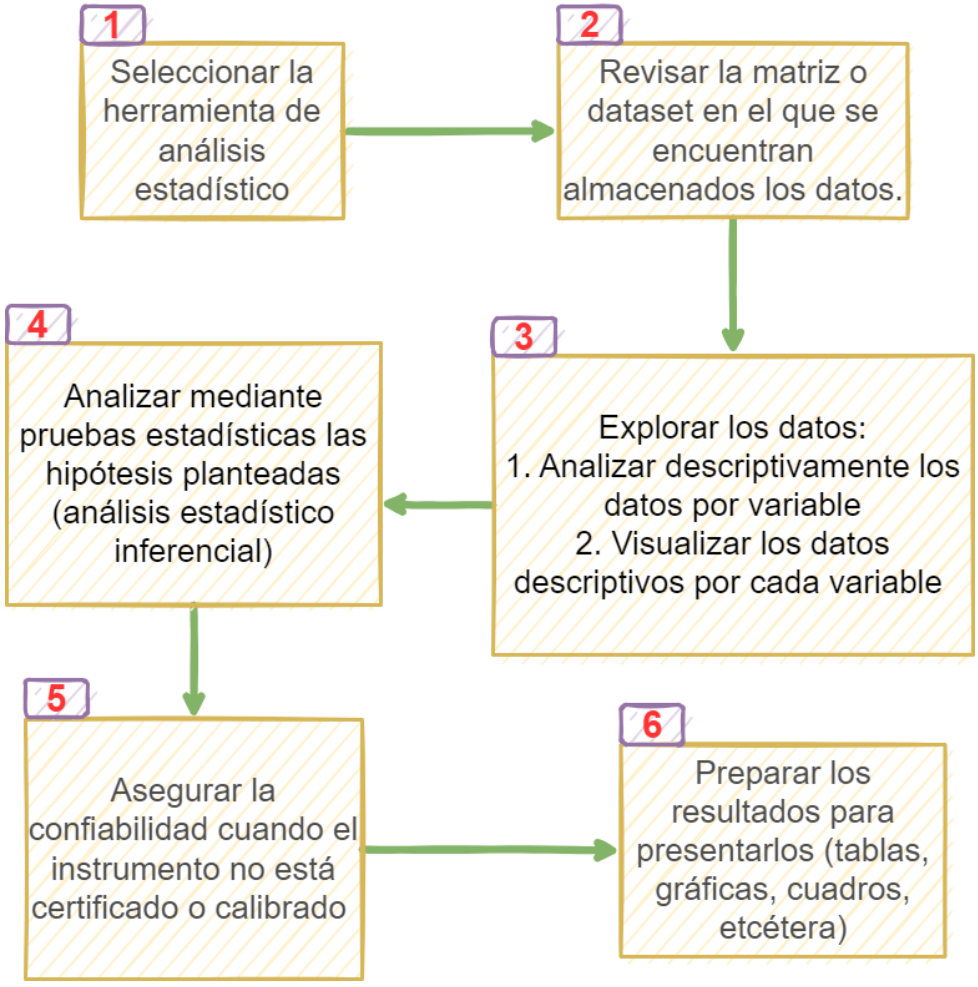

Figura 2.7 Proceso de análisis de datos cuantitativos.

2.6.1. Distribución de frecuencias

La distribución de frecuencias es un instrumento estadístico que se utiliza para contar las observaciones pertenecientes a cada categoría definida en las variables categóricas o numéricas. Las gráficas y tablas son una forma común de presentar este tipo de análisis. El propósito de las agrupaciones en frecuencias es resumir y obtener información que está oculta en los datos. Algunas veces también se puede obtener la distribución de probabilidad de las variables aleatorias. En este último caso, la idea es obtener una lista de los posibles valores de la variable aleatoria y las respectivas probabilidades de ocurrencia asociadas a cada valor [49], [67].

2.6.2. Estadística descriptiva: medidas de tendencia central y variabilidad

Las medidas de tendencia central (media, moda y mediana) son los valores centrales de la distribución de frecuencias de la información y proporcionan información sobre cómo se comportan los datos de una manera resumida y simple.

2.6.2.1 Cuartiles

Los cuartiles, deciles y percentiles son medidas estadísticas que permiten analizar el comportamiento, dispersión y distribución de los datos en grupos o categorías. Para determinar la ubicación de cada cuartil, decil o percentil, primero se ordenan las observaciones de menor a mayor. Luego, se calculan los límites de división y el conjunto de datos se fragmenta en partes iguales, es decir, que cada parte debe contener la misma cantidad de valores [67]–[69]. En el caso de los cuartiles, las observaciones se dividen en cuatro grupos con la misma cantidad de datos, como se muestra en la ecuación (2.1).

$$i = 1 + \frac{i(n-1)}{4}. \tag{2.1}$$

Donde:

i: Valor del cuartil i.

n: Total de elementos del conjunto x.

2.6.2.2 Media

La media o promedio aritmético de una distribución es un valor estadístico que representa de manera generalizada a una colección de datos, es decir, es su valor central. Se calcula sumando todos los valores del corpus de datos a analizar y se divide la suma por el número total de observaciones, ver ecuación (2.2). Es importante señalar que esta medida de tendencia central puede ser poco representativo o confiable si hay datos atípicos, lo que puede dar como resultado un valor poco representativo del grupo de datos [49], [67]–[69].

$$\bar{\bar{X}} = \frac{\sum_{j=1}^{N} x_j}{N}. \tag{2.2}$$

Donde:

x: Conjunto de números.

N: Total de elementos del conjunto x.

$\bar{\bar{X}}$: La media del vector x.

2.6.2.3 Moda

La moda es una medida abstracta que describe de forma resumida a una serie de valores mediante la observación que se repite con mayor frecuencia en esa colección de datos. Si hay dos o más valores que se repiten con la misma frecuencia, se dice que la distribución es bimodal o multimodal, respectivamente [49], [67]–[69].

2.6.2.4 Mediana

La mediana es una medida estadística que resume toda la información de un vector de datos previamente ordenado. La idea es ordenar para luego buscar el valor o par de elementos que estén en la posición central, como se presenta en la ecuación (2.3). Es decir, la mediana divide los datos ordenados en dos grupos con la misma cantidad de observaciones para encontrar el valor más representativo [49], [67]–[69].

$$\frac{n+1}{2}. \tag{2.3}$$

Donde:

n: Número de observaciones.

2.6.2.5 Rango

El rango es un valor que permite saber la variación máxima de los datos aplicando la resta del valor máximo menos el valor mínimo de un vector x [49], [67]–[69]. La ecuación (2.4) muestra cómo calcular el rango en una serie de datos.

$$Rango = Max(x) - Min(x). \tag{2.4}$$

Donde:

$Max(x)$: Valor máximo.

$Min(x)$: Valor mínimo.

2.6.2.6 Varianza

La varianza describe cuánto se alejan los datos de una serie de observaciones respecto a su valor central (media) [49], [67]–[69]. La ecuación (2.5) muestra cómo calcular la varianza.

$$Var(X) = \frac{\sum_{j=1}^{N}(x_j - \bar{X})^2}{N}. \tag{2.5}$$

Donde:

X: Colección de números.

N: Total de elementos del conjunto X.

2.6.2.7 Desviación estándar

La ecuación (2.6) indica que para calcular la desviación estándar se debe obtener la raíz cuadrada de la varianza. Este valor mide cuánto se alejan los datos entre sí. Cuanto más dispersos están los datos, mayor es el valor de la desviación estándar [49], [67]–[69].

$$\sigma_x = \sqrt{Var(x)}. \quad (2.6)$$

Donde:

x: Conjunto de números.

$Var(x)$: La varianza de x.

2.6.2.8 Correlación

La correlación encuentra si hay alguna relación lineal o no lineal entre dos variables. Esta relación no significa causalidad, pero si busca medir cuánto influye una variable sobre la otra y viceversa. La letra r representa el valor de esta medición, que se encuentra en el intervalo $[-1, 1]$ [49], [68], [69].

Existen diferentes tipos de correlación. Por ejemplo, el coeficiente de correlación de Pearson se usa para datos con relaciones lineales. Mientras que el coeficiente de correlación de Spearman es útil para variables ordinales o relaciones no lineales. La ecuación (2.7) define la ecuación para el coeficiente de correlación de Pearson [49], [69]. La Interpretación de la correlación de Pearson es la siguiente:

- $r = 1$: Hay una correlación positiva perfecta.
- $0 < r < 1$: Se trata de una correlación positiva.
- $r = 0$: Significa que no hay una relación lineal, posiblemente la relación es de tipo curvilineal.
- $-1 < r < 0$: Existe una correlación negativa.
- $r = -1$: Hay una correlación negativa perfecta.

$$p = \frac{\sigma_{XY}}{\sigma_X \sigma_Y}. \quad (2.7)$$

Donde:

σ_{XY}: Covarianza de X e Y.

σ_X: Desviación estándar de X.

σ_Y: Desviación estándar de Y.

2.7. Modelado de la información

La fase de modelado, en reconocimiento de patrones, machine learning y ciencia de datos, consiste en hacer una abstracción y generalización de los datos con el objetivo de encontrar, reconocer y representar patrones en ellos. Una vez que se obtiene esta representación de la información se pueden llevar a cabo tareas de clasificación y predicción. Todo este proceso es realizado por un algoritmo inteligente, el cual es capaz de encontrar y aprender relaciones en una colección de observaciones para hacer predicciones o clasificaciones en un futuro [47].

2.8. Aprendizaje automático y profundo

El aprendizaje automático y profundo son modelos matemáticos que se inspiran en la estructura biológica del cerebro, ver Figura 2.8 [70]. Es técnicas son capaces de aprender y emular la manera en que el cerebro procesa e interpreta la información. El aprendizaje automático tiene una gran variedad de modelos, como KNN, SVM y RNA. Mientras que el aprendizaje profundo se centra en un tipo de RNA, las redes neuronales profundas, que están compuestas por una gran cantidad de capas de neuronas [70]–[72].

Estos modelos matemáticos inteligentes también emulan otras capacidades humanas, como la capacidad de memorización y la manera de aprender. Los algoritmos inteligentes aprenden mediante ejemplos, el repaso y de manera supervisada. Es decir, que el modelo recibe una serie de información con la estructura de “descripción e interpretación” o “objeto y significado”. Mediante iteraciones (repaso), el modelo procesa toda la información y va aprendiendo a encontrar las relaciones o patrones en la información. Los algoritmos como RNA, regresión lineal y logística tienen varios pesos que se modifican durante el aprendizaje. Estos pesos representan el aprendizaje o la plasticidad neuronal del cerebro [70]–[72].

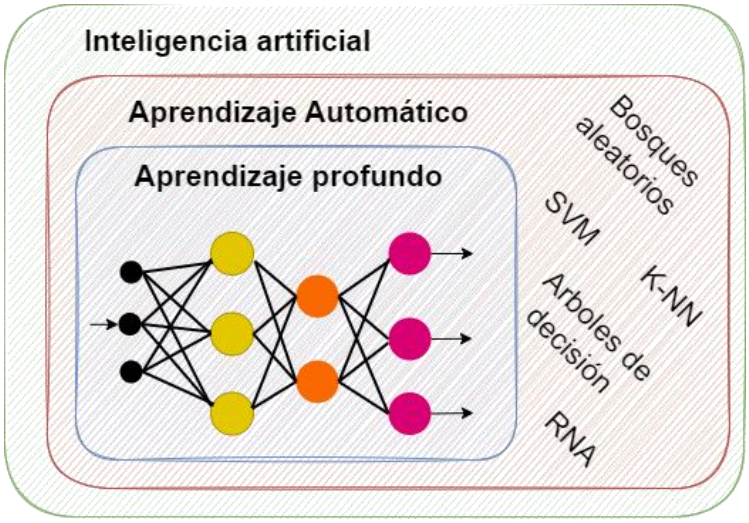

Figura 2.8 Aprendizaje automático y profundo.

2.9. Tipos de aprendizaje

Los modelos de aprendizaje automático y profundo también aprenden mediante 3 enfoques análogos a los del ser humano, de manera instruida, autodidacta y por retroalimentación, es decir, a) aprendizaje supervisado, b) aprendizaje no supervisado y c) aprendizaje por refuerzo respectivamente, ver Figura 2.9 [72] y Figura 2.10 [72]. Algunos modelos pueden aprender bajo un solo enfoque de aprendizaje o varios enfoques [70]–[72].

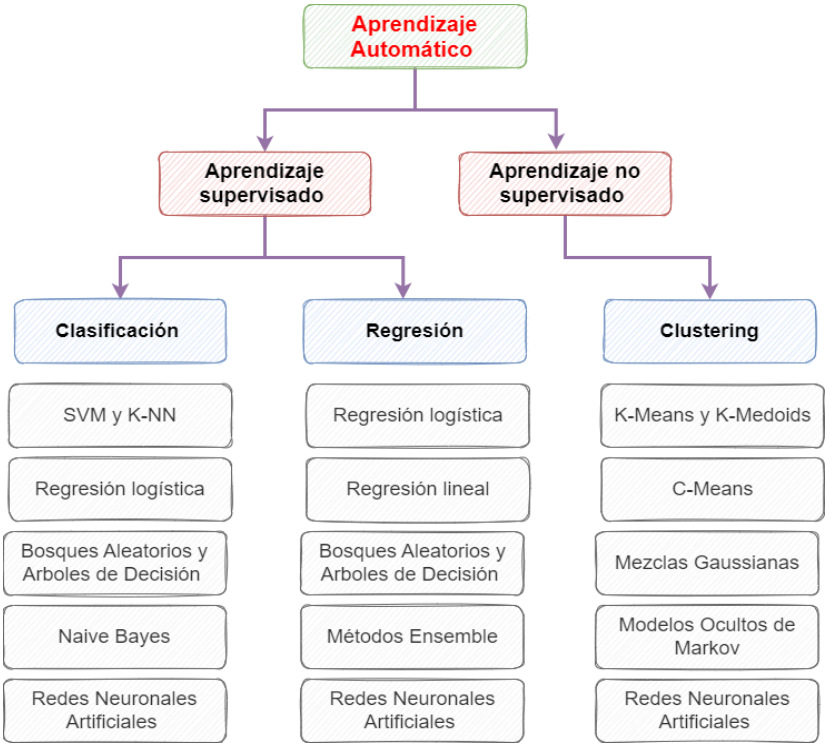

Figura 2.9 Tipos de aprendizaje y algoritmos en el área de aprendizaje automático.

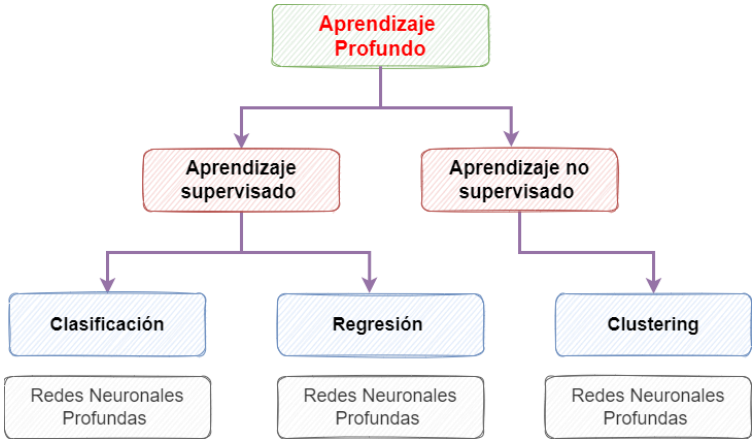

Figura 2.10 Tipos de aprendizaje y técnicas en el área de aprendizaje profundo.

2.9.1. Aprendizaje supervisado

En el aprendizaje supervisado se le proporciona información al clasificador acerca de las propiedades de un objeto o fenómeno fisco junto a su significado (etiqueta). Por ejemplo, si se le quiere enseñar al modelo a reconocer manzanas, entonces debemos proporcionarle información sobre el color, forma, tamaño, textura y peso del tipo de manzana a reconocer. La idea es proporcionar una serie de propiedades y su interpretación, después el algoritmo inteligente aprenderá a encontrar una relación entre la serie de datos y su significado mediante una abstracción. Este enfoque de aprendizaje es muy similar en la manera en cómo se les enseña y aprenden las personas [70], [73].

2.9.2. Aprendizaje no supervisado

En el aprendizaje no supervisado, el algoritmo asigna un significado o interpretación a las características de un objeto o fenómeno físico. Esto se hace proporcionando una serie de datos al clasificador, pero sin indicar su significado; el algoritmo es quien determina el significado de la información. Para encontrar este significado, el modelo realiza agrupaciones de datos mediante una relación o asociación de entre las diferentes propiedades [70], [73].

2.9.3. Aprendizaje por refuerzo

Los modelos aprenden mediante la prueba y el error. Solo se les proporciona una retroalimentación sobre si su clasificación o predicción fue correcta o no. [70], [73].

2.10. Algoritmos de clasificación y predicción

En inteligencia artificial, existe una gran variedad de modelos que simulan la capacidad de aprender del ser humano, como KNN, SVM, árboles de decisión, bosques aleatorios, regresión logística y redes neuronales artificiales. Estos clasificadores son capaces de realizar tareas como clasificar imágenes, predecir el clima, identificar y mover objetos.

2.10.1. Redes neuronales artificiales

Las computadoras resuelven de manera rápida y eficiente actividades como operaciones matemáticas, manipulación de datos, razonamiento lógico, mover objetos, consultar información y memorizar datos. Sin embargo, no son capaces de realizar de manera eficiente procesos creativos, generar ideas, aprender, tomar decisiones complejas, interpretar emociones, reconocer patrones y realizar actividades de percepción, diagnosticar el estado de salud de un individuo, entre otras

actividades. Estas últimas actividades son propias únicamente del cerebro humano o animal, el cual es capaz de realizarlas con eficiencia. Por tanto, las redes neuronales artificiales tienen el objetivo de emular a nivel lógico o físico las estructuras y conexiones neuronales del cerebro para que las computadoras puedan llevar a cabo las actividades mencionadas [74]–[76].

2.10.1.1 Neurona biológica

En la Figura 2.11 [76] se presenta la estructura de una neurona biológica y el proceso de sinapsis. Las neuronas son capaces de recibir y transmitir información de otras células mediante las dendritas, que funcionan como canales de entrada y salida de la información. Esta información es procesada por el cuerpo celular, que también funciona como una unidad de procesamiento. Después de procesar la información, esta se transporta por el axón hasta otra ramificación de dendritas para ser entregada a otras neuronas o células [74]–[76].

Las neuronas pueden disparar pulsos eléctricos o liberar químicos llamados neurotransmisores. Estas acciones químicas y eléctricas se conocen como impulsos nerviosos o potenciales de acción. Después, las neuronas transmiten esta información, la cual ya fue procesada por el soma, a otras neuronas para estimular o inhibir su actividad. Estas conexiones o uniones se llaman sinapsis y forman redes de neuronas [74]–[76].

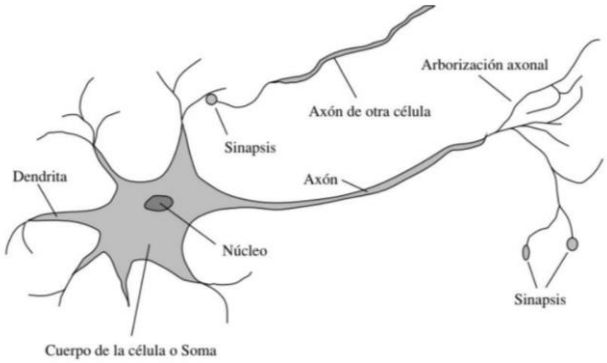

Figura 2.11 Sinapsis.

La neurona que dispara un potencial de acción se llama neurona presináptica, y la que recibe los pulsos se llama neurona postsináptica. El resultado de la capacidad de una neurona para sumar e integrar todas las señales recibidas (información) se conoce como potencial de acción. Este pulso es la probabilidad de que la neurona postsináptica dispare su propio potencial de acción. Es decir, la neurona decide enviar o no el potencial de acción bajo un umbral [74]–[76].

2.10.1.2 Neurona artificial

La estructura de una neurona artificial intenta emular cada característica, parte y habilidad de una neurona biológica. Las neuronas artificiales están formadas por

modelos matemáticos simples que trabajan de manera conjunta y forman un modelo más complejo, como se muestra en la Figura 2.12. Es decir, una neurona artificial es una unidad básica de procesamiento que está formada por estructuras matemáticas básicas y tiene la capacidad de modificar valores propios (aprendizaje).

En la Figura 2.12, el vector de entrada x y el vector de pesos w simulan las dendritas de la neurona presináptica y postsináptica. Al multiplicarse ambos vectores entre sí, emulan el proceso sináptico o de comunicación entre ambos tipos de neuronas. La función de agregación $\sum$ imita el comportamiento del soma, donde la idea es integrar las señales recibidas en la neurona. El resultado de este proceso matemático sería equivalente al potencial de acción. Después, la función de activación $f(x)$ representa el axón. Esta función tiene como objetivo recibir el potencial de acción y transformar los datos de una representación lineal a una no lineal. Finalmente, el valor obtenido se considera como la salida o pulso de la neurona, la cual está definida por $a = f(Wx + b_0)$. En las subsecciones posteriores se explica a detalle cada parte de la neurona artificial [75], [76].

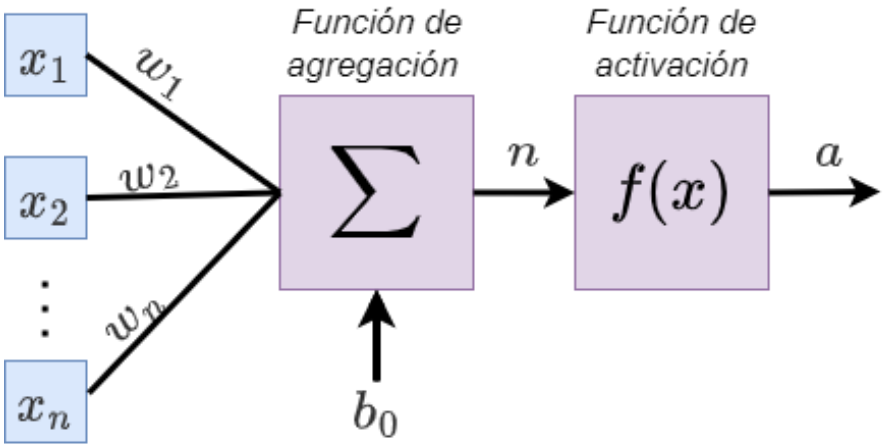

Figura 2.12 Estructura de una neurona artificial.

Entradas:

Un vector de características x representa los pulsos eléctricos o neurotransmisores emitidos por las neuronas presinápticas [75]. Este vector es definido en la ecuación (2.8).

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}. \qquad (2.8)$$

Donde:

x_n: Representa una característica o elemento del vector.

Pesos:

A cada entrada x_n se le asigna un peso w_n (valor numérico) correspondiente, ver ecuación (2.9). Este peso permite definir la importancia de la característica o entrada. Es decir, los pesos funcionan como un valor de ponderación. Además, los pesos son representados por un vector w y su objetivo es simular las dendritas de la neurona biológica. Estos valores se modifican mediante una fase de aprendizaje [75], [76].

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_n \end{bmatrix}. \quad (2.9)$$

Donde:

w_n: Elemento o peso del vector.

Función de activación:

El axón es emulado por la función de activación, cuyo objetivo es obtener la salida final de la neurona mediante un mapeo. En la Tabla 2.4 se muestran algunas de las funciones de activación más habituales en redes neuronales [75].

Tabla 2.4 Funciones de activación.

Nombre	Función
Linea.l	$a = n.$
Sigmoidea.	$a = \frac{1}{1 + e^{-n}}.$
Tangente hiperbólica.	$a = \frac{e^{n} - e^{-n}}{e^{n} + e^{-n}}.$
Softmax.	$a(n) = \frac{e^{n_j}}{\sum_{k=1}^{k} e^{n_k}}.$
Lineal positiva.	$a = 0, \ si \ \ n < 0.$ $a = n, \ si \ \ n \geq 0.$

Función de agregación:

El cuerpo celular o soma es representado por la función de agregación, ver ecuación (2.10). Esta función calcula el potencial de acción al integrar todas las señales recibidas. En otras palabras, consiste en la sumatoria de las entradas multiplicadas por los pesos (sinapsis) [75].

El resultado de esta integración de las señales de enteada se conoce como potencial de acción o campo local inducido [75].

$$b_k + \sum_{i=1}^{n} x_i * w_i. \quad (2.10)$$

Donde:

x_i: Conjunto de entradas a la neurona.

w_i: Conjunto de pesos de la neurona.

b_k: Sesgo.

n: Total de elementos.

2.10.1.3 Neurona recurrente

Las neuronas recurrentes son capaces de simular la capacidad de memorización del cerebro. Pueden recordar valores que han emitido con anterioridad, ver Figura 2.13. Es decir, la neurona recurrente recibe el vector de entradas x de la capa anterior y su propia salida de la iteración o época anterior para generar su salida [75], [76].

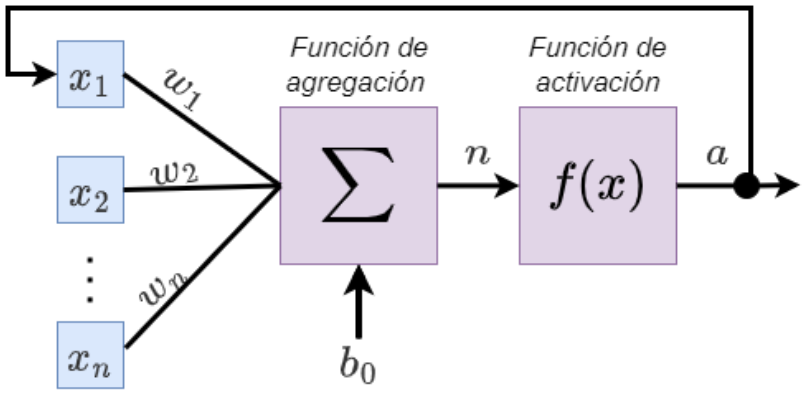

Figura 2.13 Estructura de una neurona recurrente.

2.10.1.4 Red neuronal artificial

Una red neuronal artificial es un conjunto de modelos matemáticos simples que están organizados de tal forma que simulan la estructura y comportamiento de una red neuronal biológica, con la intención de realizar activades de clasificación o regresión. La RNA puede estar compuestas por una capa de entrada, k capas ocultas y una capa de salida. Cada capa puede estar estructurada o integrada por n cantidad de neuronas, y estas capas están interconectadas entre sí [74]–[76].

La estructura fundamental de una red neuronal artificial se muestra en la Figura 2.14, esta estructura se conoce como perceptrón. Las redes neuronales perceptrón no tienen capas neuronales ocultas, por lo que las neuronas no se comunican entre sí. Es decir, cada neurona es una unidad aislada. Mientras que las Figuras 2.15 y 2.16 muestran redes neuronales multicapa. Este otro tipo de arquitectura multicapa que consta de una capa de entrada, k capas ocultas y una capa de salida. En este caso,

las neuronas en las capas ocultas están conectadas entre sí, lo que permite la interacción entre neuronas [74]–[76].

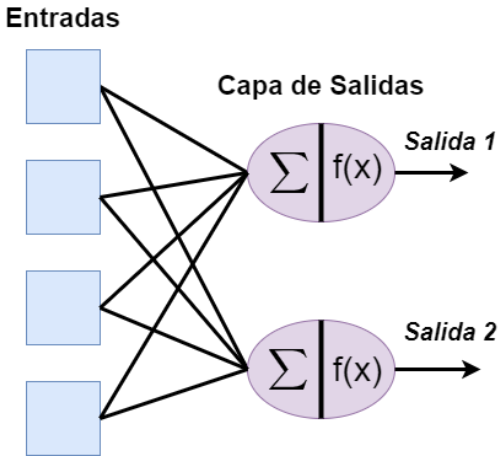

Figura 2.14 Arquitectura de una red neuronal básica.

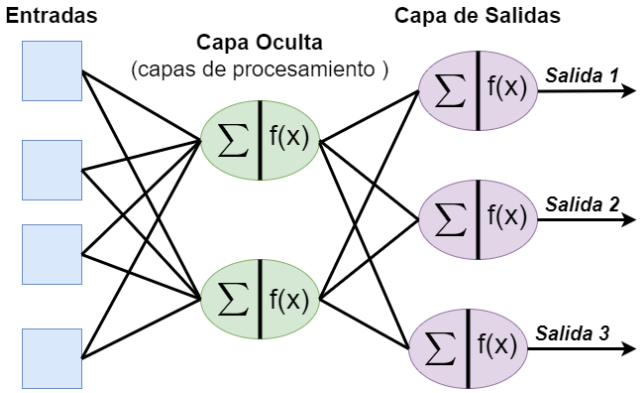

Figura 2.15 Arquitectura de una red neuronal con una capa oculta.

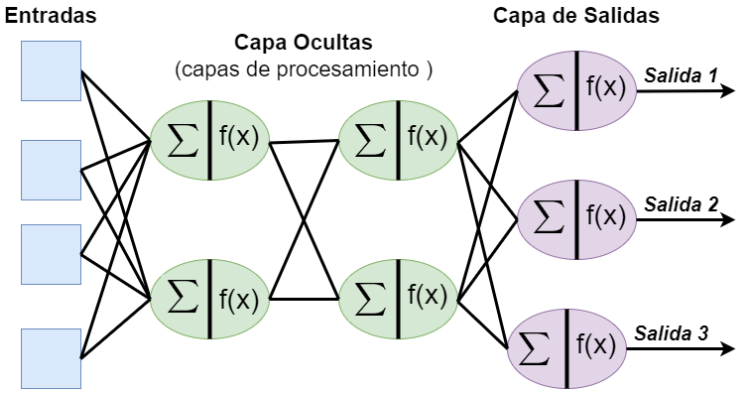

Figura 2.16 Estructura de una red neuronal multicapa.

2.10.2. **KNN**

KNN es un algoritmo de agrupamiento, de aprendizaje supervisado y no paramétrico. La idea de KNN es formar grupos de datos que comparten características similares. Esta similitud de propiedades se obtiene mediante el cálculo de la distancia entre los datos y sus atributos. El modelo de KNN no realiza una fase de entrenamiento como tal, sino que guarda las observaciones agrupadas con sus respectivas variables y distancias [47]. Después, el modelo recalcula (predicción) la distancia de las muestras nuevas con los grupos existentes.

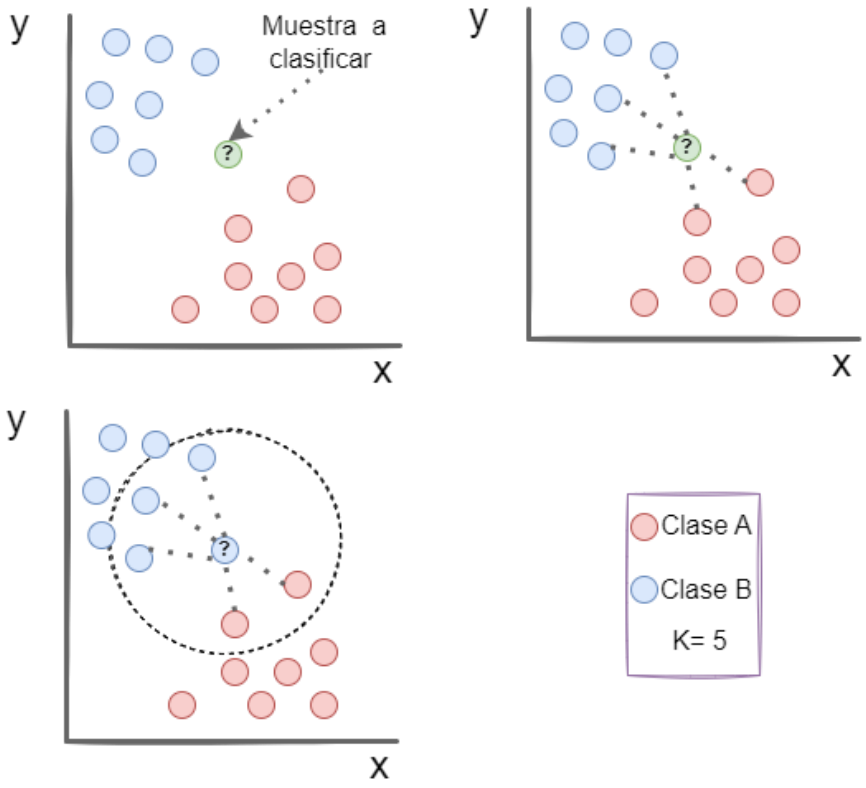

Figura 2.17 Proceso de clasificación de KNN.

La Figura 2.17 [47], [70] presenta de manera gráfica el proceso de clasificación llevado a cabo por KNN después de una fase de “entrenamiento”. El modelo recibe una observación a clasificar y se determina qué muestras de datos están más cerca de ella. Esto se lleva a cabo calculando la distancia entre la nueva observación y todas las muestras existentes. Estas distancias ayudan a identificar la posición de la observación respecto a los diferentes grupos y límites existentes. Después, se seleccionan las k muestras más cercanas y la observación es asignada al grupo o clase que aparece con mayor frecuencia (grupo más cercano) [47], [70].

Para calcular la distancia entre las muestras, se suelen utilizar diferentes tipos de distancias, como la distancia de Minkowski, la distancia de Hamming y la distancia de

Manhattan. Sin embargo, la distancia más utilizada es la distancia Euclidiana, la cual se calcula como se muestra en la ecuación (2.11).

$$d(P_1, p_2) = \sqrt{(x_2 - x_1)^2 - (y_2 - y_1)^2}. \quad (2.11)$$

Donde:

P_1: Punto 1.

P_2: Punto 2.

x_1: Coordenada en x del punto 1.

y_1: Coordenada en y del punto 1.

x_2: Coordenada en x del punto 2.

y_2: Coordenada en y del punto 2.

2.10.3. Máquina de soporte vectorial

SVM es un algoritmo de aprendizaje supervisado que se utiliza solamente para tareas de clasificación. El modelo es capaz de modelar datos linealmente y no linealmente separables. Asimismo, el modelo es eficiente para problemas binarios y multiclase [70], [73].

