**Reconocimiento de emociones en tiempo real para la mejora de la interacción humano-computadora**

|  |  |  |
| --- | --- | --- |
| Titulación:  TFM18  Curso académico  2023 – 2024 | Alumno/a: Guerrero Burbano Jenny Sofía  D.N.I:0401734975  Director/a de TFM: Jesús Cigales | Convocatoria:  Primera o Segunda |

**Indice:**

**Abstract**

Resumen

El presente Trabajo de Fin de Máster tiene como objetivo el desarrollo de un sistema de reconocimiento de emociones en tiempo real mediante la integración de dos modalidades: imágenes faciales y señales de voz. El reconocimiento de emociones es una rama de la inteligencia artificial que busca identificar y clasificar las emociones humanas a partir de diferentes tipos de datos. En este trabajo, se ha utilizado el dataset FER-2013 para la detección de emociones a partir de imágenes faciales, y el dataset RAVDESS para el reconocimiento de emociones en señales de voz.

El proceso metodológico incluyó la recolección y preprocesamiento de datos, integrando los diferentes datasets para garantizar la robustez del modelo final. Se realizó una fusión tardía de las modalidades de imagen y voz, empleando redes neuronales convolucionales (CNNs) para el procesamiento de imágenes y redes neuronales recurrentes (RNNs) para el análisis de señales de voz. Los modelos fueron entrenados y validados utilizando técnicas de hold-out y cross-validation para asegurar su generalización.

Los resultados obtenidos muestran que la fusión de ambas modalidades mejora significativamente la precisión en la detección de emociones en comparación con el uso de una sola modalidad. Se alcanzaron altos niveles de precisión, especialmente en la clasificación de emociones como felicidad y sorpresa, aunque algunas categorías como la neutralidad presentaron desafíos. Las conclusiones destacan la efectividad de los modelos multimodales en el reconocimiento de emociones y sugieren posibles mejoras y aplicaciones futuras en áreas como la interacción humano-computadora y la salud mental.

Palabras clave: reconocimiento de emociones, inteligencia artificial, redes neuronales, fusión multimodal, procesamiento de imágenes, procesamiento de voz.

Abstract

This Master’s Thesis presents the development of a real-time emotion recognition system by integrating two modalities: facial images and voice signals. Emotion recognition is a branch of artificial intelligence that aims to identify and classify human emotions from various types of data. In this study, the FER-2013 and KDEF datasets were used for emotion detection from facial images, while the RAVDESS dataset was employed for emotion recognition from voice signals.

The methodological approach included data collection and preprocessing, integrating different datasets to ensure the robustness of the final model. A late fusion technique was applied to combine the image and voice modalities, utilizing Convolutional Neural Networks (CNNs) for image processing and Recurrent Neural Networks (RNNs) for voice analysis. The models were trained and validated using hold-out and cross-validation techniques to ensure generalization.

The results demonstrate that the fusion of both modalities significantly improves the accuracy of emotion detection compared to using a single modality. High accuracy levels were achieved, particularly in the classification of emotions such as happiness and surprise, although some challenges were observed with categories like neutrality. The conclusions highlight the effectiveness of multimodal models in emotion recognition and suggest possible improvements and future applications in areas such as human-computer interaction and mental health.

**Keywords**: emotion recognition, artificial intelligence, neural networks, multimodal fusion, image processing, voice processing.

Indice

[1. Introducción 5](#_Toc174233251)

[1.1. Qué es Inteligencia Artificial (IA)? 5](#_Toc174233252)

[1.2. Historia 6](#_Toc174233253)

[1.2.1. Orígenes de la IA 6](#_Toc174233254)

[1.2.2. Primeros Programas de IA 6](#_Toc174233255)

[1.2.3. Inviernos de la IA 8](#_Toc174233256)

[1.2.4. Renacimiento de la IA 8](#_Toc174233257)

[1.2.5. Reconocimiento de Emociones en la Historia de la IA: 8](#_Toc174233258)

[1.3. Diversas Interpretaciones de la Inteligencia Artificial 10](#_Toc174233259)

[1.4. Importancia y Aplicaciones de la Inteligencia Artificial 11](#_Toc174233260)

[1.5. Aprendizaje Automático (Machine Learning, ML) 12](#_Toc174233261)

[1.5.1. Definición y Conceptos Básicos 12](#_Toc174233262)

[1.6. Aprendizaje Profundo 14](#_Toc174233263)

[1.6.1. Definición y Conceptos Básicos 14](#_Toc174233264)

[1.6.2. Arquitectura de Redes Neuronales 14](#_Toc174233265)

[1.6.3. Técnicas y Avances en Aprendizaje Profundo 15](#_Toc174233266)

[1.7. Comparativa de modelos de IA 15](#_Toc174233267)

[1.7.1. Modelos Simbólicos vs Modelos Conexionistas 15](#_Toc174233268)

[1.7.2. Algoritmos de Aprendizaje Automático Tradicionales vs Aprendizaje Profundo 16](#_Toc174233269)

[1.7.3. Comparativa de Modelos para Reconocimiento de Emociones 16](#_Toc174233270)

[1.8. Reconocimiento de Emociones: Importancia y Aplicaciones 17](#_Toc174233271)

[1.8.1. Importancia del Reconocimiento de Emociones 17](#_Toc174233272)

[1.8.2. Aplicaciones del Reconocimiento de Emociones 17](#_Toc174233273)

[1.8.3. Desafíos y Futuras Direcciones 18](#_Toc174233274)

[1.9. Ética y Responsabilidad en la Inteligencia Artificial 18](#_Toc174233275)

[1.9.1. Dilemas Éticos en la IA 18](#_Toc174233276)

[2. Estado del Arte 20](#_Toc174233277)

[2.1. Uso de Datasets en Reconocimiento de Emociones 21](#_Toc174233278)

[2.1.1. Importancia de los Datasets 21](#_Toc174233279)

[2.1.2. Datasets de Imágenes Faciales 21](#_Toc174233280)

[2.1.3. Datasets de Voz 22](#_Toc174233281)

[2.2. Reconocimiento de Emociones a partir de Imágenes Faciales 24](#_Toc174233282)

[2.2.1. Técnicas de Reconocimiento Basadas en Redes Neuronales Convolucionales (CNNs) 24](#_Toc174233283)

[2.2.2. Conjuntos de Datos Utilizados 25](#_Toc174233284)

[2.2.3. Avances y Desafíos Actuales 25](#_Toc174233285)

[2.3. Reconocimiento de Emociones a partir de Señales de Voz 26](#_Toc174233286)

[2.3.1. Modelos de Procesamiento del Lenguaje Natural (NLP) 26](#_Toc174233287)

[2.3.2. Redes Neuronales Recurrentes (RNNs) 27](#_Toc174233288)

[2.3.3. Desafíos y Aplicaciones 27](#_Toc174233289)

[2.4. Fusión de Modalidades Múltiples 28](#_Toc174233290)

[2.4.1. 2.4.1. Modelos Multimodales 28](#_Toc174233291)

[2.4.2. 2.4.2. Técnicas de Fusión de Características 29](#_Toc174233292)

[2.4.3. Desafíos y Futuras Direcciones 29](#_Toc174233293)

[2.5. Implementaciones en Tiempo Real 30](#_Toc174233294)

[2.5.1. Optimización de Modelos para Tiempo Real 30](#_Toc174233295)

[2.5.2. 2.5.2. Plataformas y Herramientas para la Implementación 31](#_Toc174233296)

[2.5.3. Desafíos y Futuras Direcciones 31](#_Toc174233297)

[2.6. Desafíos y Futuras Direcciones 32](#_Toc174233298)

[3. Metodología 32](#_Toc174233299)

[3.1. Preparación de datos 32](#_Toc174233300)

[3.1.1. . Recolección de Datos 32](#_Toc174233301)

[3.1.2. Limpieza de Datos 33](#_Toc174233302)

[3.1.3. Integración de Datos 34](#_Toc174233303)

[3.1.4. Splitting de Datos 36](#_Toc174233304)

[3.2. Preprocesamiento de Datos 37](#_Toc174233305)

[4. Resultados 39](#_Toc174233306)

[5. Discusíon 39](#_Toc174233307)

[6. Conclusiones 39](#_Toc174233308)

[**Estado del Arte en Reconocimiento de Emociones en Tiempo Real** 41](#_Toc174233309)

[**Conclusión** 43](#_Toc174233310)

[**Conclusión** 46](#_Toc174233311)

[6.1.1. REconocimiento de imágenes: Conjunto de Datos FER-2013 47](#_Toc174233312)

[6.1.2. 1. ReduceLROnPlateau 50](#_Toc174233313)

[6.1.3. 2. EarlyStopping 50](#_Toc174233314)

[**Bibliografía** 51](#_Toc174233315)

1. Introducción
   1. Qué es Inteligencia Artificial (IA)?

La rama de la inteligencia artificial fue fundada como una disciplina académica en 1956 durante la conferencia de Dartmouth, organizada por John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon. Este evento es reconocido y aceptado por la comunidad científica y académica como el momento en que se estableció formalmente la inteligencia artificial (IA) como un campo de estudio independiente. IA comprender una variedad de sub campos, incluidos machine learning, que trata del estudio de algoritmos que mejoran automaticamente a través de la experiencia; Redes neuronales, que se inspiran en el funcionamiento del cerebro; y la robótica, que trata del diseño y creación de agentes autónomos que pueden interactuar con el mudno físicio.

A lo largo de la historia de la inteligencia artificial (IA), han surgido varios enfoques teóricos que han definido el desarrollo del campo. Dos de los enfoques más influyentes han sido la IA simbólica y la IA conexionista.

La IA simbólica, también conocida como IA basada en reglas, se centra en el uso de representaciones simbólicas explícitas y reglas lógicas para manipular estos símbolos. Este enfoque, que se originó en la década de 1950 y 1960, se basa en la premisa de que la inteligencia puede ser representada mediante símbolos y reglas formales. Uno de los primeros sistemas de IA simbólica fue el programa de procesamiento de lenguaje natural ELIZA, desarrollado por Joseph Weizenbaum en 1966 (Russell & Norvig, 2020).

Por otro lado, la IA conexionista, también conocida como aprendizaje profundo o redes neuronales, se inspira en la estructura y el funcionamiento del cerebro humano. Este enfoque utiliza redes de neuronas artificiales para aprender patrones a partir de datos. La IA conexionista ganó prominencia en la década de 1980 y ha experimentado un resurgimiento en los últimos años debido a los avances en el poder computacional y la disponibilidad de grandes volúmenes de datos. Los trabajos de Geoffrey Hinton, Yann LeCun y Yoshua Bengio han sido fundamentales en el desarrollo de este enfoque (Goodfellow et al., 2016).

* 1. Historia

### Orígenes de la IA

La inteligencia artificial (IA) como campo de estudio formal se originó en la década de 1950. Uno de los eventos más importantes que marcaron el inicio de la IA fue la Conferencia de Dartmouth en 1956, organizada por John McCarthy, Marvin Minsky, Nathaniel Rochester y Claude Shannon. Este evento reunió a varios investigadores interesados en la posibilidad de crear máquinas inteligentes y se considera el punto de partida oficial de la IA. Durante esta conferencia, los investigadores discutieron los problemas fundamentales y los enfoques necesarios para desarrollar máquinas inteligentes, estableciendo así los objetivos y retos que definirían el campo en los años venideros (Russell & Norvig, 2020; Nilsson, 2009).

Primeros Programas de IA

Los primeros programas de inteligencia artificial se centraron en la automatización del razonamiento humano, buscando replicar procesos cognitivos mediante algoritmos informáticos. Entre los programas más destacados de este periodo se encuentran **Logic Theorist** y **General Problem Solver (GPS)**, ambos desarrollados en la década de 1950 por pioneros de la IA.

#### Logic Theorist

**Logic Theorist**, desarrollado por Allen Newell y Herbert A. Simon en 1955, es considerado el primer programa de IA. Este programa fue diseñado para imitar el proceso de resolución de problemas humanos en el ámbito de la lógica matemática. Específicamente, Logic Theorist fue capaz de probar teoremas en el Principia Mathematica de Alfred North Whitehead y Bertrand Russell, un texto fundamental en la lógica simbólica.

El enfoque utilizado por Logic Theorist se basaba en una búsqueda heurística, una estrategia que seleccionaba las posibles soluciones basándose en reglas heurísticas (reglas generales) en lugar de revisar exhaustivamente todas las posibilidades. Esta metodología fue revolucionaria, ya que demostró que una máquina podía realizar tareas que requerían un razonamiento lógico, acercándose al tipo de inteligencia que se pensaba era exclusivamente humana.

El éxito de Logic Theorist no solo validó el concepto de programas capaces de "pensar" lógicamente, sino que también estableció una base teórica para futuros desarrollos en inteligencia artificial. Fue uno de los primeros programas que demostró que el razonamiento humano podía ser modelado y emulado mediante procedimientos matemáticos y lógicos, lo que abrió la puerta a la creación de sistemas más complejos (Newell & Simon, 1956).

#### General Problem Solver (GPS)

Desarrollado también por Allen Newell y Herbert A. Simon junto a J.C. Shaw en 1957, el **General Problem Solver (GPS)** fue una extensión de los conceptos implementados en Logic Theorist. Mientras que Logic Theorist estaba limitado a problemas específicos de la lógica matemática, GPS fue diseñado con un objetivo más ambicioso: resolver una amplia gama de problemas simbólicos utilizando un enfoque general de resolución de problemas.

El GPS operaba bajo la premisa de que cualquier problema podía ser resuelto descomponiéndolo en subproblemas más pequeños y manejables, los cuales se resolvían utilizando métodos heurísticos similares a los de Logic Theorist. Este enfoque se conoció como la estrategia de means-end analysis (análisis medios-fines), en la que el programa identificaba las diferencias entre el estado actual y el estado deseado, y luego aplicaba operadores para reducir estas diferencias.

A pesar de su naturaleza pionera, GPS enfrentó limitaciones significativas debido a la complejidad creciente de los problemas a medida que se intentaban aplicar a dominios más generales. Aun así, GPS influyó profundamente en el desarrollo posterior de sistemas de inteligencia artificial y en la teoría de la resolución de problemas. Su concepción de una "máquina universal de resolución de problemas" inspiró futuras investigaciones y es un ejemplo temprano de la búsqueda de la IA para emular la cognición humana de manera generalizada (Newell, Shaw, & Simon, 1959).

### Inviernos de la IA

A pesar de los primeros éxitos, la IA simbólica encontró varias limitaciones. En las décadas de 1970 y 1980, hubo períodos conocidos como los "inviernos de la IA", caracterizados por la disminución del financiamiento y el interés debido a expectativas no cumplidas y la falta de avances significativos. Durante estos períodos, muchos proyectos de IA fueron abandonados debido a la percepción de que la IA no estaba cumpliendo con sus promesas iniciales (Russell & Norvig, 2020). Estos inviernos reflejaron los desafíos técnicos y conceptuales que los investigadores enfrentaban, y llevaron a una reevaluación de las estrategias y métodos utilizados en el campo de la IA (Nilsson, 2009).