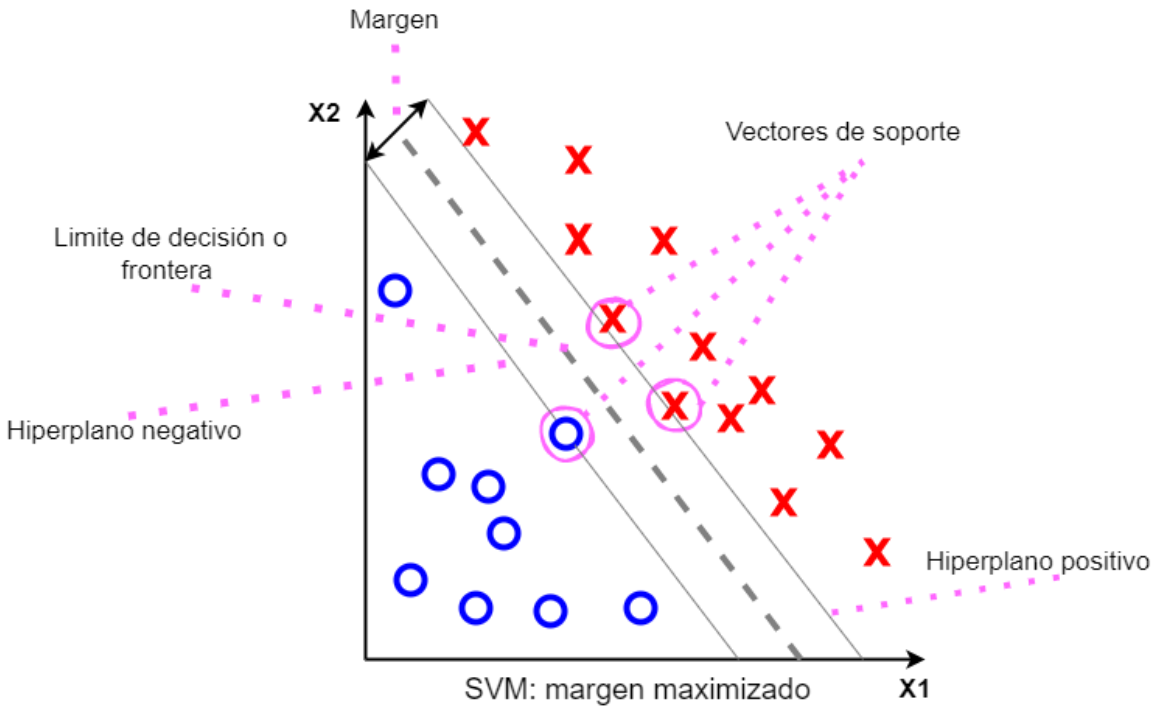

Figura 2.18 Principio de clasificación de SVM.

El modelo de máquina de vectores de soporte agrupa los datos mediante el cálculo de vectores de soporte, divisiones conocidas como fronteras y márgenes, ver Figura 2.18 [70], [73]. Los vectores de soporte son las observaciones más cercanas a la frontera, y las distancias entre ellas se suman para obtener el margen. Por otra parte, el margen es el área del espacio de características sin observaciones que separa las diferentes clases entre sí. Durante cada iteración de la fase de entrenamiento se recalculan los vectores de soporte y los márgenes. El objetivo del proceso de entrenamiento es encontrar las regiones más amplias (márgenes) que permitan separar lo mayor posible a los diferentes grupos de clases [70], [73].

2.10.4. **Arboles de decisión**

Los árboles de decisión son algoritmos de aprendizaje supervisado y no paramétricos que se utilizan para tareas de clasificación y regresión, ver Figura 2.19 [70], [73]. Estos modelos se basan en la toma de múltiples decisiones, y sus componentes son: un nodo raíz, ramas, nodos internos y nodos hoja. Los nodos hojas son las predicciones, y los nodos internos representan las decisiones del árbol [73].

Las diferentes decisiones que toma el modelo se definen con una función llamada función objetivo. La función objetivo permite medir la ganancia de información relevante, y se optimizará durante la etapa de aprendizaje del algoritmo, como se muestra en la ecuación (2.12). Dicho de otro modo, se busca maximizar la ganancia de información significativa en cada división del árbol. Por tanto, los nodos deben conservar la información más útil que permita la toma de decisiones y la discriminación. Finalmente, estas decisiones permiten encontrar las divisiones que forman subconjuntos homogéneos de datos mediante la conservación de la información relevante [73].

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j). \quad (2.12)$$

Donde:

f: Es la característica para definir la división.

D_p: Conjunto de datos del nodo padre.

D_j: Conjunto de datos del nodo hijo.

I: Medida de impureza.

N_p: Número total de muestras del nodo padre.

N_j: Número total de muestras del nodo hijo.

p: Índice nodo padre.

j: Índice nodo hijo.

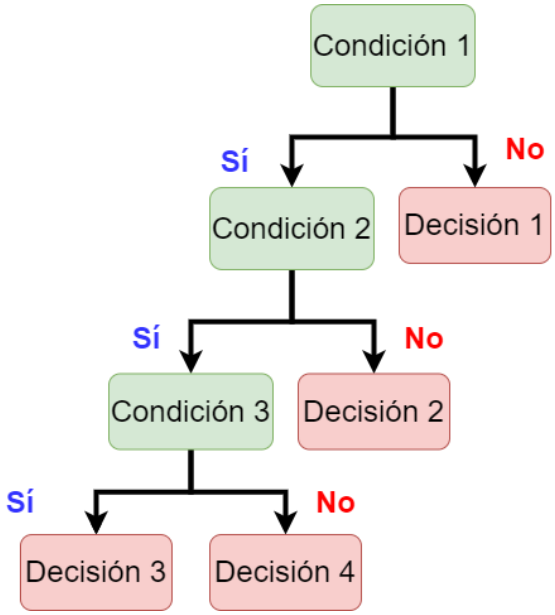

Figura 2.19 Árbol de decisión.

2.10.5. Regresión logística

Regresión logística es un modelo probabilístico de aprendizaje supervisado y multiclase que permite modelar datos que tienen un comportamiento no lineal, como se muestra en la Figura 2.20 [70], [73]. El modelo arroja una serie de resultados en el intervalo $[0, 1]$, los cuales indican la probabilidad de que suceda un evento o de que una muestra pertenezca a un grupo de datos [70], [73].

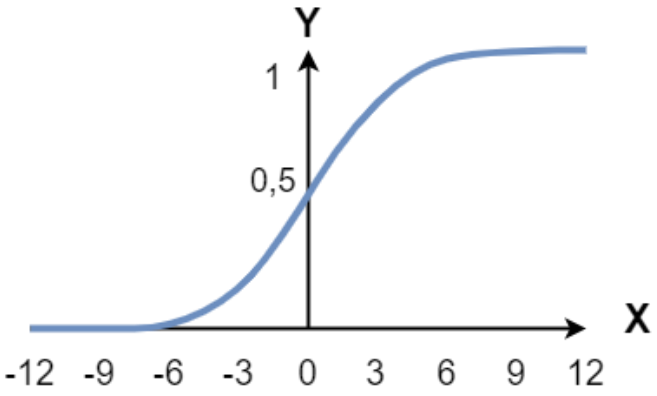

Figura 2.20 Función logística.

Para determinar la probabilidad de que una observación pertenezca a una clase en específico o la probabilidad de que suceda cierto evento [73], se puede utilizar la ecuación (2.13).

$$p(x) = \frac{1}{1 + e^{-x}}. \quad (2.13)$$

Donde:

e: Número de Euler.

x: Valor de la muestra.

2.10.6. Bosques de clasificación

El clasificador de bosques aleatorios es un modelo capaz de aprender por sí mismo y está formado por varios clasificadores basados en toma de decisiones. El modelo se utiliza para tareas de clasificación y predicción, como se muestra en la Figura 2.21 [70]. Los bosques aleatorios están formados por una colección de árboles de decisión, y cada árbol está compuesto por un nodo raíz, ramas, nodos internos y nodos hoja. El proceso de aprendizaje se vuelve más complejo, ya que hay que encontrar la cantidad óptima de árboles de decisión y a su vez encontrar la estructura óptima de cada árbol de decisión.

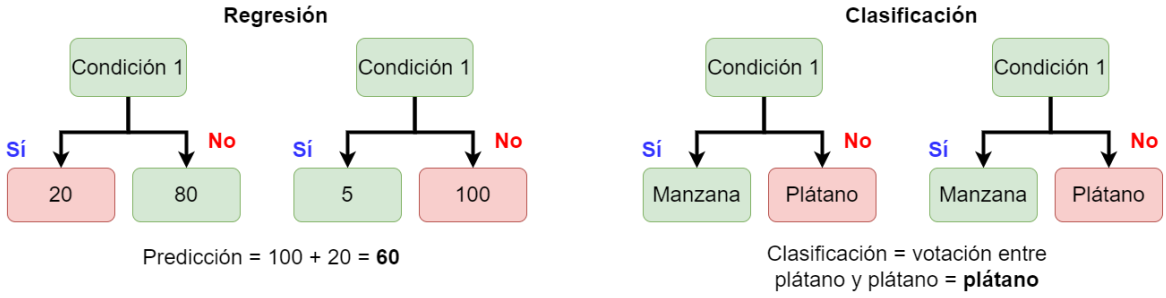

Figura 2.21 Bosques aleatorios.

Capítulo 3. Procesamiento de voz

El procesamiento de voz es un área del procesamiento de señales multidisciplinar que también se asocia a la inteligencia artificial. Se encarga de procesar, comprender, analizar, interpretar y reconocer de forma automática las señales acústicas mediante técnicas de inteligencia artificial, señales y sistemas y herramientas matemáticas [77], [78]. Estas señales de voz son procesadas automáticamente por algoritmos inteligentes con la intención de realizar actividades como reconocimiento de emociones, identificación de eventos, reconocimiento y transcripción de las palabras que una persona pronunció e identificación de la identidad de personas [77], [78].

3.1. Aplicaciones de reconocimiento de voz

La Figura 3.1 muestra que en la disciplina del procesamiento de voz se suelen realizar cuatro tipos de actividades o tareas: 1) identificación del idioma, 2) identificación de locutor y género, 3) reconcomiendo automático de voz y 4) reconocimiento de emociones. La idea en identificación del hablante es determinar la identidad de quién está hablando mientras que el reconocimiento automático de voz consiste en reconocer las palabras pronunciadas por el orador. En reconocimiento de emociones el objetivo es identificar diferentes estados emocionales a partir del tono de voz de la persona, tales estados emocionales son tristeza, alegría, enojo, etc. Finalmente, en identificación de idioma se puede considerar como el proceso de reconocer los idiomas que se hablan en una grabación de audio [77], [79]–[81].

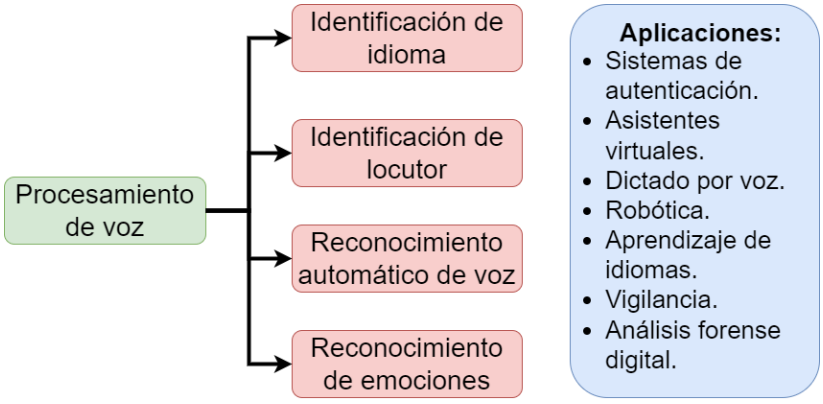

Figura 3.1 Principales diciplinas y aplicaciones del procesamiento de voz.

Como se mencionó anteriormente, el procesamiento de voz es un campo diverso con muchas aplicaciones. Las actividades de identificación de locutor y reconocimiento de voz se aplican en acceso de seguridad controlado por voz, asistentes inteligentes, aplicaciones de dictado por voz, investigaciones criminales y forenses, vigilancia, entre otras [77], [79]–[81].

3.1.1. Vigilancia

La idea es monitorear a las personas para realizar acciones como recolectar información, confirmar su ubicación, obtener pruebas o declaraciones, y verificar el cumplimiento de la ley. Estas personas pueden estar bajo investigación o en libertad condicional. La información se recopila de las conversaciones de dispositivos electrónicos, donde se reconocen las palabras pronunciadas, las personas que participan en la conversación, su estado emocional, sus acciones y su género [82].

3.1.2. Forense

El análisis forense de audio digital es el proceso de adquirir, analizar y evaluar grabaciones de audio que se utilizan como evidencia criminal. Este proceso se utiliza para llevar a cabo una variedad de tareas, incluyendo la verificación de la integridad de la grabación, la mejora de la calidad del audio, y la identificación del hablante. El resultado de este proceso puede ser utilizado para determinar si un sospechoso es culpable o inocente de un crimen [82]–[84].

3.1.3. Autenticación

El objetivo es utilizar propiedades acústicas como un tipo de contraseña biométrica o para verificar la identidad de una persona en actividades de dar acceso a las personas en aplicaciones, casas, automóviles, entre otras cosas. Sin embargo, por sí sola no es una herramienta segura, por lo que se combina con otros métodos de autenticación biométrica (construcción de sistemas de reconocimiento redundantes) [82]–[84].

3.2. Señal de voz

Una señal es un fenómeno o evento físico que contiene información y puede iniciar otros eventos. Se representan por expresiones matemáticas, de manera más específica, por medio de funciones de una o más variables independientes que indican su comportamiento, la cantidad de información que posee, tiempo de existencia y sus propiedades. Todas las señales tienen al menos una dimensión, que es la dimensión del tiempo (dominio) y se denota por t o n. Las señales pueden existir en todos los instantes de tiempo o solo en ciertos instantes de tiempo. Si existen en todos los instantes de tiempo, se les denomina señales en tiempo continuo $x(t)$. Si existen solo

en ciertos instantes de tiempo, se les denomina señales discretas $x[n]$. Las señales se miden o se recolectan por medio de sensores con el objetivo de tener una señal manipulable. Estas pueden ser capturadas por medio de micrófonos, donde la idea es convertir estos fenómenos acústicos a corrientes eléctricas, dicho de otra manera, se transforman a señales eléctricas discretas o continuas [85].

Por otra parte, las señales acústicas son ondas aleatorias que contienen información y una serie de propiedades únicas. Estas variaciones de presión de aire viajan a través del ambiente para transmitir conceptos, estados emocionales, ideas y opiniones. La señal acústica es modelada por una función unidimensional que indica la información contenida en ella y su comportamiento, donde su distribución de probabilidad cambia a través del tiempo (eventos aleatorios). Las ondas de voz son capturadas por medio de micrófonos u otros sensores que miden y convierten estas ondas en corrientes eléctricas y luego a información digital [77], [79]–[81].

3.2.1. Producción fisiológica de la voz

La Figura 3.2 [77], [86] describe el proceso de producción y comprensión del habla en los seres humanos.

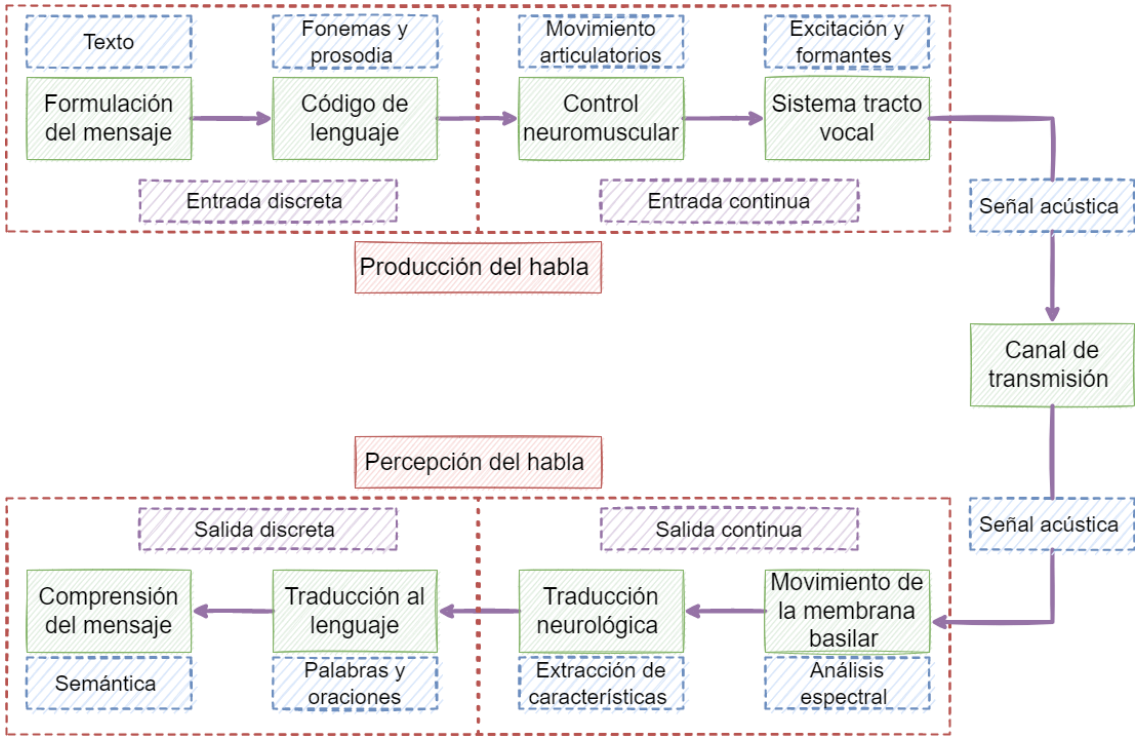

Figura 3.2 Proceso de producción y percepción del habla.

El proceso de producción y comprensión del habla en los seres humanos comienza en el cerebro, donde se forman los conceptos, ideas y emociones que se

van a comunicar. Estas ideas se traducen luego a un idioma específico, es decir, a una serie de palabras mapeadas con sus respectivos sonidos. Las palabras se convierten entonces en movimientos articulatorios, que son llevados a cabo por los órganos de fonación, respiración y articulación. Estas señales neuromusculares permiten controlar el movimiento de los órganos y producir la secuencia de sonidos formulados en el cerebro con sus respectivas propiedades, su orden, sus características únicas y significados. Como resultado, se expulsan variaciones de presión de aire por la cavidad vocal, que se dispersan en forma de ondas por el ambiente. Estas ondas son conocidas como señales acústicas [77], [79]–[81].

El proceso de comunicación (ver Figura 3.2 [77], [86]) continúa con la percepción y comprensión del habla. El sistema auditivo, que actúa como receptor, capta estas señales de voz y las procesa para que puedan ser comprendidas por el cerebro. En esta actividad, intervienen órganos como el tímpano y la membrana basilar, que amplifican y traducen las ondas en señales eléctricas, las cuales describen su amplitud e información de sus diferentes frecuencias. Estas señales eléctricas luego son transmitidas por el nervio auditivo hasta llegar al cerebro. A partir de estas señales eléctricas, se extrae información como la dirección, intensidad y características del sonido, y son traducidas a características espectrales para ser interpretadas por el cerebro humano. Estas propiedades acústicas son mapeadas en fonemas, palabras, expresiones y oraciones de un idioma específico. Finalmente, el cerebro les da una interpretación o significado a estas palabras y oraciones [77], [79]–[81].

El aparato fonético (ver Figura 3.3 [79]) del ser humano es un sistema complejo formado por una gran variedad de órganos que tienen funciones específicas [77], [79], [87].

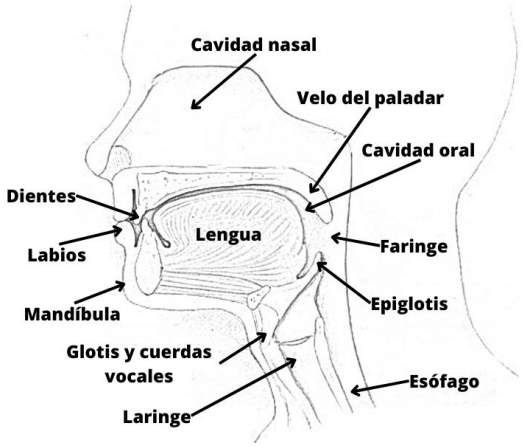

Figura 3.3 Aparato fonético del ser humano.

Los órganos del aparato fonético se pueden dividir en tres tipos: respiratorios, fonatorios y articulatorios:

- Los órganos respiratorios: los pulmones, bronquios y tráquea. Se encargan de adquirir y expulsar el aire que se utiliza para producir los sonidos del

habla [77], [79], [87].

- Los órganos fonatorios: la laringe, las cuerdas vocales y la glotis. Estos órganos se encargan de generar las características particulares de cada fonema y de darle propiedades únicas a la voz de cada individuo. Además, los órganos fonatorios obstruyen el flujo de aire emitido por los órganos de fonación para generar turbulencias o variaciones de presión de aire [77], [79], [81].
- Los órganos articulatorios: los labios, la mandíbula, los dientes, el paladar, la lengua, la faringe, velum, la cavidad oral y la cavidad nasal. Estos órganos también se encargan de añadir características particulares de cada fonema y de transmitirlos por el ambiente [77], [79], [81].

3.2.2. Características de la voz

Las señales de voz presentan diversas características acústicas que son particulares a cada individuo y fonema, y que se utilizan para su modelado. Estas propiedades acústicas incluyen la frecuencia fundamental o energía, la amplitud y la intensidad. La frecuencia fundamental f_0 es representada por el coeficiente de energía y proporciona la percepción del tono de pronunciación. Es decir, es la frecuencia a la que vibran las cuerdas vocales al pronunciar un fonema sonoro, e indica la amplitud, la cantidad de señal (intensidad) y la cantidad de información que transporta la señal. La ausencia o presencia de la frecuencia fundamental puede indicar si el fonema pronunciado es una vocal, consonante o el silencio. Asimismo, es una característica única en cada individuo y es muy útil en la identificación de locutores [79], [86], [87], [88].

La resonancia del tracto vocal emite mucha información y mejora la calidad de la voz. Esta propiedad le da a cada persona una voz distintiva, mejora el timbre, el volumen y la eficiencia de la voz. También permite identificar otras características particulares en cada individuo, como la edad, el género y el estado de salud. Esta característica es muy útil en la identificación de locutores, ya que se puede considerar una propiedad única. Los picos existentes al calcular la envolvente espectral de una señal de voz se conocen como formantes. La resonancia se obtiene definiendo la macro-forma de esta. Los formantes f_1 y f_2 son los nombres de las resonancias más bajas [79], [86], [87], [88].

3.2.3. Representación y estructura del habla

Los fonemas o fonos son las unidades básicas que representan la estructura del habla. Estas secuencias de fonemas se estructuran con un orden para formar sílabas, y una serie de una o más sílabas crean palabras. Los enunciados son finalmente producidos por una o más palabras, que dependiendo del orden y la estructura dan un significado. [77], [79]–[81]. Los fonemas se representan entre diagonales y existen diferentes tipos de fonemas, como se puede apreciar en la Tabla 3.1 [86], [89].

Tabla 3.1 Tipos de fonemas y simbología del español.

Tipo de fonema	Ejemplos:
Vocales sonoras.	/a/, /e/, /i/, /o/, /u/.
Diptongos.	/ja/, /je/, /jo/, /wa/, /we/, /wo/, /ai/, /ei/, /oi/, /au/, /eu/, /ou/, /wi/, /ju/.
Consonantes dentales.	/t/, /d/.
Consonantes interdentales.	/θ/.
Consonantes palatales.	/y/, /ña/, /ll/.
Consonantes alveolares.	/s/, /rr/, /r/, /l/, /n/.
Consonantes labiodentales.	/f/.
Consonantes bilabiales.	/p/, /b/, /m/.
Consonantes velares.	/g/, /j/, /k/.
Otras consonantes.	/c/, /h/, /q/, /ñ/, /v/, /ch/, /x/, /w./.
Fonemas plosivas.	/p/, /f/.
Fonemas no plosivas.	/m/, /l/, /r/.

3.2.4. Percepción auditiva

El oído humano está formado por tres tipos de órganos que permiten procesar las señales acústicas. El primer órgano es el oído externo y lo compone el pabellón auricular, la idea es trasladar el sonido por el conducto auditivo externo hasta el oído medio. El oído medio está estructurado por el tímpano y tres huesos pequeños: el martillo, el yunque y el estribo. Luego, el sonido se convierte en vibraciones, las altas y bajas frecuencias se amplifican. El oído interno es la última parte del oído en la que pasa la señal de voz y está formado por la cóclea, la membrana basilar y el nervio auditivo. La cóclea convierte las vibraciones en señales eléctricas, y la membrana basilar extrae información particular equivalentes a la frecuencia fundamental, amplitud, intensidad y otras propiedades. El nervio auditivo entrega estas señales eléctricas al cerebro. Finalmente, las señales son interpretadas como sonido [77].

3.3. Identificación de locutor

En actividades de procesamiento de voz se extraen características acústicas, como la frecuencia fundamental, de palabras dichas por un orador. Estas propiedades pueden usarse para determinar la identidad del hablante. La disciplina de reconocimiento de locutor tiene solo dos subáreas: identificación de locutor y verificación de hablante (ver Figura 3.4). La identificación de hablante es determinar la identidad de una persona diferenciando su voz de otras personas. Mientras que la verificación de hablante es validar que la identidad del hablante es la que dice ser [90].

En el reconocimiento de locutor de conjunto cerrado, el modelo identifica a un hablante entre un conjunto de n oradores conocidos por él. En el reconocimiento de locutor de conjunto abierto, el clasificador determina si el hablante es conocido por él. Si el modelo es de texto independiente, entonces el locutor puede proporcionar cualquier secuencia de palabras para determinar su identidad. En caso contrario, si el algoritmo es de texto dependiente, entonces el orador debe proporcionar una secuencia de palabras que se le hayan designado o que el modelo conozca.

Finalmente, se puede verificar o identificar la identidad de la persona a partir de su género o sin importar si es hombre o mujer [17], [77], [79], [81].

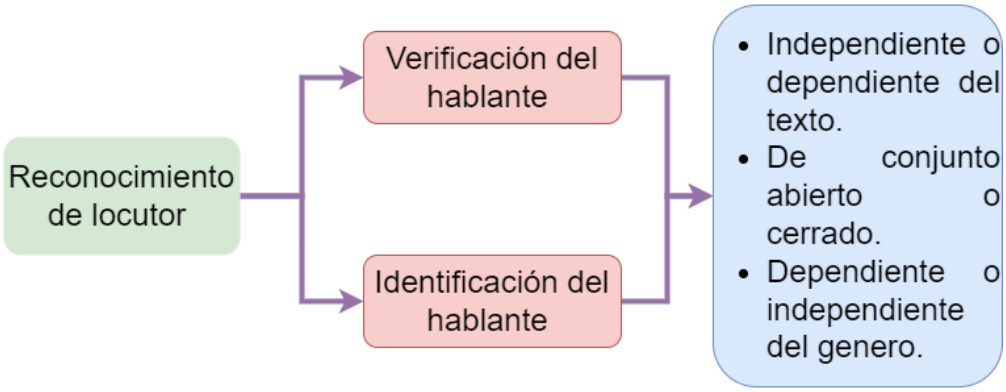

Figura 3.4 Aplicaciones del reconocimiento del hablante.

El marco de trabajo para realizar identificación y verificación del hablante es similar al proceso general de reconocimiento de patrones (ver Figura 3.5). En cada fase, se aplican técnicas especializadas para procesar, manipular y modelar las señales de voz. Las características acústicas del hablante desconocido se comparan con un patrón de referencia de locutores conocidos. Luego, bajo un umbral y una serie de probabilidades asociadas, se verifica la identidad de la persona. El umbral debe tener una alta confiabilidad y ser óptimo [81].

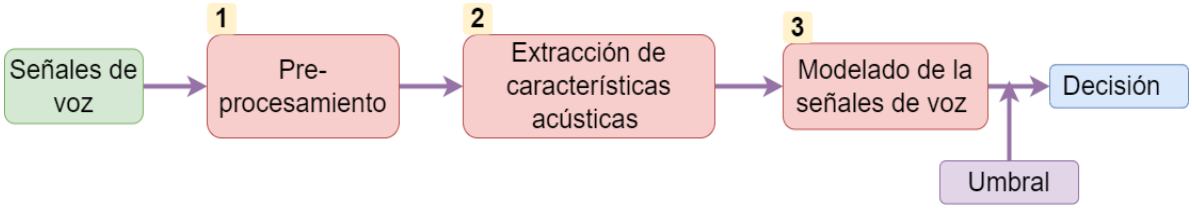

Figura 3.5 Proceso de identificación de locutor.

3.3.1. Pre-procesamiento de la señal de voz

El proceso de mejora de la señal de voz es una técnica de procesamiento que se utiliza para reducir distorsiones, resaltar sonidos de interés, incrementar las altas y bajas frecuencias, verificar la integridad, eliminar el ruido y hacer cancelaciones de eco. Muchas veces, la señal acústica se divide en pequeños fragmentos para representar las propiedades de sonidos específicos o convertir la señal de estacionaria a cuasi-estacionaria [77], [79]. Algunos métodos de extracción de características ampliamente usados son LPC, LPCC y MFCC, que también suelen hacer una fase de pre-procesamiento para adecuar las señales.

3.3.2. Técnicas de extracción de características

Existen muchos métodos de extracción de características específicos para tratar con señales de voz basados en la biología, tales como LPC, LPCC y MFCC. La extracción de características acústicas consiste en obtener información sin ruido, concisa, discriminativa y representativa de la señal de voz con el objetivo de obtener propiedades únicas. Estas técnicas de procesamiento están diseñadas para señales no estacionarias o aleatorias. En decir, el principio básico radica en convertir las ondas de voz en vectores de propiedades. Estos atributos acústicos deben ser lo suficientemente representativos para discriminar una de otras y crear grupos de propiedades similares. [79], [91].

3.3.2.1 LPC

El método de componentes de predicción lineal identifica propiedades del habla emulando la estructura del tracto vocal al producir los sonidos. El análisis LPC se considera un método de combinación lineal de P muestras [15], [92], [93]. Esta combinación lineal se lleva a cabo por medio de la ecuación (3.1).

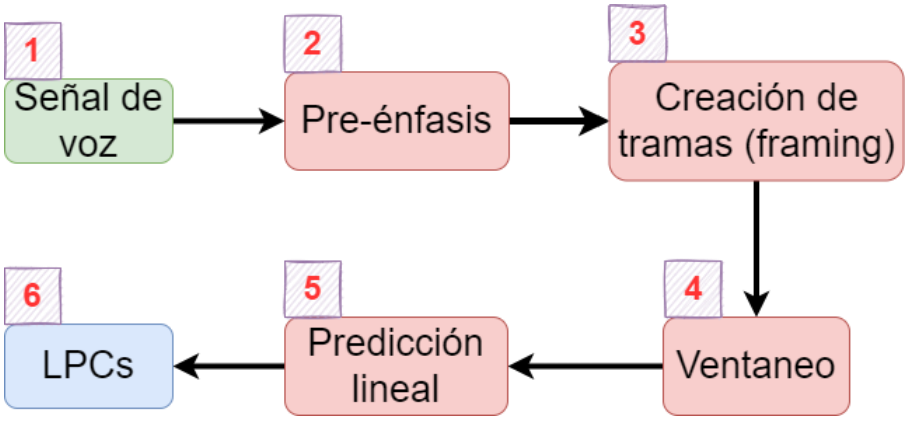

Figura 3.6 Proceso para la extracción de componentes de predicción lineal.

$$s[n] = \sum_{k=1}^{P} a_k s[n-k] + e[n]. \qquad (3.1)$$

Donde:

$e[n]$: Error de la predicción.

a_k: Coeficientes del predictor lineal de orden p.

$s[n-k]$: Las correlaciones de la señal de entrada.

p: Orden de los coeficientes lineales.

$s[n]$: Señal original.

La Figura 3.6 [15], [92], [93] detalla el proceso de análisis LPC. Primero se realiza un pre-énfasis y después la señal de voz se divide en tramas para manejar la aleatoriedad de la señal. El proceso continúa al aplicar ventanas a cada trama para reducir el ruido de la señal y mejorar su calidad. Posteriormente, se realiza un análisis de correlación para encontrar la frecuencia fundamental y el tono de una señal, con esto se logra encontrar la correlación entre la señal en el tiempo actual y anterior. Finalmente, se utiliza el método de Levinson-Durbin para extraer características acústicas lineales de los coeficientes de autocorrelación obtenidos en el paso anterior. Es decir, cada muestra en el tiempo t actual es construida como una combinación lineal de P tramas en el tiempo anterior $t-1$, esto se aprecia en la ecuación (3.1) [15], [92], [93].

3.3.2.2 LPCC

El método de LPCC es una versión mejorada de la técnica LPC. Como se puede ver en la Figura 3.7 [15], [93], se añade un paso extra después de la fase de análisis LPC. Este paso consiste en ejecutar un análisis cepstral para convertir los coeficientes LPC a coeficientes cepstrales. La idea de esto es encontrar el filtro (posiciones de los órganos del tracto vocal) de cada sonido [15], [93].

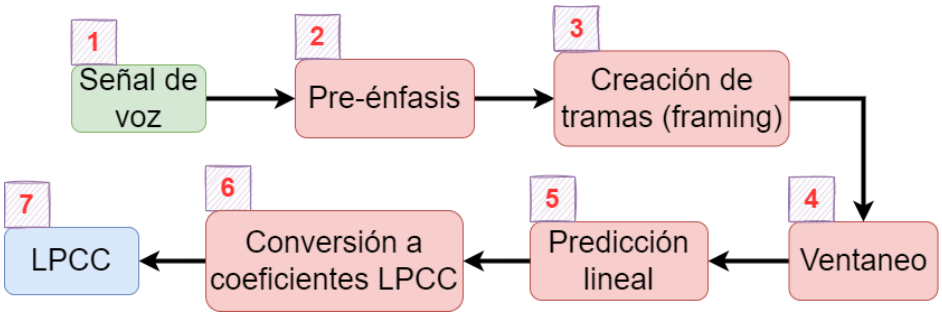

Figura 3.7 Proceso para la extracción de coeficientes cepstrales de predicción lineal.

3.3.2.3 MFCC

La técnica de MFCC emula la manera en que el oído humano percibe los sonidos. Las frecuencias se procesan de forma no lineal para imitar la sensibilidad del oído humano, que es menor a las altas frecuencias [15].

La Figura 3.8 [86] muestra en detalle el proceso para el cálculo de los MFCCs. El cálculo de los MFCCs comienza dividiendo la señal de voz en fragmentos que se traslapan. Esto significa que cada fragmento puede contener información de fragmentos anteriores o posteriores en el tiempo. Estos fragmentos se llaman tramas y tienen una duración de $25\ ms$ con un traslape entre ellas de $10\ ms$. Cada trama es multiplicada por una función de ventana de Hamming. Luego, se aplica la transformada discreta de Fourier (DFT, por sus siglas en inglés) para obtener los coeficientes espectrales. Un grupo de filtros de paso de banda triangulares son aplicados junto con la transformada inversa de Fourier y el logaritmo natural. Este proceso simula la percepción de los sonidos del oído humano, y a los coeficientes

obtenidos se les llaman coeficientes de Mel. A este vector de coeficientes calculados se le añade un coeficiente extra que contiene información de la energía. Finalmente, se calculan las variaciones en el tiempo de estos coeficientes de Mel, que se calculan al obtener la derivada de cada coeficiente cepstral [15].

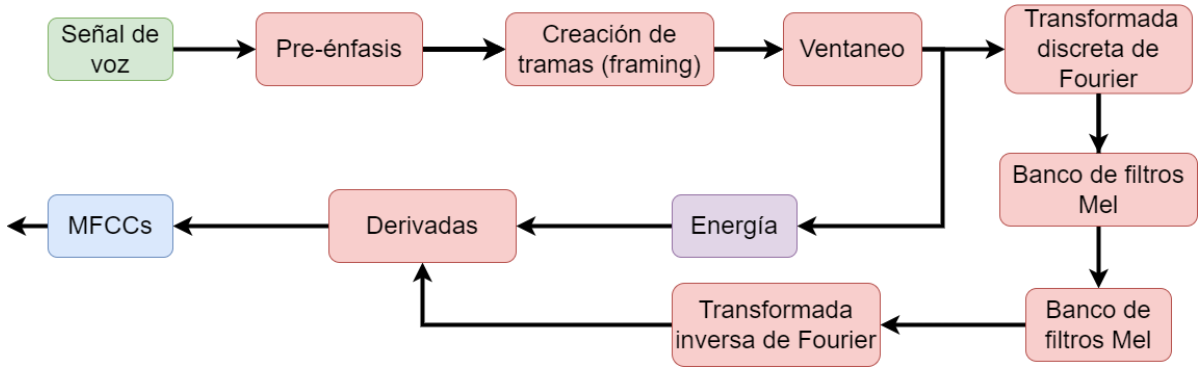

Figura 3.8 Proceso de creación de MFCCs.

3.3.3. Modelado de la señal de voz

En la fase de modelado de las señales, el objetivo es encontrar, reconocer y representar patrones en las muestras acústicas (propiedades). Una vez que se obtiene esta representación de los datos, se pueden llevar a cabo actividades como la identificación y verificación del hablante. Algunas técnicas que permiten realizar este modelado son redes neuronales artificiales, SVM, GMM, MOM, KNN, cuantificación vectorial, regresión logística, entre otras.

3.4. Reconocimiento automático de voz

El reconocimiento de voz permite identificar cuáles palabras han sido pronunciadas y ser convertidas a texto mediante modelos inteligentes [91].

3.4.1. Método clásico del reconocimiento automático de voz

La idea en el reconocimiento automático de voz es capturar las palabras que ha pronunciado alguna persona y, mediante un procesamiento de esta información, obtener una transcripción digital. Comúnmente este proceso se realiza mediante varios algoritmos inteligentes capaces de procesar, analizar, obtener propiedades y modelar señales acústicas para realizar transcripciones de voz a texto o viceversa. De manera más específica, el proceso para el reconocimiento automático de voz se realiza a partir del teorema de Bayes, donde cada componente (probabilidad) es modelado por un clasificador especifico. Esta forma se considera como el método

clásico y se describe en la ecuación (3.2) [77], [86].

$$\widehat{W} = \underset{w}{argmax}\ p(W|O) = \underset{w}{argmax}\ \frac{p(w)\,p(O|w)}{p(O)}. \quad (3.2)$$

Donde:

$p(O \mid W)$: Simboliza el modelo acústico.

$p(W)$: Representa el modelo de lenguaje.

$\widehat{W}$: Secuencia de palabras con mayor probabilidad.

$\underset{w}{argmax}$: Decodificador.

O: Observaciones.

Las señales acústicas son modeladas por un conjunto de características y propiedades acústicas definidas por $O = \{\vec{o}_1, \vec{o}_2, \dots, \vec{o}_T\}$. A partir de estas observaciones y un grupo de palabras posibles $W = \{w_1, w_2, \dots, w_m\}$, el objetivo será encontrar una secuencia de palabras $\widehat{W}$ que sea equivalente a una posible transcripción que está dada por un modelo probabilístico $\underset{w}{argmax}$ [77], [79], [86].

La ecuación anterior se puede reformular como se muestra en la ecuación (3.3).

$$\widehat{W} = \underset{w}{argmax}\ p(W|O) = \underset{w}{argmax}\ p(w)p(O|w). \quad (3.3)$$

Donde:

$p(O \mid W)$: Simboliza el modelo acústico.

$p(W)$: Representa el modelo de lenguaje.

$\widehat{W}$: Secuencia de palabras con mayor probabilidad.

$\underset{w}{argmax}$: Decodificador.

O: Observaciones.

En la ecuación (3.3), el factor $p(O \mid W)$ indica la probabilidad de que una observación acústica de una palabra dicha pertenezca a una serie de palabras conocidas. $p(W)$ se considera la probabilidad de que ocurra una palabra dado conjunto de palabras dichas, y es proporcionada por el modelo de lenguaje. La función $\underset{w}{argmax}$ obtiene la serie de palabras con la mayor probabilidad de ocurrir [77], [79].

La Figura 3.9 [77], [79], [86] describe el proceso general y clásico para la actividad de reconocimiento automático de voz basado en el teorema de Bayes. Primero, la

señal de voz es preprocesada y representada por muestras de n características u observaciones acústicas. Después de este primer paso, se realiza la clasificación y decodificación de cada muestra de voz. En esta fase se involucran otros componentes, como el modelo acústico, el diccionario de pronunciación, el modelo de lenguaje, el clasificador y decodificador [77], [79], [86].

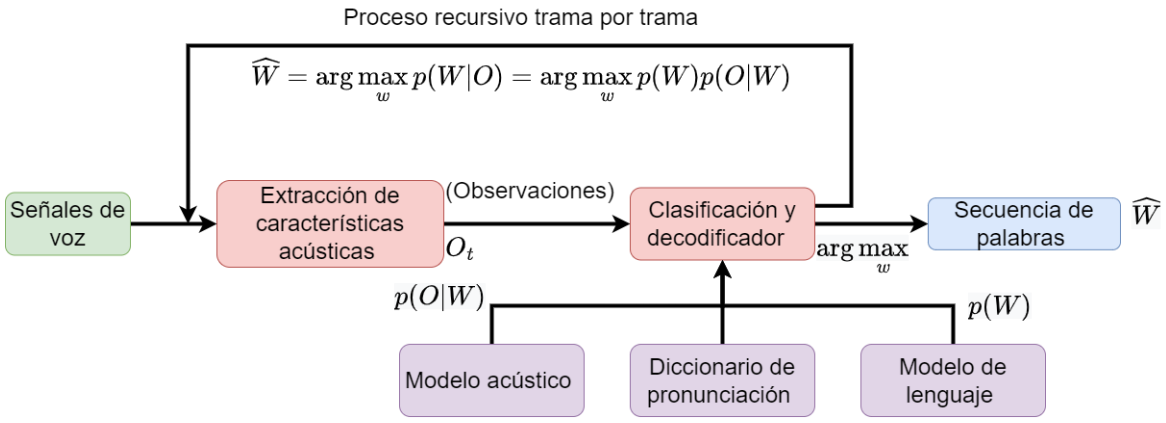

Figura 3.9 Arquitectura general de un modelo de reconocimiento automático de voz.

Por otra parte, el modelo de lenguaje proporciona la probabilidad de que una palabra (clasificación) se pronuncie dada una secuencia de palabras ya dichas (mediante $n-gramas$). Mientras que el modelo acústico proporciona la probabilidad de que una observación acústica forme parte de la estructura fonética de una palabra dicha por el hablante. El diccionario de pronunciación se encarga de mapear cada fonema en una secuencia de fonemas que forman la pronunciación de la palabra. Este diccionario contiene todas las palabras que el sistema de reconocimiento de voz conoce. Por último, el decodificador genera un conjunto de transcripciones candidatas basándose en las probabilidades de ocurrencia. Para seleccionar la mejor solución, cada transcripción candidata pasa por un proceso de clasificación donde se selecciona la palabra que obtuvo la mayor probabilidad de ocurrencia en el proceso de clasificación [77], [79], [86].

3.4.1.1 Modelo acústico

Es una abstracción relacionada en términos de probabilidad de que una serie de sonidos forme parte de la construcción de una palabra, donde las palabras están formadas por un conjunto de fonemas. El modelo acústico consiste en calcular la probabilidad de que una sucesión de observaciones acústicas $O = \{\vec{o}_1, \vec{o}_2, \dots, \vec{o}_T\}$ pertenezcan a una serie de palabras conocidas $W = \{w_1, w_2, .., w_m\}$ [77], [79], [86].

3.4.1.2 Modelo de lenguaje

El modelo de lenguaje representa las probabilidades de que una palabra pueda ser dicha, a partir de un conjunto de palabras pronunciadas. Es decir, es la posibilidad de que ocurra una palabra dado el contexto de conversación [77], [79].

3.4.1.3 Diccionario de pronunciación

Es una lista de palabras que se desean reconocer. A su vez, estas palabras están mapeadas a una representación sonora por medio de los fonemas. [77], [79], [86].

3.4.1.4 Decodificación

El decodificador traduce una secuencia de observaciones acústicas a una serie de palabras. Este proceso comienza al recibir una secuencia de características acústicas y las clasifica según al fonema que representa ese sonido. Para ello, utiliza el modelo acústico, que contiene un mapeo de los posibles sonidos y fonemas. Una vez que los sonidos han sido clasificados, el decodificador busca en el modelo de lenguaje las posibles palabras que estén asociadas a la secuencia de sonidos ya clasificados. Este proceso crea oraciones o una secuencia de palabras que son una posible transcripción de la señal de voz. Finalmente, el clasificador proporciona una secuencia de palabras candidatas $\widehat{W}$ con su respectiva probabilidad para después seleccionar la que tiene la mayor posibilidad de coincidir con la señal procesada [77], [79], [86].

3.4.1.5 Modelado y clasificación de patrones

Como se mencionó anteriormente, el objetivo es encontrar y reconocer patrones en las observaciones acústicas proporcionadas. Algunas técnicas que permiten realizar este modelado basado en el teorema de Bayes fusionados con métodos como RNA, las mezclas gaussianas, MOM y el MMG-MOM.

3.5. Principales estudios relacionados

La identificación del hablante y el reconocimiento automático de voz se utilizan en una variedad de aplicaciones, incluyendo asistentes personales, sistemas de telefonía, sistemas de autenticación e identificación, acceso a instalaciones físicas, transacciones bancarias y análisis forense. Debido a su gran variedad de aplicaciones, existen muchos trabajos relacionados con el reconocimiento e identificación de voz, en los que se intentan unir ambas tareas a través de modelos duales o multitarea. También, estas actividades se han llevado a cabo de forma independiente.

Un ejemplo de trabajo en este campo es el desarrollado por Ravanelli y Bengio [94], donde se presenta una nueva arquitectura de red neuronal convolucional artificial llamada SincNet. Esta red neuronal es un modelo multitarea que es capaz de realizar reconocimiento automático de voz, identificación y verificación de locutor. Se probó en los conjuntos de datos TIMIT (con 462 locutores), LibriSpeech (con $2{,}484$ hablantes) y DIRHA, todos los dataset en el idioma inglés. Se probaron diferentes configuraciones de SincNet, cada una con un método diferente de extracción de características, como MFCC, FBANK y RAW. SincNet-Raw obtuvo los mejores resultados, con una precisión de CER (o tasa de error de caracteres) del 0.96% en identificación de voz y una precisión de ERR (o tasa de error) del 0.51% en verificación de locutor en el conjunto de datos LibriSpeech. También obtuvo una precisión de WER del 37.2% en el corpus de datos DIRHA.

Los investigadores Tang et al. [34] diseñaron un clasificador recurrente multitarea para reconocimiento de voz y hablantes. Este esquema unificado realiza ambas tareas de manera simultánea empleando dos redes neuronales recurrentes con capas LSTM (Long Short-Term Memory), donde la salida de cada modelo sirve como información adicional para el otro modelo. Como técnicas de extracción de características se usaron MFCC, FBANKS (o coeficientes cepstrales de frecuencia Mel) e $i-vectores$. Por otro lado, se implementaron LDA y PLDA (o análisis discriminante lineal probabilístico) con el propósito de reducir la dimensión de los vectores de propiedades. El algoritmo presentado obtuvo un WER de 7.41% (promedio) en los diferentes corpus de datos en tareas de reconocimiento de voz. Mientras que en tareas de reconocimiento de locutor obtuvo un ERR del 0.57% al usar la técnica de $i-vectores$ y PLDA.

El trabajo publicado por Zhao et al. [30] propone un clasificador End-to-End multitarea para el reconocimiento de voz en el idioma tibetano. Este modelo evita el uso del diccionario de pronunciación y la segmentación de palabras para nuevos dialectos. También, este modelo permite entrenar tres tareas (de procesamiento de voz) simultáneamente con un solo modelo, ya que comparte componentes como el clasificador y método de representación de características acústicas. El nuevo esquema multitarea sugerido tiene el objetivo de realizar identificación de locutor, reconocimiento de voz e identificación de dialecto.

Andra y Usagawa [95] crearon un sistema de transcripción e identificación de hablantes para el idioma indonesio usando RNN y una arquitectura End-to-End. El sistema primero identifica los segmentos de voz de cada locutor (diarización), luego identifica el orador y transcribe el habla a texto. La separación del habla se realiza mediante una RNA de clustering profundo, mientras que el proceso de transcripción e identificación del hablante se realiza con un modelo RNN y End-to-End, respectivamente. El sistema obtuvo una tasa de error de palabra o WER promedio de 16.59%. Por otro lado, la investigación presentada por Squartini et al. [96] logra un reconocimiento de voz y locutor en escenarios ruidosos a través de la ecualización de histograma multicanal. Se implementaron diferentes clasificadores, como el MOM, GMM-UBM (GMM - Universal Background Model) y GMM, los cuales fueron entrenados con MFCCs.

En el artículo de Lu et al. [97], proponen un algoritmo de reconocimiento de voz y locutor de múltiples hablantes para aplicaciones de streaming. Los FBANKs fueron empleados para alimentar diferentes clasificadores, como una red neuronal convolucional con capas LSTM y un modelo End-to-End basado en una RNN-T (o transductor de red neuronal recurrente). Asimismo, en esta investigación se propone un nuevo enfoque llamado SURIT (Streaming Unmixing, Recognition and Identification Transducer). Bajo este nuevo enfoque, se consigue un WER de 10.1% y un SER (Symbol Error Rate) del 6.7%. Kanda et al. [98] desarrollaron un modelo para unir el recuento de hablantes, el reconocimiento de voz y la identificación de hablantes en el habla superpuesta de cualquier número de hablantes. El esquema SA-ASR (Speaker-Attributed Automatic Speech Recognition) sugerido está basado en una arquitectura End-to-End que unifica las diferentes tareas de procesamiento de voz en un solo modelo. El clasificador también aprende con un entrenamiento de

salida serializada (SOT, por sus siglas en inglés) utilizando un Mel-filterbank.

En Shafey et al. [99] presentan un enfoque novedoso para abordar el reconocimiento de voz y la diarización del hablante mediante un sistema dual. Este sistema une ambas tareas en un solo modelo y utiliza una RNN-T como clasificador. Se evaluó el desempeño de su sistema en un corpus robusto de conversaciones médicas entre médicos y pacientes. Este nuevo enfoque mejoró la tasa de error de diarización a nivel de palabra (WDER, por sus siglas en inglés) de un 15.8% a un 2.2%. Finalmente, es importante señalar que se utilizan señales tanto lingüísticas como acústicas. En la misma línea, Mao et al. [100] realizaron reconocimiento de voz y diarización de múltiples hablantes en audios con conversaciones largas, utilizando un algoritmo inteligente multitarea que une la fase de aprendizaje para aprender las dos tareas de manera conjunta. El modelo presentado por los investigadores está basado en un esquema End-To-End.

En el trabajo publicado por Manurung et al. [23] construyeron un prototipo de aplicación de software para el reconocimiento de hablantes en el análisis de audio forense digital. El prototipo implementa un clasificador de red neuronal de cuantificación vectorial de aprendizaje (NN-LVQ, por sus siglas en inglés), el cual obtiene una exactitud de clasificación del 73.33%. Adicionalmente, el método de MFCC fue usado para la extracción de características acústicas. En el estudio de S. Singh y Singh [24] llevan a cabo un reconocimiento forense de hablantes independiente del idioma a partir de la agrupación de expresiones. Las relaciones entre las expresiones fueron representadas por MFCC. Los algoritmos de agrupación empleados en esta investigación son K-medoid, Gustafson y Kessel, Fuzzy C-means y Gath-Geva. Los resultados obtenidos fueron una exactitud del 95.2%, 97.3%, 98.5% y 99.7%, y un EER del 3.62%, 2.91%, 2.82% y 2.61%, respectivamente.

Los autores Kakade y Salunke [101] implementaron un sistema de reconocimiento e identificación de voz en ambientes ruidosos y en tiempo real. Este sistema se aplica con propósitos forenses y de seguridad. Para entrenar los modelos inteligentes, se utiliza un conjunto de datos en inglés y personalizado, el cual es independiente del texto y de conjunto abierto. En esta investigación, primero se ejecutó una fase de pre-procesamiento y extracción de características mediante el método de MFCC. Posteriormente, los clasificadores implementados fueron cuantización vectorial (VQ, por sus siglas en inglés) y DTW (o deformación dinámica del tiempo. El sistema obtuvo una exactitud del 82% en tareas de identificación de locutor. En ese mismo sentido, Naveen M y Ponraj [102] realizan un reconocimiento de voz con identificación de género y diarización del hablante mediante el algoritmo GMM. Es importante señalar que las señales de voz fueron representadas a partir de la obtención de los MFCCs. Como resultado, el sistema sugerido logró una precisión del 92% en el reconocimiento de voz.

Por otra parte, Tiwari et al. [103] desarrollaron un asistente virtual para el control y la programación de tareas en el hogar. El asistente virtual es capaz de reconocer comandos de voz para controlar o monitorear dispositivos en un hogar. También, realiza reconocimiento automático de locutor y reconocimiento automático de voz. Los MFCCs se utilizaron como vector de características. Mediante el método de VQ y el análisis de componentes principales se hizo una reducción de la dimensionalidad de

este vector de características. El asistente virtual, en identificación de locutor, es capaz de alcanzar una precisión superior al 92% utilizando GMM como clasificador. Cabe señalar que el reconocimiento automático de voz fue llevado a cabo con herramientas creadas por Google.

Finalmente, Rajasekhar y Hota [104] hicieron un estudio de reconocimiento de voz, locutor y de las emociones mediante el uso de MFCCs. Las emociones por identificar fueron ira, felicidad, miedo y tristeza. La base de datos se actualizó a medida que avanzaba el estudio. Finalmente, el algoritmo de SVM fue empleado para realizar las distintas tareas de reconocimiento y procesamiento de voz. Se muestra que el modelo de SVM puede ser eficaz para actividades de procesamiento de voz.

Tabla 3.2 Resumen de las técnicas utilizadas en los trabajos relacionados.

Trabajo/ técnicas	Métodos de extracción de características	Técnicas de reducción de características	Modelos de clasificación
Ravanelli y Bengio [94].	MFCC, FBANK y RAW.	No aplica.	Modelo SincNet, DNN y CNN.
Tang et al. [34].	MFCC, FBANKS y i-Vectores.	LDA y PLDA.	RNN con capas LSTM.
Zhao et al. [30].	MFCC.	No aplica.	Modelo End-to-End (WaveNet-CTC).
Andra y Usagawa [95].	LPC y MFCC.	No aplica.	RNA, RNN con capas LSTM y arquitectura End-to-End.
M & Ponraj [102].	MFCC.	No aplica.	GMM.
Squartini et al. [96].	MFCC.	No aplica.	HMM, GMM y GMM-UBM.
Lu et al. [97].	FBANK.	Convoluciones.	CNN con capas LSTM, RNN-T y arquitectura End-to-End.
Kanda et al. [98].	Mel-filterbank.	No aplica.	CNN, capas LSTM, RNN y arquitectura End-to-End.
Shafey et al. [99].	Log Mel-filter bank energies.	No aplica.	RNN-T con arquitectura CTC.
Mao et al. [100].	Time-Depth Separable (TDS) Convolution.	No aplica.	Arquitectura End-to-End.
Manurung et al. [23].	MFCC.	No aplica.	Learning Vector Quantization Neural Network (JST-LVQ).
S. Singh y Singh [24].	MFCC.	No aplica.	Modelos de clasificación: K-medoid, Fuzzy C-means, Gustafson y Kessel, Gath-Geva clustering. y GMM.
Kakade y Salunke [101].	MFCC.	No aplica.	VQ y DTW.
Tiwari et al. [103].	MFCC.	PCA.	GMM.
Rajasekhar y Hota [104].	MFCC.	No aplica.	SVM.

La Tabla 3.2 presenta un resumen de las principales técnicas de extracción de características, reducción de dimensionalidad y modelos de clasificación utilizados en los trabajos relacionados ya mencionados. Se puede analizar que en el área del procesamiento de voz se dejan de lado métodos importantes como la selección y fusión de características. Estas técnicas pueden ayudar a tener una mejor representación de la información. También, técnicas como la normalización, que ayudan a reducir complicaciones en los datos con ruido, no se usan.

3.5.1. Contribuciones y limitaciones de estudios relacionados

A continuación, se describen las principales contribuciones y limitaciones de los trabajos relacionados, ver Tabla 3.3.

Tabla 3.3 Contribuciones y limitaciones de los trabajos relacionados.

	Contribución	Limitación
R Ravanelli y Bengio [94].	Se presenta una nueva arquitectura de CNN (SincNet) que está basada en un esquema End-to-End. Este modelo es más rápido computacionalmente y presenta mayor precisión que una CNN convencional.	El clasificador propuesto no realiza ninguna fase de selección de características previa. Tampoco se realiza una comparación de tiempo sobre el rendimiento del modelo propuesto contra trabajos relacionados.
Tang et al. [34].	Este artículo diseña un algoritmo unificado para realizar reconocimiento de voz y reconocimiento de locutor simultáneamente.	El corpus de datos utilizado contiene pocos hablantes (clases). También, no se realizan fases de normalización, selección ni fusión de características.
Zhao et al. [30].	Se desarrolla un esquema End-to-End multitarea que evita el uso del diccionario de pronunciación. Para llevar a cabo las diferentes tareas, el modelo comparte componentes como el clasificador y el método de representación de propiedades acústicas.	Este trabajo no se lleva a cabo sobre grabaciones de audio con ruido o de mala calidad. Tampoco se realiza una fase de selección de características. Finalmente, el conjunto de datos utilizado contiene pocas clases y archivos de audio.
Andra y Usagawa [95].	Esta investigación sugiere un sistema mejorado para generar transcripciones e identificar hablantes en Bahasa Indonesia mediante la diarización.	Se usa un clasificador para cada tarea de procesamiento de voz, lo cual es más costoso computacionalmente. Tampoco se realiza ninguna fase de selección de atributos para mejorar el rendimiento de los modelos.
M y Ponraj [102].	Se propone un modelo de mezcla Gaussiana para identificar el hablante y su género de manera conjunta.	No se realiza ninguna fase de selección ni fusión de características para mejorar el rendimiento de los modelos.
Squartini et al. [96].	La presencia de canales acústicos multicanal se utilizan para mejorar las capacidades de modelado estadístico en tareas de reconocimiento e identificación de voz.	El dataset contiene pocas clases. Tampoco se presenta una comparación del tiempo de ejecución del modelo propuesto contra trabajos relacionados. Finalmente, el conjunto de datos es muy pequeño.

Tabla 3.4 Contribuciones y limitaciones de los trabajos relacionados (continuación).

	Contribución	Limitación
Shafey et al. [99].	Presentan un enfoque novedoso para abordar el reconocimiento de voz y la diarización del hablante mediante un sistema dual y una RNN-T.	No se presenta una comparación del desempeño y tiempo de ejecución entre el modelo propuesto y trabajos relacionados.
Mao et al. [100].	Se realiza un reconocimiento de voz y una diarización de múltiples hablantes mediante un algoritmo multitarea que combina la fase de aprendizaje para aprender las dos tareas de manera conjunta.	El estudio no se enfoca en el procesamiento de grabaciones de audio con ruido o de mala calidad.
Manurung et al. [23].	Se diseña un prototipo de aplicación para el reconocimiento de hablantes en el análisis de audio forense digital. Esto se lleva a cabo con el clasificador de red neuronal de cuantificación vectorial de aprendizaje (NN-LVQ).	El dataset utilizado solo contiene 5 hablantes (clases) y pocos archivos de audio, por lo que es difícil medir el rendimiento real del prototipo propuesto.
S. Singh y Singh [24].	Este estudio examinó métodos de agrupamiento para mejorar la medición de la similitud entre pronunciaciones en la identificación de hablantes.	El clasificador no se evaluó exhaustivamente para su aplicación en múltiples contextos y en la vida real.
Kakade y Salunke [101].	Implementaron un sistema de reconocimiento e identificación de voz en ambientes ruidosos y en tiempo real. Este sistema se aplicó con propósitos forenses y de seguridad.	La aplicación de software desarrollada no se evaluó exhaustivamente para su aplicación en escenarios de diferentes regiones, idiomas y acentos.
Tiwari et al. [103].	Desarrollaron un asistente virtual para el control y la programación de tareas en el hogar. El asistente es capaz de identificar al orador y reconocer comandos de voz para controlar o monitorear dispositivos en un hogar.	El conjunto de datos es poco robusto, es decir, contiene pocos datos y clases. Los resultados de exactitud en la clasificación de locutor son muy bajos para que el modelo pueda ser validado y aplicado en escenarios críticos y reales.
Rajasekhar y Hota [104].	Realizaron un estudio de reconocimiento de voz, identificación de locutor y reconocimiento de emociones utilizando MFCCs y SVM.	Los autores no presentan los resultados del rendimiento del modelo sugerido en las diferentes tareas de clasificación.
Kanda et al. [98].	Se propone un esquema que unifica las diferentes tareas de procesamiento de voz en un solo modelo, como el recuento de hablantes, el reconocimiento de voz y la identificación de hablantes.	No se realiza una comparación del tiempo de ejecución y del rendimiento del modelo propuesto contra trabajos relacionados. Además, el conjunto de datos no muestra información con diferentes tipos de ruido.
Lu et al. [97].	Se diseña un enfoque novedoso denominado Streaming Unmixing Recognition and Identification Transducer (SURIT) para llevar a cabo de manera conjunta las tareas de reconocimiento de locutor y habla.	El modelo no se evalúa de manera exhaustiva para su aplicación en escenarios reales. Finalmente, el corpus de datos no contempla diferentes tonos, acentos y estados emocionales.

3.5.2. Comparación entre los trabajos relacionados y la propuesta de investigación

En la Tabla 3.5 se muestra una matriz comparativa de las características, contribuciones y alcance entre los trabajos relacionados y el trabajo propuesto.

Tabla 3.5 Comparativa entre los trabajos relacionados y el trabajo propuesto.

Variables/trabajo	Nuestro modelo	Ravanelli & Bengio [94].	Rajasekhar & Hota [104].	Tang et al. [34].
Año de publicación.	2023	2019	2018	2016
Propone una nueva función de activación.	Sí	No	No	No
Identificación de locutor.	Sí	Sí	Sí	Sí
Corpus de datos en español.	Sí	No	No	No
Con propósitos forenses.	Sí	No	No	No
Aplicado a ambientes ruidosos.	Sí	Sí	Sí	No
Combinación de características.	Sí	No	No	No
Uso y obtención de características estadísticas.	Sí	No	No	No
Propone un nuevo esquema de RNA.	Sí	Sí	No	No
Selección/fusión de características.	Sí	Sí	No	No

Tabla 3.6 Comparativa entre los trabajos relacionados y el trabajo propuesto (continuación).

Variables/modelo	Zhao et al. [30].	Andra & Usagawa [95].	Tiwari et al. [103].	M & Ponraj [102].
Año de publicación.	2019	2021	2018	2020
Propone una nueva función de activación.	Sí	No	No	No
Identificación de locutor.	Sí	Sí	Sí	Sí
Corpus de datos en español.	No	No	No	No
Con propósitos forenses.	No	No	No	No
Aplicado a ambientes ruidosos.	No	Sí	Sí	Sí
Combinación de características.	No	No	No	No
Uso y obtención de características estadísticas.	No	No	No	No
Propone un nuevo esquema de RNA.	No	No	No	No
Selección/fusión de características.	No	No	No	No

Tabla 3.7 Comparativa entre los trabajos relacionados y el trabajo propuesto (continuación).

Variables/modelo	Squartini et al. [96].	Lu et al. [97].	Kakade & Salunke [101].	Kanda et al. [98].
Año de publicación.	2012	2021	2019	2020
Propone una nueva función de activación.	No	No	No	No
Identificación de locutor.	Sí	Sí	Sí	Sí
Corpus de datos en español.	No	No	No	No
Con propósitos forenses.	No	No	Sí	No
Aplicado a ambientes ruidosos.	Sí	Sí	Sí	No
Combinación de características.	No	No	No	No
Uso y obtención de características estadísticas.	No	No	No	No
Propone un nuevo esquema de RNA.	No	No	No	No
Selección/fusión de características.	No	No	No	No

Tabla 3.8 Comparativa entre los trabajos relacionados y el trabajo propuesto (continuación).

Variables/modelo	Shafey et al. [99].	Mao et al. [100].	S. Singh & Singh [24].	Manurung et al. [23].
Año de publicación.	2019	2020	2020	2018
Propone una nueva función de activación.	No	No	No	No
Identificación de locutor.	Sí	Sí	Sí	Sí
Corpus de datos en español.	No	No	No	No
Con propósitos forenses.	No	No	Sí	Sí
Aplicado a ambientes ruidosos.	No	No	Sí	Sí
Combinación de características.	No	No	No	No
Uso y obtención de características estadísticas.	No	No	No	No
Propone un nuevo esquema de RNA.	No	No	No	No
Selección/fusión de características.	No	No	No	No

Capítulo 4. Método y propuesta de investigación

En esta sección se presenta la descripción del método y la propuesta de investigación. Asimismo, se detalla el marco de trabajo para llevar a cabo el modelado de las señales acústicas mediante modelos inteligentes.

4.1. Descripción de la propuesta

Se presenta un modelo de reconocimiento de locutor basado en un esquema de red neuronal artificial que implementa nuevas unidades de procesamiento de soporte (llamadas neuronas de soporte) y funciones de activación paramétricas. El objetivo de estas nuevas funcionalidades es permitir que las redes neuronales artificiales sean capaces de optimizar de manera automática los parámetros de las funciones de activación paramétricas y presenten un mejor rendimiento en el modelado de datos ruidosos. El esquema propuesto es capaz de reconocer con alta confiabilidad y precisión la identidad de la persona que pronunció una determinada expresión en grabaciones de audio con diferentes tipos de calidad de sonido y diversos tipos de ruido ambiental en escenarios forenses.

También se desarrollan dos variantes simplificadas de la función DPReLU (o unidad lineal rectificada paramétrica dinámica) y basadas en ReLU, con la finalidad de reducir el costo computacional (cálculos matemáticos) al optimizar únicamente las pendientes requeridas de la función DPReLU mediante parámetros entrenables.

4.2. Modelo o esquema general de investigación

Según Hernández Sampieri et al. [50], el método científico es un conjunto de pasos ordenados, repetibles y empíricos que se aplican para comprender un fenómeno, responder alguna interrogante y dar solución a un problema. El método científico puede tener distintos tipos enfoques como cuantitativo, cualitativo y mixto. Además, cada enfoque de investigación se puede abordar con distintos diseños de investigación. Es decir, que los enfoques de investigación pueden ser guiados por diferentes métodos o estrategias para probar las hipótesis, obtener información, responder preguntas, analizar datos y responder al planteamiento de problema [49].

La Figura 4.1 [50] presenta el proceso del enfoque cuantitativo propuesto por Hernández Sampieri et al. [105], el cual es un proceso sistemático y demostrativo. Las fases en este método se realizan de manera ordenada, secuencial y no se pueden omitir, aunque el orden es estricto, es posible modifícalo o adaptarlo.

Este trabajo de investigación ha sido desarrollado siguiendo el método científico con un enfoque cuantitativo, debido a que se utilizan herramientas matemáticas para analizar, describir, explicar y modelar señales de voz mediante variables numéricas. Este proceso es realizado aplicando un plan de diseño cuantitativo de tipo experimental puro y transaccional. Se sigue un plan experimental porque se manipulan las variables independientes (características acústicas), para después analizar los efectos de esta manipulación en el desempeño del modelo propuesto al clasificar las categorías (variable dependiente) de las señales de voz. Adicionalmente, la investigación se considera experimental de tipo puro dado que la manipulación de variables es realizada en dos grupos de datos (conjunto de entrenamiento y validación). Lo anterior tiene el propósito de comparar el rendimiento obtenido por el modelo de clasificación sobre estos dos subconjuntos de datos. También se puede considerar de tipo transaccional porque la recolección de datos solo se realiza una única vez en un tiempo determinado.