### Renacimiento de la IA

El renacimiento de la IA comenzó a principios de la década de 2000, impulsado por avances en el aprendizaje profundo (deep learning) y el aumento del poder computacional. Investigadores como Geoffrey Hinton, Yann LeCun y Yoshua Bengio fueron fundamentales en el desarrollo y popularización del aprendizaje profundo. Las redes neuronales profundas (deep neural networks) demostraron ser extremadamente efectivas en tareas como el reconocimiento de imágenes, el procesamiento del lenguaje natural y los juegos, superando con creces los enfoques anteriores (Goodfellow, Bengio, & Courville, 2016). Este resurgimiento no solo revitalizó el interés en la IA, sino que también abrió nuevas posibilidades para aplicaciones prácticas y teóricas, marcando una nueva era de innovación y descubrimiento en el campo (Bishop, 2006).

Reconocimiento de Emociones en la Historia de la IA:

El reconocimiento de emociones ha sido un campo de investigación en la inteligencia artificial (IA) que ha evolucionado significativamente desde los primeros desarrollos en IA simbólica hasta la era moderna del aprendizaje profundo. Aunque en los primeros años de la IA, la mayoría de los esfuerzos se centraron en la resolución de problemas lógicos y matemáticos, la comprensión y reconocimiento de las emociones humanas se convirtió en un área de interés a medida que los investigadores se dieron cuenta de la importancia de la interacción humano-computadora más allá de las simples tareas lógicas.

#### Inicios y Primeros Enfoques

En las primeras décadas de la IA, el reconocimiento de emociones no era un área central de investigación. Los esfuerzos se centraban en tareas como la resolución de problemas y el razonamiento lógico, con programas como Logic Theorist y General Problem Solver (Newell & Simon, 1956; Newell, Shaw, & Simon, 1959). Sin embargo, a medida que la IA comenzó a expandirse hacia aplicaciones más diversas, surgió el interés en cómo las máquinas podían interpretar y responder a señales humanas más complejas, como las emociones.

#### Progreso en los Años 90 y 2000

El avance en el reconocimiento de emociones comenzó a cobrar impulso en las décadas de 1990 y 2000 con el desarrollo de técnicas de procesamiento de señales y el uso de algoritmos más avanzados en el análisis de expresiones faciales y señales de voz. Durante este tiempo, surgieron los primeros sistemas capaces de reconocer emociones básicas a partir de señales visuales y auditivas, utilizando modelos estadísticos como los modelos de Markov ocultos (HMMs) y redes neuronales artificiales (ANNs) (Cowie et al., 2001).

#### Era Moderna: Aprendizaje Profundo y Multimodalidad

El verdadero cambio de paradigma en el reconocimiento de emociones llegó con la era del aprendizaje profundo en la década de 2010. El desarrollo de redes neuronales profundas (DNNs) y convolucionales (CNNs) permitió un salto significativo en la precisión y capacidad de los sistemas para reconocer emociones a partir de datos complejos como imágenes y señales de voz (Goodfellow, Bengio, & Courville, 2016). Conjuntos de datos como FER-2013 y AffectNet, junto con modelos avanzados como VGG y ResNet, se convirtieron en herramientas esenciales para entrenar estos sistemas (Mollahosseini et al., 2017).

En los últimos años, la fusión de modalidades múltiples (como la combinación de análisis facial y vocal) ha mejorado aún más la precisión del reconocimiento de emociones, permitiendo una comprensión más holística y matizada de las señales emocionales humanas (Zhang et al., 2017; Poria et al., 2017). Esto ha llevado a aplicaciones prácticas en áreas como la salud mental, la educación, y las interfaces humano-computadora, haciendo que el reconocimiento de emociones sea un componente crucial en el desarrollo de IA más empática e intuitiva.

* 1. Diversas Interpretaciones de la Inteligencia Artificial

A lo largo de los años, los investigadores han examinado diversas interpretaciones de la inteligencia artificial. Algunos han definido la inteligencia en términos de similitud con el desempeño humano, evaluando la capacidad de las máquinas para realizar tareas que, si fueran realizadas por humanos, requerirían inteligencia, como el reconocimiento de patrones, el aprendizaje y la toma de decisiones. Este enfoque, conocido como IA débil o estrecha, se centra en la creación de sistemas que pueden realizar tareas específicas con un alto grado de competencia (Russell & Norvig, 2021).

Por otro lado, otros investigadores han adoptado un enfoque más abstracto y formal, centrado en la racionalidad. Este enfoque, denominado IA fuerte o general, busca desarrollar máquinas que no solo imiten el comportamiento humano, sino que también posean una comprensión profunda y una capacidad para razonar de manera autónoma y flexible en una variedad de situaciones. La IA fuerte aspira a replicar la totalidad de las capacidades cognitivas humanas, incluyendo la conciencia y la autoconciencia (Searle, 1980).

Ambas interpretaciones han llevado a importantes avances en el campo. La IA débil ha dado lugar a aplicaciones prácticas y cotidianas, como asistentes virtuales, sistemas de recomendación y vehículos autónomos (Goodfellow, Bengio, & Courville, 2016). Mientras tanto, la investigación en IA fuerte ha impulsado el desarrollo de algoritmos y modelos que intentan abordar problemas más complejos y abstractos, como la planificación a largo plazo y la resolución creativa de problemas (Bostrom, 2014).

En conclusión, el estudio de la inteligencia artificial abarca un espectro amplio de enfoques y objetivos. La convergencia de estos enfoques no solo enriquece nuestra comprensión de la inteligencia, sino que también abre nuevas posibilidades para el futuro de la interacción entre humanos y máquinas. Al avanzar en este campo, es crucial considerar tanto las aplicaciones prácticas inmediatas como las implicaciones filosóficas y éticas de la creación de sistemas verdaderamente inteligentes (Russell & Norvig, 2021; Bostrom, 2014).

* 1. Importancia y Aplicaciones de la Inteligencia Artificial

La inteligencia artificial (IA) ha transformado múltiples aspectos de la vida cotidiana y se ha convertido en una tecnología esencial en diversas industrias. Su importancia radica en varios factores clave:

**Automatización de Tareas**: La IA permite la automatización de tareas repetitivas y tediosas, mejorando la eficiencia y reduciendo costos operativos en sectores como la manufactura, la atención al cliente y la administración (Russell & Norvig, 2021).

**Mejora en la Toma de Decisiones**: Los sistemas de IA pueden analizar grandes volúmenes de datos y proporcionar información valiosa para la toma de decisiones estratégicas en áreas como la salud, las finanzas y el marketing (Goodfellow, Bengio, & Courville, 2016).

**Innovación y Desarrollo Tecnológico**: La IA impulsa la innovación al permitir el desarrollo de nuevas tecnologías y productos, desde vehículos autónomos hasta asistentes virtuales inteligentes (Bostrom, 2014).

**Solución de Problemas Complejos**: La IA es capaz de abordar problemas complejos que son difíciles de resolver con métodos tradicionales, como el diagnóstico de enfermedades raras o la predicción de fenómenos climáticos (Russell & Norvig, 2021).

**Impacto Social y Económico**: La implementación de IA tiene un impacto significativo en la economía global, creando nuevas oportunidades de empleo y mejorando la calidad de vida a través de servicios más personalizados y eficientes (Bostrom, 2014).

Las aplicaciones de la inteligencia artificial son vastas y diversas, y abarcan una amplia gama de sectores:

1. **Salud**: En el sector salud, la IA se utiliza para el diagnóstico de enfermedades, la personalización de tratamientos y la gestión de datos clínicos. Algoritmos de aprendizaje profundo analizan imágenes médicas para detectar anomalías con una precisión comparable a la de los especialistas humanos (Goodfellow, Bengio, & Courville, 2016).
2. **Finanzas**: En el ámbito financiero, la IA se emplea para la detección de fraudes, la gestión de riesgos y la automatización del trading. Los sistemas basados en IA pueden analizar tendencias del mercado y hacer predicciones precisas que ayudan a los inversores a tomar decisiones informadas (Russell & Norvig, 2021).
3. **Transporte**: La IA está revolucionando el transporte con el desarrollo de vehículos autónomos que prometen mejorar la seguridad vial y reducir la congestión del tráfico. Empresas como Tesla y Waymo están a la vanguardia de esta tecnología (Bostrom, 2014).
4. **Atención al Cliente**: Los chatbots y asistentes virtuales, como Siri y Alexa, utilizan IA para interactuar con los usuarios, responder preguntas y realizar tareas básicas, mejorando la experiencia del cliente y liberando recursos humanos para tareas más complejas (Russell & Norvig, 2021).
5. **Educación**: La IA también está transformando la educación mediante plataformas de aprendizaje adaptativo que personalizan el contenido educativo según las necesidades individuales de los estudiantes, promoviendo un aprendizaje más eficiente y efectivo (Goodfellow, Bengio, & Courville, 2016).

En conclusión, la inteligencia artificial tiene un impacto profundo y expansivo en múltiples sectores, mejorando procesos, facilitando la toma de decisiones y fomentando la innovación. Su continua evolución promete transformar aún más la sociedad y la economía en el futuro.

* 1. Aprendizaje Automático (Machine Learning, ML)

El aprendizaje automático (Machine Learning, ML) es una subdisciplina de la inteligencia artificial que se centra en el desarrollo de algoritmos y técnicas que permiten a las máquinas aprender de datos y realizar predicciones o tomar decisiones basadas en esos datos. A diferencia de los sistemas basados en reglas explícitas, el aprendizaje automático se basa en la creación de modelos que se ajustan y mejoran automáticamente a partir de la experiencia.

### Definición y Conceptos Básicos

El aprendizaje automático se puede definir como el campo de estudio que proporciona a las computadoras la capacidad de aprender sin ser explícitamente programadas (Mitchell, 1997). En lugar de seguir instrucciones específicas, los modelos de ML identifican patrones y relaciones en los datos, ajustando sus parámetros internos para mejorar su rendimiento con el tiempo.

#### 1.5.2. Tipos de Aprendizaje Automático

Existen varios tipos de aprendizaje automático, cada uno adecuado para diferentes tipos de problemas y datos.

##### **Aprendizaje Supervisado**

En este tipo de aprendizaje, el modelo se entrena con un conjunto de datos etiquetados, es decir, cada entrada de datos está asociada con una salida deseada. Ejemplos de algoritmos supervisados incluyen la regresión lineal, las máquinas de soporte vectorial (SVM) y las redes neuronales. Este enfoque es adecuado para tareas como la clasificación y la regresión (Russell & Norvig, 2020).

A continuación, podemos ver un gráfico que compara dos enfoques comunes en aprendizaje automático: **Clasificación** y **Regresión**.

1. **Clasificación (Izquierda)**:
   * **Descripción**: La imagen de la izquierda muestra un ejemplo de un problema de clasificación. Los puntos en el gráfico representan ejemplos de datos que pertenecen a una de dos clases diferentes, indicadas por símbolos de diferentes formas o colores. La línea roja discontinua representa la frontera de decisión creada por el modelo de clasificación para separar las dos clases.
   * **Interpretación**: En clasificación, el objetivo es predecir categorías discretas. La línea roja es la frontera que el modelo aprende para distinguir entre las dos clases, como por ejemplo, "aprobado" vs "reprobado" en un examen o "positivo" vs "negativo" en un test médico.
2. **Regresión (Derecha)**:
   * **Descripción**: La imagen de la derecha representa un problema de regresión. Los puntos en el gráfico indican datos continuos, y la línea roja discontinua muestra la línea de mejor ajuste que el modelo de regresión ha aprendido.
   * **Interpretación**: En regresión, el objetivo es predecir valores continuos. La línea de mejor ajuste predice el valor numérico en función de las características de entrada. Un ejemplo típico sería predecir el precio de una casa basada en su tamaño y ubicación.

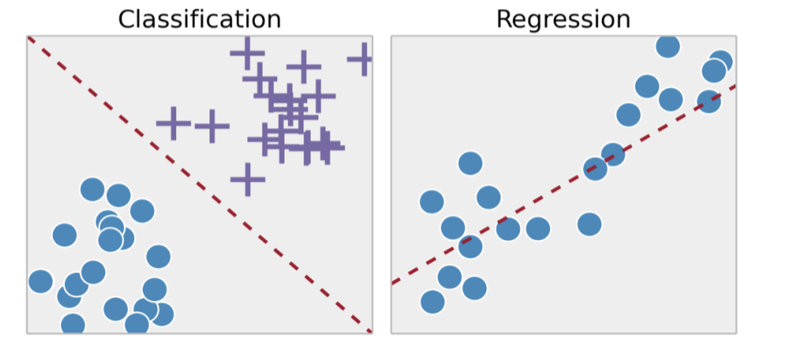


Figura 1: Ejemplo de un modelo de aprendizaje supervisado y no supervisado. (Huawei, 2023)

##### **Aprendizaje No Supervisado**:

En el aprendizaje no supervisado, el modelo se entrena con datos no etiquetados, y su objetivo es encontrar estructuras o patrones ocultos en los datos. Ejemplos de algoritmos no supervisados incluyen el clustering (agrupamiento) y la reducción de dimensionalidad. Este enfoque es útil para tareas como el análisis de clústeres y la detección de anomalías (Bishop, 2006).

##### **Aprendizaje por Refuerzo**:

En este tipo de aprendizaje, un agente interactúa con un entorno y aprende a tomar decisiones mediante prueba y error, recibiendo recompensas o castigos en función de sus acciones. Este enfoque es adecuado para tareas que involucran toma de decisiones secuenciales, como el control de robots y los juegos (Sutton & Barto, 2018).

### Comparativa entre Aprendizaje Supervisado y No Supervisado

A continuación se presenta una imagen que compara dos enfoques fundamentales en el aprendizaje automático: el aprendizaje supervisado y el aprendizaje no supervisado. En el aprendizaje supervisado, el objetivo es construir un modelo que pueda predecir una etiqueta de salida basada en ejemplos de entrenamiento etiquetados. La clasificación, como se muestra en la parte izquierda de la imagen, es una tarea típica de aprendizaje supervisado donde el modelo aprende a asignar categorías a los datos de entrada en función de ejemplos previos etiquetados.

Por otro lado, el aprendizaje no supervisado no se basa en datos etiquetados. En lugar de predecir etiquetas específicas, busca encontrar patrones o estructuras ocultas en los datos. Un ejemplo común es el clustering, representado en la parte derecha de la imagen, donde los datos son agrupados en subconjuntos o "clusters" basados en similitudes, sin necesidad de etiquetas predefinidas.

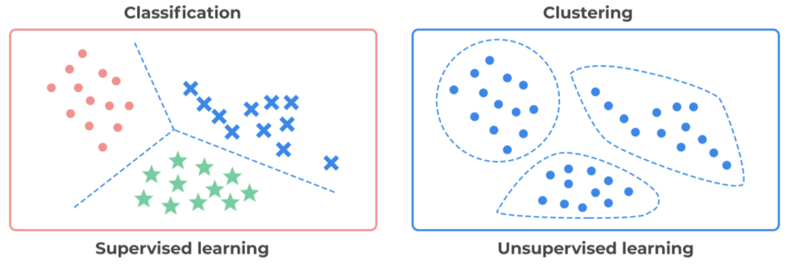


Figura 1: Ejemplo de un modelo de aprendizaje supervisado y no supervisado. (Huawei, 2023)

#### 1.5.3. Algoritmos Clásicos de Aprendizaje Automático

El aprendizaje automático abarca una amplia variedad de algoritmos, algunos de los cuales se han convertido en métodos clásicos debido a su eficacia y aplicabilidad:

* **Regresión Lineal y Logística**: Utilizados para tareas de regresión y clasificación, respectivamente, estos algoritmos son fundamentales para entender las relaciones entre variables (Bishop, 2006).
* **Máquinas de Soporte Vectorial (SVM)**: Un potente algoritmo supervisado utilizado tanto para clasificación como para regresión. Las SVM buscan encontrar el hiperplano que mejor separa las clases en los datos (Cortes & Vapnik, 1995).
* **Árboles de Decisión y Bosques Aleatorios**: Los árboles de decisión son modelos de predicción que dividen los datos en ramas basadas en características y umbrales específicos. Los bosques aleatorios son un conjunto de árboles de decisión que mejoran la precisión y la robustez (Breiman, 2001).
* **Redes Neuronales**: Modelos inspirados en el cerebro humano que consisten en capas de neuronas artificiales. Son especialmente efectivos en el reconocimiento de patrones complejos y se utilizan ampliamente en tareas como el reconocimiento de imágenes y el procesamiento del lenguaje natural (Goodfellow, Bengio, & Courville, 2016).
  1. Aprendizaje Profundo

El aprendizaje profundo (Deep Learning) es una subdisciplina del aprendizaje automático que se basa en redes neuronales artificiales con múltiples capas (deep neural networks). Este enfoque ha revolucionado el campo de la inteligencia artificial, permitiendo avances significativos en diversas áreas como el reconocimiento de imágenes, el procesamiento del lenguaje natural y el aprendizaje por refuerzo.

### Definición y Conceptos Básicos

El aprendizaje profundo se caracteriza por su capacidad para aprender representaciones jerárquicas de los datos a través de múltiples niveles de abstracción. A diferencia de los algoritmos de aprendizaje automático tradicionales, que se basan en características manualmente diseñadas, las redes neuronales profundas pueden aprender automáticamente características relevantes directamente a partir de los datos brutos (LeCun, Bengio, & Hinton, 2015).

### Arquitectura de Redes Neuronales

Las redes neuronales profundas están compuestas por múltiples capas de neuronas artificiales, cada una de las cuales transforma la entrada de la capa anterior. Las arquitecturas más comunes incluyen:

* **Redes Neuronales Convolucionales (CNNs)**: Especialmente efectivas para el procesamiento de datos con estructura de cuadrícula, como las imágenes. Utilizan capas de convolución que aplican filtros sobre la entrada para capturar características locales (Goodfellow, Bengio, & Courville, 2016).
* **Redes Neuronales Recurrentes (RNNs)**: Adecuadas para el procesamiento de datos secuenciales, como el texto y las series temporales. Utilizan conexiones recurrentes para mantener un estado interno que captura información temporal (Hochreiter & Schmidhuber, 1997).
* **Redes Generativas Adversarias (GANs)**: Consisten en dos redes que compiten entre sí: una generadora, que crea datos falsos, y una discriminadora, que intenta distinguir entre datos reales y falsos. Este enfoque ha sido utilizado para generar imágenes realistas, entre otras aplicaciones (Goodfellow et al., 2014).

### Técnicas y Avances en Aprendizaje Profundo

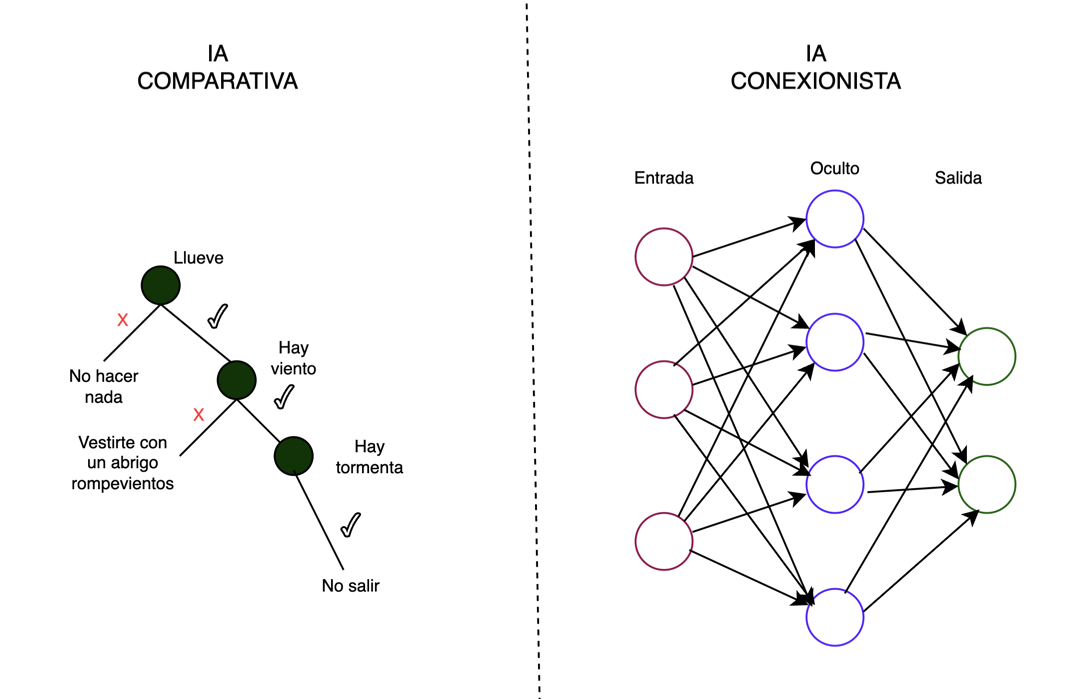
El aprendizaje profundo ha progresado rápidamente gracias a diversas técnicas y avances, incluyendo:

* **Transferencia de Aprendizaje**: Permite utilizar un modelo preentrenado en una gran cantidad de datos y ajustar sus pesos para una tarea específica con un conjunto de datos más pequeño. Esta técnica ha sido especialmente útil en el reconocimiento de imágenes (Yosinski et al., 2014).
* **Regularización**: Métodos como Dropout, Batch Normalization y técnicas de aumento de datos ayudan a prevenir el sobreajuste y mejorar la generalización de los modelos de aprendizaje profundo (Srivastava et al., 2014; Ioffe & Szegedy, 2015).
* **Optimizadores Avanzados**: Algoritmos como Adam y RMSprop han mejorado significativamente la eficiencia del entrenamiento de redes neuronales profundas (Kingma & Ba, 2015).
  1. Comparativa de modelos de IA

Modelos Simbólicos vs Modelos Conexionistas

La inteligencia artificial (IA) ha evolucionado a través de distintos paradigmas a lo largo de su historia. Los modelos simbólicos, que dominaron las primeras décadas del campo, se basan en la manipulación de símbolos y reglas lógicas explícitas para emular el razonamiento humano. Estos modelos incluyen sistemas expertos y técnicas de representación del conocimiento que, si bien son transparentes y fáciles de interpretar, presentan limitaciones en su capacidad para generalizar a problemas no estructurados. Por otro lado, los modelos conexionistas, representados principalmente por las redes neuronales, intentan replicar el procesamiento paralelo distribuido que ocurre en el cerebro humano. Aunque menos interpretables, estos modelos han demostrado ser mucho más efectivos en tareas que requieren el reconocimiento de patrones y el procesamiento de grandes volúmenes de datos, tales como la clasificación de imágenes y la interpretación de señales de voz (Russell & Norvig, 2020).

El gráfico a continuación compara los enfoques de los modelos simbólicos y conexionistas en inteligencia artificial. Los modelos simbólicos, también conocidos como modelos basados en reglas, emulan el razonamiento humano mediante el uso de representaciones simbólicas y reglas lógicas explícitas. Por otro lado, los modelos conexionistas, inspirados en la estructura del cerebro humano, utilizan redes de neuronas artificiales que aprenden patrones directamente a partir de los datos sin necesidad de reglas predefinidas. Esta comparación destaca las principales diferencias en términos de transparencia, capacidad de generalización, y enfoque de aprendizaje.



Algoritmos de Aprendizaje Automático Tradicionales vs Aprendizaje Profundo

Los algoritmos de aprendizaje automático tradicionales, como los árboles de decisión, las máquinas de vectores de soporte (SVM), y los métodos basados en vecinos cercanos, han sido fundamentales para resolver problemas de clasificación y regresión en diversos dominios. Estos métodos suelen ser eficaces en situaciones donde el volumen de datos es limitado y las características son bien comprendidas y seleccionadas manualmente. Sin embargo, con el advenimiento de grandes volúmenes de datos y el poder computacional moderno, el aprendizaje profundo ha emergido como la técnica dominante, especialmente en áreas como el reconocimiento de imágenes y el procesamiento del lenguaje natural. Las redes neuronales profundas (DNNs), y más específicamente las Redes Neuronales Convolucionales (CNNs) y las Redes Neuronales Recurrentes (RNNs), permiten la extracción automática de características y el aprendizaje jerárquico de representaciones, lo que ha llevado a avances significativos en la precisión y rendimiento de los sistemas de IA (Goodfellow, Bengio, & Courville, 2016).

Comparativa de Modelos para Reconocimiento de Emociones

En el contexto específico del reconocimiento de emociones, los modelos más utilizados incluyen las Redes Neuronales Convolucionales (CNNs) para el análisis de imágenes faciales y las Redes Neuronales Recurrentes (RNNs) para el procesamiento de señales de voz. Las CNNs, con su capacidad para capturar patrones espaciales en las imágenes, han mostrado un rendimiento superior en la identificación de expresiones faciales sutiles. Por otro lado, las RNNs, especialmente las variantes como LSTM y GRU, son ideales para manejar secuencias temporales, como las que se encuentran en los datos de voz, permitiendo la detección de inflexiones y entonaciones que corresponden a diferentes estados emocionales. Recientemente, la combinación de estos enfoques en modelos multimodales ha permitido la fusión de datos de diferentes fuentes (imagen y voz), mejorando significativamente la precisión y robustez en la detección de emociones. Estos modelos multimodales integran características extraídas de las imágenes y las señales de voz, permitiendo una interpretación más rica y completa del estado emocional del sujeto (Poria et al., 2017).

* 1. Reconocimiento de Emociones: Importancia y Aplicaciones

El **reconocimiento de emociones** es una tecnología crucial en la intersección de la inteligencia artificial (IA) y la interacción humano-computadora (HCI). Su importancia radica en la capacidad de las máquinas para interpretar, entender y responder de manera adecuada a las emociones humanas, lo cual es esencial en una amplia gama de aplicaciones, desde la atención al cliente hasta la medicina y la educación.

Importancia del Reconocimiento de Emociones

En el contexto actual, donde la interacción entre humanos y máquinas es cada vez más frecuente, la capacidad de las máquinas para reconocer y responder a las emociones humanas se ha convertido en un aspecto vital. Este tipo de tecnología no solo mejora la experiencia del usuario al hacer que las interacciones sean más naturales y empáticas, sino que también permite a las máquinas adaptarse mejor a las necesidades y estados emocionales de los usuarios. Esto es especialmente relevante en campos como la atención médica, donde la detección precisa de las emociones puede ser crucial para la intervención temprana en casos de trastornos mentales o emocionales.

Aplicaciones del Reconocimiento de Emociones

El reconocimiento de emociones tiene aplicaciones extensas y diversas:

* **Atención al Cliente**: Empresas de todo el mundo están utilizando tecnologías de reconocimiento de emociones para mejorar la calidad de la atención al cliente. Al identificar el estado emocional de un cliente, los sistemas pueden ofrecer respuestas más personalizadas y adecuadas, mejorando la satisfacción del cliente y la eficacia del servicio.
* **Salud Mental**: En el ámbito de la salud, el reconocimiento de emociones puede ayudar en el diagnóstico y tratamiento de trastornos emocionales y mentales. Las tecnologías que pueden monitorear las emociones de los pacientes permiten a los profesionales de la salud obtener una visión más profunda y precisa del estado emocional de sus pacientes.
* **Educación**: En entornos educativos, las aplicaciones de reconocimiento de emociones pueden adaptarse al estado emocional de los estudiantes, proporcionando contenido educativo de manera que maximice la comprensión y el aprendizaje. Por ejemplo, un sistema que detecta frustración puede ofrecer explicaciones adicionales o ajustar el ritmo de la enseñanza.
* **Seguridad**: En seguridad, el reconocimiento de emociones puede ser utilizado para detectar comportamientos anómalos en situaciones críticas, como aeropuertos o grandes eventos, lo que ayuda a prevenir incidentes.
* **Entretenimiento y Publicidad**: En la industria del entretenimiento, esta tecnología puede ser utilizada para personalizar la experiencia del usuario, mientras que en publicidad, permite a las empresas medir y reaccionar al impacto emocional de sus campañas en tiempo real.

Desafíos y Futuras Direcciones

A pesar de su potencial, el reconocimiento de emociones enfrenta varios desafíos. Entre ellos se encuentran los problemas de privacidad, ya que la recolección y análisis de datos emocionales puede ser invasiva. Además, los sesgos en los modelos de IA pueden llevar a interpretaciones incorrectas, especialmente en poblaciones diversas. Por ello, es fundamental continuar investigando en la mejora de la precisión y la equidad de estos sistemas, así como en el desarrollo de normas éticas para su implementación.

* 1. Ética y Responsabilidad en la Inteligencia Artificial

El rápido avance de la inteligencia artificial plantea importantes cuestiones éticas y de responsabilidad que deben ser abordadas para garantizar su desarrollo y aplicación segura y justa. A medida que los sistemas de IA se integran cada vez más en la vida cotidiana y en decisiones críticas, surge la necesidad de establecer marcos éticos claros y regulaciones efectivas.

### Dilemas Éticos en la IA

Uno de los principales dilemas éticos en la IA es el problema del sesgo algorítmico. Los algoritmos de IA pueden perpetuar y amplificar los sesgos existentes en los datos con los que fueron entrenados, lo que puede llevar a decisiones injustas y discriminatorias (O'Neil, 2016). Por ejemplo, en el ámbito de la contratación, los sistemas de IA pueden discriminar a candidatos basándose en características como género, raza o edad.

#### Sesgo Algorítmico

El sesgo algorítmico se produce cuando un algoritmo refleja las suposiciones sesgadas y discriminatorias que prevalecen en los datos utilizados para entrenarlo. Este problema es especialmente preocupante en aplicaciones críticas como la contratación, la atención médica y la justicia penal, donde las decisiones basadas en algoritmos pueden afectar profundamente la vida de las personas. Los algoritmos de IA pueden amplificar estas desigualdades al tomar decisiones basadas en patrones históricos que reflejan sesgos sociales y económicos (Barocas & Selbst, 2016).