Por otra parte, este estudio es considerado de tipo descriptivo porque se busca recolectar muestras de voz (datos) para encontrar y especificar las propiedades (características acústicas) que mejor describen a estas señales acústicas, lo que permite modelar las señales de la manera más adecuada.

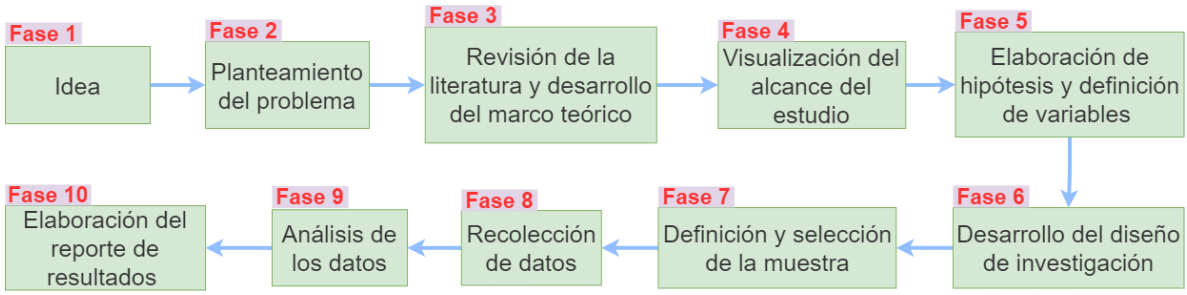

Figura 4.1 Método de investigación con enfoque cuantitativo.

4.3. Marco de trabajo propuesto

La Figura 4.2 proporciona una descripción detallada y secuencial de los procedimientos del marco de trabajo seguido para el procesamiento, análisis y modelado de las señales de voz (datos).

En el primer paso de este marco de trabajo, se realiza el pre-procesamiento de las señales acústicas que incluye la obtención, integración y limpieza inicial de los datos. Luego, se lleva a cabo una transformación de los datos para su adecuado uso. El procesamiento también implica la reducción, extracción, fusión y selección de características, así como la división del conjunto de datos en subconjuntos de

entrenamiento y prueba. El objetivo es mantener la información relevante y preparar los datos para su modelado. También, se busca mejorar la calidad de la señal de voz al reducir las distorsiones y eliminar el ruido.

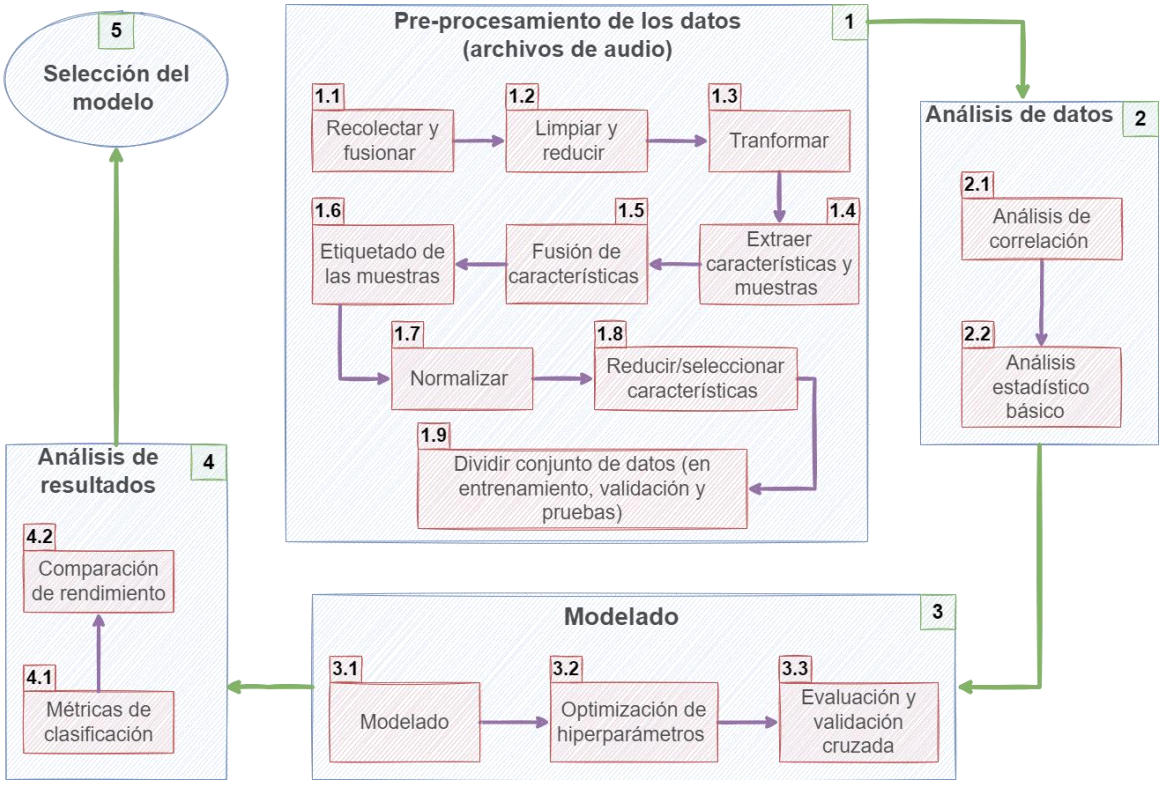

Figura 4.2 Marco general de trabajo.

La segunda fase se enfoca en el análisis de los datos para resumir las propiedades de cada variable, identificar relaciones entre variables y comprender su comportamiento para crear características estadísticas. Este análisis y proceso de creación de nuevas variables se realiza mediante técnicas matemáticas básicas y el estudio de correlaciones. En la fase 3, se aplican diferentes esquemas de redes neuronales artificiales. Para este proceso de modelado, se normalizan los datos, se optimizan los hiperparámetros y se evalúa el rendimiento del modelo utilizando validación cruzada. Este procedimiento se repite para todos los modelos y sus diversas configuraciones (experimentos).

Posteriormente, en la fase 4 se lleva a cabo la optimización, comparación y análisis de los resultados obtenidos por las diferentes configuraciones de modelos. Finalmente, basándose en el análisis de los resultados, se selecciona la configuración óptima o el modelo con el mejor rendimiento y se propone como solución en la fase 5. Aunque el marco de trabajo se propone como secuencial, es posible regresar a fases anteriores y convertirse en un proceso iterativo o cíclico en cada subfase. Es decir, no es estrictamente un marco de trabajo secuencial.

4.4. Arquitectura del modelo de actividades reconocimiento de locutor

En la Figura 4.3 se muestra la arquitectura del modelo general propuesto para la identificación de locutor. Se ha diseñado un modelo más robusto al incorporar las fases de combinación y selección de características acústicas. La combinación de diferentes métodos de extracción de características acústicas proporciona información adicional que permite obtener una representación de datos más precisa y reducir las tasas de error del modelo. Con relación a esto, se ha logrado mantener la simplicidad y minimizar el costo computacional al integrar en un solo modelo las etapas de extracción, fusión, selección, normalización y clasificación de características. Esto permite realizar todas estas tareas de manera conjunta y secuencial en la identificación de locutor.

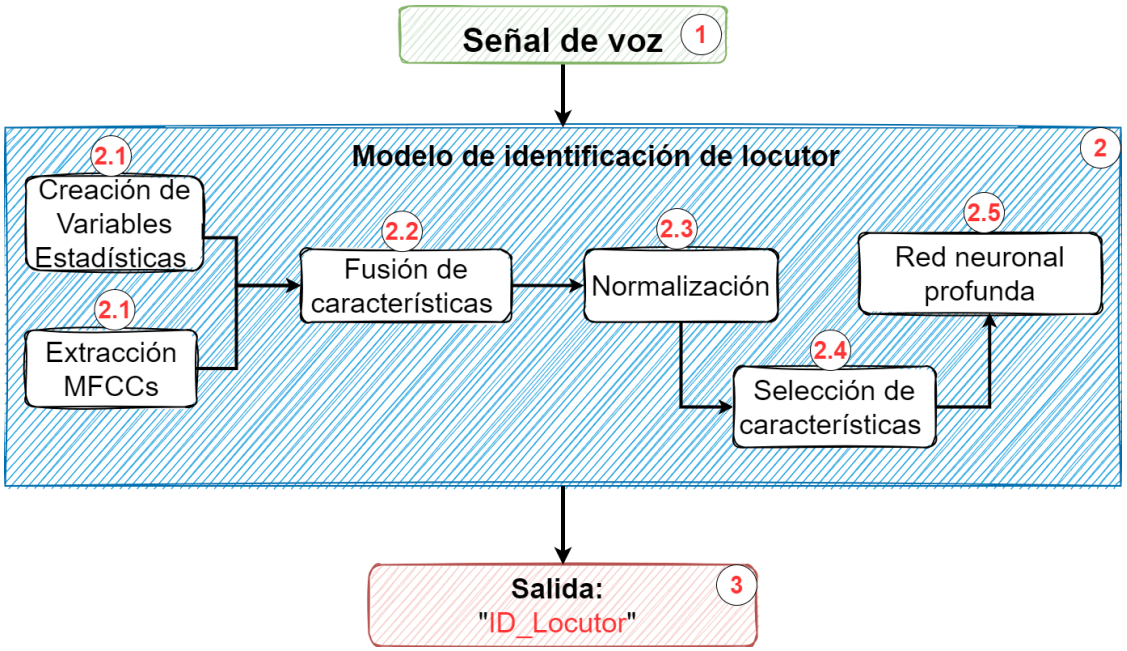

Figura 4.3 Modelo de identificación de locutor.

Por otro lado, el proceso de reconocimiento de locutor se divide en tres pasos, como se muestra en Figura 4.3. En la primera etapa, se recibe la señal acústica que se procesará posteriormente. En la segunda etapa, se realiza un modelado de la señal de voz para llevar a cabo la identificación de locutor. Esta fase incluye diferentes subprocesos, como la extracción de características, la fusión, la normalización, la selección de características y el proceso de clasificación. En la última etapa, se determina qué locutor tiene participación en la muestra de voz proporcionada inicialmente. El modelo propuesto ofrece una estructura simple y eficiente para realizar la identificación de locutor de manera robusta en situaciones de datos con ruido.

4.4.1. Aplicación de escritorio HAAF (Herramienta de Análisis Acústico Forense)

En este proyecto también se propone la aplicación de escritorio llamada HAAF (Herramienta de Análisis Acústico Forense). HAAF es desarrollada para llevar a cabo con fines prácticos la actividad de identificación de locutor en audios utilizados como evidencia criminal y en el idioma español. Las herramientas y tecnologías utilizadas para el desarrollo de esta aplicación de escritorio son: Python, Tkinter, Librosa y Pygame.

La aplicación HAAF está compuesta por el mejor esquema (modelo óptimo) de red neuronal artificial como modelo de clasificación e identificador de hablantes. La aplicación cuenta con una ventana principal de inicio de sesión (ver Figura 4.4). Si el usuario que desea usar la aplicación no está registrado en el sistema, entonces el sistema permite registrar nuevos usuarios, pidiendo información como su nombre, apellidos, nombre de usuario, correo electrónico y contraseña (ver Figura 4.5). Estas funciones añaden una capa de seguridad y confiabilidad de la información.

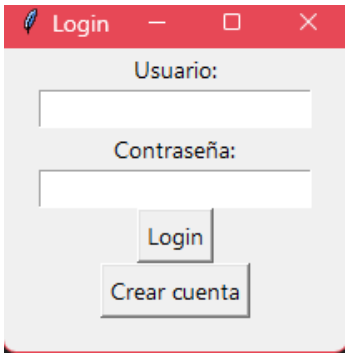

Figura 4.4 Ventada de inicio de sesión.

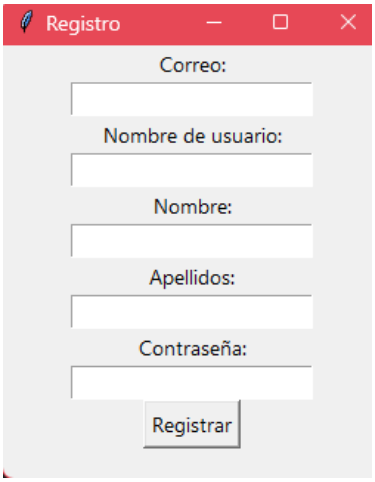

Figura 4.5 Ventada de registro de usuarios.

Una vez que el usuario pasa por la fase de inicio de sesión y/o registro de cuenta, se le redirecciona a la ventana principal donde se le otorgan tres herramientas o funciones principales (ver Figura 4.6):

- La primera herramienta, que aparece en la sección azul, permite cargar el audio y ser analizado con un reproductor de audio.
- La segunda funcionalidad, que se encuentra en el recuadro de color rosa, permite llevar a cabo el análisis del audio cargado en la aplicación en la sección 1. Es decir, se lleva a cabo la identificación del hablante que tiene participación en la muestra de voz proporcionada.
- Finalmente, en la sección 3 de color verde, se puede llevar un entrenamiento o reentrenamiento del modelo de clasificación a partir de un conjunto de datos en formato .csv.

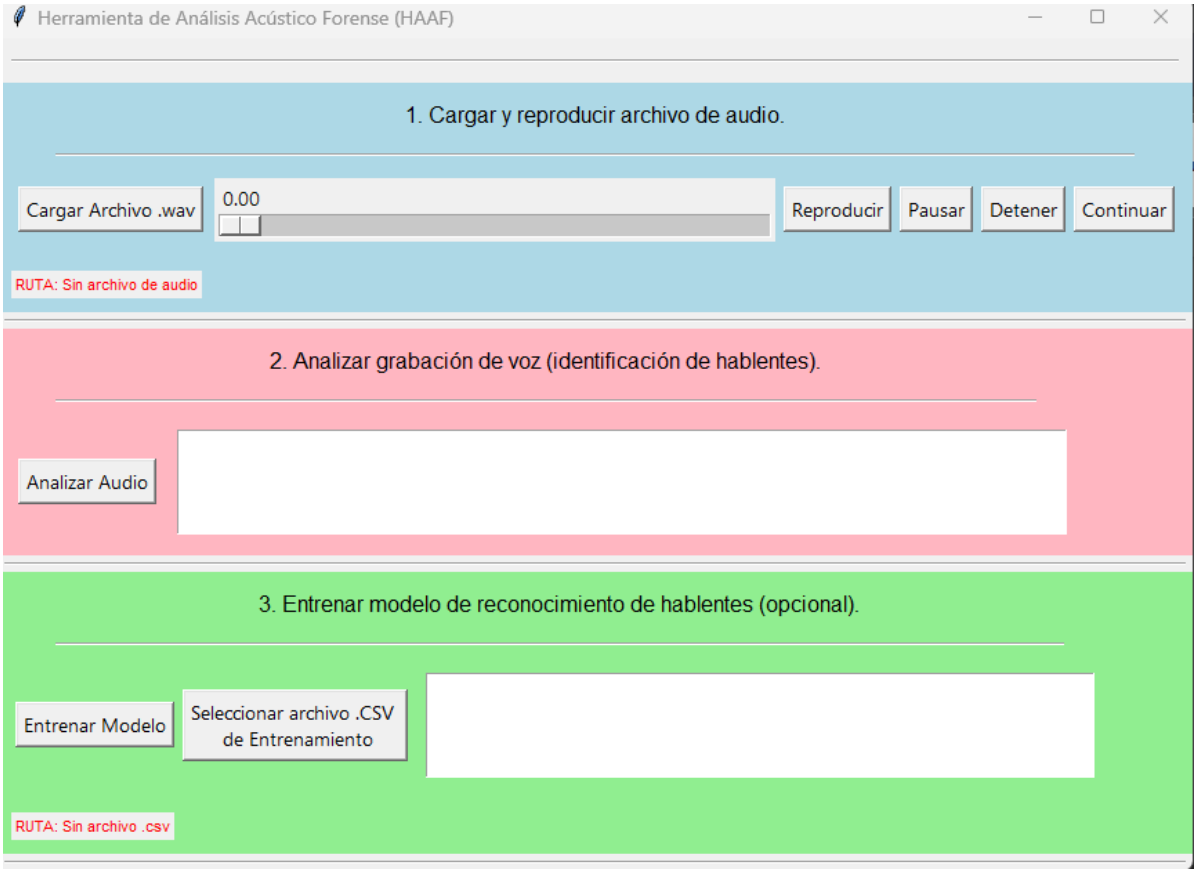

Figura 4.6 Ventada principal (herramientas).

La Figura 4.7 presenta el menú principal de la aplicación después de utilizar las distintas funcionalidades que el usuario puede llevar a cabo dentro del sistema. Es importante mencionar que esta versión de la aplicación HAAF se presenta como un prototipo funcional que puede ser mejorado en distintos aspectos.

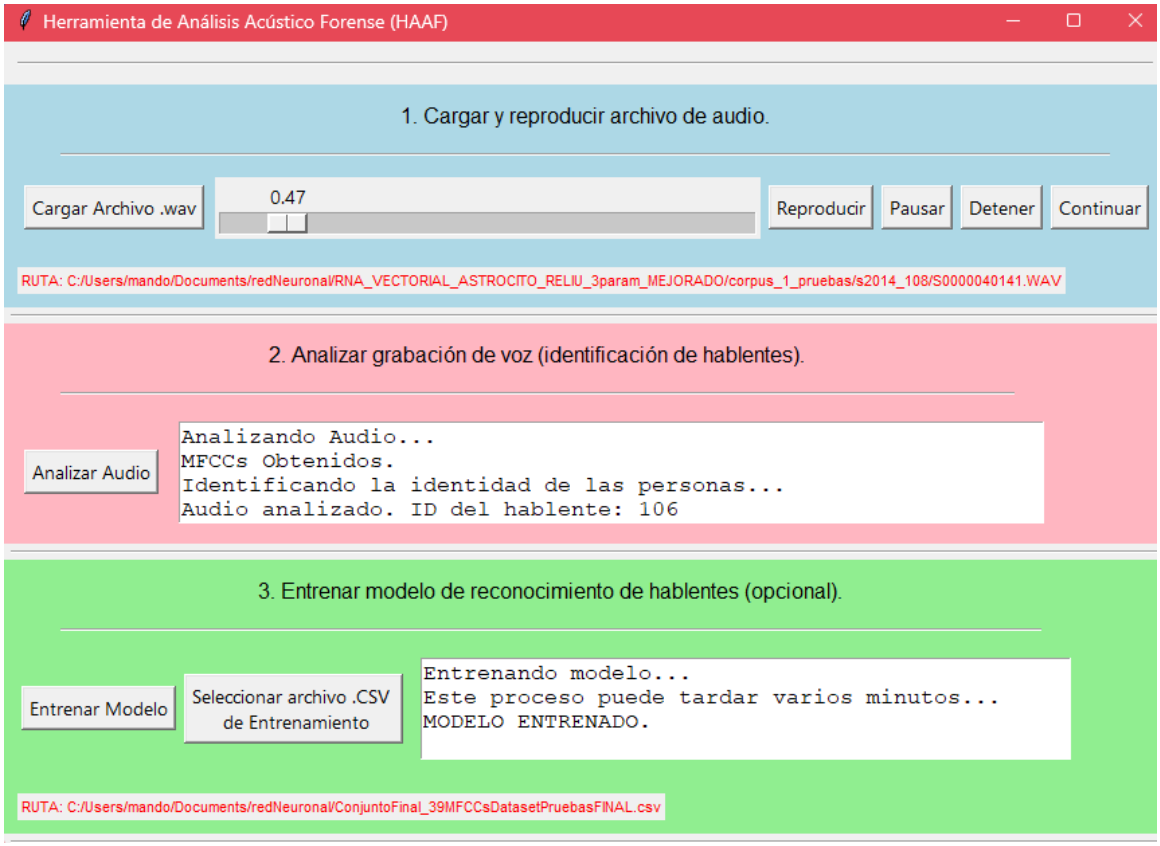

Figura 4.7 Ejecución del análisis de grabaciones de voz.

4.5. Esquema de red neuronal artificial propuesto

En este trabajo se propone una arquitectura de red neuronal artificial inspirada y basada en las RNA-Glía (ver Figura 4.8) [37], [106]. Esta nueva variante se llama Red Neuronal Artificial con Neuronas de Soporte (RNA-NS). La RNA-NS asigna a las neuronas artificiales convencionales una nueva tarea: cumplir una función de soporte y no estar comunicadas con otras unidades o neuronas. Esto da como resultado un nuevo tipo de unidades de procesamiento llamadas neuronas artificiales de soporte. Las unidades artificiales de soporte se encargan solamente de optimizar automáticamente los parámetros de las funciones de activación paramétricas. Por tanto, a cada neurona artificial estándar se le asignará una cantidad i de neuronas de soporte.

El objetivo de la RNA-NS es regular el potencial de acción de cada neurona y realizar un modelado más acorde al comportamiento de los datos, mediante la inclusión de una nueva unidad artificial de procesamiento y funciones de activación paramétricas. Además, otra de las intenciones de la arquitectura sugerida es permitir que las neuronas artificiales convencionales también puedan emular el comportamiento de las células biológicas de apoyo, como las células gliales o los astrocitos. De esta manera, se pueden tener unidades de cómputo especializadas para tareas específicas en los procesos internos de una red neuronal artificial.

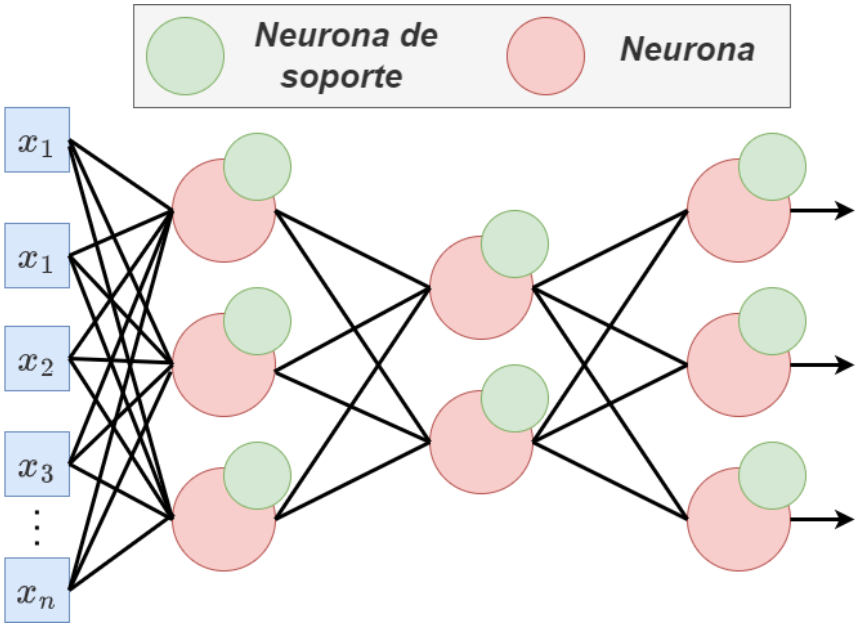

Figura 4.8 Red neuronal artificial con neuronas de soporte.

Esta propuesta busca que las redes neuronales artificiales puedan aprender automáticamente los parámetros entrenables de las funciones de activación paramétricas, regular los potenciales de acción y lograr que las funciones de activación se ajusten al comportamiento de los datos para un mejor modelado. Esto se traducirá en una mayor robustez en datos con ruido y un mejor rendimiento en el modelado de señales acústicas.

Notación:

b_k: Sesgo aplicado a la neurona k.

p_{ki}: Sesgo aplicado a la neurona de soporte i asociada a la neurona k.

x_j: Entrada j.

w_{kj}: Peso j de la neurona k.

m_{kij}: Peso j de la neurona de soporte i asociada a la neurona k.

$v_k(n)$: Potencial acción de la neurona k.

$z_{ki}\ (n)$: Potencial acción de la neurona de soporte i asociada a la neurona k.

$h_{ki}(z_{ki}(n))$: Función de activación de la neurona de soporte i asociada a la neurona k.

$\varphi_k(v_k\ (n))$: Función de activación de la neurona k.

y_k: Salida de la neurona k.

η: Tasa de aprendizaje.

a_{ki}: Salida de la neurona de soporte i asociada a la neurona k. También se puede interpretar como el parámetro asociado a la función de activación de la neurona k, el cual será optimizado por la neurona de soporte i que está asociada a la neurona k.

$\delta_k^{(l)}(n)$: Error local de la neurona k.

$\Omega_{ki}^{(l)}(n)$: Error local de la neurona de soporte i asociada a la neurona k.

La Figura 4.9 describe la estructura de una neurona artificial convencional, mientras que la Figura 4.10 presenta los componentes de la neurona artificial de soporte. A diferencia de las unidades artificiales convencionales, las unidades artificiales de soporte no tienen un vector de entradas x_n porque se asume que todas sus entradas valen 1. Esto significa que las neuronas de soporte no reciben señales de otras unidades. Las neuronas de soporte solo tienen un vector de pesos m_{ki} y un sesgo p_{ki}, que debe inicializarse bajo otra distribución de probabilidad o simplemente en 0. Las unidades artificiales de soporte pueden ayudar a entrenar parámetros de funciones paramétricas como PReLU, FReLU, DPReLU, soft++, entre otras.

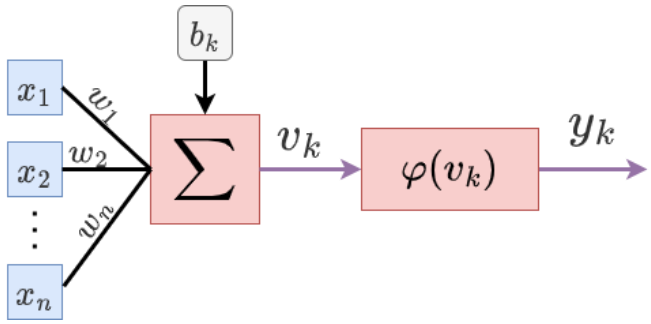

Figura 4.9 Estructura de una neurona artificial convencional.

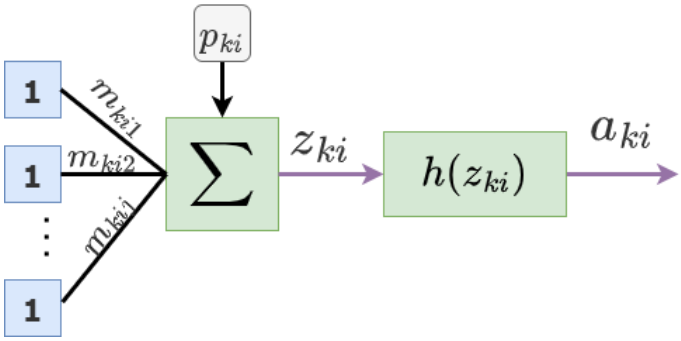

Figura 4.10 Estructura de una neurona artificial de soporte.

Definir múltiples pesos, cada uno asociado a su respectiva entrada en las neuronas de soporte, donde todas las entradas tienen un valor de 1, permite que el

modelo converja más rápidamente y con un mejor rendimiento (ver Figura 4.10). Por otra parte, mantener las entradas con un valor constante de 1 en las unidades de soporte ocasiona que se distribuya la responsabilidad de la convergencia de las neuronas de soporte entre varios pesos y un sesgo, en lugar de un solo peso. Esto hace que la salida de la neurona también se mantenga constante (en cada propagación hacia delante) y no dependa de conexiones que puedan afectar la convergencia del modelo debido a la aleatoriedad de las entradas. También, se brinda estabilidad al proceso de convergencia y contribuye a un mejor desempeño general del modelo. Por último, dado que este trabajo no propone un método específico para la inicialización de los pesos, la estrategia sugerida ayuda a solucionar parcialmente el problema de la inicialización de los pesos y los sesgos en las unidades de soporte.

La función de activación en las neuronas de soporte permite definir el rango de búsqueda para la optimización del parámetro deseado en la función de activación paramétrica de la neurona estándar. Realizar una delimitación precisa al utilizar la función de activación correcta facilita la búsqueda del valor óptimo del parámetro, acelerando así la convergencia y mejorando el rendimiento del modelo. Por ejemplo, si se utiliza la función sigmoide, el rango de búsqueda es $[0, 1]$. Si se utiliza una tangente hiperbólica, el rango de búsqueda está en $[-1, 1]$. Se sugiere no usar funciones paramétricas en las unidades estándar. En caso de requerir una función paramétrica, los parámetros deben ser definidos y ajustados manualmente.

En la Figura 4.11 se describe la interacción entre neuronas estándar y neuronas de soporte. En esta interacción, es necesario usar una función de activación paramétrica para poder asociar unidades convencionales con unidades de soporte, donde se asigna una neurona de soporte por cada parámetro que se desee optimizar en la función de activación paramétrica de la unidad estándar. Cada neurona de soporte está asociada únicamente con una neurona artificial convencional, y cada unidad estándar puede estar asociada con i unidades de soporte, esto dependiendo de los parámetros a optimizar en la función de activación de la neurona artificial estándar. Entonces, la salida de una unidad de soporte será el valor óptimo de un parámetro requerido en la función de activación de la unidad convencional.

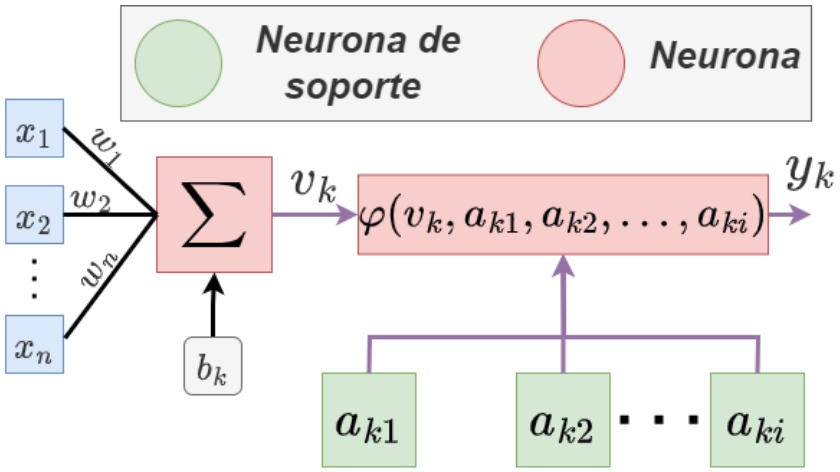

Figura 4.11 Asociación entre neuronas estándar y neuronas de soporte.

La Figura 4.12 muestra un ejemplo de la estructura del esquema de red neuronal artificial que se sugiere. La RNA-NS puede estar compuesta por L capas de unidades de procesamiento interconectadas entre sí (una capa de salida y capas ocultas) y una capa de entrada. Las capas ocultas y la capa de salida contienen k neuronas convencionales y cada una puede estar asociada a i neuronas de soporte. En cada capa se puede incluir o no unidades de soporte. Es decir, la red propuesta puede estar formada por una combinación de capas con neuronas de soporte y de capas sin unidades de soporte (ver Figuras 4.12 y 4.13). Bajo esta propuesta no se incluyen neuronas de soporte en capas softmax. En resumen, si en una capa de neuronas se utilizan con una función de activación paramétrica y se desea entrenar sus parámetros, entonces a cada parámetro (de cada unidad) se le asigna una neurona de soporte para realizar su respectivo proceso de aprendizaje u optimización.

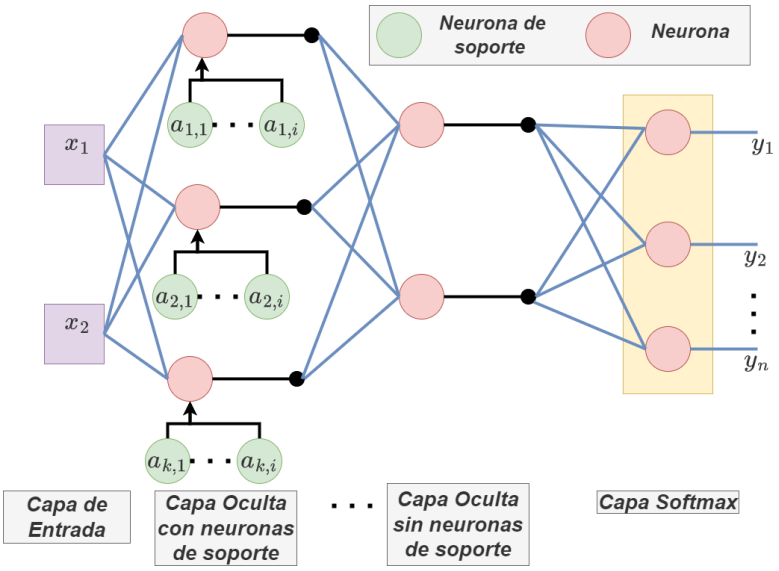

Figura 4.12 Arquitectura de la red neuronal artificial propuesta.

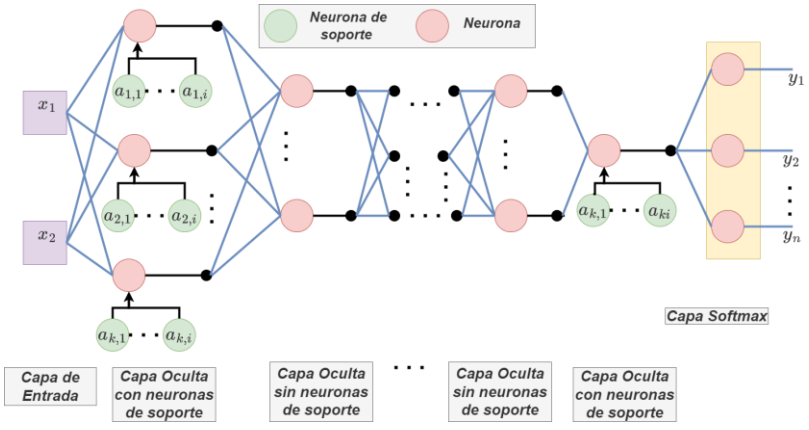

Figura 4.13 Arquitectura de la red neuronal artificial propuesta con L capas ocultas.