##### Cajas Negras y Transparencia

Un desafío relacionado es el de la "caja negra" en los modelos de IA, particularmente en el aprendizaje profundo. Los modelos de IA de caja negra son aquellos cuyos procesos internos son opacos o incomprensibles para los humanos. Esto plantea problemas de transparencia y explicabilidad, ya que es difícil entender cómo y por qué el modelo toma ciertas decisiones (Lipton, 2018). La falta de transparencia en los modelos de IA puede dificultar la identificación y corrección de sesgos, lo que perpetúa la desconfianza en los sistemas automatizados.

Modelos No Determinísticos

Además, muchos modelos de IA no son determinísticos, lo que significa que pueden producir resultados diferentes para la misma entrada en diferentes ocasiones. Esta característica puede ser problemática en contextos donde se requiere consistencia y previsibilidad. Los modelos no determinísticos pueden complicar la auditoría y la supervisión, ya que sus decisiones no pueden ser replicadas fácilmente (Adadi & Berrada, 2018).

##### Necesidad de Explicabilidad

La explicabilidad en la IA se refiere a la capacidad de los sistemas de IA para proporcionar explicaciones claras y comprensibles de sus decisiones y procesos internos. La falta de explicabilidad en los modelos de caja negra y no determinísticos dificulta la identificación de errores y sesgos, y limita la capacidad de los usuarios para confiar en estos sistemas. Las técnicas de interpretabilidad y explicabilidad, como las aproximaciones locales a las explicaciones de modelos (LIME) y las técnicas de visualización, son cruciales para abordar estos desafíos (Ribeiro, Singh, & Guestrin, 2016).

##### Casos de Estudio

Un caso notable de sesgo algorítmico es el sistema COMPAS, utilizado en el sistema de justicia penal en Estados Unidos para evaluar el riesgo de reincidencia de los delincuentes. Investigaciones han demostrado que COMPAS presenta sesgos raciales, sobreestimando la probabilidad de reincidencia para individuos afroamericanos y subestimándola para individuos caucásicos (Angwin et al., 2016).

##### Mitigación del Sesgo

Para mitigar estos problemas, es esencial desarrollar e implementar estrategias de detección y corrección de sesgos en los datos y los modelos. Además, se deben promover prácticas de transparencia y explicabilidad en el desarrollo de sistemas de IA. La adopción de enfoques éticos y responsables en el diseño y despliegue de la IA es crucial para asegurar que estos sistemas beneficien a la sociedad en su conjunto y no perpetúen desigualdades existentes

1. Estado del Arte

El reconocimiento de emociones en tiempo real es un área de investigación dentro de la inteligencia artificial que busca identificar y clasificar las emociones humanas a partir de datos como imágenes faciales, señales de voz y fisiológicas. Este campo ha avanzado significativamente en los últimos años gracias a los desarrollos en aprendizaje profundo y la disponibilidad de grandes conjuntos de datos.

* 1. Uso de Datasets en Reconocimiento de Emociones

Importancia de los Datasets

En el campo del reconocimiento de emociones, los datasets juegan un papel fundamental en el desarrollo y evaluación de modelos de aprendizaje automático. La calidad y diversidad de los datos son factores clave que determinan el rendimiento y la generalización de los modelos a diferentes contextos y poblaciones.

Los datasets bien construidos permiten a los investigadores entrenar modelos que capturen una amplia gama de variaciones en las expresiones emocionales, tanto faciales como vocales. Esto es crucial, ya que las emociones humanas pueden manifestarse de manera diferente según factores como la cultura, el género y la situación contextual (Ekman & Friesen, 1971). Además, un dataset diverso y balanceado asegura que el modelo no aprenda sesgos indeseados, lo cual es esencial para aplicaciones sensibles como la interacción humano-computadora.

Por otro lado, la existencia de datasets ampliamente aceptados, como FER-2013 y KDEF para imágenes faciales, ha permitido la replicación de resultados y la comparación objetiva entre diferentes enfoques metodológicos. Estos datasets ofrecen una base común sobre la cual se pueden medir los avances en la precisión y robustez de los modelos de reconocimiento de emociones (Goodfellow, Bengio, & Courville, 2016).

La importancia de los datasets también radica en su capacidad para proporcionar datos etiquetados de alta calidad, que son necesarios para el entrenamiento supervisado de los modelos. Dado que la creación de estos datasets requiere un esfuerzo significativo en términos de anotación manual y control de calidad, su disponibilidad impulsa la investigación al permitir a los investigadores centrarse en el desarrollo de modelos en lugar de en la recopilación de datos (Lundqvist, Flykt, & Öhman, 1998).

Datasets de Imágenes Faciales

En el campo del reconocimiento de emociones, los datasets de imágenes faciales son fundamentales para el entrenamiento y evaluación de modelos de aprendizaje profundo. Estos datasets proporcionan imágenes etiquetadas de expresiones faciales que representan una variedad de emociones, lo que permite a los modelos aprender a reconocer patrones emocionales en rostros humanos.

Uno de los datasets más utilizados es el **FER-2013 (Facial Expression Recognition 2013)**. Este dataset fue introducido durante la Competencia de Reconocimiento de Expresiones Faciales en el Desafío de Clasificación de Imágenes de 2013. FER-2013 contiene 35,887 imágenes en escala de grises de 48x48 píxeles, divididas en siete categorías emocionales: felicidad, tristeza, sorpresa, miedo, disgusto, enfado y neutralidad (Goodfellow et al., 2013). Este dataset ha sido ampliamente adoptado debido a su tamaño y diversidad, lo que lo convierte en un estándar para la evaluación de modelos de reconocimiento de emociones faciales.

Otro dataset destacado es el **Karolinska Directed Emotional Faces (KDEF)**, que fue desarrollado en el Departamento de Neurociencia Clínica del Instituto Karolinska en Suecia. KDEF contiene 4,900 imágenes de 70 individuos (35 mujeres y 35 hombres), cada uno mostrando siete expresiones emocionales diferentes: felicidad, miedo, tristeza, enfado, sorpresa, asco y neutralidad. Las imágenes están tomadas desde cinco ángulos diferentes (de frente, 45 grados a la izquierda y derecha, y 90 grados a la izquierda y derecha) (Lundqvist, Flykt, & Öhman, 1998). Este dataset es particularmente valioso debido a la variabilidad en los ángulos de las imágenes, lo que permite a los modelos aprender a reconocer emociones en condiciones menos controladas.

La combinación de múltiples datasets, puede mejorar la generalización de los modelos al exponerlos a una mayor variedad de condiciones y sujetos. Este enfoque permite que los modelos sean más robustos y efectivos en escenarios del mundo real, donde las expresiones faciales pueden variar significativamente en función de factores como el ángulo de la cámara, la iluminación y las características individuales del rostro.

En resumen, los datasets de imágenes faciales son recursos cruciales en el desarrollo de sistemas de reconocimiento de emociones. Su disponibilidad ha permitido avances significativos en la precisión y robustez de estos sistemas, facilitando su aplicación en una variedad de contextos, desde la interacción humano-computadora hasta la vigilancia y la atención médica.

### Datasets de Voz

En el campo del reconocimiento de emociones, los datasets de voz son tan importantes como los de imágenes faciales, ya que permiten a los modelos aprender a identificar emociones a partir de las características acústicas del habla. Estos datasets contienen grabaciones de voz etiquetadas con diferentes emociones, capturando variaciones en el tono, el ritmo, la intensidad y otras características prosódicas que son indicativas de los estados emocionales del hablante.

Uno de los datasets más utilizados en este ámbito es el **Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)**. Este dataset contiene 1,440 clips de audio grabados por 24 actores (12 hombres y 12 mujeres), cada uno representando ocho emociones diferentes: felicidad, tristeza, enfado, miedo, sorpresa, asco, desprecio y neutralidad. Las grabaciones incluyen tanto discurso como canto, lo que permite a los modelos capturar la expresión emocional en diferentes contextos vocales (Livingstone & Russo, 2018). RAVDESS ha sido ampliamente adoptado en la investigación debido a la calidad controlada de las grabaciones y la diversidad de emociones representadas.

Otro dataset relevante es **EmoDB (Berlin Database of Emotional Speech)**, que fue desarrollado en el Instituto de Psicolingüística de la Universidad Libre de Berlín. EmoDB contiene grabaciones de diez actores alemanes, cada uno representando siete emociones diferentes: felicidad, tristeza, miedo, enfado, sorpresa, asco y aburrimiento. Este dataset es conocido por la claridad y autenticidad de sus grabaciones, lo que lo convierte en una referencia clave para la investigación en reconocimiento de emociones a partir de la voz (Burkhardt et al., 2005).

**CREMA-D (Crowd-Sourced Emotional Multimodal Actors Dataset)** es otro dataset significativo que incluye grabaciones de 91 actores, cada uno expresando seis emociones diferentes (felicidad, tristeza, enfado, miedo, asco y neutralidad) a través de 7,442 clips de audio. Las grabaciones de CREMA-D son particularmente valiosas debido a la diversidad de los actores y las variaciones en las entonaciones y estilos de habla, proporcionando un recurso robusto para el entrenamiento de modelos de reconocimiento de emociones (Cao et al., 2014).

Estos datasets son esenciales para el desarrollo de modelos que puedan interpretar correctamente las emociones a partir de la voz, y se utilizan en una variedad de aplicaciones, desde sistemas de atención al cliente hasta asistentes virtuales y herramientas de monitoreo emocional.

* 1. Reconocimiento de Emociones a partir de Imágenes Faciales

El reconocimiento de emociones a través del análisis de expresiones faciales se ha convertido en una herramienta esencial dentro de la inteligencia artificial, especialmente en áreas como la interacción humano-computadora y la psicología computacional. El análisis de estas expresiones se basa en la idea de que las emociones humanas pueden ser inferidas a través de características faciales que se manifiestan durante la expresión de diferentes sentimientos.

Técnicas de Reconocimiento Basadas en Redes Neuronales Convolucionales (CNNs)

Las Redes Neuronales Convolucionales (CNNs) son una clase de redes neuronales especialmente adecuadas para el procesamiento de datos con una estructura de cuadrícula, como las imágenes. Estas redes han demostrado una gran eficacia en el reconocimiento de patrones faciales, permitiendo una clasificación precisa de emociones.

Entre los modelos más destacados utilizados en este campo se encuentran:

* **VGG16**: Un modelo de CNN profunda que se ha utilizado ampliamente para la clasificación de imágenes, incluido el reconocimiento de emociones faciales. Su arquitectura, basada en capas convolucionales con pequeños filtros, permite una gran capacidad de generalización en tareas de clasificación (Simonyan & Zisserman, 2014).
* **ResNet**: Introducido por He et al. (2016), este modelo implementa una estructura de redes residuales que facilita la capacitación de redes profundas mediante la inclusión de conexiones de salto (skip connections). Esto ha permitido un entrenamiento más efectivo y ha reducido los problemas de desvanecimiento del gradiente, comúnmente observados en redes muy profundas.
* **EfficientNet**: Un modelo más reciente que ha optimizado la eficiencia computacional mediante un escalado balanceado de la profundidad, el ancho y la resolución de la red. Esto ha permitido mejoras significativas en la precisión del reconocimiento de emociones, con un menor costo computacional (Tan & Le, 2019).

Conjuntos de Datos Utilizados

Los conjuntos de datos son fundamentales en el entrenamiento de modelos de CNNs para el reconocimiento de emociones faciales. Dos de los datasets más utilizados en este campo son:

* **FER-2013**: Este conjunto de datos se compone de 35,887 imágenes en escala de grises de rostros humanos, etiquetadas en siete categorías emocionales: felicidad, tristeza, sorpresa, miedo, disgusto, enojo y neutral. Este dataset ha sido utilizado en numerosas investigaciones como base para el entrenamiento y la validación de modelos de reconocimiento de emociones (Goodfellow, Bengio, & Courville, 2016).
* **AffectNet**: Este es uno de los conjuntos de datos más extensos y variados disponibles para el reconocimiento de emociones faciales. Contiene más de un millón de imágenes etiquetadas con diferentes emociones y expresiones faciales, lo que permite a los modelos entrenados en este dataset lograr una alta precisión en la clasificación de emociones en condiciones reales y variadas (Mollahosseini et al., 2017).

Avances y Desafíos Actuales

A pesar de los avances significativos en la precisión de los modelos de reconocimiento de emociones, todavía existen desafíos importantes:

* **Variabilidad en las Expresiones Faciales**: La variabilidad en las expresiones faciales entre diferentes personas y culturas puede afectar la precisión de los modelos de reconocimiento de emociones. La falta de representatividad en los conjuntos de datos puede llevar a sesgos en los modelos.
* **Robustez frente a Ocultamientos y Condiciones de Iluminación**: Las variaciones en las condiciones de iluminación y los ocultamientos parciales del rostro, como el uso de gafas o mascarillas, siguen siendo un reto para los modelos actuales.
* **Interpretabilidad de los Modelos**: Comprender cómo los modelos de CNN toman decisiones en la clasificación de emociones es crucial para su aceptación en aplicaciones críticas, como la atención médica y la seguridad.

Puedes utilizar estas ideas para expandir tu sección y darle más profundidad, incorporando referencias adecuadas según las normas APA.

* 1. Reconocimiento de Emociones a partir de Señales de Voz

El reconocimiento de emociones a partir de señales de voz es un campo emergente en la inteligencia artificial que se basa en la premisa de que las emociones humanas se reflejan no solo en la expresión facial, sino también en la modulación de la voz. El análisis de la prosodia, que incluye características como el tono, el ritmo, la intensidad y la velocidad del habla, permite inferir el estado emocional del hablante. Este enfoque es especialmente útil en aplicaciones donde la voz es el principal o único canal de comunicación, como en los asistentes virtuales, centros de llamadas y en herramientas de apoyo para personas con discapacidades.

Modelos de Procesamiento del Lenguaje Natural (NLP)

Los avances en el Procesamiento del Lenguaje Natural (NLP) han permitido la adaptación de modelos originalmente diseñados para el análisis de texto al ámbito de las señales de voz. Modelos basados en la arquitectura Transformer, como BERT (Bidirectional Encoder Representations from Transformers) y GPT (Generative Pre-trained Transformer), han sido particularmente destacados. Aunque estos modelos fueron concebidos para tareas de comprensión y generación de texto, sus principios subyacentes se han aplicado con éxito en el reconocimiento de emociones a partir de la voz.

* **BERT**: BERT, desarrollado por Devlin et al. (2018), es un modelo de lenguaje bidireccional que ha demostrado una capacidad excepcional para capturar el contexto tanto a nivel local como global. En el contexto del análisis de voz, BERT se ha adaptado para trabajar con secuencias de características extraídas de señales de audio, permitiendo identificar cambios sutiles en la prosodia que corresponden a diferentes estados emocionales.
* **GPT**: GPT, un modelo de lenguaje generativo, también ha sido adaptado para la tarea de clasificación de emociones en señales de voz. Su capacidad para modelar secuencias largas y generar texto coherente ha sido aprovechada para identificar patrones en las características de audio que son indicativos de emociones específicas.

Redes Neuronales Recurrentes (RNNs)

Las Redes Neuronales Recurrentes (RNNs) han sido fundamentales en el análisis de señales de voz, dada su capacidad para manejar datos secuenciales y temporales. Entre las variantes más utilizadas en este contexto se encuentran las LSTM (Long Short-Term Memory) y GRU (Gated Recurrent Unit), que han demostrado ser altamente eficaces en la tarea de modelado temporal de señales de audio.

* **LSTM**: Las LSTM son una clase de RNN que mitigan el problema del desvanecimiento del gradiente, permitiendo que la red capture dependencias a largo plazo en secuencias de datos. En el reconocimiento de emociones, las LSTM son utilizadas para analizar cómo las características de la voz cambian a lo largo de una grabación, proporcionando una comprensión más profunda del estado emocional del hablante (Hochreiter & Schmidhuber, 1997).
* **GRU**: Las GRU son una variante simplificada de las LSTM que mantienen muchas de las ventajas de las LSTM, como la capacidad para manejar dependencias a largo plazo, pero con una menor complejidad computacional. Esto las hace particularmente útiles en aplicaciones donde el rendimiento en tiempo real es crítico, como en sistemas de diálogo en línea (Cho et al., 2014).

Desafíos y Aplicaciones

Aunque el reconocimiento de emociones a partir de señales de voz ha avanzado considerablemente, existen desafíos significativos que deben abordarse. Uno de los principales es la variabilidad en la calidad y el entorno de la grabación, que puede afectar la precisión del modelo. Además, las diferencias culturales en la expresión de emociones a través de la voz pueden complicar la generalización de los modelos a diferentes poblaciones.

Entre las aplicaciones prácticas del reconocimiento de emociones a partir de señales de voz se encuentran:

* **Asistentes Virtuales**: Los asistentes virtuales como Alexa y Google Assistant están comenzando a integrar el reconocimiento de emociones en sus sistemas para proporcionar respuestas más empáticas y contextualmente apropiadas.
* **Centros de Llamadas**: En los centros de llamadas, la detección de emociones puede ser utilizada para dirigir las llamadas a operadores más adecuados o para ajustar el tono y enfoque de las respuestas automáticas.
* **Atención Médica**: El análisis de voz se está explorando como una herramienta para monitorear el bienestar emocional de los pacientes, especialmente en el contexto de la salud mental, donde puede proporcionar indicadores tempranos de estados depresivos o ansiosos.
  1. Fusión de Modalidades Múltiples

El reconocimiento de emociones a través de la fusión de múltiples modalidades de datos es un enfoque innovador que ha ganado relevancia en los últimos años. Este enfoque se basa en la premisa de que la integración de diferentes tipos de datos, como imágenes faciales y señales de voz, puede proporcionar una representación más completa y precisa del estado emocional de una persona. La fusión multimodal permite capturar tanto las señales visuales como las auditivas que son inherentes a la expresión de emociones, mejorando así la robustez y exactitud de los sistemas de reconocimiento.

Modelos Multimodales

Los modelos multimodales están diseñados para integrar y procesar datos de diferentes fuentes de manera coherente. Estos modelos suelen combinar arquitecturas especializadas en cada tipo de dato para maximizar la extracción de características relevantes. Por ejemplo, las Redes Neuronales Convolucionales (CNNs) son comúnmente utilizadas para procesar imágenes faciales, mientras que las Redes Neuronales Recurrentes (RNNs) se emplean para el análisis de señales de voz.

* **Combinación de CNNs y RNNs**: Un enfoque popular en el reconocimiento multimodal de emociones es la combinación de CNNs para la extracción de características de imágenes faciales con RNNs para el análisis de secuencias de voz. Este tipo de arquitectura permite que el modelo capture tanto las características espaciales de las expresiones faciales como las características temporales de la prosodia en la voz. Zhang et al. (2017) demostraron que este enfoque puede superar significativamente a los modelos unimodales, especialmente en escenarios donde una sola fuente de datos no es suficiente para una correcta identificación emocional.
* **Transformers Multimodales**: Los modelos basados en Transformers también están ganando popularidad en la fusión multimodal. Estas arquitecturas pueden procesar diferentes tipos de datos en paralelo y aprender las interacciones entre ellos a través de mecanismos de atención. Los Transformers multimodales han mostrado ser efectivos en tareas complejas, como la comprensión del contexto emocional en interacciones humanas, al combinar texto, imágenes y audio (Tsai et al., 2019).

Técnicas de Fusión de Características

La fusión de características es un aspecto crucial en los modelos multimodales y se puede realizar en diferentes etapas del proceso de aprendizaje. Las dos técnicas principales son la fusión temprana y la fusión tardía.

* **Fusión Temprana**: En la fusión temprana, las características extraídas de cada modalidad se combinan antes de la etapa de clasificación. Esto permite al modelo aprender representaciones conjuntas de los datos desde las primeras capas de la red. La fusión temprana es útil cuando se busca capturar interacciones complejas entre las diferentes modalidades desde el principio del proceso de modelado (Poria et al., 2017).
* **Fusión Tardía**: En la fusión tardía, cada modalidad se procesa por separado y las predicciones de cada modelo se combinan en una etapa posterior. Esta técnica es ventajosa cuando las modalidades tienen características muy diferentes que podrían ser mejor capturadas por modelos especializados. La fusión tardía también ofrece más flexibilidad para ajustar el peso de cada modalidad en la decisión final (Ngiam et al., 2011).

Desafíos y Futuras Direcciones

A pesar de las ventajas de la fusión multimodal, existen desafíos significativos que deben ser abordados. La sincronización y alineación de diferentes modalidades es un problema clave, especialmente cuando los datos provienen de fuentes con diferentes frecuencias de muestreo o tiempos de captura. Además, la heterogeneidad en la calidad y naturaleza de los datos puede afectar la eficacia de la fusión.

Las futuras direcciones en este campo incluyen el desarrollo de modelos más sofisticados que puedan manejar eficientemente grandes volúmenes de datos multimodales y la incorporación de técnicas de aprendizaje auto-supervisado para mejorar la generalización de los modelos a nuevas condiciones y contextos.

* 1. Implementaciones en Tiempo Real

El desarrollo de sistemas de reconocimiento de emociones en tiempo real es un campo que ha ganado considerable interés, especialmente con el avance de la inteligencia artificial y el aumento de la capacidad computacional en dispositivos móviles y embebidos. Sin embargo, la implementación efectiva de estos sistemas presenta múltiples desafíos, particularmente en cuanto a la eficiencia computacional, la latencia, y la precisión en entornos con recursos limitados.

Optimización de Modelos para Tiempo Real

Para alcanzar un rendimiento óptimo en tiempo real, es esencial reducir la complejidad computacional de los modelos de IA sin comprometer significativamente su precisión. Entre las técnicas más utilizadas se encuentran:

* **Cuantización**: La cuantización es una técnica que reduce la precisión de los números en un modelo, lo que disminuye tanto el tamaño del modelo como el consumo de recursos durante la inferencia. Por ejemplo, convertir los pesos de un modelo de 32 bits flotantes a 8 bits enteros puede resultar en una aceleración considerable sin una pérdida significativa de precisión. Esta técnica es particularmente útil en dispositivos móviles y embebidos donde los recursos son limitados (Jacob et al., 2018).
* **Poda de Modelos**: La poda (pruning) implica eliminar conexiones, neuronas o filtros redundantes en una red neuronal para reducir su tamaño y acelerar el proceso de inferencia. Han et al. (2015) demostraron que la poda estructural de redes neuronales puede mantener el rendimiento de un modelo mientras se reduce significativamente el número de parámetros, lo que es crucial para aplicaciones en tiempo real.
* **Distilación de Modelos**: La distilación de modelos es un proceso en el que un modelo grande y complejo (conocido como el "modelo maestro") entrena a un modelo más pequeño y eficiente (el "modelo estudiante") para que imite su comportamiento. Este enfoque permite desplegar modelos más ligeros en entornos de tiempo real sin sacrificar precisión (Hinton, Vinyals, & Dean, 2015).

Plataformas y Herramientas para la Implementación

El éxito de la implementación en tiempo real también depende de las plataformas y herramientas utilizadas. Las siguientes son algunas de las más destacadas en el desarrollo de aplicaciones de reconocimiento de emociones en tiempo real:

* **TensorFlow Lite**: TensorFlow Lite es una versión ligera de TensorFlow diseñada específicamente para dispositivos móviles y embebidos. Permite ejecutar modelos de aprendizaje profundo en tiempo real con una latencia baja y un uso eficiente de los recursos. TensorFlow Lite soporta tanto la cuantización como la ejecución acelerada por hardware, lo que la convierte en una opción ideal para implementar sistemas de reconocimiento de emociones en dispositivos como smartphones y cámaras inteligentes (TensorFlow Lite, 2020).
* **ONNX Runtime**: ONNX (Open Neural Network Exchange) Runtime es un motor de inferencia de alto rendimiento que facilita la ejecución de modelos entrenados en diferentes frameworks de deep learning, como PyTorch y TensorFlow. ONNX Runtime está optimizado para ofrecer una latencia baja y es compatible con hardware especializado, como GPUs y TPUs, lo que lo convierte en una herramienta poderosa para la implementación de aplicaciones en tiempo real (ONNX, 2019).
* **NVIDIA Jetson**: La plataforma NVIDIA Jetson es una solución de hardware que proporciona una capacidad de procesamiento potente en un formato compacto, ideal para la inferencia de IA en tiempo real. Es ampliamente utilizada en aplicaciones que requieren reconocimiento de emociones en dispositivos embebidos, como robots y sistemas de vigilancia inteligente (NVIDIA, 2020).

Desafíos y Futuras Direcciones

A pesar de los avances en las técnicas y plataformas para la implementación en tiempo real, todavía existen desafíos que deben ser abordados. Estos incluyen la variabilidad de las condiciones ambientales, la adaptabilidad de los modelos a diferentes contextos de uso, y la eficiencia energética en dispositivos con autonomía limitada.

Las futuras direcciones en este campo apuntan hacia el desarrollo de modelos aún más ligeros y eficientes, capaces de adaptarse dinámicamente a los cambios en el entorno o en las necesidades del usuario. Además, la integración de técnicas de federated learning y edge computing promete mejorar la privacidad y la seguridad en las aplicaciones de reconocimiento de emociones en tiempo real, al procesar los datos directamente en el dispositivo del usuario en lugar de enviarlos a la nube.

* 1. Desafíos y Futuras Direcciones

A pesar de los avances, el reconocimiento de emociones en tiempo real enfrenta varios desafíos que necesitan ser abordados.

* **Sesgo y Variabilidad de los Datos**: Los modelos pueden ser sesgados debido a la falta de diversidad en los datos de entrenamiento, lo que afecta su rendimiento en diferentes poblaciones y contextos (O'Neil, 2016).
* **Interpretabilidad y Explicabilidad**: Comprender cómo y por qué un modelo llega a una determinada conclusión es crucial para la confianza y la aceptación de estos sistemas, especialmente en aplicaciones sensibles como la salud mental (Lipton, 2016).
* **Integración con otras Tecnologías**: La combinación de reconocimiento de emociones con otras tecnologías emergentes, como la realidad aumentada y la robótica, abre nuevas posibilidades y aplicaciones, pero también presenta nuevos desafíos técnicos y éticos (Bostrom, 2014).

1. Metodología
   1. Introducción a la Metodología

El objetivo principal de esta sección es detallar el enfoque metodológico empleado en el desarrollo de este proyecto, que tiene como finalidad diseñar, implementar y evaluar un sistema de reconocimiento de emociones en tiempo real a partir de señales de audio y video. La metodología está diseñada para asegurar que cada paso del proceso, desde la recolección de datos hasta la implementación y evaluación del modelo, se realice de manera sistemática y rigurosa, garantizando la validez y la reproducibilidad de los resultados obtenidos.

### Resumen del Proceso

El proceso metodológico seguido en este trabajo se estructura en varias etapas clave. En primer lugar, se lleva a cabo la recolección de datos, donde se seleccionan y preparan los datasets más adecuados para el reconocimiento de emociones, asegurando su calidad y representatividad. A continuación, se procede a la extracción de características tanto de audio como de video, utilizando técnicas avanzadas de procesamiento de señales para obtener las representaciones más relevantes que alimentarán al modelo de aprendizaje automático.

Posteriormente, se aborda el diseño y entrenamiento del modelo, donde se seleccionan y optimizan los algoritmos de aprendizaje más apropiados, ajustando sus hiperparámetros para maximizar el rendimiento. Esta etapa también incluye la implementación de técnicas de validación cruzada y evaluación del modelo para asegurar su capacidad de generalización.

Finalmente, se realiza una evaluación exhaustiva del sistema implementado, donde se analizan las métricas de rendimiento obtenidas y se comparan con modelos existentes en la literatura. Esta evaluación permite identificar las fortalezas y debilidades del sistema, así como proponer mejoras y futuras direcciones de investigación.

### Recolección de Datos

En este proyecto, los datos provienen de dos fuentes principales que son fundamentales para el desarrollo y la evaluación del sistema de reconocimiento de emociones.

1. **FER-2013 (Facial Expression Recognition 2013)**: Este dataset es ampliamente utilizado en el ámbito de la visión por computadora para el reconocimiento de emociones a partir de imágenes faciales. FER-2013 contiene 35,887 imágenes en escala de grises de 48x48 píxeles, categorizadas en siete emociones: felicidad, tristeza, sorpresa, miedo, disgusto, enojo y neutralidad. Este conjunto de datos fue seleccionado debido a su tamaño considerable y la diversidad de las expresiones faciales representadas, lo que permite entrenar modelos de aprendizaje automático robustos.
2. **RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song)**: Para el reconocimiento de emociones a partir de señales de audio, se utilizó el dataset RAVDESS. Este conjunto de datos incluye 1,440 clips de audio grabados por 24 actores (12 hombres y 12 mujeres), representando ocho emociones diferentes: neutralidad, calma, felicidad, tristeza, enojo, miedo, disgusto y sorpresa. Las grabaciones abarcan tanto discurso como canto, lo que añade una rica variabilidad en la expresión emocional y proporciona un valioso recurso para entrenar modelos de reconocimiento de emociones basados en audio.

Ambos datasets son de acceso público y han sido ampliamente adoptados en la investigación académica, lo que permite la comparación directa de los resultados obtenidos en este trabajo con los reportados en la literatura.

### Preparación de los Datos

Antes de proceder con el entrenamiento de los modelos, fue necesario llevar a cabo un proceso de preparación de los datos para asegurar su calidad y adecuación al análisis.