4.5.1. **Regla de aprendizaje**

En este trabajo también se propone un algoritmo y el proceso matemático para realizar las correcciones de pesos y sesgos tanto para las neuronas convencionales como para las neuronas de soporte en la RNA-NS. Para definir la regla de aprendizaje de los pesos y sesgos de las neuronas estándar y de soporte, se utilizó el algoritmo de propagación hacia atrás, la función de costo de entropía cruzada, el método de gradiente estocástico descendente y la regla delta. En este procedimiento se asume que las unidades de soporte solo existen en las capas ocultas y que la capa de salida será de tipo softmax.

Los pasos secuenciales para actualizar los sesgos y pesos de las diferentes unidades de procesamiento de la RNA-NS mediante la propagación hacia atrás son los siguientes:

1. Realizar una propagación hacia adelante (predicción).
2. Calcular el error cometido de la RNA-NS mediante la función de error (entropía cruzada).
3. Calcular el error de la capa de salida y propagarlo hacia atrás, capa por capa, hasta la primera capa oculta de la red (retropropagación del error).
4. Calcular las deltas (gradiente de error) para las neuronas estándar de la capa actual.
5. Calcular las deltas (gradiente de error) para las neuronas de soporte de la capa actual.
6. Repetir los pasos 4 y 5 hasta procesar la última capa de la capa actual.
7. Recorrer la red, capa por capa, hacia delante.
8. Actualizar los sesgos y pesos de las neuronas convencionales.
9. Actualizar los sesgos y pesos de las neuronas de soporte.
10. Repetir los pasos 8 y 9 hasta procesar la última capa.
11. Repetir los pasos 1 a 10 hasta la última iteración de entrenamiento.

La ecuación (4.1) define el nuevo cálculo del gradiente local (error local) para las neuronas convencionales, mientras que la ecuación (4.2) presenta la forma para calcular el error local de las neuronas de soporte. Se puede observar que para actualizar los pesos y sesgos de las unidades estándar no existen cambios significativos respecto al método clásico de la regla delta y propagación hacia atrás.

$$\delta_k^{(l)}(n) = \left[\begin{array}{c} y_k^{(L)}(n) - d_k^{(L)}(n) \\ \hline \dfrac{\partial \varphi_k^l\left(v_k^{(l)}(n),\ a_{ki}^{(l)}\right)}{\partial v_k^{(l+1)}(n)} \end{array} * \sum_k \delta_k^{(l+1)}(n) * w_{kj}^{(l+1)}(n) \right]. \qquad (4.1)$$

Donde:

$d_k^{(L)}(n)$: Salida esperada de la neurona k perteneciente a la capa de salida.

$y_k^{(L)}(n)$: Salida de la neurona k perteneciente a la capa de salida.

$\frac{\partial \varphi_k^l\left(v_k^{(l)}(n)\right)}{\partial v_k^{(l+1)}(n)}$: Derivada parcial de la función de activación de la neurona k respecto a su potencial de acción.

$\delta_k^{(l+1)}(n)$: Delta de la capa $l+1$ de la neurona k.

$w_{kj}^{(l+1)}(n)$: Peso j de la neurona k perteneciente de a la capa $l+1$.

La ecuación (4.2) define el proceso para calcular el gradiente local de una neurona de soporte i asociada a la neurona k de la capa oculta l. Es importante señalar que solo se considera el cálculo para neuronas de soporte que se encuentren en capas ocultas y que estas no reciben conexiones de otras unidades de procesamiento.

$$\Omega_{ki}^{(l)}(n) = \left[h'_{ki}\left(z_{ki}^{(l)}(k)\right) * \frac{\partial \varphi_k^l\left(v_k^{(l)}(n),\ a_{ki}^{(l)}\right)}{\partial a_{ki}^{(l)}(n)} \ * \sum_k \left(\delta_k^{(l+1)}(n) * w_{kj}^{(l+1)}(n)\right)\right]. \quad (4.2)$$

Donde:

$h'_{ki}\left(z_{ki}^{(l)}(k)\right)$: Derivada de la función de activación de la neurona de soporte i asociada a la neurona k de la capa oculta l.

$\frac{\partial \varphi_k^l\left(v_k^{(l)}(n)\right)}{\partial a_{ki}^{(l)}(n)}$: Derivada parcial de la función de activación de la neurona k respecto a la salida de la neurona de soporte i asociada a la neurona k de la capa oculta l.

$\delta_k^{(l+1)}(n)$: Delta de la capa $l+1$ (capa siguiente) de la neurona k.

$w_{kj}^{(l+1)}(n)$: Peso j de la neurona k perteneciente a la capa $l+1$ (capa siguiente).

La ecuación (4.3) y la ecuación (4.4) resumen las correcciones de los pesos para las unidades convencionales y de soporte, respectivamente. Existen dos tasas de aprendizaje: una para las neuronas de soporte θ y otra para las unidades estándar η. Las tasas de aprendizaje pueden tener el mismo o diferente valor, dependiendo del problema. Es posible que cada tipo de unidad deba aprender a un ritmo diferente, por lo que es importante seleccionar los mejores valores para cada uno.

$$\Delta w_{kj} = w_{kj} + \Big(\eta * \delta_k(n) * y_j(n)\Big). \quad (4.3)$$

Donde:

Δw_{kj}: Corrección del peso j correspondiente a la neurona k.

w_{kj}: Peso actual j de la neurona k.

$\delta_k(n)$: Delta de la neurona k.

$y_j(n)$: Entra j de la neurona k.

η: Tasa de aprendizaje de las neuronas convencionales.

$$\Delta m_{kij} = m_{kij} + \big(\theta * \Omega_{ki}(n) \big). \tag{4.4}$$

Donde:

Δm_{kij}: Corrección del peso j correspondiente a la neurona de soporte i asociada a la neurona k.

m_{kij}: Peso actual j correspondiente a la neurona de soporte i asociada a la neurona k.

$\Omega_{ki}(n)$: Delta de la neurona de soporte i ligada a la neurona k.

θ: Tasa de aprendizaje de las neuronas de soporte.

4.5.2. Funciones de activación paramétricas

Como se mencionó anteriormente, las neuronas de soporte permiten optimizar los parámetros de las funciones de activación paramétricas. En este trabajo, se utilizan tres diferentes funciones con parámetros entrenables basadas en la función ReLU (o unidad lineal rectificada): la Unidad Lineal Rectificada Paramétrica (PReLU, por sus siglas en ingles), AReLU (o unidad lineal rectificada adaptable) y la Unidad Lineal Rectificada Paramétrica Modificada (MPReLU, por sus siglas en ingles).

Las diferentes funciones de activación que se diseñan en este trabajo (AReLU y MPReLU) permiten regular el potencial de acción de la neurona estándar. Esto se logra mediante parámetros entrenables que ajustan las pendientes positivas y negativas de las funciones. Ajustar las pendientes de las funciones hace posible que las funciones se ajusten mejor al comportamiento de los datos.

4.5.2.1 Función de activación ReLU

La función ReLU permite activar o inhibir la participación de una neurona bajo un umbral con valor de 0. Si el potencial de acción es mayor que el umbral, se transmite a otras neuronas. En caso contrario, la unidad queda inactiva [107], [108]. Esto puede observase en la Figura 4.14 [107] y la ecuación (4.5). Lo interesante de esta función de activación es que intenta imitar la característica de los estados de activación o inactivación de las neuronas biológicas, dependiendo del valor del potencial de acción [107], [108].

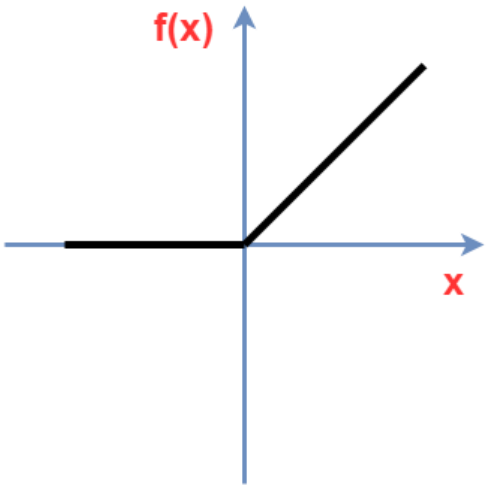

Figura 4.14 Función de activación ReLU.

$$ReLU(x) = \begin{cases} x, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}. \quad (4.5)$$

Donde:

x: Potencial de acción.

4.5.2.2 Función de activación PReLU

PReLU es una variante existente y muy conocida de ReLU y LReLU. Esta función introduce un parámetro entrenable para modificar la pendiente de la parte negativa de la función de activación [109], como se presenta en la ecuación (4.6) y Figura 4.15.

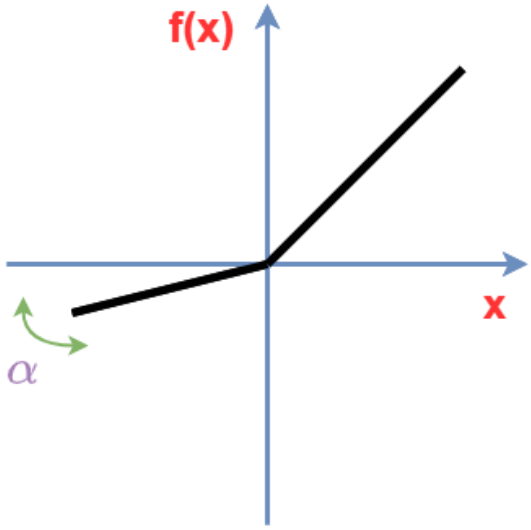

Figura 4.15 Función de activación PReLU.

$$PReLU(x,\alpha) = \begin{cases} x, & si\ x \geq 0 \\ \alpha x, & si\ x < 0 \end{cases}. \quad (4.6)$$

Donde:

x: Potencial de acción.

α: Parámetro entrenable que modifica la parte negativa de la función PReLU y puede ser controlado por una neurona de soporte.

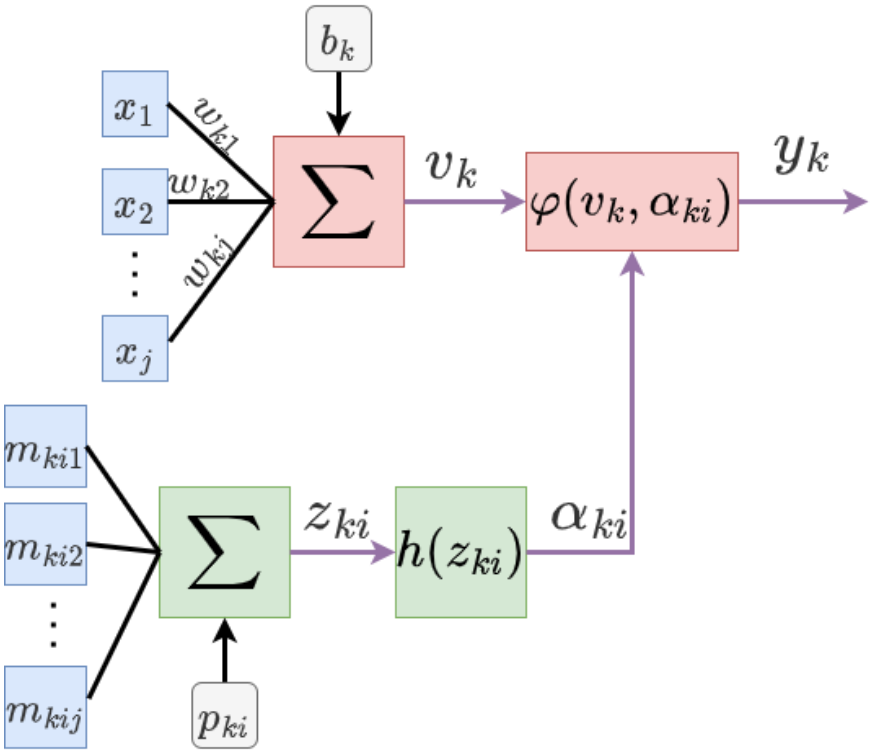

Figura 4.16 Estructura de la asociación entre neuronas de soporte y neuronas estándar con función de activación PReLU.

La Figura 4.16 muestra el proceso y la estructura utilizados para entrenar el parámetro α de la función de activación PReLU. Cada neurona de soporte está vinculada a una neurona convencional. La tarea de la unidad de soporte es optimizar el parámetro α correspondiente a la neurona estándar. Si una capa contiene 10 neuronas convencionales, habrá un total de 10 parámetros α que deben ser optimizados. Esto implica la presencia de 10 unidades de soporte, lo que da lugar a un total de 20 unidades de procesamiento (10 neuronas estándar y 10 de soporte). La salida de la unidad de soporte i perteneciente a la neurona k representa el valor óptimo de su parámetro α_{ki} correspondiente.

La ecuación (4.7) presenta la derivada parcial de la función PReLU respecto al potencial de acción.

$$\frac{\partial PReLU(x,\alpha)}{\partial x} = \begin{cases} 1, & si\ x \geq 0 \\ \alpha, & si\ x < 0 \end{cases}. \quad (4.7)$$

Donde:

x: Potencial de acción.

α: Parámetro entrenable que modifica la parte negativa de la función PReLU y es controlado por una neurona de soporte.

Por otro lado, la ecuación (4.8) presenta la derivada parcial de la función PReLU respecto al parámetro α.

$$\frac{\partial PReLU(x,\alpha)}{\partial \alpha} = \begin{cases} 0, & si\ x \geq 0 \\ x, & si\ x < 0 \end{cases}. \quad (4.8)$$

Donde:

x: Potencial de acción.

α: Parámetro entrenable que modifica la parte negativa de la función PReLU y puede ser optimizado por una neurona de soporte.

4.5.2.3 Función de activación DPReLu

DPReLU es una función de activación paramétrica basada en ReLU muy flexible y adaptable. Esta función permite ajustar la función ReLU original en los ejes x e y, y también permite ajustar las pendientes positiva y negativa de la función [107]. Podemos apreciar su comportamiento en la Figura 4.17 [105] y matemáticamente en la ecuación (4.9).

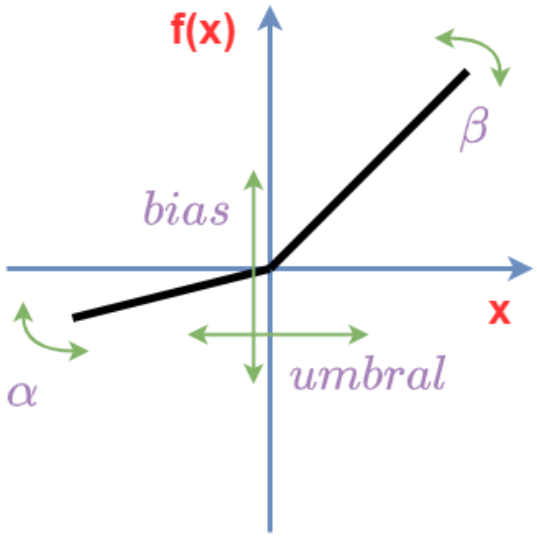

Figura 4.17 Función de activación DPReLU.

Las funciones MPReLU y AReLU son dos variantes simplificadas de DPReLU que se presentan en este trabajo. Ambas funciones tienen el objetivo de modificar únicamente las pendientes de la función ReLU, por lo que son una versión sintetizada de DPReLU que evitan cálculos innecesarios de los parámetros que no se pretenden modificar. Las funciones mencionadas se describen en detalle en las siguientes subsecciones.

$$DPReLU(x, \beta, \alpha, umbral, bias) = \begin{cases} \beta(x - umbral) + bias, & si\ x \geq 0 \\ \alpha(x - umbral) + bias, & si\ x < 0 \end{cases}. \quad (4.9)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la parte positiva (pendiente) de la función.

α: Parámetro entrenable que modifica la pendiente negativa de la función.

$Umbral$: Parámetro entrenable que desplaza la función en el eje x.

$Bias$: Parámetro entrenable que desplaza la función en el eje y.

4.5.2.4 Función de activación MPReLU

Se propone la función MPReLU como una variante de la función PReLU y una versión simplificada de la función DPReLU que solo modifica uno de sus cuatro parámetros. Como se ilustra en la Figura 4.18, DPReLU permite modificar las pendientes negativa y positiva de la función ReLU, así como desplazarla en el eje x e y. Dado que solo se pretende modificar un parámetro (pendiente positiva), se simplificó DPReLU para reducir la cantidad de cálculos y el costo computacional. A diferencia de PReLU, que modifica la pendiente negativa de ReLU, MPReLU modifica la parte positiva de la pendiente para regular el potencial de acción [107].

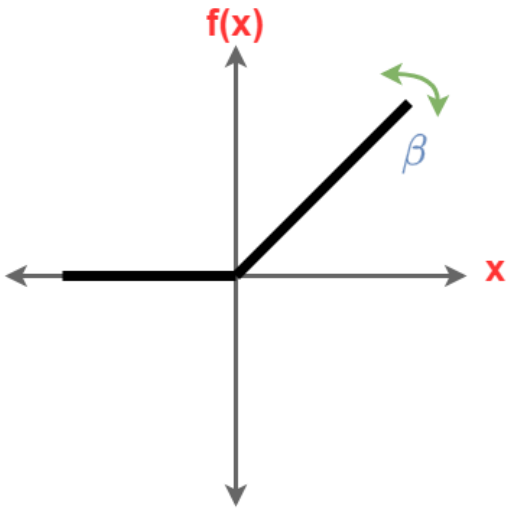

Figura 4.18 Modified Parametric Rectified Linear Unit (MPReLU).

En la función MPReLU, la parte negativa es idéntica a la función ReLU, mientras que la parte positiva se reemplaza por la entrada multiplicada por un parámetro entrenable. El propósito de esta función es ajustar únicamente la parte positiva de la función ReLU, es decir, modificar la pendiente de la parte positiva en lugar de la negativa. Esto permite que la neurona esté inactiva o activa, y además de amplificar o reducir el potencial de acción. La ecuación (4.10) describe la función MPReLU.

$$MPReLU(x,\beta) = \begin{cases} \beta x, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}. \quad (4.10)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la parte positiva (pendiente) de la función MPReLU y es controlado por una neurona de soporte.

La ecuación (4.11) presenta la derivada parcial de la función MPReLU respecto al potencial de acción.

$$\frac{\partial MPReLU(x,\beta)}{\partial x} = \begin{cases} \beta, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}. \quad (4.11)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la parte positiva (pendiente) de la función MPReLU y puede ser controlado por una neurona de soporte.

Por otro lado, la ecuación (4.12) presenta la derivada parcial de la función MPReLU respecto al parámetro β.

$$\frac{\partial MPReLU(x,\beta)}{\partial \beta} = \begin{cases} x, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}. \quad (4.12)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la parte positiva (pendiente) de la función MPReLU y es controlado por una neurona de soporte.

El parámetro beta ayuda a regular el potencial de acción al aumentarlo o disminuirlo en la parte positiva de la función.

Para optimizar el parámetro beta de la función MPReLU, se sigue un proceso similar al de la función PReLU, como se muestra en la Figura 4.19. La diferencia radica en que en este caso el parámetro α se representa como β y se encarga de ajustar la pendiente positiva de la función en lugar de la pendiente negativa, como se ilustra en la Figura 4.18. Igualmente, la salida de la unidad de soporte i perteneciente a la neurona k representa el valor óptimo de su parámetro β_{ki} correspondiente.

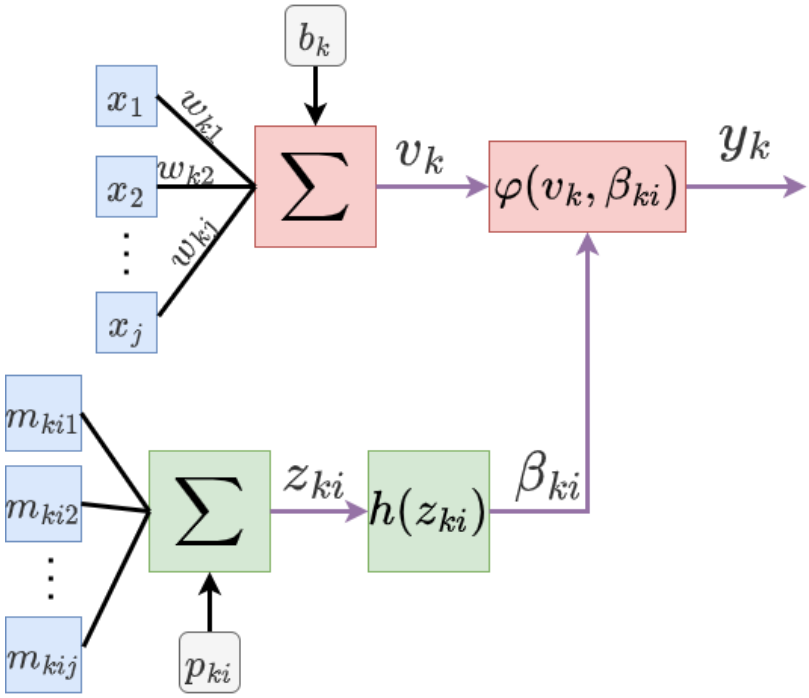

Figura 4.19 Estructura de la asociación entre neuronas de soporte y neuronas estándar con función de activación MPReLU.

4.5.2.5 Función de activación AReLU

AReLU también se propone como otra versión de DPReLU, donde solo se modifican dos parámetros: la pendiente positiva y negativa. La idea es presentar una versión más sencilla para reducir la cantidad de cálculos y costo computacional, ya que los otros parámetros de DPReLU no se modifican [107]. Asimismo, es posible regular el potencial de acción (incrementándolo o disminuyéndolo) tanto para la parte positiva como negativa de la función. De esta manera, se puede regular el potencial de acción específicamente, dependiendo de si es negativo o positivo.

Como se mencionó, la función AReLU (ver Figura 4.20) se propone como una extensión de las funciones PReLU y ReLU, y como una versión sintetizada de DPReLU. Como se observa en la ecuación (4.13), se introducen los parámetros alfa y beta para ajustar las pendientes de las partes negativa y positiva, respectivamente. El objetivo es lograr una mejor adaptación de la función al comportamiento de los datos, permitiendo un modelado no lineal más adecuado.

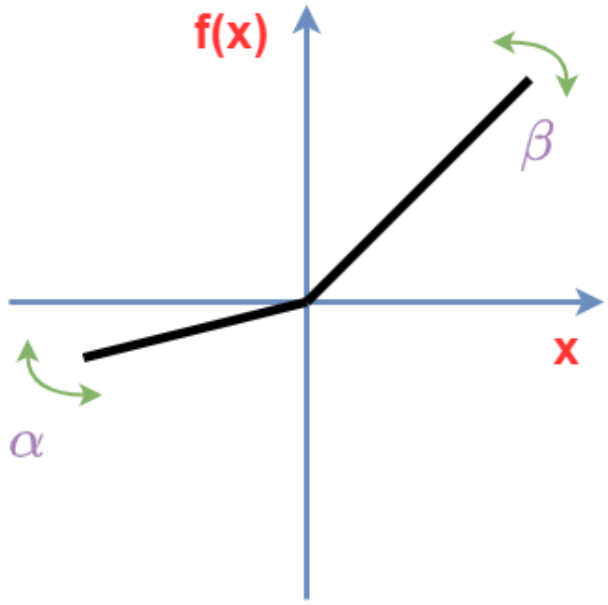

Figura 4.20 Adaptive Rectified Linear Unit (AReLU).

$$AReLU(x,\alpha,\beta) = \begin{cases} \beta x, & si\ x \geq 0 \\ \alpha x, & si\ x < 0 \end{cases}. \quad (4.13)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la pendiente positiva de la función AReLU y es controlado por una neurona de soporte.

α: Parámetro entrenable que modifica la pendiente negativa de la función AReLU y es controlado por una neurona de soporte.

La ecuación (4.14) presenta la derivada parcial de la función AReLU respecto al potencial de acción.

$$\frac{\partial AReLU(x,\alpha,\beta)}{\partial x} = \begin{cases} \beta, & si\ x \geq 0 \\ \alpha, & si\ x < 0 \end{cases}. \quad (4.14)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la pendiente positiva de la función AReLU y es controlado por una neurona de soporte.

α: Parámetro entrenable que modifica la pendiente negativa de la función AReLU y es controlado por una neurona de soporte.

Por otro lado, la ecuación (4.15) presenta la derivada parcial de la función AReLU respecto al parámetro α.

$$\frac{\partial AReLU(x,\alpha,\beta)}{\partial \alpha} = \begin{cases} 0, & si\ x \geq 0 \\ x, & si\ x < 0 \end{cases}. \quad (4.15)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la pendiente positiva de la función AReLU y es controlado por una neurona de soporte.

α: Parámetro entrenable que modifica la pendiente negativa de la función AReLU y es controlado por una neurona de soporte.

Y finalmente, la ecuación (4.16) presenta la derivada parcial de la función AReLU respecto al parámetro β.

$$\frac{\partial AReLU(x,\alpha,\beta)}{\partial \beta} = \begin{cases} x, & si\ x \geq 0 \\ 0, & si\ x < 0 \end{cases}. \quad (4.16)$$

Donde:

x: Potencial de acción.

β: Parámetro entrenable que modifica la pendiente positiva de la función AReLU y es controlado por una neurona de soporte.

α: Parámetro entrenable que modifica la pendiente negativa de la función AReLU y es controlado por una neurona de soporte.

El proceso y la estructura para entrenar los parámetros α y β de esta función de activación se pueden visualizar en la Figura 4.21. En este caso, cada neurona estándar está asociada con dos neuronas de soporte. Ambas unidades de soporte se conectan a un único par de parámetros alfa y beta, que se optimizan conjuntamente. Es decir, el parámetro α estará vinculado a una neurona de soporte específica, mientras que el parámetro β también estará vinculado a una unidad de soporte. Cada neurona de soporte se especializa únicamente en un solo parámetro. Por lo tanto, por cada neurona AReLU, se tendrán dos neuronas de soporte. Si una capa contiene 5 unidades AReLU, entonces habrá un total de 15 unidades de procesamiento (5 neuronas convencionales y 10 de soporte). Esto se debe a que en esta capa hay 10 parámetros extras a optimizar. Es importante destacar que las neuronas de soporte son las únicas unidades que no reciben señales de otras unidades. La idea es mantener los parámetros alfa y beta constantes para mantener la estabilidad de la red neuronal.

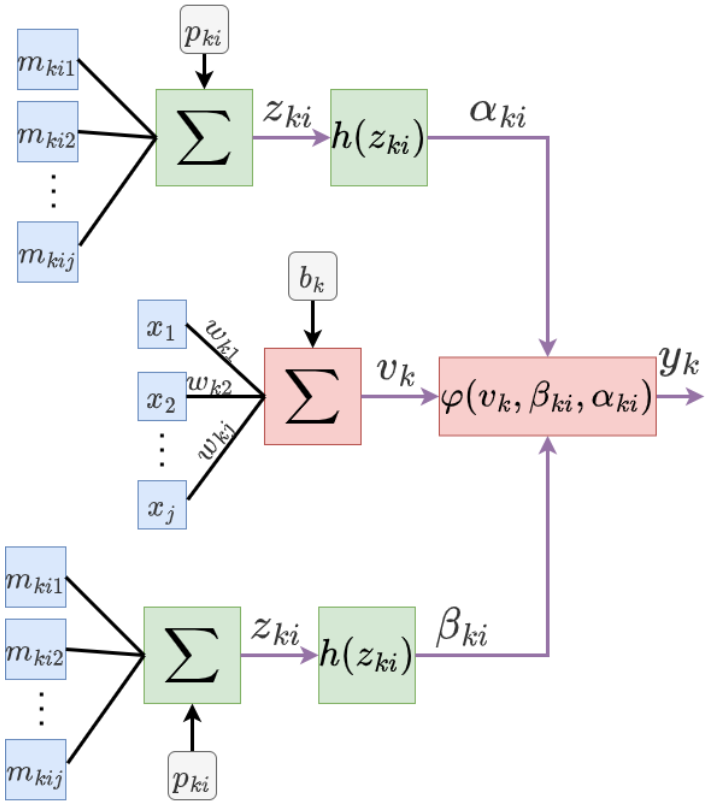

Figura 4.21 Estructura de la asociación entre neuronas de soporte y neuronas estándar con función de activación AReLU.

4.6. Identificación de locutor

En esta subsección se describe de manera secuencial cada fase para llevar a cabo el pre-procesamiento y recolección de los datos, extracción y selección de características, y el modelado de las señales acústicas para realizar la actividad de identificación del hablante. También, se presenta información de la arquitectura de los diferentes esquemas de red neuronal artificial propuestos.

4.6.1. Resumen y recolección de los datos

Para llevar a cabo este proyecto, se utilizó un conjunto de datos proporcionado y creado por el Laboratorio de Procesamiento de Voz de la Maestría en Ciencias del Procesamiento de la Información de la Universidad Autónoma de Zacatecas.

4.6.1.1 Selección de la muestra

Se aplicó el método propuesto por Hernández Sampieri [49] para formar el conjunto de datos empleado en esta investigación (ver Figura 4.22 [49]). El método consta de 5 fases, las cuales son identificar los elementos de muestreo o análisis,

definir concretamente la población o universo, seleccionar el tipo de estrategia de muestreo (probabilística o no probabilística), calcular el tamaño de la muestra apropiado y finalmente seleccionar bajo una estrategia las muestras de análisis.

La población o universo está formado por estudiantes de ingeniería de Software de la Universidad Autónoma de Zacatecas, que hablan español de forma nativa y no son de origen extranjero. La edad de los estudiantes oscila entre los 18 y 26 años, tanto de sexo masculino como femenino. Se toman en cuenta alumnos de todos los semestres (un total de 10 semestres).

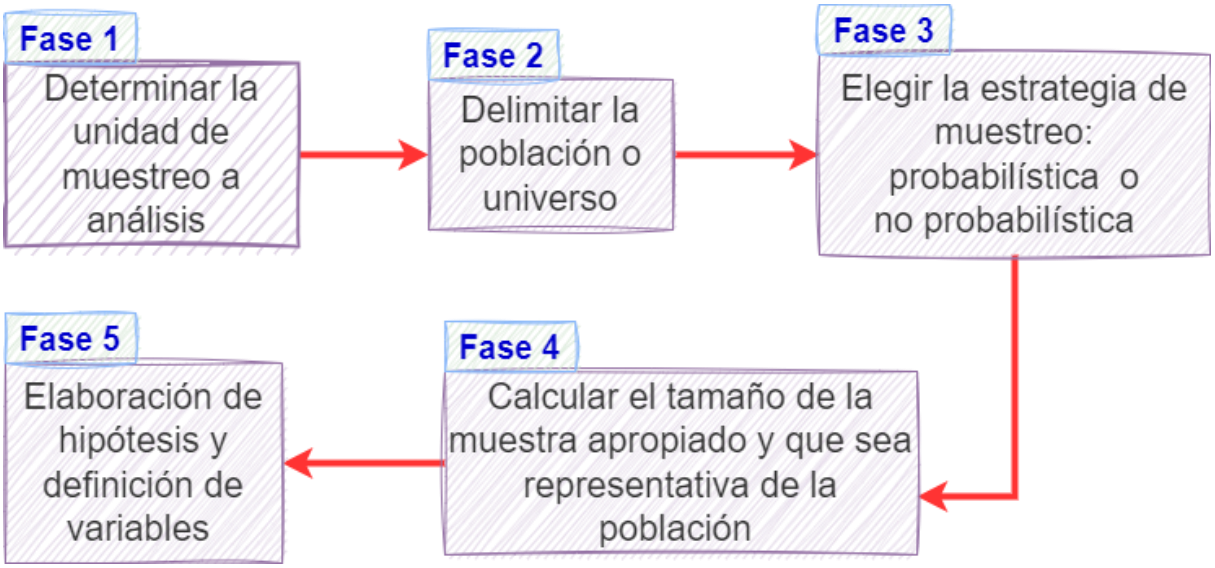

Figura 4.22 Proceso de selección de la muestra de datos.

Se seleccionó una estrategia de muestreo de tipo probabilístico, consecuentemente los alumnos fueron seleccionados de manera aleatoria. El tamaño de la muestra calculada fue de 158 alumnos conformada por hombres y mujeres, con un porcentaje de error máximo tolerable del 5% y un nivel de confianza del 95%. Lo anterior fue a consideración del total los alumnos inscritos en el programa académico mencionado (universo).

4.6.1.2 Recolección de datos

Para la fase de recolección de datos, se envió un correo electrónico a cada alumno seleccionado con las instrucciones para generar y enviar las grabaciones. El proceso consistió en proporcionar a los alumnos un conjunto de frases ya definidas, posteriormente se les pidió a los alumnos que pronunciaran las oraciones con diversos tonos de voz y fueran grabadas con distintos dispositivos tales como celulares, computadoras o audífonos con micrófono integrado. También se les indicó a los participantes que cada frase fuera grabada en diferentes escenarios: sin ruido ambiental y con ruido ambiental. Por último, los audios fueron guardados o convertidos en formato .wav y enviados a una plataforma de almacenamiento.

4.6.1.3 Descripción y división del corpus de datos

El corpus de datos utilizado en este proyecto consta de 158 clases, que representan a diferentes personas. De estas clases, 122 corresponden a personas del sexo masculino y 36 corresponden a personas del sexo femenino. Cada clase está etiquetada con el nombre de la persona correspondiente. El corpus contiene un total de 4,911 archivos de audio. De estos, 3,395 se utilizan para el entrenamiento, lo cual representa el 70% del dataset. Del total de archivos de entrenamiento, se reservan 758 (15% del total de archivos) para la fase de validación. Por otro lado, se utilizan las otras 758 grabaciones restantes para el subconjunto de pruebas, lo cual corresponde al 15% restante del conjunto de datos original. En los subconjuntos de entrenamiento, validación y pruebas se incluyen las mismas personas o clases.