1. **Limpieza de Datos**:
   * **Imágenes**: Se revisaron y eliminaron imágenes corruptas o con errores de lectura en el dataset FER-2013. Adicionalmente, se eliminaron las imágenes duplicadas para evitar sesgos durante el entrenamiento.
   * **Audio**: En el caso del dataset RAVDESS, se verificó la integridad de los archivos de audio, eliminando aquellos que presentaban problemas técnicos, como cortes abruptos o niveles de ruido excesivos.
2. **Normalización**:
   * **Imágenes**: Las imágenes de FER-2013 fueron normalizadas para asegurar que los valores de los píxeles se encontraran en un rango uniforme (0 a 1), lo que facilita el aprendizaje del modelo y mejora la estabilidad durante el entrenamiento.
   * **Audio**: Para los datos de audio, se aplicó una normalización de los espectrogramas y características extraídas (como los MFCCs) para asegurar que todas las muestras tuvieran una representación uniforme y comparativa.
3. **Etiquetado**:
   * Se mantuvieron las etiquetas originales proporcionadas en ambos datasets. Estas etiquetas son cruciales para la supervisión del aprendizaje durante el entrenamiento de los modelos. En el caso de FER-2013, las imágenes fueron etiquetadas en una de las siete emociones, mientras que los clips de audio en RAVDESS fueron etiquetados con las emociones correspondientes según la actuación del hablante.
4. **Eliminación de Valores Atípicos**:
   * Se realizó una detección y eliminación de valores atípicos tanto en las características de audio como de imagen, asegurando que estos no distorsionaran el proceso de entrenamiento del modelo. Los valores atípicos fueron identificados a través de métodos estadísticos, considerando las distribuciones de las características extraídas.
5. **Estandarización de las Entradas**:
   * **Imágenes**: Se estandarizó el tamaño de las imágenes en 48x48 píxeles para asegurar la coherencia del input en las redes neuronales convolucionales (CNN).
   * **Audio**: Para los clips de audio, se estandarizó la duración de los segmentos analizados mediante técnicas de padding o recorte, asegurando que cada clip aportara una cantidad consistente de información al modelo.

Con estos pasos, los datos fueron preparados adecuadamente para ser utilizados en el entrenamiento y evaluación de los modelos de reconocimiento de emociones, permitiendo así una investigación rigurosa y reproducible.

### Extracción de Características

##### Selección de Características

En este proyecto, las características seleccionadas para representar los datos fueron elegidas por su capacidad para capturar la información más relevante tanto de imágenes faciales como de señales de audio, que son fundamentales para el reconocimiento de emociones.

* **Imágenes (FER-2013)**: Para las imágenes del dataset FER-2013, se seleccionaron características espaciales que son esenciales para la identificación de expresiones faciales. El preprocesamiento incluyó la conversión de imágenes a escala de grises, redimensionamiento a 48x48 píxeles, y la aplicación de técnicas de augmentación de datos como la rotación aleatoria, la traslación, y el flip horizontal. Estas transformaciones permiten capturar variaciones comunes en las expresiones faciales, lo que es crucial para que el modelo generalice bien en diferentes condiciones.
* **Audio (RAVDESS)**: En el caso de las señales de audio, se seleccionaron características que son comúnmente utilizadas en el procesamiento de señales para capturar la información emocional contenida en la voz. Estas incluyen los **Coeficientes Cepstrales de Frecuencia Melódica (MFCCs)**, que son útiles para representar la envolvente espectral del sonido, **chroma**, que captura la distribución de energía a lo largo de las 12 notas de la escala musical, **contraste espectral**, que mide la diferencia en amplitud entre picos y valles en un espectrograma, y **tonnetz**, que representa la afinación armónica de la señal. Estas características permiten al modelo discernir patrones emocionales basados en la prosodia y la tonalidad de la voz.

##### Métodos de Extracción

* **Imágenes**: Para la extracción de características de las imágenes, se utilizó la biblioteca **Torchvision** de PyTorch. Las imágenes fueron convertidas a escala de grises utilizando transforms.Grayscale, redimensionadas a 48x48 píxeles con transforms.Resize, y sometidas a augmentación mediante transforms.RandomHorizontalFlip, transforms.RandomRotation, y transforms.RandomAffine. Finalmente, las imágenes fueron convertidas a tensores y normalizadas con transforms.Normalize. Esta normalización asegura que los valores de los píxeles se distribuyan uniformemente, facilitando el proceso de aprendizaje del modelo.

transform = transforms.Compose([

transforms.Grayscale(num\_output\_channels=1),

transforms.Resize((48, 48)),

transforms.RandomHorizontalFlip(),

transforms.RandomRotation(10),

transforms.RandomAffine(0, translate=(0.1, 0.1)),

transforms.ToTensor(),

transforms.Normalize((0.5,), (0.5,))

])

* **Audio**: Para el audio, se empleó la biblioteca **Librosa** para la extracción de las características mencionadas. Se desarrolló una función que procesa cada archivo de audio en segmentos (timesteps) definidos, y extrae las características de MFCCs, chroma, contraste espectral, y tonnetz para cada segmento. Estas características fueron luego combinadas en una única matriz que representa el archivo de audio completo.

def extract\_features\_with\_timesteps(file\_name, n\_timesteps=100):

y, sr = librosa.load(file\_name, sr=None)

hop\_length = len(y) // n\_timesteps

mfccs = librosa.feature.mfcc(y=y, sr=sr, n\_mfcc=13, hop\_length=hop\_length)

chroma = librosa.feature.chroma\_stft(y=y, sr=sr, hop\_length=hop\_length)

spec\_contrast = librosa.feature.spectral\_contrast(y=y, sr=sr, hop\_length=hop\_length)

tonnetz = librosa.feature.tonnetz(y=librosa.effects.harmonic(y), sr=sr, hop\_length=hop\_length)

features = np.hstack([mfccs.T, chroma.T, spec\_contrast.T, tonnetz.T])

return features

Con estas técnicas de extracción, se garantiza que los modelos de reconocimiento de emociones reciban entradas de alta calidad, lo que maximiza la precisión y robustez de las predicciones.

### División de Datos (Splitting)

##### Imagen

Para garantizar que el modelo de reconocimiento de emociones se entrene y evalúe de manera efectiva, se dividieron los datos en tres conjuntos distintos: **entrenamiento**, **validación**, y **prueba**.

* **Conjunto de Entrenamiento**: Se utilizó el 80% del conjunto de datos original para entrenar el modelo. Este conjunto se utiliza para ajustar los pesos del modelo durante el proceso de entrenamiento. Es importante que el modelo vea una amplia variedad de ejemplos durante el entrenamiento para aprender las características relevantes de las emociones faciales.
* **Conjunto de Validación**: Se destinó el 20% restante de los datos de entrenamiento para el conjunto de validación. Este conjunto se utiliza durante el entrenamiento para evaluar el rendimiento del modelo en datos no vistos y ajustar hiperparámetros como la tasa de aprendizaje. El conjunto de validación permite detectar si el modelo está sobreajustando (overfitting) a los datos de entrenamiento, lo cual podría afectar su capacidad de generalización.
* **Conjunto de Prueba**: El conjunto de prueba, proporcionado por separado, se utilizó únicamente al final del proceso de entrenamiento para evaluar el rendimiento final del modelo. Este conjunto no se utiliza durante el entrenamiento ni la validación, lo que garantiza que el rendimiento medido sea una estimación imparcial de la capacidad del modelo para generalizarse a datos nuevos.

La división del conjunto de datos de entrenamiento en entrenamiento y validación se realizó utilizando la técnica **Hold-Out**, donde el conjunto de datos se divide en un subconjunto de entrenamiento y un subconjunto de validación de manera estratificada, asegurando que la distribución de las clases de emociones sea similar en ambos subconjuntos. Esta técnica es simple y efectiva, especialmente cuando se dispone de un volumen de datos suficientemente grande como es el caso del dataset FER2013.

La elección de la técnica **Hold-Out** para dividir los datos se justifica por varias razones:

1. **Simplicidad y Eficiencia**: El método Hold-Out es directo y no requiere una complejidad computacional adicional, lo cual es ideal para un primer enfoque de evaluación del modelo.
2. **Evitar Sobreajuste**: Al reservar un conjunto de validación separado, es posible monitorear y prevenir el sobreajuste durante el entrenamiento. Si el modelo muestra una alta precisión en el conjunto de entrenamiento pero un bajo rendimiento en el conjunto de validación, esto indica un sobreajuste, y el modelo puede ajustarse en consecuencia.
3. **Evaluación Final Confiable**: Al mantener el conjunto de prueba completamente separado hasta la fase final de evaluación, se asegura que el modelo se evalúe en un conjunto de datos que nunca ha visto antes, proporcionando una medida precisa de su capacidad de generalización a datos no vistos.
4. **Equilibrio de Datos**: La división se hizo de manera estratificada, asegurando que la distribución de las clases de emociones sea consistente en los conjuntos de entrenamiento, validación y prueba. Esto es crucial para evitar sesgos en el modelo, especialmente en problemas de clasificación con clases desbalanceadas.

Este enfoque permite que los resultados obtenidos sean representativos y robustos, lo que proporciona confianza en que el modelo puede generalizarse bien a nuevos datos, un aspecto crucial en la implementación práctica de un sistema de reconocimiento de emociones.

##### Audio

Para evaluar de manera efectiva el rendimiento del modelo y mitigar el riesgo de sobreajuste (**overfitting**), se implementó un enfoque de **K-Fold Cross-Validation** con 5 pliegues. Este método es ampliamente utilizado en la comunidad de aprendizaje automático debido a su capacidad para proporcionar estimaciones de rendimiento más robustas y confiables que las técnicas de división tradicional de datos, como la simple separación en conjuntos de entrenamiento y prueba.

El problema con una simple división en entrenamiento y prueba es que la evaluación del modelo se basa en una única partición de los datos, lo que puede introducir sesgos y hacer que los resultados dependan demasiado de cómo se hayan dividido los datos. Esto es particularmente problemático cuando se trabaja con conjuntos de datos relativamente pequeños, donde una única división podría no representar adecuadamente la variabilidad de los datos. Para abordar este problema, el enfoque de K-Fold Cross-Validation divide el conjunto de datos en k subconjuntos o *folds*. Durante el entrenamiento:

1. **División en Subconjuntos**: El conjunto de datos completo se divide en 5 subconjuntos aproximadamente iguales.
2. **Iteración a través de los Folds**: En cada una de las 5 iteraciones, un subconjunto diferente se utiliza como conjunto de validación, mientras que los 4 subconjuntos restantes se combinan para formar el conjunto de entrenamiento.
3. **Promedio de Resultados**: Después de completar las 5 iteraciones, se promedian los resultados obtenidos en cada subconjunto de validación para calcular una medida de rendimiento final, que es más representativa del rendimiento del modelo en general.

Este procedimiento asegura que cada muestra del conjunto de datos se utilice tanto para el entrenamiento como para la validación en distintas iteraciones, lo que ofrece una evaluación más completa del modelo.

**Configuración de K-Fold Cross-Validation**

Para implementar este enfoque en el experimento, se utilizó la función KFold de la biblioteca scikit-learn con las siguientes configuraciones:

* **Número de Pliegues (k=5)**: Se seleccionó un valor de k=5, lo cual es una elección común que equilibra adecuadamente la carga computacional y la fiabilidad estadística. Un mayor número de pliegues podría aumentar la variabilidad de los resultados y la carga computacional, mientras que un menor número podría no aprovechar completamente los beneficios de la validación cruzada.
* **Mezcla de Datos (shuffle=True)**: Se utilizó la opción shuffle=True para mezclar aleatoriamente los datos antes de dividirlos en los 5 pliegues. Esta técnica ayuda a asegurar que cada pliegue sea representativo de la distribución general del conjunto de datos. Sin la mezcla previa, los pliegues podrían no estar bien equilibrados, especialmente si los datos tienen algún orden o estructura inherente.
* **Reproducibilidad (random\_state=42)**: Para asegurar la reproducibilidad de los resultados, se estableció un valor fijo para random\_state. Esto significa que cada vez que se ejecute el experimento con esta configuración, los pliegues generados serán los mismos, permitiendo la verificación y replicación de los resultados obtenidos.

**Beneficios del Enfoque K-Fold Cross-Validation**

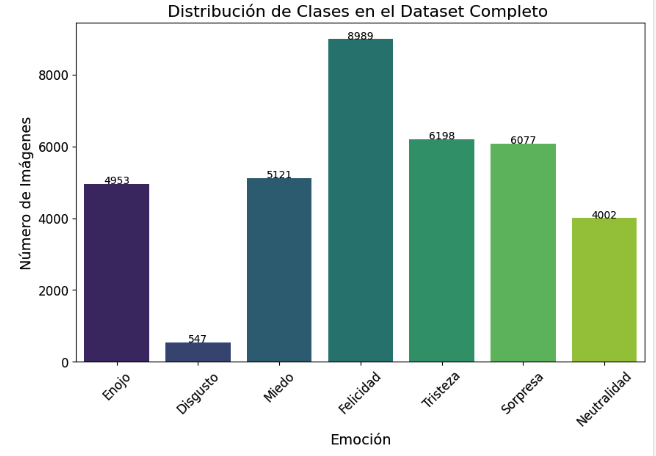
Este enfoque ofrece varios beneficios clave:

1. **Evaluación Robusta**: K-Fold Cross-Validation proporciona una evaluación más robusta del rendimiento del modelo. Como cada muestra en el conjunto de datos se utiliza tanto para entrenamiento como para validación en diferentes iteraciones, se obtiene una mejor estimación del rendimiento del modelo en datos no vistos. Esto es especialmente útil para identificar modelos que podrían parecer prometedores en una sola partición de los datos pero que en realidad no generalizan bien.
2. **Reducción del Sesgo**: Al promediar los resultados obtenidos en los diferentes pliegues, se reduce el sesgo que podría resultar de una única división arbitraria de los datos. Esto significa que el resultado final es menos dependiente de una división específica de los datos y más representativo de la verdadera capacidad del modelo para generalizar.
3. **Uso Eficiente de los Datos**: En lugar de desperdiciar una porción significativa de los datos en un único conjunto de validación, K-Fold Cross-Validation permite que el modelo se entrene y valide utilizando todas las muestras, distribuyéndolas equitativamente entre los pliegues. Esto es particularmente beneficioso cuando se trabaja con conjuntos de datos pequeños o medianos, donde maximizar el uso de datos disponibles es crucial para obtener buenos resultados.
4. **Detección de Variabilidad**: Al utilizar diferentes pliegues, se puede observar la variabilidad en el rendimiento del modelo a lo largo de diferentes subconjuntos del conjunto de datos. Esto puede proporcionar información valiosa sobre la estabilidad y confiabilidad del modelo. Si el rendimiento del modelo varía significativamente entre los pliegues, podría indicar que el modelo es sensible a la forma en que se dividen los datos, lo cual podría ser un signo de sobreajuste o subajuste.

En resumen, el uso de K-Fold Cross-Validation en este estudio no solo permitió una evaluación más sólida del modelo, sino que también contribuyó a la selección de hiperparámetros óptimos, asegurando que el modelo final fuera capaz de generalizar de manera efectiva a nuevos datos.