Todos los archivos son guardados con codificación PCM, 8 bits por muestra, una tasa de muestreo de 16 khz en un solo canal. La mayoría de los audios fueron grabados con varios tipos de ruido: distorsión de micrófono, lluvia, autos, animales, audios de alto y bajo volumen y ruido ambiental en general. Finalmente, es importante mencionar que estos archivos de audio se encuentran en el idioma español.

Hay algunas limitaciones en este corpus de datos. En primer lugar, la base de datos está restringida a una suma de 158 hablantes. En segundo lugar, se actúa nuestro conjunto de datos, ya que las conversaciones no se dan en un ambiente natural. En tercer lugar, no se consideran diferentes tipos de tonos. Finalmente, se recolecta información de personas de una única región de la parte norte central de México, por lo que no se toman en cuenta diferentes tipos de acentos.

4.6.2. Procesamiento de los datos

A continuación, se describe en detalle cada fase del procesamiento de los datos presentados en la Figura 2.5. Se proporciona información sobre la preparación de los datos para su uso, la eliminación del ruido, la corrección de inconsistencias, la eliminación de información redundante e irrelevante, normalización de los datos y preservar únicamente la información realmente significativa. Estas acciones incluyen la normalización de los datos, la extracción de características y la selección de características.

4.6.2.1 Extracción y fusión de características

Se ha utilizado el paquete Librosa para llevar a cabo la extracción de características utilizando la función *librosa.feature.mfcc().* Para capturar los MFCC, se han utilizado ventanas (también conocidas como frames) de 88 ms, con una superposición de 35 ms. La elección de este método se justifica debido a su capacidad para extraer información representativa y eliminar elementos no deseados, como ruidos ambientales y ruido introducido por el micrófono. Adicionalmente, es poco es sensible a factores como emociones, volumen y tono. Cabe destacar que esta técnica se basa en principios biológicos, ya que este método al igual que el oído humano es poco sensible a las alta frecuencias, lo que contribuye a mejorar su rendimiento en la representación de la información. Finalmente, la técnica realiza un pre-procesamiento.

Una vez que se extraen las características, se tienen tres subconjuntos de muestras: entrenamiento, validación y pruebas, cada uno con 126 características. Estos subconjuntos varían en el número de muestras (tramas) que contienen. El subconjunto de entrenamiento consta de $429{,}309$ muestras, el de validación contiene $96{,}138$ muestras, y el de pruebas consta de $97{,}286$ muestras.

Las características extraídas incluyen el coeficiente de energía, 39 MFCCs, 40 deltas y 40 doble-deltas. Además, se obtuvieron variables estadísticas como la media, mediana, moda, desviación estándar, varianza, máximo y mínimo de la variable de energía. Estas variables estadísticas se calcularon por archivo de audio a partir de las tramas obtenidas en cada uno de ellos.

En cuanto a los parámetros definidos para la extracción de los MFCCs, se utilizaron los siguientes valores: $nmfcc = 39,\ htk = True,\ nfft = 0.088 * tasa\ de\ muestreo,$ y $hoplength = 0.035 * tasa\ de\ muestreo$.

4.6.2.2 Normalización

El escalamiento, la normalización y la estandarización de las características son técnicas de pre-procesamiento utilizadas para reducir el ruido de los datos y lograr que los modelos converjan más rápidamente. En este estudio, se emplea el escalado estándar utilizando la función $StandardScaler()$ de Scikit-Learn para reducir el ruido de los datos. Asimismo, esta función permite que los datos adopten una distribución normal, lo cual facilita el aprendizaje de los pesos y sesgos en redes neuronales artificiales.

4.6.2.3 Selección de características

Se implementa el algoritmo de selección secuencial hacia atrás y KNN de manera anidada para seleccionar las características más significativas, es decir, aquellas que aportan información representativa y discriminativa.

Procedimiento para la selección de características:

- Paso 1: Entrenar un modelo óptimo de KNN. Para un rendimiento óptimo del algoritmo, se implementaron las técnicas de validación cruzada y búsqueda de cuadrículas de manera anidada. Posteriormente, se utilizaron curvas de validación para diagnosticar y resolver problemas de sub-ajuste o sobre-ajuste.
- Paso 2: Seleccionar las n características más significativas mediante los modelos de SBS y KNN.

El rango de valores utilizado para encontrar el valor óptimo de k para el algoritmo de KNN fue de 1 a 27, teniendo en cuenta únicamente los números impares. Finalmente, se determinaron los valores óptimos para obtener el modelo con el mejor rendimiento de KNN utilizando técnicas de validación cruzada y búsqueda de cuadrícula de manera anidada, los parámetros óptimos se presentan en la Tabla 4.1.

En la Tabla 4.2 se presentan las características más significativas obtenidas mediante el algoritmo de SBS y KNN (de manera anidada). La cantidad de

características que permitió que los modelos alcanzaran el mejor rendimiento fue de 46 variables.

Tabla 4.1 Parámetros óptimos para KNN.

parámetro	Valor
k	3
p	2

Tabla 4.2 Características seleccionadas con KNN y SBS de manera anidada.

Características seleccionadas	
Energia_0.	
Mfcc_1, Mfcc_2, Mfcc_3, Mfcc_4, Mfcc_5, Mfcc_6, Mfcc_7, Mfcc_8, Mfcc_9, Mfcc_10, Mfcc_11, Mfcc_12, Mfcc_13, Mfcc_14, Mfcc_15, Mfcc_16, Mfcc_17, Mfcc_18, Mfcc_19, Mfcc_20, Mfcc_21, Mfcc_22, Mfcc_23, Mfcc_24, Mfcc_25, Mfcc_26, Mfcc_27, Mfcc_29, Mfcc_30, Mfcc_33, Mfcc_35, Mfcc_39.	
DeltaMfcc_3, DeltaMfcc_4.	
DobleDeltaMfcc_2, DobleDeltaMfcc_4, DobleDeltaMfcc_5, DobleDeltaMfcc_6.	
DobleDeltaMfcc_7, Energia_media, Energia_mediana, Energia_std, Energia_var, Energia_min, Energia_max.	
Número óptimo de características.	**46.**

4.6.3. Modelado y clasificación

Para llevar a cabo la tarea de identificación de locutores en archivos de audio en el idioma español, considerando diferentes tipos de ruido ambiental, se proponen, se utilizan, y se comparan varias configuraciones de redes neuronales artificiales. El objetivo es encontrar el esquema óptimo que logre un equilibrio adecuado entre sesgo y varianza, teniendo en cuenta tanto el rendimiento como el tiempo de ejecución para llevar a cabo la comparación de los modelos. Las dependencias necesarias para este proyecto incluyen los siguientes paquetes y bibliotecas: Numpy, Pandas, SciPy, CUDA, Scikit-Learn, Librosa, Rapids de Nvidia y Matplotlib. Finalmente, el lenguaje de programación principal utilizado para este proyecto es Python.

4.6.3.1 Modelos de RNA propuestos

En esta sección se ofrece una descripción de las características, parámetros y propiedades de los diversos esquemas y configuraciones de RNA utilizados para llevar a cabo la tarea de identificación de locutores en el idioma español. En primer lugar, se describe una configuración de RNA basada en una arquitectura convencional llamada como RNA tradicional con ReLU. Posteriormente se presentan tres configuraciones de RNA basadas en la nueva arquitectura propuesta que incorpora neuronas de soporte junto con diferentes funciones de activación (PReLU, MPReLU y AReLU). En otras palabras, se utilizan distintas configuraciones de redes neuronales artificiales con neuronas de soporte. El objetivo es comparar el rendimiento de los

modelos RNA ya existentes respecto a la arquitectura propuesta de RNA con neuronas de soporte.

4.6.3.2 RNA tradicional con ReLU

En la Figura 4.23 se puede observar la representación gráfica de la estructura del modelo de RNA con una arquitectura tradicional. La RNA se compone de una capa de entrada y dos capas totalmente conectadas de neuronas. La capa de entrada recibe un vector de entrada con un tamaño de 46 elementos (características). Cada capa utiliza un número distinto de neuronas e implementa una función de activación diferente. La primera capa de unidades, que consta de 384 neuronas, implementa la función de activación ReLU. La segunda capa, que es la capa de salida, está compuesta por 158 neuronas y usa la función de activación softmax.

Para entrenar la RNA, se utiliza el algoritmo del gradiente estocástico descendente con una tasa de aprendizaje de 0.01 y se realizan 15 épocas de entrenamiento. Los pesos se inicializan de manera aleatoria utilizando el método de Xavier y una distribución uniforme. Además, los sesgos de las neuronas se inicializan con un valor de 0. Por último, se empleó la función de costo de entropía cruzada para evaluar el error de la red.

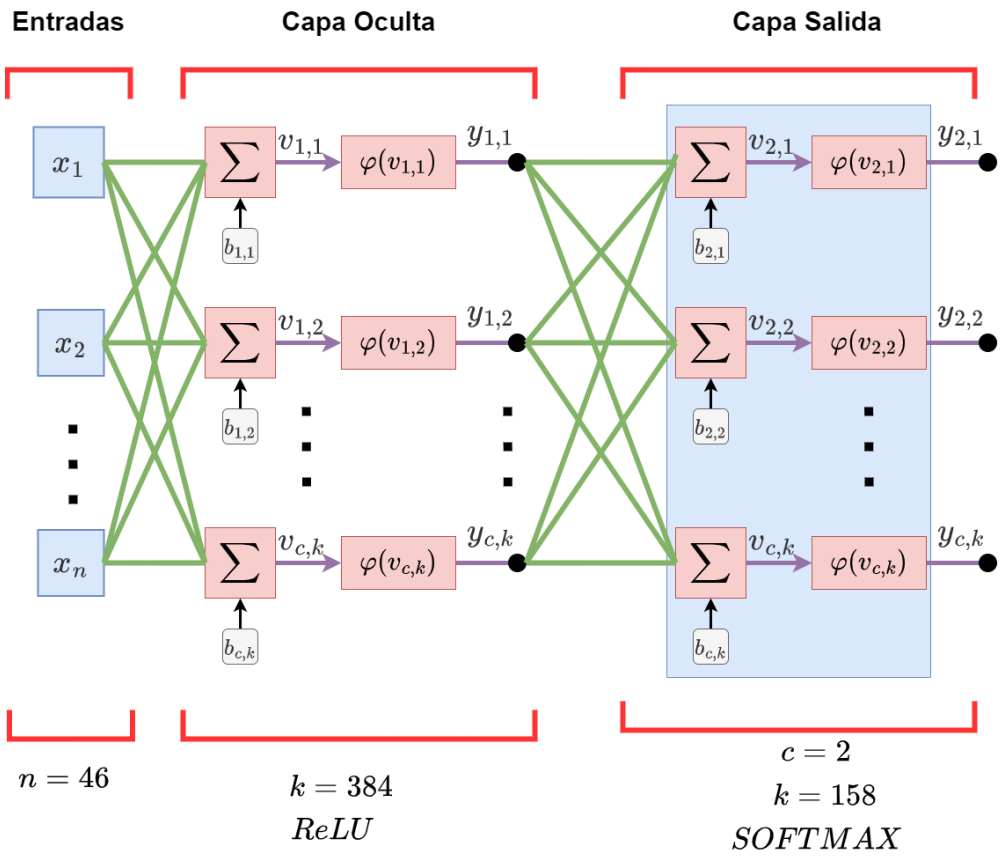

Figura 4.23 **Arquitectura de red neuronal artificial convencional con función de activación ReLU.**

4.6.3.3 RNA con neuronas de soporte y PReLU

La Figura 4.24 describe la estructura de una de las configuraciones propuesta basada en el modelo de RNA con neuronas de soporte y funciones de activación paramétricas. La RNA se compone de una capa de entrada y dos capas totalmente conectadas de neuronas. La capa de entrada recibe un vector de entrada con un tamaño de 46 características. Cada capa de neuronas utiliza un número distinto de unidades e implementa una función de activación diferente. En la primera capa, se encuentran 768 unidades en total, de las cuales 384 son neuronas estándar y 384 neuronas de soporte. Las neuronas estándar utilizan la función de activación PReLU. La segunda capa, que es la capa de salida, está compuesta por 158 neuronas, 0 neuronas de soporte y usa la función de activación softmax. Este modelo tiene como objetivo ajustar la pendiente en la parte negativa de la función ReLU para obtener un modelado más adecuado. En este caso, cada neurona en la capa oculta realiza un modelado distinto con respecto a las otras neuronas de la misma capa.

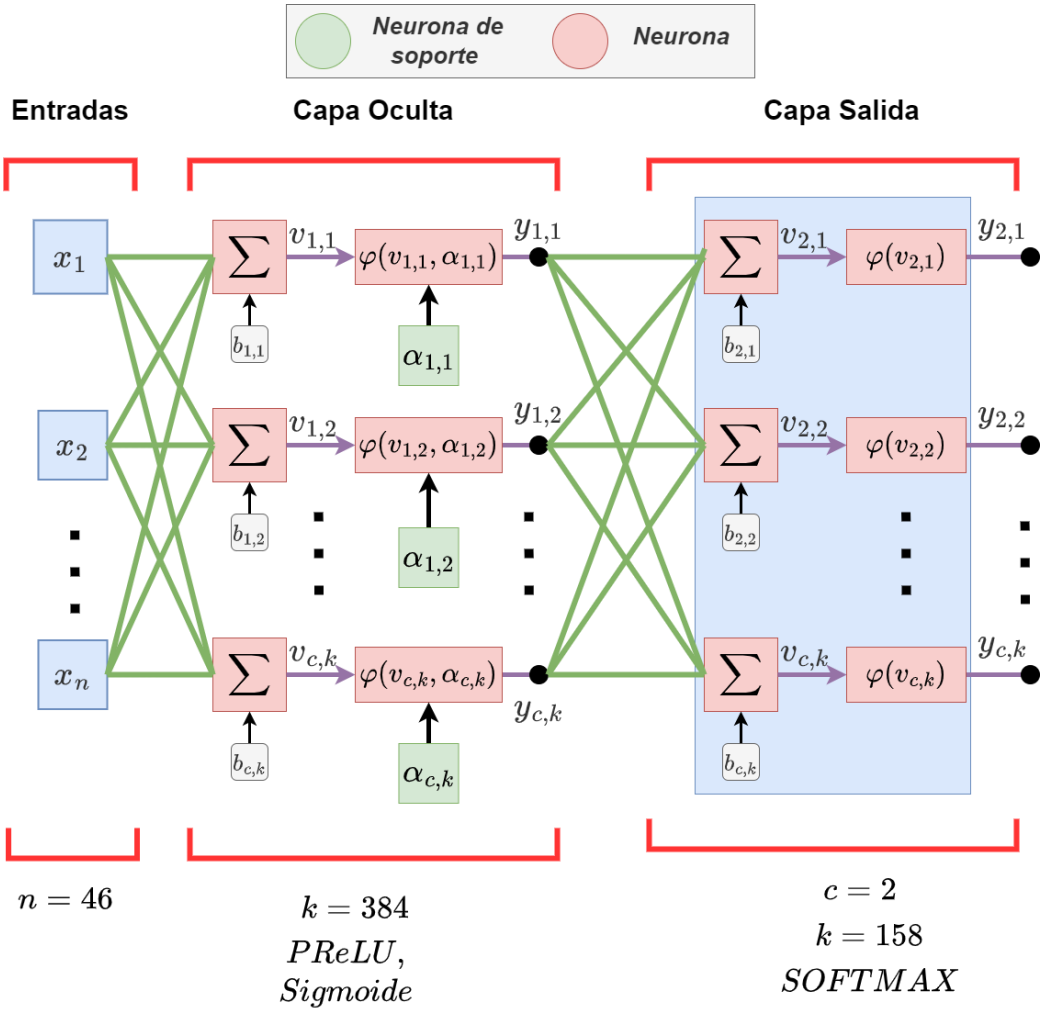

Figura 4.24 **Arquitectura de red neuronal artificial con función de activación PReLU y neuronas de soporte.**

La función de activación PReLU tiene un parámetro α entrenable y único para cada neurona, lo que significa que cada neurona convencional tendrá un valor de alfa distinto. Además, en la capa oculta, cada neurona convencional está asociada a una neurona de soporte para optimizar su propio parámetro α respectivo. Las neuronas de soporte se inicializan con 384 pesos y un sesgo, y no reciben señales de otras neuronas. Por lo tanto, su entrada asociada tiene automáticamente un valor de 1. Finalmente, las neuronas de soporte utilizan la función sigmoide como función de activación no paramétrica. Al utilizar la función sigmoide se está delimitando la búsqueda del paramero alfa en el rango $[0,1]$.

Por otra parte, los pesos de las neuronas de soporte y neuronas estándar se inicializan de forma aleatoria utilizando el método de Xavier y una distribución uniforme. Los sesgos de todas las unidades de procesamiento se inicializan con un valor de 0. Para entrenar la RNA con neuronas de soporte y PReLU, se emplea el algoritmo del gradiente estocástico descendente. Se utiliza una tasa de aprendizaje de 0.01 para las neuronas convencionales y de 0.0009 para las neuronas de soporte. Por último, se realizan 15 épocas de entrenamiento. Debido a que es un problema de clasificación, se usa la función de pérdida de entropía cruzada.

4.6.3.4 RNA con neuronas de soporte y MPReLU

El modelo de RNA presentado en la Figura 4.25 corresponde a una segunda propuesta que utiliza neuronas de soporte junto con la función MPReLU.

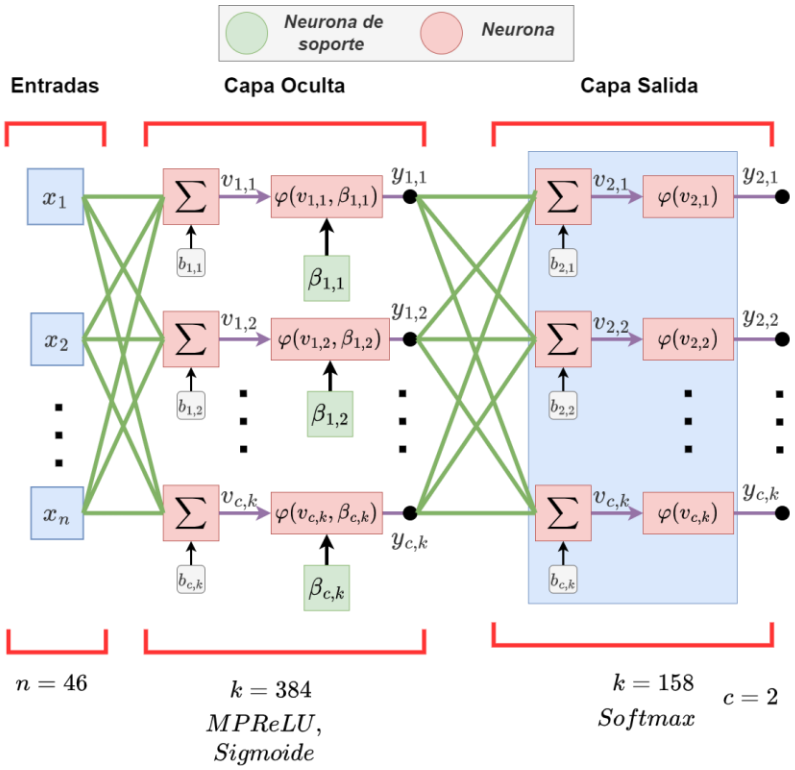

Figura 4.25 Arquitectura de red neuronal artificial con función de activación MPReLU y neuronas de soporte.

Esta RNA con neuronas de soporte y MPReLU se compone de una capa de entrada y dos capas de neuronas conectadas. La primera capa de neuronas consta de 768 unidades, con 384 neuronas estándar y 384 neuronas de soporte que utilizan la función de activación MPReLU y la función de activación sigmoide, respectivamente. La segunda capa de salida consta de 158 neuronas y utiliza la función de activación softmax. El enfoque actual se centra en ajustar exclusivamente la pendiente positiva de la función ReLU. Ahora la función de activación incluye un parámetro β (en lugar de alfa) que también puede ser entrenado. El comportamiento de este parámetro y su proceso de optimización es parecido al de una RNA con neuronas de soporte y la función PReLU. Por último, el rango de búsqueda definido de β también se encuentra en el rango $[0,1]$.

Las neuronas de soporte se inician con 384 pesos y un sesgo. Su entrada asociada se establece automáticamente en 1 y los sesgos se fijan en 0. Tanto las neuronas de soporte como las convencionales tienen pesos iniciales aleatorios, los cuales se generan utilizando el método de Xavier y una distribución uniforme. La RNA-NS se entrena mediante el algoritmo del gradiente estocástico descendente durante 15 épocas, con tasas de aprendizaje de 0.01 para las neuronas convencionales y de 0.0009 para las neuronas de soporte. Finalmente, se usa la función de pérdida de entropía cruzada para calcular el error cometido por la RNA-NS.

4.6.3.5 RNA con neuronas de soporte y AReLU

La estructura representada en la Figura 4.26 ilustra una tercera configuración propuesta del modelo de RNA que emplea neuronas de soporte y funciones de activación paramétricas. El modelo de RNA también incluye una capa de entrada seguida de dos capas completamente conectadas de neuronas. La capa de entrada recibe un vector de entrada con una dimensión de 46 elementos. Cada capa de neuronas utiliza una cantidad distinta de unidades y aplica una función de activación específica a cada tipo de neurona. En la primera capa, se encuentran en total $1{,}152$ unidades; de las cuales 384 son neuronas estándar y 768 neuronas de soporte. Las neuronas estándar utilizan la función de activación AReLU. Por otro lado, la segunda capa, que corresponde a la capa de salida, está conformada por 158 neuronas, sin ninguna neurona de soporte, y utiliza la función de activación softmax.

En este caso, la función de activación AReLU cuenta con dos parámetros entrenables, α y β, los cuales también son únicos para cada neurona. Esto implica que cada neurona convencional poseerá diferentes valores para beta y alfa respecto a las otras neuronas de la misma capa. En la capa oculta, cada neurona convencional está vinculada a dos neuronas de soporte que optimizan sus propios parámetros α y β. Cada parámetro es optimizado por una neurona de soporte específica. Al igual que las configuraciones anteriores, las neuronas de soporte se inician con 384 pesos y un sesgo, y no reciben señales de otras neuronas, lo que también resulta en una entrada asociada que se establece automáticamente en 1.

Además, estas neuronas emplean la función sigmoide como su función de activación no paramétrica, limitando el rango de búsqueda de alfa y beta entre 0 y 1. Para este modelo, se busca ajustar las pendientes tanto en la parte positiva como en

la negativa de la función ReLU para lograr un modelado más preciso de los datos. Debido a que este ajuste varía en cada neurona, podría considerarse como tener diferentes funciones de activación por capa, aunque en realidad son variaciones de una función específica.

Además, los pesos de las neuronas de soporte y neuronas estándar se inicializan de manera aleatoria utilizando el método de Xavier y una distribución uniforme. Los sesgos de las neuronas se establecen inicialmente en 0. Durante el proceso de entrenamiento de la RNA con neuronas de soporte, se emplea el algoritmo del gradiente estocástico descendente. Para las neuronas convencionales, se utiliza una tasa de aprendizaje de 0.01, mientras que, para las neuronas de soporte, se utiliza una tasa de aprendizaje de 0.0009. Por último, se llevan a cabo 15 épocas de entrenamiento usando la función de pérdida de entropía cruzada.

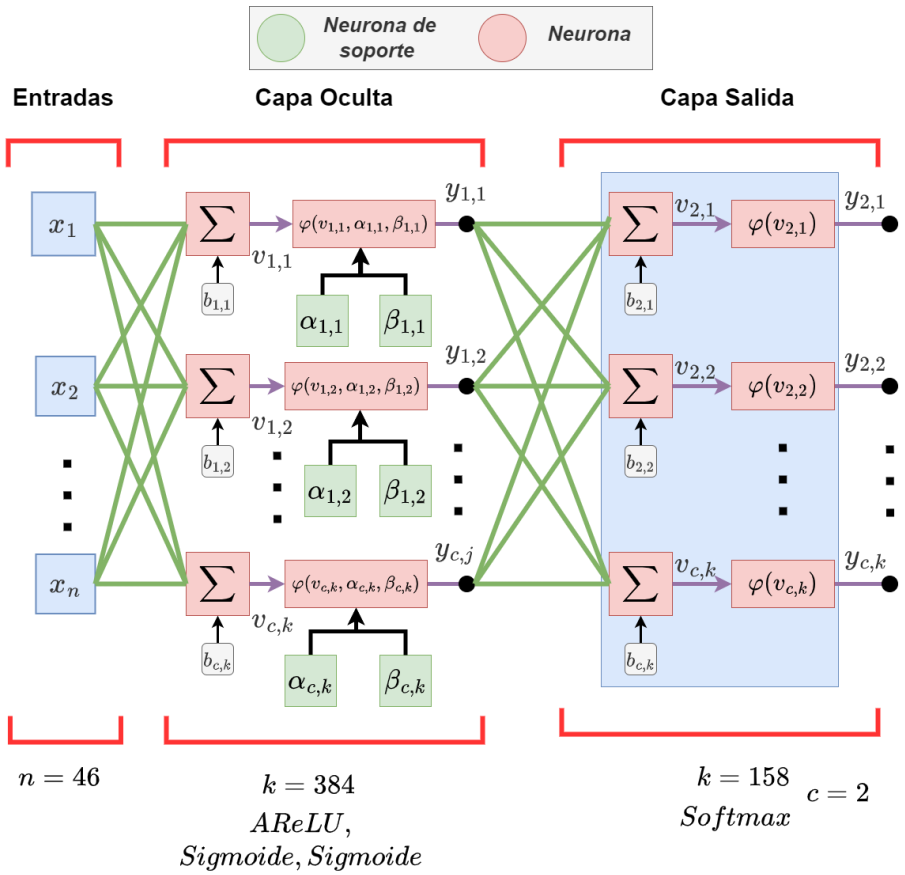

Figura 4.26 Arquitectura de red neuronal artificial con función de activación AReLU y neuronas de soporte.

4.6.4. Clasificación de archivos de audio

En este caso, se extraen las tramas (muestras) de cada archivo de audio y después se clasifican de manera independiente. Finalmente, se aplica la técnica de

voto mayoritario para clasificar el archivo de audio según la clase que esté más presente en la clasificación de esas muestras. La técnica de voto mayoritario permite proporcionar una clasificación basada en la opinión (votación) de otros clasificadores, es decir, que el modelo seleccionará la clase (locutor) con más ocurrencias en las tramas extraídas de la grabación, esto tiene la finalidad de mejorar el puntaje de clasificación, reducir problemas de memorización y que el modelo tenga mejor capacidad de generalización [110].

Capítulo 5. Resultados

El conjunto de datos, que ha sido preprocesado y descrito en las secciones anteriores, se utilizó para entrenar los modelos de clasificación propuestos. La exactitud y el puntaje F1 [111] se seleccionaron como las principales métricas de evaluación. Se utilizaron el recall y la precisión [111] como métricas adicionales en el análisis de datos.

En este trabajo se evalúa el desempeño de los modelos desarrollados en tareas de clasificación de tramas de voz, así como de clasificación de archivos de audio.

5.1. Tiempos de ejecución

La fase de extracción/selección de características y los diferentes modelos implementados de RNA se llevaron a cabo en una computadora Alienware x17 R2 con Windows 11. Este equipo de cómputo cuenta con un procesador 12th Gen Intel(R) Core (TM) i7-12700H a 2.3 GHz y 32 GB de RAM DDR5 a 4,800 MHz. Para acelerar el rendimiento de los modelos de KNN y SBS, se utilizó una tarjeta NVIDIA GeForce RTX 3080 Ti con 16 GB de memoria GDDR6 y 10,240 núcleos CUDA de procesamiento. También, se utilizaron la biblioteca CUDA 12.0 y Rapids 23.04 en Ubuntu 22.04.2 LTS mediante wsl para acelerar las operaciones matriciales.

La Tabla 5.1 muestra el tiempo de ejecución correspondiente a las fases de entrenamiento y de predicciones (en validación y pruebas) de cada modelo implementado en tareas de clasificación de tramas de voz. Donde el modelo de SBS solo se ejecutó en el conjunto de entrenamiento.

Tabla 5.1 Tiempos de ejecución (en segundos) en actividades de entrenamiento y predicción de tramas de los diferentes esquemas de RNA.

Modelo	Entrenamiento	Validación	Pruebas
SBS.	36,842.4882	No aplica	No aplica
KNN con 126 características.	89.8235	20.4360	20.8416
KNN con 46 características.	**6.7742**	**1.6877**	**1.8400**
RNA con ReLu.	**5,833.4268**	**18.2566**	**18.1478**
RNA con MPReLu y usando neuronas de soporte.	8,056.3175	102.1927	111.3356
RNA con AReLu y usando neuronas de soporte.	10,398.1446	71.6324	66.1856
RNA con PReLu y usando neuronas de soporte.	8,579.1032	80.5841	96.8911

La Tabla 5.2 presenta los tiempos de ejecución para la fase de entrenamiento y prueba de cada modelo implementado en las tareas de clasificación de archivos de voz utilizando la técnica de voto mayoritario.

Tabla 5.2 Tiempos de ejecución (en segundos) de los diferentes esquemas de redes neuronales artificiales en tareas de clasificación de archivos de audio.

Modelo	Entrenamiento	Validación	Pruebas
RNA con ReLu.	**5,833.4268**	**17.5524**	**17.7801**
RNA con MPReLu y usando neuronas de soporte.	8,056.3175	87.6023	74.4506
RNA con AReLu y usando neuronas de soporte.	10,398.1446	51.9547	50.7540
RNA con PReLu y usando neuronas de soporte.	8,579.1032	65.8540	65.1504

5.2. Resultados de KNN y SBS en la fase de selección de características

Las Tablas 5.3 y 5.4 presentan los resultados obtenidos por el modelo KNN y SBS antes y después de llevar a cabo la selección de características, respectivamente. Se puede ver que KNN obtiene mejores resultados al usar 46 características.

Tabla 5.3 Clasificación de muestras de voz utilizando KNN con 46 características acústicas.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9929	0.6674	0.6660
Precisión.	0.9940	0.6641	0.6618
Recall.	0.9943	0.6284	0.6252
F1-score.	0.9941	0.6199	0.6173

Al analizar y comparar en detalle las Tablas 5.3 y 5.4, se puede observar que el modelo KNN muestra un incremento en el sub-ajuste durante la fase de entrenamiento. Sin embargo, en las etapas de validación y pruebas, el modelo tiende a tener una mejor capacidad de generalización presentando un menor sobre-aprendizaje y sub-ajuste.

Tabla 5.4 Clasificación de muestras de voz utilizando KNN con las 46 características acústicas más relevantes.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9647	0.7810	0.7741
Precisión.	0.9624	0.7527	0.7432
Recall.	0.9665	0.7603	0.7479
F1-score.	0.9641	0.7464	0.7356

Realizar una selección de características, conservando únicamente aquellas variables que aportan información relevante, ha demostrado mejorar la capacidad de

generalización de los modelos, aunque con un ligero incremento en el sobre-aprendizaje. Este fenómeno es evidente en la Figura 5.1, donde se aplicaron de forma anidada los modelos KNN y SBS. En este sentido, en la Figura 5.1 se observa que, al utilizar una combinación de más de 46 variables de predicción, el modelo KNN tiende a perder su capacidad de abstracción, y lo mismo ocurre al utilizar una combinación de características menor a 46.

Una ventaja adicional al realizar esta selección de variables radica en que se obtienen modelos menos complejos, los cuales convergen más rápidamente y requieren menos tiempo computacional como se aprecia en la Tabla 5.2. Finalmente, la combinación y solución óptima de características relevantes, siendo un conjunto de 46 variables, se presenta en la Tabla 4.2.

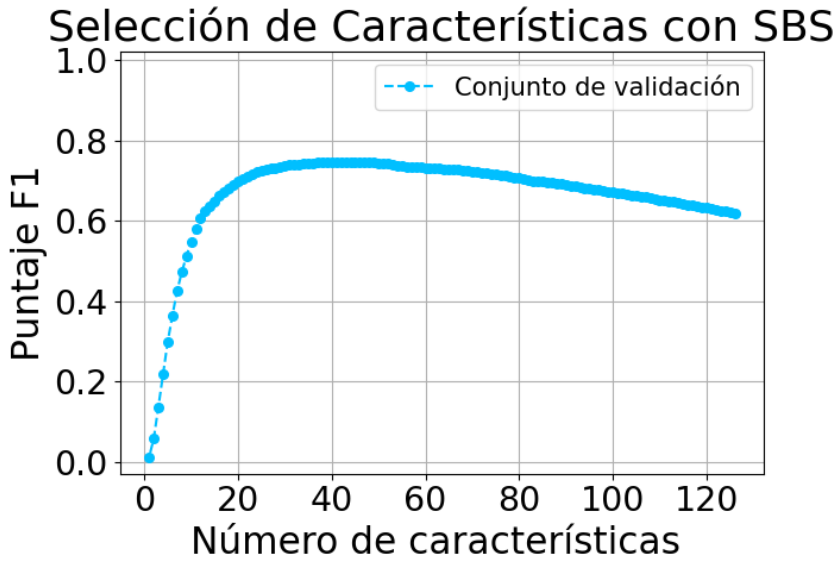

Figura 5.1 Rendimiento del modelo KNN en el proceso de selección de características con SBS.

5.3. Resultados de la red neuronal artificial convencional con función de activación ReLU

La Tabla 5.5 ilustra el rendimiento promedio de todas las clases en la identificación de tramas de voz por hablante al utilizar una RNA convencional que implementa neuronas ReLU en las capas ocultas y una capa de salida Softmax. El modelo fue entrenado con las 46 características más representativas. Este modelo, durante la etapa de entrenamiento, logra clasificar correctamente (exactitud) el 93.98% de las tramas, de un total de $429{,}309$ muestras. Para la fase de validación y pruebas, el puntaje obtenido es de 85.06% y 86.75%, respectivamente.

Por otra parte, cuando el clasificador predice que una trama pertenece a un hablante específico (precisión), acierta el 83.38% de las veces en la fase de validación

y el 85.40% en pruebas. En adicción, el esquema aprende a realizar esta tarea con una precisión del 94.38%. Siendo capaz de distinguir las tramas de otros locutores (recall) en un 93.95% de los casos. Durante la etapa de validación, logra esta distinción en un 82.78% y en pruebas alcanza un 85.54% de aciertos en esta tarea. Finalmente, logra identificar y detectar las muestras con sus respectivos locutores en un 99.83%. En la fase de validación y pruebas, logra mantener este equilibrio en un 82.32% y 84.87% respectivamente.

El modelo de RNA convencional muestra un bajo sub-jauste y un alto sobre-aprendizaje al identificar correctamente las tramas con sus respectivos locutores. Sin embargo, es normal que se presenten problemas de sobre-aprendizaje en un escenario real (fase de pruebas), ya que los datos suelen contener otros patrones o, en este caso, diferentes tonos de voz o tipos de ruido.

Tabla 5.5 Clasificación de muestras de voz utilizando un esquema de RNA convencional con función de activación ReLU.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9398	0.8506	0.8675
Precisión.	0.9438	0.8338	0.8540
Recall.	0.9395	0.8278	0.8554
F1-score.	0.9404	0.8232	0.8487

La Tabla 5.6 muestra los resultados del modelo al clasificar archivos de audio con sus respectivos locutores utilizando la técnica de voto mayoritario sobre una clasificación individual de tramas de voz. La exactitud del clasificador, medida como el porcentaje de predicciones correctas sobre el total de archivos de audio, es del 99.88% en el período de entrenamiento, 97.49% en la etapa de validación y 97.88% en la fase de pruebas.

Tabla 5.6 Clasificación de archivos de audio utilizando un esquema de RNA convencional con función de activación ReLU.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9988	0.9749	0.9788
Precisión.	0.9983	0.9833	0.9835
Recall.	0.9984	0.9693	0.9759
F1-score.	0.9983	0.9660	0.9739

En cuanto a la precisión del modelo, en la fase de entrenamiento se alcanza un valor de 0.9983, lo que significa que cuando el modelo predice que un archivo de audio pertenece a un hablante específico, acierta el 99.83% de las veces. Para la etapa de validación, se obtiene una confiabilidad de 98.33%, mientras que en la fase de pruebas se logra una confiabilidad del 98.35%. En resumen, el modelo demuestra una alta precisión en la clasificación de los archivos de audio en todas las etapas, con valores superiores al 97% en términos de exactitud. Finalmente, el algoritmo muestra una alta confiabilidad en la predicción de los locutores, con porcentajes superiores al

98%.

La Tabla 5.6 también indica que el modelo logra diferenciar las clases en entrenamiento con un puntaje de recall del 99.84%, en validación con un 96.93% y en pruebas con un 97.59%. La RNA logra identificar y diferenciar con alta confianza (puntaje F1, un equilibrio entre precisión y recall), al aprender a clasificar archivos de audio, con un puntaje F1 del 99.83%. Este equilibrio de confianza se refleja en un puntaje F1 del 96.60% para la validación y del 97.39% en las pruebas.

Al comparar las Tablas 5.5 y 5.6, podemos observar que, al realizar una predicción final del archivo de audio mediante la técnica de voto mayoritario, el modelo logra una mejor abstracción de los datos, lo que disminuye los problemas de sobre-aprendizaje y sub-ajuste. No obstante, aún se presentan problemas leves de memorización en el modelo. Mientras que, al realizar una clasificación por tramas de voz, existen ligeros problemas de alto sesgo y se presentan problemas alta varianza, es decir, problemas de sub-ajuste y sobre-aprendizaje.

5.4. Resultados de la red neuronal artificial con función de activación PReLU y neuronas de soporte

El desempeño de la RNA con PreLU y neuronas de soporte en la clasificación de tramas de voz se muestra en la Tabla 5.7. En este caso, ajustar la pendiente negativa de la función base (ReLU) no mejora significativamente el rendimiento del modelo. El modelo logra clasificar correctamente las tramas de voz en un 94.39%, 86.13% y 87.29% para los conjuntos de entrenamiento, validación y pruebas, respectivamente. En otras palabras, este esquema también presenta buena capacidad de abstracción.

Se consigue una confiabilidad del 94.80%, 84.30% y 86.31% al clasificar estas tramas en los mismos conjuntos. La capacidad de la RNA-NS para distinguir entre las diferentes clases de muestras también es baja, ya que obtiene un recall de 94.09% en entrenamiento, 83.45% en validación y 85.53% en pruebas. Finalmente, con esta configuración se logra un promedio entre confiabilidad y capacidad de distinción entre hablantes del 94.31% en entrenamiento, 83.17% en validación y 85.23% en pruebas.

Tabla 5.7 Clasificación de muestras de voz utilizando el esquema de RNA con función de activación PReLU y neuronas de soporte.

Métricas	Entrenamiento	Validación	Pruebas
Exactitud.	0.9439	0.8613	0.8729
Precisión.	0.9480	0.8430	0.8631
Recall.	0.9409	0.8345	0.8553
F1-score.	0.9431	0.8317	0.8523

De manera similar a los modelos y experimentos anteriores, se puede observar que el modelo mejora su capacidad de clasificación al utilizar la técnica de voto mayoritario después de clasificar las tramas individualmente de un archivo de audio, como se muestra en la Tabla 5.8. El clasificador tiene una alta tasa de aciertos en sus

predicciones. En el corpus de entrenamiento, el esquema logra acertar correctamente el 99.88% de las predicciones, mientras que en las etapas de pruebas y validación obtiene un 96.83% y un 97.88% de aciertos, respectivamente. Estos aciertos son en relación con el total de ejemplos. La precisión de estas predicciones es muy alta, alcanzando el 99.87% en entrenamiento, el 97.78% en validación y el 98.44% en pruebas. En cuanto a la capacidad del modelo para reconocer cada clase individualmente, se obtienen valores elevados de recall, con un 99.86% en entrenamiento, un 95.03% en validación y un 97.00% en pruebas. En promedio, se logra obtener un equilibrio entre precisión y recall (puntaje F1), con valores del 99.86% en entrenamiento, 94.35% en validación y 96.80% en pruebas.

Tabla 5.8 Clasificación de archivos de audio utilizando el esquema de RNA con función de activación PReLU y neuronas de soporte.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9988	0.9683	0.9788
Precisión.	0.9987	0.9778	0.9844
Recall.	0.9986	0.9503	0.9700
F1-score.	0.9986	0.9435	0.9680

La inactividad de la neurona cuando el potencial de acción es negativo o igual a cero continúa brindando resultados eficientes, lo que indica que no es necesario ajustar la parte negativa de la función ReLU para este problema en particular. Por otro lado, regular la intensidad del potencial de acción cuando es positivo resulta beneficioso para el proceso de modelado, ya que conduce a tasas de error más bajas.

5.5. Resultados de la red neuronal artificial con función de activación MPReLU y neuronas de soporte

En la Tabla 5.9 se puede observar el rendimiento promedio en la identificación de tramas de voz por hablante, mientras que en Tabla 5.10 se aprecian los resultados (macro promedio) de clasificación de archivos de audio por locutor. En ambos casos, se empleó una red neuronal artificial con neuronas de soporte que utilizan la función de activación MPReLU en las capas ocultas y una capa de salida Softmax, cuya estructura se presentó en la Figura 4.25Figura 4.25 Arquitectura de red neuronal artificial con función de activación MPReLU y neuronas de soporte.. En el proceso de entrenamiento del modelo, se utilizaron las 46 características más informativas para mejorar el desempeño y la precisión de las predicciones.

Este esquema de RNA-NS utilizado en la clasificación de tramas de voz (ver Tabla 5.9) muestra un alto rendimiento en los conjuntos de entrenamiento, validación y pruebas. Alcanza una exactitud del 95.66%, 87.50% y 88.89% respectivamente, lo que significa que acierta correctamente un porcentaje similar de las 429,309 muestras totales en cada una de estas fases. El modelo también logra clasificar las tramas con una alta confianza (presión) del 96.05% en el periodo de entrenamiento, del 85.63% en la fase de validación y del 87.77% en la etapa de pruebas. También se aprecia que

llevar a cabo un ajuste en el proceso de modelado individualmente de cada neurona ha permitido que el modelo presente una mayor robustez en señales de voz con ruido.

Tabla 5.9 Clasificación de muestras de voz utilizando el esquema de RNA con función de activación MPReLU y neuronas de soporte.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9566	0.8749	0.8889
Precisión.	0.9605	0.8563	0.8778
Recall.	0.9531	0.8459	0.8709
F1-score.	0.9558	0.8438	0.8693

En términos de diferenciación de locutores (recall), la RNA-NS obtiene una puntuación de 95.31% en el conjunto de entrenamiento, lo que indica su buena habilidad para aprender a distinguir correctamente a los hablantes entre ellos. Sin embargo, es importante destacar que existe cierto grado de memorización en el aprendizaje del modelo. En el dataset de validación, presenta una capacidad de diferenciación de hablantes o recall de 84.60%, mientras que en la fase de pruebas esta capacidad se eleva al 87.09%.

Finalmente, el clasificador logra un equilibrio satisfactorio entre confianza y capacidad de identificación, medido por el puntaje F1, en los corpus de entrenamiento, validación y pruebas. En el conjunto de entrenamiento, alcanza un puntaje F1 del 95.55%, en el dataset de validación es del 84.38% y en el conjunto de pruebas es del 86.93%.

Las neuronas de soporte se muestran útiles para encontrar de manera automática y precisa los valores óptimos de los parámetros de las funciones de activación. Los resultados evidencian una estabilidad en el proceso de aprendizaje del esquema presentado, y se observa que el modelo tiende a converger, logrando una mejor minimización del error del clasificador. Por lo tanto, estas neuronas pueden ser un componente que contribuya al proceso de búsqueda de valores de ciertos parámetros, lo que a su vez puede mejorar el rendimiento de los modelos inteligentes en determinados casos.

Tabla 5.10 Clasificación de archivos de audio utilizando el esquema de RNA con función de activación MPReLU y neuronas de soporte.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9988	0.9722	0.9868
Precisión.	0.9983	0.9815	0.9910
Recall.	0.9977	0.9630	0.9838
F1-score.	0.9979	0.9597	0.9828

Por otro lado, al analizar la Tabla 5.10, se puede observar que el esquema de RNA-NS propuesto tiene un rendimiento superior en la tarea de clasificación por archivos. En términos de exactitud, en el corpus de entrenamiento el modelo logra

clasificar correctamente el 99.88% de todas las muestras. En el conjunto de validación, alcanza una tasa de acierto del 97.22% sobre el total de muestras, y en el dataset de pruebas acierta el 98.68% de las predicciones realizadas.

Al realizar una predicción, el modelo muestra una alta fiabilidad (precisión) de 99.83%, 98.15% y 99.10% en los corpus de entrenamiento, validación y pruebas, respectivamente. Asimismo, el modelo demuestra una alta capacidad para distinguir una clase respecto a las demás (recall) con valores de 99.79% en entrenamiento, 95.97% en validación y 98.28% en pruebas. Los problemas de memorización tienden a disminuir con la técnica de voto mayoritario para una clasificación final.

En cuanto a los puntajes F1 obtenidos en los conjuntos de entrenamiento, validación y pruebas, estos son de 99.79%, 95.97% y 98.28% respectivamente. Estos resultados también indican que el modelo posee una alta capacidad para clasificar de manera confiable y acertada. Esta configuración de RNA-NS presenta el mejor desempeño respecto a los otros modelos.

Con base en la información presentada en las Tablas 5.9 y 5.10, se puede deducir que el modelo aborda de manera eficaz los problemas de sesgo y varianza al aplicar la técnica de voto mayoritario para obtener una clasificación final del archivo de audio. Al ajustar exclusivamente la pendiente positiva de la función de activación ReLU, se obtienen resultados superiores en términos de exactitud y un equilibrio mejorado entre confiabilidad y recall. Esto se debe a una reducción ligera en el sobre-aprendizaje y una mayor tasa de clasificaciones correctas en comparación con una arquitectura de RNA convencional. No obstante, esta diferencia se presenta de manera más significativa en la clasificación por muestras y de manera sutil en la clasificación por voto mayoritario (clasificación por archivo de audio).

5.6. Resultados de la red neuronal artificial con función de activación AReLU y neuronas de soporte

La Tabla 5.11 describe el rendimiento del modelo de RNA con AReLU y neuronas de soporte en la clasificación de tramas de voz. El clasificador logra identificar correctamente el 95.69% de las tramas de voz en el conjunto de entrenamiento. En el conjunto de validación, el modelo es correcto el 87.05% de las veces que predice, mientras que en el conjunto de pruebas acierta el 88.41% de las veces.

Al realizar una predicción (ver Tabla 5.11), la RNA-NS muestra una alta fiabilidad, con una confianza de acierto del 96.19%, 85.44% y 87.24% en los conjuntos de entrenamiento, validación y pruebas, respectivamente. También logra distinguir una clase de las demás con un recall del 95.71% en el corpus de entrenamiento, del 85.05% en el dataset de validación y del 87.26% en el conjunto de pruebas. Finalmente, consigue un puntaje promedio entre confiabilidad y capacidad de distinción entre clases del 95.90% en fase de entrenamiento, 84.72% en la etapa de validación y 86.73% en el subconjunto de pruebas.

Por otra parte, se presentan los resultados obtenidos por la RNA con AReLU y neuronas de soporte en la clasificación de archivos por locutor en la Tabla 5.12. En

este caso, la frecuencia con que la RNA-NS acierta correctamente una predicción es del 99.94%, 97.09% y 98.41% en las fases de entrenamiento, pruebas y validación, respectivamente. Lo resultados anteriores son en relación del total de ejemplos. También, la posibilidad (precisión) de que estas predicciones sean correctas es del 99.92% en el corpus de entrenamiento, del 98.0661% en el conjunto de validación y del 98.43% en el dataset de pruebas. Con una capacidad promedio del modelo para reconocer cada clase individualmente (recall) del 99.91% en entrenamiento, 95.87% en validación y 98.39% en pruebas. El promedio armónico del clasificador entre precisión y recall es del 99.91% en entrenamiento, 95.44% en validación y 98.34% en pruebas, respectivamente.

Tabla 5.11 Clasificación de muestras de voz utilizando el esquema de RNA con función de activación AReLU y neuronas de soporte.

Métricas	Entrenamiento	Validación	Pruebas
Exactitud.	0.9569	0.8705	0.8841
Precisión.	0.9619	0.8544	0.8724
Recall.	0.9571	0.8505	0.8726
F1-score.	0.9590	0.8472	0.8673

Tabla 5.12 Clasificación de archivos de audio utilizando el esquema de RNA con función de activación AReLU y neuronas de soporte.

Métrica	Entrenamiento	Validación	Pruebas
Exactitud.	0.9994	0.9709	0.9841
Precisión.	0.9992	0.9806	0.9843
Recall.	0.9991	0.9587	0.9839
F1-score.	0.9991	0.9544	0.9834

Formar un conjunto de datos equilibrado, con una mayor cantidad de archivos de audio y de locutores, puede contribuir a reducir los problemas de generalización y memorización del modelo al clasificar tramas de voz. El garantizar la igualdad de condiciones en las fases de entrenamiento, validación y pruebas puede ayudar a mitigar los problemas de sobre-aprendizaje. Esto implica que los modelos deben enfrentarse a los mismos tipos de ruido ambiental durante la validación y las pruebas que ha experimentado durante su proceso de aprendizaje. La falta de robustez en el conjunto de datos y una división aleatoria del mismo no garantizan las condiciones de igualdad mencionadas.

5.7. Comparación de resultados

La Figura 5.2 presenta una comparación del rendimiento de los modelos en la clasificación de tramas de voz durante la fase de pruebas. Se observa que todos los modelos presentan problemas de sobre-aprendizaje y un ligero problema de generalización. Sin embargo, el modelo que exhibe menor sub-ajuste y menos problemas de memorización es la RNA con MPReLU y neuronas de soporte, ya que

logra clasificar correctamente una mayor cantidad de muestras, alcanzando una exactitud del 88.89%. La RNA con MPReLU y neuronas de soporte es el modelo que tiene una mayor robustez al modelar datos con ruido y de mala calidad.

Los problemas de sobre-aprendizaje que se observan pueden atribuirse a que el modelo, en un escenario real (como en este caso de prueba), se enfrenta a variaciones en el tono de voz, diferentes tipos de ruido y cambios en la calidad del sonido que no fueron adecuadamente identificados durante la fase de entrenamiento. Esto puede ser resultado de una división aleatoria del conjunto de datos y de la falta de robustez de este. Como solución, resulta necesario disponer de un corpus de datos más robusto que garantice condiciones equitativas tanto en la fase de entrenamiento como en la de pruebas. No obstante, es común que los algoritmos inteligentes se enfrenten a desafíos de este tipo al ser aplicados en escenarios reales, donde los patrones de datos varían y se presentan cambios en el tono de voz y distintos tipos de ruido que el clasificador no ha tenido oportunidad de conocer previamente. Por tanto, es normal que los modelos muestren un rendimiento inferior en un escenario real. A pesar de enfrentarse a estos cambios en los patrones y presencia de ruido, los experimentos propuestos siguen demostrando un buen desempeño.

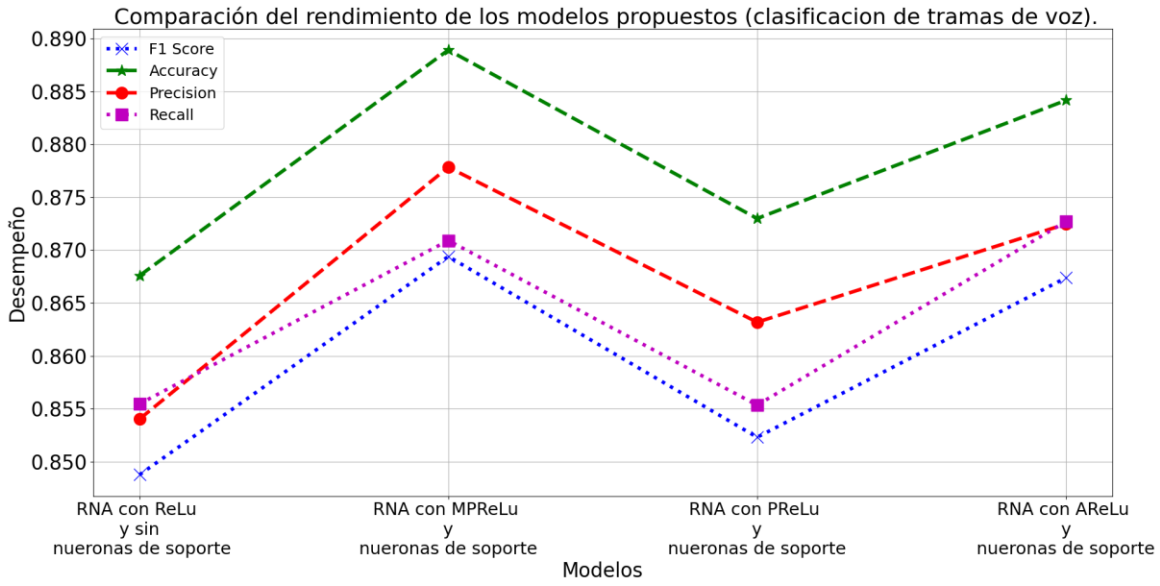

Figura 5.2 Comparación del rendimiento de los modelos en la clasificación de tramas de voz durante la fase de pruebas.

Por otro lado, en la Figura 5.3 se muestra una comparación del rendimiento de los modelos en la clasificación de archivos de voz durante la fase de pruebas utilizando la técnica de voto mayoritario después de una clasificación previa por tramas. La aplicación de la técnica de voto mayoritario ayudó a reducir los problemas de sesgo y varianza, a la vez que proporcionó mayor robustez a los clasificadores. Entre los algoritmos inteligentes que fueron evaluados, el modelo de RNA con MPReLU y

neuronas de soporte continuó siendo el modelo con mejor desempeño, logrando este buen desempeño con una exactitud del 98.68%.

En tareas de clasificación por tramas, se observa que la RNA con MPReLU y neuronas de soporte supera a la RNA convencional con una ventaja de alrededor del 2.1% de exactitud. Al aplicar la técnica de voto mayoritario en la clasificación por archivo de audio, esta diferencia se reduce aproximadamente al 0.8% de exactitud. Los resultados obtenidos de reducción del error (exactitud) superan el mínimo esperado (0.5%), logrando de esta manera el objetivo de reducción del error de los RNA en tareas de identificación de locutor en escenarios ruidosos. Las mejoras propuestas permiten que las RNA logren una mayor tasa de predicciones correctas.

Basándonos en la métrica de exactitud y los resultados presentados en las Figuras 5.2 y 5.3, el clasificador de RNA con MPReLU y neuronas de soporte ha sido seleccionado como el clasificador óptimo para tareas de clasificación de locutores en archivos de audio en el idioma español con de diferentes tipos de ruido ambiental. Se seleccionó la métrica de exactitud como principal debido a que permite realizar una comparación más equitativa entre las RNA tradicionales y las RNA propuestas. La exactitud mide la proporción de clasificaciones correctas sobre el total de muestras existentes en el corpus de datos, lo cual nos brinda una medida objetiva del desempeño de los modelos bajo este contexto de rendimiento.

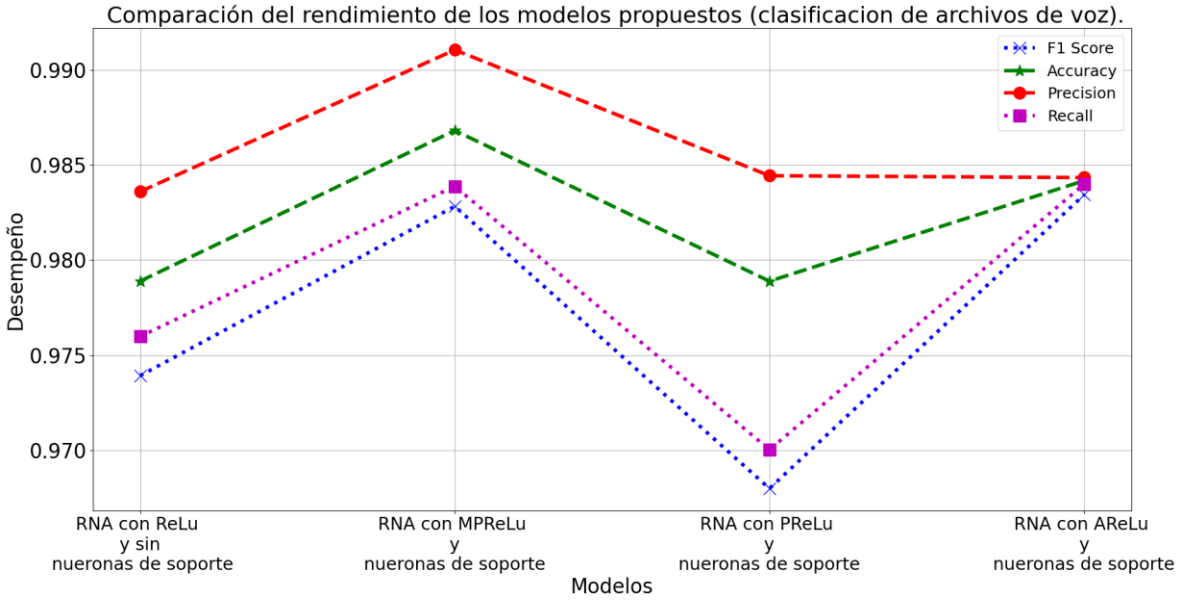

Figura 5.3 Comparación del rendimiento de los modelos en la clasificación de archivos de voz durante la fase de pruebas.

La Figura 5.4 presenta la comparación del proceso de minimización del error (loss) en el conjunto de entrenamiento después de 15 épocas, mientras que la Figura 5.5 ilustra el mismo proceso en el dataset de validación. Los esquemas que logran una mejor minimización del error en el conjunto de entrenamiento son la RNA con

MPReLU y neuronas de soporte, y la RNA con AReLu y neuronas de soporte, alcanzando el valor mínimo de error en 15 épocas. Por otro lado, en el corpus de validación, los modelos tienden a converger entre las 6 o 7 épocas de entrenamiento, pero después de esta convergencia, el error tiende a incrementar en cada modelo.

En el conjunto de validación, los clasificadores que demuestran una mayor estabilidad, un menor incremento del error después de la convergencia y una mejor minimización del error son la RNA con MPReLU y neuronas de soporte, así como la RNA con AReLU y neuronas de soporte. Por otra parte, la arquitectura de RNA convencional que utiliza la función ReLU tanto en el corpus de entrenamiento como en el de pruebas muestra una mayor tasa de error, mayores problemas de sobre-ajuste y sub-ajuste. Se puede ver que realizar ajustes en la parte positiva de la función ReLU permite obtener un mejor modelado de los datos. También se aprecia que regular el potencial de acción mediante la función de activación ReLU y parámetros entrenables puede ser una alternativa para mejorar los problemas de sobre-ajuste y sub-ajuste. Sin embargo, la mejora en ambos problemas es sutil para esta situación.

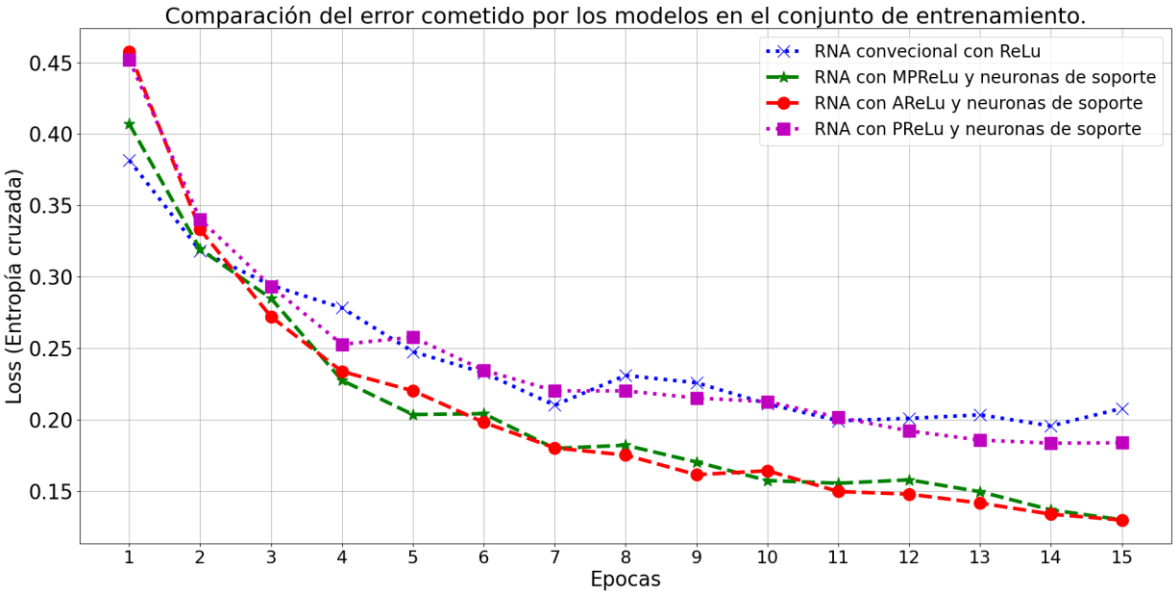

Figura 5.4 Comparación del error cometido por los modelos de RNA en el conjunto de entrenamiento.

En esta comparación de rendimiento se aprecia que la regulación del potencial de acción por medio de funciones de activación paramétricas añade a las redes neuronales artificiales la capacidad de procesar información ruidosa o con un mal procesamiento de datos, desempeñándose ligeramente mejor a una RNA con funciones de activación no paramétricas. Esto podría habilitar el uso de datos sin procesar o ser un correctivo de última instancia en datos de mala calidad. Sin embargo, este análisis recalca la importancia de un buen procesamiento de los datos, por lo que también es necesario añadir componentes a las RNA que ayuden a filtrar

de manera automática la información relevante, eliminando toda aquella información que afecte el proceso de modelado. Un ejemplo de esto son las redes neuronales convolucionales. Por último, este análisis y comparación de rendimiento también recuerda que crear modelos más robustos en el manejo de información de mala calidad añade un mayor costo computacional y tiempo de ejecución, por lo que también se presenta el desafío de desarrollar algoritmos optimizados y simplificados.

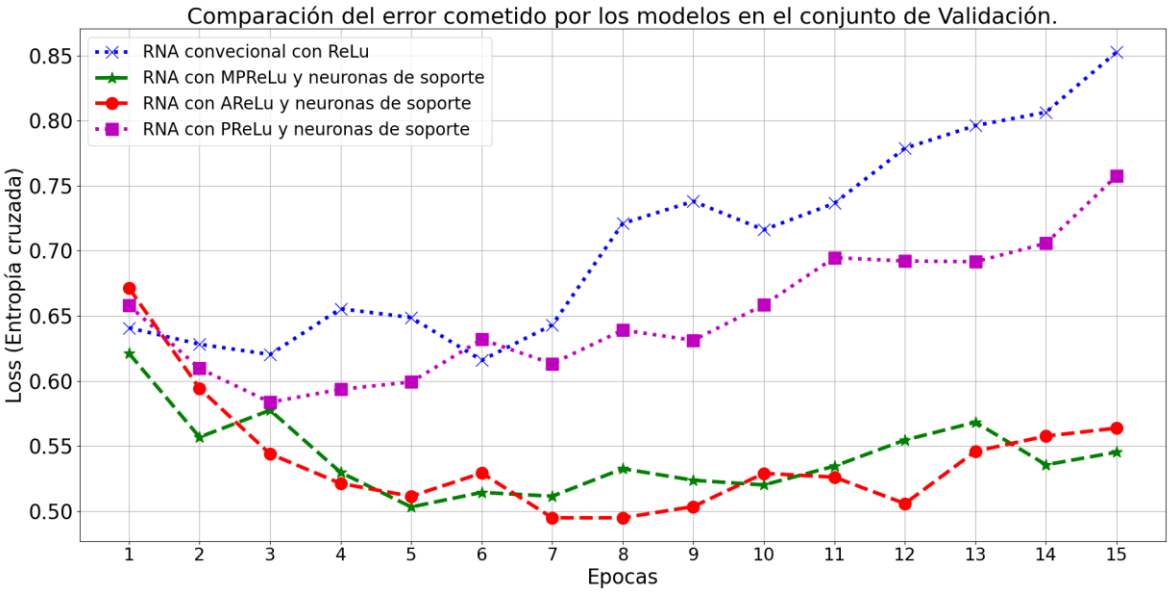

Figura 5.5 Comparación del error cometido por los modelos de RNA en el conjunto de validación.

Tabla 5.13 Comparación del error (loss) cometido por los diferentes esquemas de redes neuronales artificiales.

Modelo	Entrenamiento	Validación
RNA con ReLu.	0.2080	0.8526
RNA con MPReLu y usando neuronas de soporte.	0.1300	0.5451
RNA con AReLu y usando neuronas de soporte.	0.1298	0.5636
RNA con PReLu y usando neuronas de soporte.	0.1839	0.7575

En la Tabla 5.13 se puede ver el error (loss) cometido por los modelos durante el proceso de entrenamiento en los conjuntos de entrenamiento y validación. El modelo de RNA con MPReLu y neuronas de soporte fue el que obtuvo los puntajes de error más bajos, con un loss de 0.1300 en el corpus de entrenamiento y 0.5451 en el dataset de validación. Al utilizar variantes de funciones de activación paramétricas y neuronas de soporte, es posible reducir significativamente el error cometido en comparación con las RNA convencionales y lograr un modelado más ajustado al comportamiento de los datos que pueden contener ruido después de un buen pre-procesamiento. Se logra reducir la tasa de error en un 0.078 en la fase de entrenamiento y un 0.3075 en la etapa de validación.

5.8. Discusión

Usar neuronas artificiales con el propósito exclusivo de entrenar los parámetros de las funciones paramétricas, llamadas neuronas de soporte, puede ser un enfoque eficiente para ajustar dichos parámetros de forma automática y simultánea. Esto evita la necesidad de ajustarlos manualmente, lo que puede mejorar el rendimiento de las redes neuronales convencionales en ciertos casos. No obstante, este enfoque también presenta desafíos, como la complejidad computacional y el desafío adicional de encontrar valores óptimos para los parámetros. Esto se debe a que el entrenamiento simultáneo de los parámetros se convierte en un problema más amplio y complejo de búsqueda.

En esta nueva propuesta es importante destacar la importancia de delimitar de manera adecuada el rango o espacio de búsqueda de los parámetros entrenables de las funciones de activación paramétricas al seleccionar el tipo de función adecuada en las neuronas de soporte. Esto facilita una mejor y más rápida convergencia (con menor cantidad de épocas) del clasificador, evitando que la RNA quede atrapada en mínimos locales y en el espacio de búsqueda de los posibles valores óptimos de los parámetros. En otras palabras, esto ayuda a optimizar eficientemente el rendimiento del modelo y obtener resultados más precisos.

Además, la tasa de aprendizaje es un factor importante al optimizar los pesos y sesgos de las neuronas, así como encontrar de manera simultánea los parámetros entrenables de las funciones. En ciertas situaciones, es posible que las neuronas convencionales y las neuronas de soporte deban aprender a diferentes ritmos. Esto puede lograrse ajustando los valores de la tasa de aprendizaje de manera individual para cada tipo de neurona o unidad. Ajustar las tasas de aprendizaje por tipo de unidad permite adaptar el proceso de aprendizaje a las características específicas de cada nuevo componente del modelo, lo que facilita la convergencia hacia soluciones óptimas en este problema de optimización más complejo.

Una de las limitaciones de esta propuesta radica en requerir un mayor costo computacional y tiempo de ejecución para obtener mejores resultados de desempeño. Sin embargo, el avance en los componentes de hardware como CPU, GPU y RAM puede contribuir a solucionar este tipo de problemas en el futuro, ya que estos en cada nueva generación ofrecen un mayor poder bruto de procesamiento. Adicionalmente, esta nueva arquitectura de RNA implica la necesidad de definir y configurar los valores óptimos de más y nuevos hiperparámetros. Esto requiere un análisis cuidadoso y más laborioso para encontrar los valores de los nuevos hiperparámetros al maximizar el rendimiento de la arquitectura propuesta.