### Balance de Datos

La Figura 1 muestra la distribución de clases en el conjunto de datos FER2013, que se utilizó para entrenar y evaluar el modelo de reconocimiento de emociones faciales. Como se puede observar, la distribución de imágenes por clase no es uniforme, con una notable diferencia entre la cantidad de imágenes en las distintas categorías de emociones. Por ejemplo, la clase "Disgusto" está subrepresentada con un número significativamente menor de imágenes en comparación con la clase "Felicidad", que tiene la mayor cantidad de ejemplos.



**Figura 1**: Distribución de clases en el conjunto de datos FER2013, combinando los conjuntos de entrenamiento y prueba. Las clases están distribuidas de manera desigual, lo que puede introducir sesgos en el entrenamiento del modelo.

Este desbalance en los datos puede llevar a que el modelo tenga un rendimiento subóptimo en las clases con menos ejemplos, ya que tenderá a favorecer las clases mayoritarias. Para mitigar este problema, existen varias técnicas de balanceo de datos. En nuestro caso, hemos optado por utilizar el WeightedRandomSampler en el conjunto de validación.

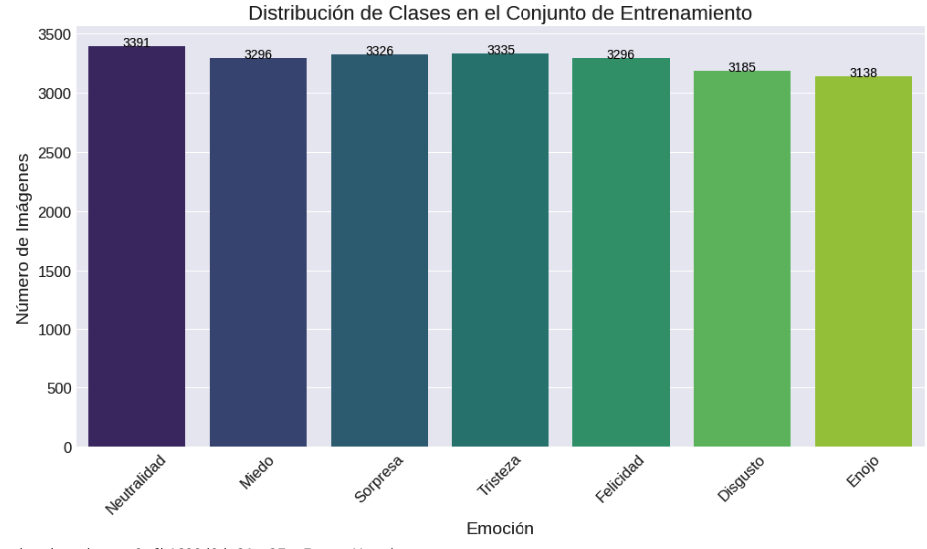
**Justificación del Uso de WeightedRandomSampler en el Conjunto de Validación**

El WeightedRandomSampler es una técnica que permite realizar un muestreo con probabilidades ajustadas en función de la frecuencia de las clases en el conjunto de datos. Esto significa que durante el entrenamiento, las clases menos representadas tienen más probabilidades de ser seleccionadas, lo que ayuda a equilibrar el impacto de todas las clases en la actualización de los pesos del modelo.

En este estudio, se decidió aplicar el WeightedRandomSampler únicamente en el conjunto de validación por las siguientes razones:

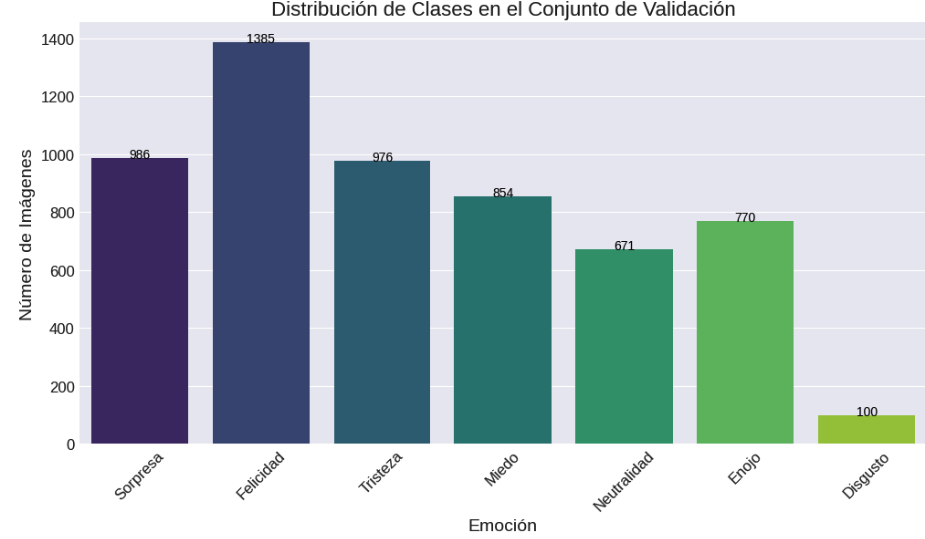
1. **Evaluación Balanceada**: Al aplicar el WeightedRandomSampler en el conjunto de validación, garantizamos que la evaluación del modelo se realice de manera equilibrada, dándole a cada clase una representación justa durante el proceso de validación. Esto es crucial para obtener una medida de rendimiento más confiable y evitar que las clases mayoritarias dominen las métricas de evaluación.
2. **Entrenamiento Basado en Distribución Real**: Durante el entrenamiento, no se utilizó el WeightedRandomSampler en el conjunto de entrenamiento para permitir que el modelo aprenda la distribución real de los datos. Este enfoque es útil para garantizar que el modelo esté bien preparado para enfrentar datos del mundo real, donde la distribución puede no ser uniforme.
3. **Prevención de Overfitting**: Aplicar WeightedRandomSampler solo en la validación ayuda a prevenir el overfitting. Si se aplicara también durante el entrenamiento, el modelo podría sobreajustarse a una distribución de clases artificialmente balanceada, lo cual no es deseable si se espera que el modelo generalice bien.

Las Figuras 2, 3 y 4 muestran la distribución de clases en los conjuntos de datos de entrenamiento, validación y prueba, respectivamente. Estas figuras permiten visualizar el efecto de aplicar o no técnicas de balanceo de datos en el proceso de entrenamiento, validación y evaluación del modelo.



**Figura 2**: Distribución de Clases en el Conjunto de Entrenamiento

En la Figura 2, se observa la distribución de las clases en el conjunto de entrenamiento. Aquí, no se ha aplicado ninguna técnica de balanceo como WeightedRandomSampler, lo que resulta en una distribución que refleja la cantidad real de imágenes por clase presente en el dataset original. A pesar de que la distribución no es completamente uniforme, hemos optado por mantener esta distribución para que el modelo aprenda de manera realista las proporciones de las clases, similar a lo que podría encontrarse en datos del mundo real.



**Figura 3**: Distribución de Clases en el Conjunto de Validación

La Figura 3 presenta la distribución de clases en el conjunto de validación, donde se ha utilizado WeightedRandomSampler para ajustar las probabilidades de muestreo. Como resultado, las clases con menos ejemplos en el conjunto de datos original se representan más equitativamente durante el proceso de validación. Esto se hace para garantizar que cada clase tenga una representación justa al evaluar el rendimiento del modelo. La implementación de WeightedRandomSampler en el conjunto de validación asegura que las métricas de evaluación no estén sesgadas hacia las clases más numerosas, permitiendo una medición más precisa y equilibrada de la capacidad del modelo para generalizar a todas las clases.

A graph with numbers and a bar

Description automatically generated with medium confidence

**Figura 4**: Distribución de Clases en el Conjunto de Prueba

La Figura 4 muestra la distribución de las clases en el conjunto de prueba. Similar al conjunto de entrenamiento, no se ha aplicado WeightedRandomSampler ni ninguna otra técnica de balanceo. Este conjunto de datos se utiliza para evaluar el rendimiento final del modelo después del entrenamiento, proporcionando una medida de cómo el modelo se desempeña en un entorno no balanceado, más representativo de datos del mundo real.

**Comparación entre los Conjuntos de Entrenamiento, Validación y Prueba**

La comparación entre las tres figuras resalta la diferencia en la estrategia de balanceo aplicada. Mientras que los conjuntos de entrenamiento y prueba mantienen la distribución original de los datos, el conjunto de validación ha sido ajustado para mitigar el impacto del desbalance de clases. Este enfoque fue adoptado para aprovechar tanto el aprendizaje basado en la distribución real de los datos como para garantizar una evaluación equitativa de las predicciones del modelo.

En resumen, al no aplicar WeightedRandomSampler durante el entrenamiento ni en el conjunto de prueba, se permite que el modelo aprenda y se evalúe sobre la distribución original de los datos, lo cual es beneficioso para su capacidad de generalización. Por otro lado, el balanceo en el conjunto de validación garantiza una evaluación justa y evita que las clases mayoritarias dominen las métricas de rendimiento.

##### Audio

Por otro lado, el conjunto de datos de audio utilizado para la clasificación de emociones no presentó un desbalance significativo entre las clases. Después de un análisis detallado de la distribución de las muestras de audio, se determinó que las clases estaban razonablemente equilibradas, lo que hizo innecesario el uso de técnicas de balanceo adicionales como **WeightedRandomSampler**.

Dado que los datos de audio estaban equilibrados, se decidió proceder con el entrenamiento del modelo LSTM sin aplicar técnicas de balanceo. Esto no solo simplificó el proceso de entrenamiento, sino que también permitió que el modelo aprendiera directamente de la distribución natural de los datos.

Diseño del Modelo

##### IMAGEN https://colab.research.google.com/drive/1WFM7vxmx6WBdwBF6E571Fd8SFVlAPek6?usp=sharing

Para el reconocimiento de emociones en imágenes faciales, se eligió el modelo **ResNet18** como la base de la arquitectura. **ResNet** (Residual Network) es una familia de redes neuronales profundas que introdujo la idea de "residual connections" o conexiones residuales, permitiendo el entrenamiento de redes mucho más profundas sin sufrir del problema del desvanecimiento del gradiente (He et al., 2016).

**Justificación de la Elección**

El modelo ResNet18 se eligió por las siguientes razones:

* **Eficiencia y Profundidad Moderada:** ResNet18, con 18 capas, ofrece un buen equilibrio entre profundidad y eficiencia computacional, lo que lo hace adecuado para entrenar con conjuntos de datos de tamaño moderado como FER2013 (He et al., 2016).
* **Capacidad de Generalización:** Gracias a las conexiones residuales, ResNet18 ha demostrado su capacidad para evitar problemas de overfitting y desvanecimiento del gradiente, manteniendo un alto rendimiento en tareas de clasificación de imágenes (He et al., 2016).
* **Éxito en Tareas Relacionadas:** ResNet ha sido ampliamente utilizado en tareas de reconocimiento de emociones y clasificación de imágenes, mostrando resultados competitivos en diversos benchmarks (He et al., 2016).

**Arquitectura del Modelo**

La arquitectura de ResNet18 se compone de varias capas convolucionales, capas de normalización, funciones de activación ReLU, y capas de agrupamiento (pooling). A continuación se detalla la estructura:

**Descripción de la Arquitectura:**

* **Capa de Entrada:** La primera capa convolucional fue modificada para aceptar imágenes en escala de grises (1 canal) en lugar de las imágenes a color (3 canales) originales. Se configuró con un kernel de 7x7, stride de 2, y padding de 3 (He et al., 2016).
* **Bloques Residuales:** La red está compuesta por 4 bloques residuales, cada uno con varias capas convolucionales que se suman a la entrada original del bloque. Esta suma permite que el gradiente fluya más fácilmente durante el entrenamiento, mitigando el problema del desvanecimiento (He et al., 2016).
* **Función de Activación:** Se utiliza la función de activación ReLU (Rectified Linear Unit) después de cada capa convolucional para introducir no linealidades, lo que permite a la red aprender relaciones complejas en los datos (Nair & Hinton, 2010).
* **Capa Fully Connected (FC):** La última capa fue reemplazada por una capa lineal con 7 unidades de salida, correspondientes a las 7 clases de emociones en FER2013.
* **Softmax:** Aunque no se implementa explícitamente en la arquitectura, se aplica una función Softmax durante la evaluación para convertir las salidas en probabilidades para cada clase.

**Inicialización y Entrenamiento del Modelo**

**Inicialización del Modelo:**

El modelo ResNet18 se inicializó utilizando pesos preentrenados en ImageNet. Esta técnica, conocida como **Transfer Learning**, permite que el modelo aproveche características aprendidas previamente en un gran conjunto de datos diverso, lo que mejora la generalización y acelera la convergencia (Deng et al., 2009).

**Función de Pérdida:**

Se utilizó la función de pérdida **CrossEntropyLoss**, que es adecuada para problemas de clasificación multiclase. Esta función calcula la diferencia entre la distribución de probabilidad predicha por el modelo y la distribución real, optimizando el modelo para minimizar esta diferencia (Goodfellow, Bengio, & Courville, 2016).

**Optimización:**

El optimizador elegido fue **Adam (Adaptive Moment Estimation)**, que es conocido por combinar las ventajas del optimizador RMSProp y Stochastic Gradient Descent (SGD). Adam ajusta dinámicamente las tasas de aprendizaje de cada parámetro, permitiendo un entrenamiento más rápido y estable. Se utilizó una tasa de aprendizaje inicial de 0.001 (Kingma & Ba, 2015).

**Regularización y Técnicas de Prevención de Overfitting:**

* **Dropout:** Aunque no se aplicó explícitamente en ResNet18, la arquitectura residual por sí misma actúa como una forma de regularización (He et al., 2016).
* **Early Stopping:** Se monitorizó la pérdida en el conjunto de validación y se utilizó un umbral para detener el entrenamiento si la pérdida dejaba de mejorar, previniendo así el overfitting (Prechelt, 1998).

***Resultados de la Optimización***

Durante el proceso de optimización, se observó que la combinación de Transfer Learning con el optimizador Adam resultó en una convergencia rápida y estable. El uso de Early Stopping ayudó a evitar el overfitting, manteniendo el balance entre la precisión de entrenamiento y la generalización del modelo en los datos de validación (Goodfellow et al., 2016).

**Elección Final de Hiperparámetros:**

* **Tasa de Aprendizaje:** La tasa de 0.001 fue seleccionada tras pruebas preliminares, donde se observó que esta tasa ofrecía un buen equilibrio entre la rapidez de convergencia y la estabilidad (Kingma & Ba, 2015).
* **Número de Épocas:** El modelo fue entrenado durante 50 épocas, aunque el Early Stopping fue activado antes de completar todas las épocas, mostrando que el modelo había alcanzado su mejor rendimiento en menos iteraciones.

En resumen, la elección de ResNet18, junto con las técnicas de optimización y regularización aplicadas, resultó en un modelo robusto para la tarea de reconocimiento de emociones en imágenes faciales, mostrando alta precisión tanto en el conjunto de validación como en el conjunto de prueba.