Por otra parte, se proponen nuevas variantes de las funciones paramétricas basadas en ReLU y DPReLU que se combinan con las neuronas de soporte. La finalidad es ajustar las funciones de activación mediante parámetros entrenables para adaptar el modelado de manera más precisa al comportamiento de los datos. Esto se logra obteniendo diferentes variaciones de modelado en cada capa de la RNA-NS, específicamente ajustando las pendientes de la función de activación base (ReLU) por neurona de cada capa. Los resultados de este estudio muestran que esta

estrategia puede mejorar el rendimiento de las RNA, aunque se requiere un ligero incremento en el costo computacional y tiempo de ejecución.

Este trabajo se distingue de otros estudios existentes en varias formas. En primer lugar, se propone una nueva arquitectura de red neuronal artificial que utiliza nuevas unidades de procesamiento y nuevas variantes de funciones de activación paramétricas para mejorar el análisis y el modelado de señales acústicas. En segundo lugar, se presenta el proceso de entrenamiento mediante propagación hacia atrás y la regla delta para la RNA con neuronas de soporte. En tercer lugar, se busca mejorar la capacidad de las RNA para modelar señales acústicas con ruido ambiental y en el idioma español. Por último, se pretende presentar una arquitectura de RNA más robusta, es decir, capaz de abordar problemas de sub-ajuste y sobre-aprendizaje al modelar datos con ruido o de mala calidad. Finalmente, se realiza una comparación entre modelos de RNA existentes y la arquitectura propuesta.

Tabla 5.14 Ventajas y desventajas del modelo propuesto de identificación de locutor en comparación con modelos tradicionales.

Modelo	Ventajas	Desventajas
Modelo de identificación de locutor propuesto usando una red neuronal con neuronas de soporte y función de activación MPReLU.	1. Mayor robustez al contemplar una fase de selección y fusión de características. 2. Reducción del error de clasificación del modelo. 3. Entrenamiento automático de parámetros entrenables internos de las RNA. 4. Reducción de problemas, como sobre-aprendizaje y sub-ajuste. 5. Mayor robustez del modelo a datos ruidosos. 6. El modelo puede converger en una menor cantidad de épocas de entrenamiento.	1. Mayor costo computacional. 2. Mayor complejidad de componentes, cálculos matemáticos y de parámetros por definir y optimizar. 3. Conlleva un mayor tiempo de ejecución del modelo tanto para entrenamiento como para predicciones.
Modelo de identificación de locutor basado en una red neuronal tradicional.	1. Menor costo computacional. 2. Requiere menor tiempo de ejecución. 3. Un modelo con menos complejidad de componentes y de cálculos matemáticos.	1. Tasas de error mayores. 2. Incremento de los posibles problemas de sobre-ajuste y sub-ajuste. 3. Menor robustez al modelar datos ruidosos. 4. Posiblemente necesita mayor cantidad de épocas de entrenamiento.

Para validar esta nueva arquitectura sugerida, es necesario que en estudios futuros se evalúe su rendimiento en conjuntos de datos de señales acústicas más diversos y robustos. Esto incluye datos en diferentes idiomas y regiones, con una variedad de ruidos y tonos de voz. Adicionalmente, es importante evaluar su desempeño en otros tipos de datos o señales, como corpus de datos médicos, en tareas de clasificación de imágenes y datos genéricos. El rendimiento logrado por esta

nueva variante de RNA es un buen punto de inicio que la habilita como una posible propuesta de nueva arquitectura de RNA mejorada. Sin embargo, es necesario llevar a cabo una fase de validación exhaustiva para confirmar su efectividad y aplicabilidad en escenarios reales y en diversos contextos. La Tabla 5.14 muestra una comparativa de las ventajas y desventajas del modelo propuesto de identificación de locutor en comparación con modelos tradicionales.

Capítulo 6. Conclusiones

En este estudio se presentó un esquema de RNA que incorpora funciones de activación paramétricas y una variante de neuronas artificiales denominadas neuronas de soporte. Estas neuronas de soporte están diseñadas específicamente para optimizar los parámetros de las funciones de activación paramétricas. El objetivo de esta propuesta fue mejorar el rendimiento de las RNA en tareas de identificación de locutor con fines forenses en archivos de audio en español y considerando diferentes tipos de ruido ambiental. Por otra parte, se propusieron dos nuevos tipos de funciones de activación paramétricas basadas en la función DPReLU que son: MPReLU y AReLU. Para este problema en particular, estas variantes permiten un modelado más adecuado al comportamiento de los datos y brindan mayor robustez en la representación de datos con presencia de ruido. También, se compararon diferentes configuraciones de RNA que utilizan funciones de activación paramétricas y neuronas de soporte contra las RNA convencionales, con la intención de seleccionar el modelo óptimo.

Los modelos propuestos exhibieron problemas de generalización y sobre-aprendizaje. Para abordar estos problemas, se llevó a cabo una selección de características con el objetivo de reducir el sub-ajuste y la memorización en tareas de identificación de tramas de voz. El esquema de RNA que demostró el mejor rendimiento en actividades de clasificación de locutor por archivo de audio, logrando un equilibrio óptimo entre sesgo y varianza, fue la RNA con MPReLU y neuronas de soporte. Este modelo alcanzó una exactitud del 98.68% y un puntaje F1 del 98.28%. Esta investigación posibilitará el desarrollo y la implementación de diversas herramientas de software destinadas a brindar apoyo en el análisis de información en investigaciones criminales y forenses que implican el uso de muestras de voz grabadas. Estas herramientas podrán ser utilizadas en tareas de vigilancia y seguridad para facilitar el procesamiento y la interpretación de los datos obtenidos como evidencia criminal.

Este trabajo demostró que las RNA con neuronas de soporte pueden ser eficaces para aprender de manera conjunta y automática los parámetros de las funciones de activación paramétricas, así como los pesos y sesgos de las neuronas artificiales convencionales. Además, al incorporar nuevas variantes de funciones paramétricas basadas en ReLU y DPReLU, se logra una mayor robustez en el modelado de señales acústicas con presencia de ruido. Por otro lado, la hipótesis es aceptada ya que el esquema de red neuronal artificial propuesto (RNA-NS con función MPReLU) logra

reducir el error aproximadamente un 0.8% (de exactitud) en comparación con una arquitectura de RNA convencional, esto en tareas de identificación de locutor en entornos ruidosos. Este resultado obtenido de reducción del error supera el mínimo esperado (0.5%) en la hipótesis planteada. También, se logra reducir la tasa de error en un 0.078 en la fase de entrenamiento y un 0.3075 en la etapa de validación. En este sentido, los resultados obtenidos también respaldan la propuesta de utilizar RNA con funciones paramétricas y neuronas de soporte como una posible arquitectura de RNA mejorada.

Finalmente, considerar fases como normalización y selección/fusión de características permite conservar la información relevante y reducir problemas de información con ruido. Esto añade robustez a los clasificadores de procesamiento de voz, reduciendo los problemas de generalización. Normalmente, estas fases no se consideran en modelos de identificación de locutor o reconocimiento de voz, ya que se asume que las características extraídas son la información más representativa y no requieren pasos adicionales, como la normalización y selección de características acústicas. Sin embargo, este trabajo ha demostrado que, aunque el objetivo de un método de extracción de características es conservar la información más relevante y tratar problemas de ruido, no garantiza que toda la información obtenida sea importante y totalmente libre de ruido. Por lo tanto, en algunas situaciones es necesario realizar procesos adicionales para asegurar este objetivo y tratar de mejor manera el ruido de los datos. Los resultados logrados en este trabajo de investigación respaldan la propuesta de considerar estas fases como relevantes en modelos de procesamiento de voz.

Como trabajo futuro, sería interesante evaluar y validar exhaustivamente la nueva arquitectura de RNA en diferentes tipos de datos y tareas, como en datasets de datos médicos, datos genéricos, datos con diversos tipos de ruido y clasificación de imágenes. También, sería relevante explorar su desempeño en la actividad de identificación de locutores en diversos idiomas. Esto podría proporcionar una visión más amplia de la aplicabilidad y eficacia de la arquitectura propuesta en diferentes dominios y escenarios. Por otra parte, sería beneficioso explorar la incorporación de capas convolucionales $1-dimensión$ en la nueva arquitectura de RNA con la intención de abordar de mejor manera los problemas de sobrea-prendizaje y eliminación de ruido en señales acústicas con presencia de ruido.

Referencias

[1] R. Mohd Hanifa, K. Isa, and S. Mohamad, “A review on speaker recognition: Technology and challenges,” *Comput. Electr. Eng.*, vol. 90, p. 107005, 2021, doi: https://doi.org/10.1016/j.compeleceng.2021.107005.

[2] V. M. Patel, N. K. Ratha, and R. Chellappa, “Cancelable Biometrics: A review,” *IEEE Signal Process. Mag.*, vol. 32, no. 5, pp. 54–65, 2015, doi: 10.1109/MSP.2015.2434151.

[3] A. Ross *et al.*, “Some Research Problems in Biometrics: The Future Beckons,” *2019 Int. Conf. Biometrics*, pp. 1–8, 2019, doi: 10.1109/ICB45273.2019.8987307.

[4] Z. Zhang, “Mechanics of human voice production and control,” *J. Acoust. Soc. Am.*, vol. 140, no. 4, pp. 2614–2635, 2016, doi: 10.1121/1.4964509.

[5] N. Singh, R. A. Khan, and R. Shree, “Applications of Speaker Recognition,” *Procedia Eng.*, vol. 38, pp. 3122–3126, 2012, doi: https://doi.org/10.1016/j.proeng.2012.06.363.

[6] J. Benesty, M. Mohan, and Y. Arden, *Springer Handbook of Speech Processing*, 1st ed. Berlin Heidelberg: Springer Berlin, Heidelberg, 2008. doi: https://doi.org/10.1007/978-3-540-49127-9.

[7] L. Rabiner and R. Schafer, *Theory and Applications of Digital Speech Processing*, 1st ed. USA: Prentice Hall Press, 2010. doi: 10.5555/1841670.

[8] K. B. Bhangale and K. Mohanaprasad, “A review on speech processing using machine learning paradigm,” *Int. J. Speech Technol.*, vol. 24, no. 2, pp. 367–388, 2021, doi: 10.1007/s10772-021-09808-0.

[9] S. Latif, R. Rana, S. Khalifa, R. Jurdak, J. Qadir, and B. W. Schuller, “Deep Representation Learning in Speech Processing: Challenges, Recent Advances, and Future Trends,” *ArXiv*, p. 25, 2021, doi: https://doi.org/10.48550/arXiv.2001.00378.

[10] J. P. Campbell, “Speaker recognition: a tutorial,” *Proc. IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997, doi: 10.1109/5.628714.

[11] Z. Bai and X.-L. Zhang, “Speaker recognition based on deep learning: An overview,” *Neural Networks*, vol. 140, pp. 65–99, 2021, doi: https://doi.org/10.1016/j.neunet.2021.03.004.

[12] S. P. Todkar, S. S. Babar, R. U. Ambike, P. B. Suryakar, and J. R. Prasad, “Speaker Recognition Techniques: A Review,” in *2018 3rd International Conference for Convergence in Technology (I2CT)*, 2018, pp. 1–5. doi: 10.1109/I2CT.2018.8529519.

[13] R. V Pawar, R. M. Jalnekar, and J. S. Chitode, “Review of various stages in speaker recognition system, performance measures and recognition toolkits,” *Analog Integr. Circuits Signal Process.*, vol. 94, no. 2, pp. 247–257, 2018, doi: 10.1007/s10470-017-1069-1.

[14] L. Deng and X. Li, “Machine Learning Paradigms for Speech Recognition: An Overview,” *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 21, no. 5, pp. 1060–1089, 2013, doi: 10.1109/TASL.2013.2244083.

[15] M. Cutajar, E. Gatt, I. Grech, O. Casha, and J. Micallef, “Comparative study of automatic speech recognition techniques,” *IET Signal Process.*, vol. 7, no. 1, pp. 25–46, 2013, doi: https://doi.org/10.1049/iet-spr.2012.0151.

[16] J. Padmanabhan and M. Premkumar, “Machine Learning in Automatic Speech Recognition: A Survey,” *IETE Tech. Rev.*, vol. 32, pp. 1–12, 2015, doi: 10.1080/02564602.2015.1010611.

[17] S. S. Tirumala, S. R. Shahamiri, A. S. Garhwal, and R. Wang, “Speaker identification features extraction methods: A systematic review,” *Expert Syst. Appl.*, vol. 90, pp. 250–271, 2017, doi: https://doi.org/10.1016/j.eswa.2017.08.015.

[18] H. Gupta and D. Gupta, “LPC and LPCC method of feature extraction in Speech Recognition System,” in *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*, 2016, pp. 498–502. doi: 10.1109/CONFLUENCE.2016.7508171.

[19] A. Vibhute, “Feature Extraction Techniques in Speech Processing A Survey,” *Int. J. Comput. Appl.*, vol. 107, pp. 1–8, 2014, doi: 10.5120/18744-9997.

[20] D. Wang, X. Wang, and S. Lv, “An Overview of End-to-End Automatic Speech Recognition,” *Symmetry (Basel).*, vol. 11, no. 8, 2019, doi: 10.3390/sym11081018.

[21] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, “An Overview of Noise-Robust Automatic Speech Recognition,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 4, pp. 745–777, 2014, doi: 10.1109/TASLP.2014.2304637.

[22] M. Zakariah, M. K. Khan, and H. Malik, “Digital multimedia audio forensics: past, present and future,” *Multimed. Tools Appl.*, vol. 77, no. 1, pp. 1009–1040, 2018, doi: 10.1007/s11042-016-4277-2.

[23] D. B. Manurung, B. Dirgantoro, and C. Setianingsih, “Speaker Recognition For Digital Forensic Audio Analysis Using Learning Vector Quantization Method,” in *2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS)*, 2018, pp. 221–226. doi: 10.1109/IOTAIS.2018.8600852.

[24] S. Singh and P. Singh, “Speaker specific feature based clustering and its applications in language independent forensic speaker recognition,” *Int. J. Electr. Comput. Eng.*, vol. 10, pp. 3508–3518, 2020, doi: 10.11591/ijece.v10i4.pp3508-3518.

[25] F. Ye and J. Yang, “A Deep Neural Network Model for Speaker Identification,” *Appl. Sci.*, vol. 11, no. 8, pp. 1–18, 2021, doi: 10.3390/app11083603.

[26] Z. Liu, Z. Wu, T. Li, J. Li, and C. Shen, “GMM and CNN Hybrid Method for Short Utterance Speaker Recognition,” *IEEE Trans. Ind. Informatics*, vol. 14, no. 7, pp. 3244–3252, 2018, doi: 10.1109/TII.2018.2799928.

[27] N. Simić *et al.*, “Speaker Recognition Using Constrained Convolutional Neural Networks in Emotional Speech,” *Entropy*, vol. 24, no. 3, 2022, doi: 10.3390/e24030414.

[28] G. Pop and D. Burileanu, “Speech enhancement for forensic purposes,” *UPB Sci. Bull. Ser. C Electr. Eng. Comput. Sci.*, vol. 81, no. 3, pp. 41–52, 2019, [Online]. Available: https://www.scientificbulletin.upb.ro/rev_docs_arhiva/fullcd2_532416.pdf

[29] M. Ravanelli and Y. Bengio, “Speaker Recognition from Raw Waveform with SincNet,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 1021–1028. doi: 10.1109/SLT.2018.8639585.

[30] Y. Zhao, J. Yue, X. Xu, L. Wu, and X. Li, “End-to-End-Based Tibetan Multitask Speech Recognition,” *IEEE Access*, vol. 7, pp. 162519–162529, 2019, doi: 10.1109/ACCESS.2019.2952406.

[31] O. Siohan and D. Rybach, “Multitask learning and system combination for automatic speech recognition,” in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 589–595. doi: 10.1109/ASRU.2015.7404849.

[32] S. Ruder, “An Overview of Multi-Task Learning in Deep Neural Networks *,” *ArXiv*, p. 14, 2017,

doi: https://doi.org/10.48550/arXiv.1706.05098.

[33] A. Becerra, J. I. de la Rosa, E. González, A. D. Pedroza, and N. I. Escalante, “Training deep neural networks with non-uniform frame-level cost function for automatic speech recognition,” *Multimed. Tools Appl.*, vol. 77, no. 20, pp. 27231–27267, 2018, doi: 10.1007/s11042-018-5917-5.

[34] Z. Tang, L. Li, and D. Wang, “Multi-task recurrent model for speech and speaker recognition,” in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016, pp. 1–4. doi: https://doi.org/10.48550/arXiv.1603.09643.

[35] X. Zhang *et al.*, “An Artificial Neuron Based on a Threshold Switching Memristor,” *IEEE Electron Device Lett.*, vol. 39, no. 2, pp. 308–311, 2018, doi: 10.1109/LED.2017.2782752.

[36] I. Omelianenko, *Hands-On Neuroevolution with Python*, 1st ed. Packt, 2019. [Online]. Available: https://www.packtpub.com/product/hands-on-neuroevolution-with-python/9781838824914

[37] P. Mesejo, O. Ibáñez, E. Fernández-Blanco, F. Cedrón, A. Pazos, and A. B. Porto-Pazos, “Artificial Neuron–Glia Networks Learning Approach Based on Cooperative Coevolution,” *Int. J. Neural Syst.*, vol. 25, no. 04, p. 1550012, 2015, doi: 10.1142/S0129065715500124.

[38] F. Cedron, S. Alvarez-Gonzalez, A. Pazos, and A. B. Porto-Pazos, “Use of Multiple Astrocytic Configurations within an Artificial Neuro-Astrocytic Network,” *Proceedings*, vol. 21, no. 1, 2019, doi: 10.3390/proceedings2019021046.

[39] C. Ikuta, Y. Uwate, and Y. Nishio, “Artificial Neuron-Glia Network Including Random Glia Pulse Generation,” *IEEE Work. Nonlinear Circuit Networks*, vol. 1, pp. 27–29, 2015.

[40] R. C. Maher, “Audio forensic examination,” *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 84–94, 2009, doi: 10.1109/MSP.2008.931080.

[41] S. J. Russell and P. Norvig, *Artificial Intelligence: a modern approach*, 4th editio. Pearson, 2021.

[42] P. Ponce, *INTELIGENCIA ARTIFICIAL Con Aplicaciones a la Ingeniería*, Primera ed. Alfaomega, 2010.

[43] V. Mathivet, *Inteligencía Artificial para desarrolladores conceptos e implementación en Java*. 2017.

[44] A. Majeed and S. O. Hwang, “Data-Driven Analytics Leveraging Artificial Intelligence in the Era of COVID-19: An Insightful Review of Recent Developments,” *Symmetry (Basel).*, vol. 14, no. 1, 2022, doi: 10.3390/sym14010016.

[45] R. Kusters *et al.*, “Interdisciplinary Research in Artificial Intelligence: Challenges and Opportunities,” *Front. Big Data*, vol. 3, 2020, doi: 10.3389/fdata.2020.577974.

[46] J. García Herrero, A. BERLANGA DE JESÚS, M. Á. PATRICIO GUISADO, and W. R. PADILLA, *CIENCIA DE DATOS - Técnicas analíticas y aprendizaje estadístico*, Primera ed. México, D.F: Alfaomega, Altaria Editorial, 2018.

[47] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, NY, 2006. doi: 10.1007/978-3-030-57077-4_11.

[48] E. Paradis, B. O’Brien, L. Nimmon, G. Bandiera, and M. A. (Tina) Martimianakis, “Design: Selection of Data Collection Methods,” *J. Grad. Med. Educ.*, vol. 8, no. 2, pp. 263–264, 2016, doi: 10.4300/JGME-D-16-00098.1.

[49] R. Hernandez Sampieri, *FUNDAMENTOS DE INVESTIGACIÓN*. México: Mc Graw Hill, 2016.

[50] R. Hernández Sampieri, C. Feránadez Collado, and P. Baptizta Lucio, *Metodología de la investigación*, 6ta edició. México, D.F: MC Graw Hill, 2014.

[51] J. Huang, Y.-F. Li, and M. Xie, “An empirical analysis of data preprocessing for machine learning-based software cost estimation,” *Inf. Softw. Technol.*, vol. 67, pp. 108–127, 2015, doi: https://doi.org/10.1016/j.infsof.2015.07.004.

[52] J. Han, M. Kamber, and J. Pei, “3 - Data Preprocessing,” in *Data Mining (Third Edition)*, Third Edit., J. Han, M. Kamber, and J. Pei, Eds. Boston: Morgan Kaufmann, 2012, pp. 83–124. doi: https://doi.org/10.1016/B978-0-12-381479-1.00003-4.

[53] A. Subasi, “Chapter 2 - Data preprocessing,” in *Practical Machine Learning for Data Analysis Using Python*, A. Subasi, Ed. Academic Press, 2020, pp. 27–89. doi: https://doi.org/10.1016/B978-0-12-821379-7.00002-3.

[54] Ø. Due Trier, A. K. Jain, and T. Taxt, “Feature extraction methods for character recognition-A survey,” *Pattern Recognit.*, vol. 29, no. 4, pp. 641–662, 1996, doi: https://doi.org/10.1016/0031-3203(95)00118-2.

[55] L. A. Guyon, Isabelle and Nikravesh, Masoud and Gunn, Steve R. and Zadeh, *Feature Extraction: Foundations and Applications*, First. Springer Berlin, Heidelberg, 2006. doi: https://doi.org/10.1007/978-3-540-35488-8.

[56] F. P. Shah and V. Patel, “A review on feature selection and feature extraction for text classification,” in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, 2016, pp. 2264–2268. doi: 10.1109/WiSPNET.2016.7566545.

[57] B. Ghojogh *et al.*, “Feature Selection and Feature Extraction in Pattern Analysis: A Literature Review.” arXiv, p. 14, 2019. doi: https://doi.org/10.48550/arXiv.1905.02845.

[58] S. Khalid, T. Khalil, and S. Nasreen, “A survey of feature selection and feature extraction techniques in machine learning,” in *2014 Science and Information Conference*, 2014, pp. 372–378. doi: 10.1109/SAI.2014.6918213.

[59] M. Ringnér, “What is principal component analysis?,” *Nat. Biotechnol.*, vol. 26, no. 3, pp. 303–304, 2008, doi: 10.1038/nbt0308-303.

[60] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, “Linear discriminant analysis: A detailed tutorial,” *AI Commun.*, vol. 30, pp. 169–190, 2017, doi: 10.3233/AIC-170729.

[61] G. Wang, Q. Ding, and Z. Hou, “Independent component analysis and its applications in signal processing for analytical chemistry,” *TrAC Trends Anal. Chem.*, vol. 27, no. 4, pp. 368–376, 2008, doi: https://doi.org/10.1016/j.trac.2008.01.009.

[62] P. Comon, “Independent component analysis, A new concept?,” *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994, doi: https://doi.org/10.1016/0165-1684(94)90029-9.

[63] B. Remeseiro and V. Bolon-Canedo, “A review of feature selection methods in medical applications,” *Comput. Biol. Med.*, vol. 112, p. 103375, 2019, doi: https://doi.org/10.1016/j.compbiomed.2019.103375.

[64] G. Chandrashekar and F. Sahin, “A survey on feature selection methods,” *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 16–28, 2014, doi: https://doi.org/10.1016/j.compeleceng.2013.11.024.

[65] A. Jović, K. Brkić, and N. Bogunović, “A review of feature selection methods with applications,” in *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2015, pp. 1200–1205. doi:

10.1109/MIPRO.2015.7160458.

[66] S. Piramuthu, “Evaluating feature selection methods for learning in data mining applications,” *Eur. J. Oper. Res.*, vol. 156, no. 2, pp. 483–494, 2004, doi: https://doi.org/10.1016/S0377-2217(02)00911-6.

[67] R. A. Johnson, *Probabilidad y Estadística Para Ingenieros*, Octava Edi. Pearson, 2012.

[68] M. R. Spiegel, J. J. Schiller, and R. A. Srinivasan, *Probability and Statistics*, 4th editio. McGraw Hill, 2012.

[69] O. Sánchez, *PROBABILIDAD Y ESTADISTICA*, 4th Edició. McGraw-Hill, 2022.

[70] A. VANNIEUWENHUYZE, *Inteligencia artificial fácil Machine Learning y Deep Learning prácticos*, Primera ed. España: ENI, 2020.

[71] MathWorks, “Deep Learning: Tres cosas que es necesario saber,” 2022. https://la.mathworks.com/discovery/deep-learning.html

[72] MathWorks, “Machine Learning: Tres cosas que es necesario saber,” 2022. https://la.mathworks.com/discovery/machine-learning.html

[73] S. Raschk and V. Mirjalili, *Python Machine Learning*, Segunda ed. España: MARCOMBO, SA, 2019.

[74] V. Mathivet, *Inteligencia Artificial para desarrolladores Conceptos e implementación java*, 2da ed. Ediciones ENI, 2017.

[75] H. B. Demuth, M. H. Beale, O. De Jess, and M. T. Hagan, *Neural Network Design*, 2nd ed. Stillwater, OK, USA: Martin Hagan, 2014.

[76] B. M. Del Brío and A. Sanz Molina, *Redes neuronales y Sistemas borrosos*, Tercera ed. RA-MA y Alfaomegs, 2007.

[77] L. R. Rabiner and R. W. Schafer, “Introduction to Digital Speech Processing,” *Found. Trends® Signal Process.*, vol. 1, no. 1–2, pp. 1–194, 2007, doi: 10.1561/2000000001.

[78] P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, “Chapter 1 - Introduction to Emotion, Electroencephalography, and Speech Processing,” in *Introduction to EEG- and Speech-Based Emotion Recognition*, P. A. Abhang, B. W. Gawali, and S. C. Mehrotra, Eds. Academic Press, 2016, pp. 1–17. doi: https://doi.org/10.1016/B978-0-12-804490-2.00001-4.

[79] T. Bäckström *et al.*, *Introduction to Speech Processing*, 2nd ed. 2022. doi: 10.5281/zenodo.6821775.

[80] M. R. Schroeder, “The Speech Signal,” in *Computer Speech: Recognition, Compression, Synthesis*, Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 105–108. doi: 10.1007/978-3-662-03861-1_7.

[81] B. H. Juang, M. M. Sondhi, and L. R. Rabiner, “Digital Speech Processing,” in *Encyclopedia of Physical Science and Technology (Third Edition)*, Third Edit., R. A. Meyers, Ed. New York: Academic Press, 2003, pp. 485–500. doi: https://doi.org/10.1016/B0-12-227410-5/00178-2.

[82] R. Mohd Hanifa, K. Isa, and S. Mohamad, “A review on speaker recognition: Technology and challenges,” *Comput. Electr. Eng.*, vol. 90, no. April 2020, p. 107005, 2021, doi: 10.1016/j.compeleceng.2021.107005.

[83] B. Basharirad and M. Moradhaseli, “Speech emotion recognition methods: A literature review,”

AIP Conf. Proc., vol. 1891, no. October 2017, 2017, doi: 10.1063/1.5005438.

[84] R. V. Pawar, R. M. Jalnekar, and J. S. Chitode, “Review of various stages in speaker recognition system, performance measures and recognition toolkits,” *Analog Integr. Circuits Signal Process.*, vol. 94, no. 2, pp. 247–257, 2018, doi: 10.1007/s10470-017-1069-1.

[85] M. Wickert, *Signal & Systems For DUMMIES*, 1st ed. John Wiley and Sons, 2013.

[86] “Reconocimiento de voz a través de técnicas híbridas utilizando modelos Markovianos y nuevos tipos de redes neuronales,” 2017.

[87] J. Wolfe, M. Garnier, and J. Smith, “Vocal tract resonances in speech, singing, and playing musical instruments.,” *HFSP J.*, vol. 3, no. 1, pp. 6–23, 2009, doi: 10.2976/1.2998482.

[88] S. Z. Li and A. Jain, Eds., “Fundamental Frequency, Pitch, F0 BT - Encyclopedia of Biometrics,” Boston, MA: Springer US, 2009, p. 592. doi: 10.1007/978-0-387-73003-5_775.

[89] M. Ruiz Huillca, “MONOGRAFÍA: Los fonemas del español Examen.” Perú, p. 93, 2018. [Online]. Available: https://repositorio.une.edu.pe/bitstream/handle/20.500.14039/4361/MONOGRAFÍA - RUIZ HUILLCA MARUJA - FCSYH.pdf?sequence=5&isAllowed=y

[90] J. P. Campbell, “Speaker recognition: A tutorial,” *Proc. IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997, doi: 10.1109/5.628714.

[91] L. D. Dong Yu, *Automatic Speech Recognition: A Deep Learning Approach*, 1st ed. Springer London, 2015. doi: https://doi.org/10.1007/978-1-4471-5779-3.

[92] M. A. Anusuya and S. K. Katti, “Front end analysis of speech recognition: a review,” *Int. J. Speech Technol.*, vol. 14, no. 2, pp. 99–145, 2011, doi: 10.1007/s10772-010-9088-7.

[93] D. O’Shaughnessy, “Linear predictive coding,” *IEEE Potentials*, vol. 7, no. 1, pp. 29–32, 1988, doi: 10.1109/45.1890.

[94] M. Ravanelli and Y. Bengio, “SPEECH AND SPEAKER RECOGNITION FROM RAW WAVEFORM WITH SINCNET,” *arXiv Prepr. arXiv1812.05920*, p. 5, 2018, doi: https://doi.org/10.48550/arXiv.1812.05920.

[95] M. B. Andra and T. Usagawa, “Improved Transcription and Speaker Identification System for Concurrent Speech in Bahasa Indonesia Using Recurrent Neural Network,” *IEEE Access*, vol. 9, pp. 70758–70774, 2021, doi: 10.1109/ACCESS.2021.3077441.

[96] S. Squartini, E. Principi, R. Rotili, and F. Piazza, “Environmental robust speech and speaker recognition through multi-channel histogram equalization,” *Neurocomputing*, vol. 78, no. 1, pp. 111–120, 2012, doi: https://doi.org/10.1016/j.neucom.2011.05.035.

[97] L. Lu, N. Kanda, J. Li, and Y. Gong, “Streaming Multi-Talker Speech Recognition with Joint Speaker Identification,” in *Proc. Interspeech 2021*, 2021, pp. 1782–1786. doi: 10.21437/Interspeech.2021-207.

[98] N. Kanda *et al.*, “Joint Speaker Counting, Speech Recognition, and Speaker Identification for Overlapped Speech of any Number of Speakers,” in *Proc. Interspeech 2020*, 2020, pp. 36–40. doi: 10.21437/Interspeech.2020-1085.

[99] L. El Shafey, H. Soltau, and I. Shafran, “Joint Speech Recognition and Speaker Diarization via Sequence Transduction,” in *Proc. Interspeech 2019*, 2019, pp. 396–400. doi: 10.21437/Interspeech.2019-1943.

[100] H. H. Mao, S. Li, J. McAuley, and G. W. Cottrell, “Speech Recognition and Multi-Speaker

Diarization of Long Conversations," in *Proc. Interspeech 2020*, 2020, pp. 691–695. doi: 10.21437/Interspeech.2020-3039.

[101] M. N. Kakade and D. B. Salunke, "An Automatic Real Time Speech-Speaker Recognition System: A Real Time Approach," in *ICCCE 2019*, 2020, pp. 151–158.

[102] N. M and A. S. Ponraj, "Speech Recognition with Gender Identification and Speaker Diarization," in *2020 IEEE International Conference for Innovation in Technology (INOCON)*, 2020, pp. 1–4. doi: 10.1109/INOCON50539.2020.9298241.

[103] V. Tiwari, M. F. Hashmi, A. Keskar, and N. C. Shivaprakash, "Virtual home assistant for voice based controlling and scheduling with short speech speaker identification," *Multimed. Tools Appl.*, vol. 79, no. 7, pp. 5243–5268, 2020, doi: 10.1007/s11042-018-6358-x.

[104] A. Rajasekhar and M. K. Hota, "A Study of Speech, Speaker and Emotion Recognition Using Mel Frequency Cepstrum Coefficients and Support Vector Machines," in *2018 International Conference on Communication and Signal Processing (ICCSP)*, 2018, pp. 114–118. doi: 10.1109/ICCSP.2018.8524451.

[105] R. Sampieri Hernández, C. Fernández Collado, and M. del P. Baptista Lucio, *Metodología de la investigación*, 6ta edició. México D.F: MC Graw Hill, 2014.

[106] S. Alvarez-Gonzalez, F. Cedron, A. Pazos, and A. B. Porto-Pazos, "Artificial glial cells in artificial neuronal networks: a systematic review," *Artif. Intell. Rev.*, 2023, doi: 10.1007/s10462-023-10586-1.

[107] D. Yang, K. M. Ngoc, I. Shin, and M. Hwang, "DPReLU: Dynamic Parametric Rectified Linear Unit and Its Proper Weight Initialization Method," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, 2023, doi: 10.1007/s44196-023-00186-w.

[108] Y. Bai, "RELU-Function and Derived Function Review," *SHS Web Conf.*, vol. 144, p. 02006, 2022, doi: 10.1051/shsconf/202214402006.

[109] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2015 Inter, pp. 1026–1034, 2015, doi: 10.1109/ICCV.2015.123.

[110] T. B. Chandra, K. Verma, B. K. Singh, D. Jain, and S. S. Netam, "Coronavirus disease (COVID-19) detection in Chest X-Ray images using majority voting based classifier ensemble," *Expert Syst. Appl.*, vol. 165, p. 113909, 2021, doi: https://doi.org/10.1016/j.eswa.2020.113909.

[111] C. Iwendi *et al.*, "COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm," *Front. Public Heal.*, vol. 8, 2020, doi: 10.3389/fpubh.2020.00357.

Anexos

Esta sección presenta productos científicos e información adicional relacionada con el trabajo de investigación propuesto, como artículos científicos publicados, conferencias y coloquios.

6.1. Productos de investigación