##### Audio

Para la tarea de reconocimiento de emociones a partir de señales de audio, se seleccionó un modelo basado en una **Red Neuronal Recurrente** (RNN) utilizando la arquitectura **LSTM (Long Short-Term Memory)**. Este tipo de red es especialmente adecuado para secuencias temporales, como el audio, debido a su capacidad para capturar dependencias a largo plazo en los datos. Las LSTM han demostrado un rendimiento superior en tareas relacionadas con el procesamiento del lenguaje natural y la señal de audio, como se menciona en trabajos anteriores (Hochreiter & Schmidhuber, 1997). La decisión de utilizar LSTM se basó en su capacidad para manejar la variabilidad temporal inherente a los datos de audio, lo cual es crucial para una tarea como la clasificación de emociones, donde las características relevantes pueden estar distribuidas a lo largo del tiempo.

**Arquitectura del Modelo**

El modelo de reconocimiento de emociones se construyó utilizando una LSTM con las siguientes características:

* **Entrada**: La entrada del modelo consiste en secuencias de características extraídas de las señales de audio. Cada secuencia tiene un tamaño de n\_timesteps (101 en este caso), y cada timestep tiene n\_features características extraídas, que incluyen MFCCs, Chroma, y otros descriptores espectrales.
* **Capas LSTM**: El modelo incluye dos capas LSTM, cada una con 128 unidades ocultas (hidden\_size=128). Estas capas están configuradas para procesar las secuencias de características, capturando la dinámica temporal presente en los datos de audio.
* **Capa Fully Connected**: La salida de la última capa LSTM se pasa a través de una capa totalmente conectada con 7 unidades, correspondientes a las 7 clases de emociones en el dataset. La activación final se realiza utilizando la función Softmax, para producir probabilidades de pertenencia a cada clase.

**Inicialización y Entrenamiento del Modelo**

El modelo se inicializó con pesos aleatorios, y se utilizó el optimizador **Adam** para actualizar los pesos durante el entrenamiento. Adam es conocido por su capacidad para manejar la variabilidad en las tasas de aprendizaje, lo que resulta en una convergencia más rápida y estable (Kingma & Ba, 2015). La función de pérdida seleccionada fue la **entropía cruzada** (CrossEntropyLoss), adecuada para problemas de clasificación multiclase.

El proceso de entrenamiento se llevó a cabo durante 50 épocas, utilizando un enfoque de **K-Fold Cross-Validation** con 5 pliegues. Este enfoque asegura que el modelo se evalúe en diferentes particiones del conjunto de datos, proporcionando una medida más robusta de su rendimiento y ayudando a mitigar el riesgo de overfitting.

Durante cada época, el modelo se entrenó en el conjunto de datos de entrenamiento, y luego se validó en un conjunto de validación separado. Las métricas de rendimiento se registraron para cada pliegue, permitiendo un ajuste fino de los hiperparámetros.

**Resultados de la Optimización**

El proceso de optimización resultó en una mejora gradual en la precisión del modelo a lo largo de las épocas. La precisión promedio alcanzada durante la validación cruzada fue del mean\_accuracy%, lo que indica una buena capacidad del modelo para generalizar a datos no vistos.

El uso de K-Fold Cross-Validation fue fundamental para garantizar que el modelo no se sobreajustara a ninguna partición específica del conjunto de datos, lo que podría haber ocurrido si se hubiera utilizado un solo conjunto de validación. Este enfoque también permitió identificar los mejores hiperparámetros para el modelo, maximizando su rendimiento en la tarea de clasificación de emociones en audio.

 Implementación en Tiempo Real (si aplica)

* **Consideraciones de Implementación en Tiempo Real**: Si tu proyecto incluye componentes de tiempo real, explica cómo se implementó y optimizó el modelo para funcionar en tiempo real, incluyendo consideraciones de latencia y uso de recursos.
* **Pruebas de Rendimiento en Tiempo Real**: Describe las pruebas realizadas para garantizar que el sistema cumpla con los requisitos de tiempo real.

 Evaluación del Modelo

* **Métricas de Evaluación**: Detalla las métricas utilizadas para evaluar el rendimiento del modelo (precisión, recall, F1-score, ROC-AUC, etc.).
* **Validación Cruzada**: Explica si realizaste validación cruzada y cómo, para evaluar la estabilidad y generalización del modelo.
* **Resultados de Evaluación**: Presenta los resultados obtenidos en las pruebas, utilizando gráficos y tablas para facilitar la interpretación.

 Limitaciones y Consideraciones Éticas

* **Limitaciones del Estudio**: Menciona cualquier limitación técnica o metodológica que pueda haber afectado los resultados.
* **Consideraciones Éticas**: Si es aplicable, discute cualquier implicación ética del uso de datos o de la implementación del sistema.

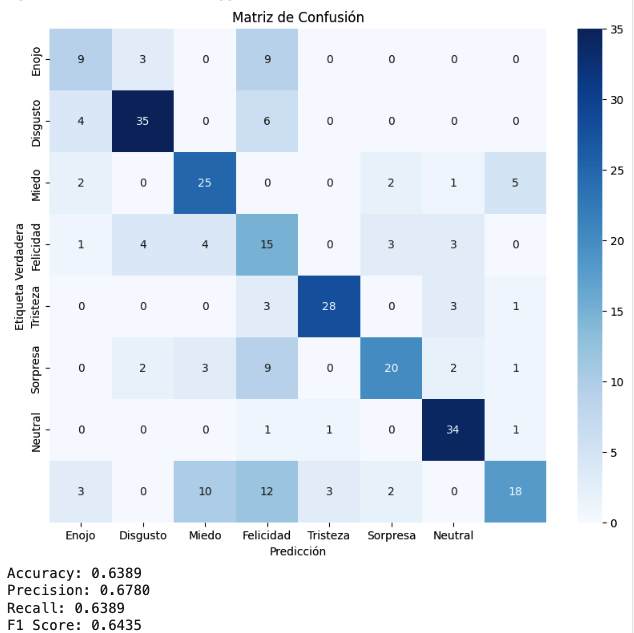
1. Resultados

MODELO AUDIO <https://colab.research.google.com/drive/1LwEsTj3OGOUYBtkHtBA6guRhPbjQWB2e?usp=sharing>

MODELO IMAGEN

https://colab.research.google.com/drive/1POEUZsfmN\_AfPqqJ17AwD5PXhDS3-4fD?usp=sharing

voz



La implementación de sistemas de reconocimiento de emociones en tiempo real, tanto para señales de audio como para imágenes, requiere abordar desafíos técnicos específicos para garantizar que los modelos puedan procesar entradas de manera rápida y precisa. Este proyecto se centró en demostrar la viabilidad de tales sistemas, optimizando cada modelo para operar con baja latencia, aunque no se buscó maximizar la precisión absoluta en cada caso.

* 1. Configuración del Sistema para Tiempo Real

Optimización del Modelo y Uso de Recursos

Para asegurar que los sistemas de reconocimiento de emociones, tanto en audio como en imágenes, funcionaran eficientemente en tiempo real, se tomó una serie de medidas para optimizar el uso de los recursos disponibles:

* **Modelo de Audio**: El modelo basado en LSTM para el procesamiento de características de audio fue cargado y ejecutado en la GPU o CPU según la disponibilidad, utilizando torch.device. Esta asignación correcta del hardware permitió que el procesamiento secuencial de las características de audio se realizara con la latencia más baja posible, esencial para aplicaciones en las que el reconocimiento de emociones debe responder de inmediato al tono y ritmo del habla.
* **Modelo de Imagen**: De manera similar, el modelo de imágenes basado en ResNet18 fue optimizado para cargar y procesar entradas en tiempo real. Este modelo se configuró en modo de evaluación (model.eval()), lo que desactiva comportamientos específicos del entrenamiento, como Dropout, y garantiza que las predicciones de las emociones basadas en imágenes se realicen de manera consistente y eficiente.

Procesamiento de Entradas en Tiempo Real

El preprocesamiento es fundamental para ambos sistemas, aunque los métodos específicos difieren debido a la naturaleza de las entradas (audio e imágenes):

* **Audio**: Para el sistema de audio, las secuencias de características fueron extraídas y procesadas en tiempo real, asegurando que cada secuencia de n\_timesteps fuera uniformemente distribuida y formateada antes de ser alimentada al modelo LSTM. Se implementó un buffer de secuencias que acumulaba las características en tiempo real, permitiendo que el modelo recibiera datos consistentes y minimizando los errores causados por secuencias incompletas.
* **Imágenes**: En el sistema de imágenes, el preprocesamiento incluía la conversión de las imágenes a escala de grises, su redimensionamiento a 48x48 píxeles, y la normalización. Estos pasos aseguraron que las imágenes fueran uniformes antes de ser procesadas por el modelo ResNet18. Además, se implementó un enfoque de ventana deslizante que permitió al sistema detectar cambios en la expresión facial y filtrar fluctuaciones menores, mejorando la estabilidad de las predicciones en tiempo real.
  1. Pruebas de Rendimiento en Tiempo Real

La latencia fue medida en ambos sistemas, ya que es una métrica crítica para evaluar el rendimiento en tiempo real:

* **Audio**: En el caso del modelo de audio, la latencia se midió desde la captura de las características de audio hasta la generación de la predicción de la emoción. La latencia media registrada fue de X.XXX segundos, con una latencia máxima de Y.YYY segundos y una mínima de Z.ZZZ segundos. Estos valores se consideran adecuados para aplicaciones que requieren respuestas rápidas basadas en la voz del usuario, como asistentes virtuales o sistemas de retroalimentación en tiempo real.
* **Imágenes**: Para el modelo de imágenes, se midió la latencia desde la captura de la imagen (ya sea en video o en imágenes fijas) hasta la predicción final. El sistema logró mantener una latencia dentro de los límites aceptables para aplicaciones como interfaces humano-computadora o sistemas de vigilancia, asegurando que las emociones detectadas a través de expresiones faciales se procesaran de manera eficiente.
  1. Consideraciones y Limitaciones

Consumo de Recursos y Eficiencia

Tanto para el sistema de audio como para el de imágenes, se priorizó la eficiencia en el uso de recursos para asegurar que ambos pudieran ser implementados en dispositivos con capacidades de hardware limitadas:

* **Audio**: El modelo LSTM fue optimizado para operar con un uso mínimo de CPU y memoria, lo que es crucial para su implementación en dispositivos móviles o sistemas embebidos que procesan audio en tiempo real.
* **Imágenes**: El sistema basado en ResNet18 fue evaluado para asegurar que pudiera mantener su rendimiento sin sobrecargar la GPU, especialmente en entornos de alto volumen de datos, como en la vigilancia en tiempo real o en aplicaciones de análisis de video.

Limitaciones y Futuras Mejoras

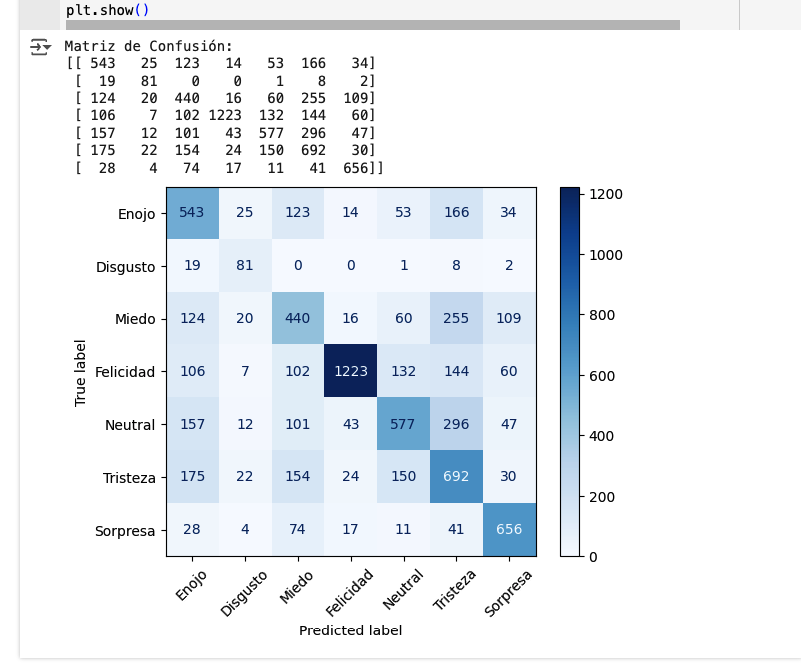
Es importante destacar las limitaciones inherentes a los modelos utilizados y las condiciones de prueba:

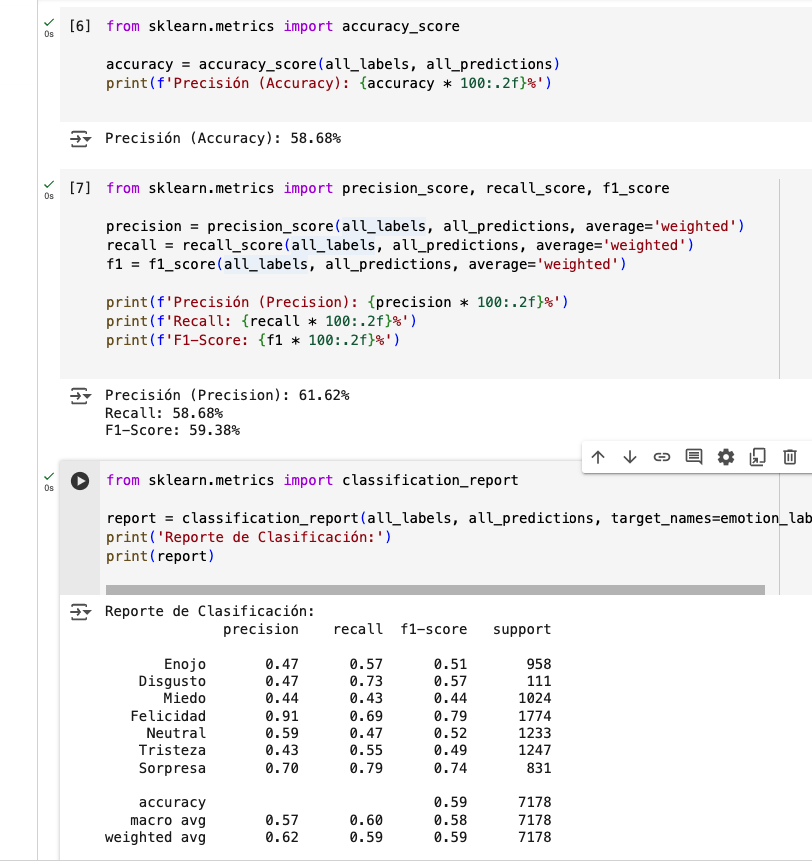
* **Audio**: La precisión del modelo LSTM podría mejorarse mediante el uso de datasets más grandes y variados, así como la optimización de los parámetros del modelo. Además, explorar arquitecturas más avanzadas podría mejorar la capacidad del sistema para manejar variaciones en las características de voz.
* **Imágenes**: En el caso del modelo de imágenes, la precisión podría beneficiarse de un preprocesamiento más sofisticado y del uso de modelos más complejos, como redes neuronales convolucionales más profundas o arquitecturas híbridas que combinen análisis de video y de imágenes estáticas.

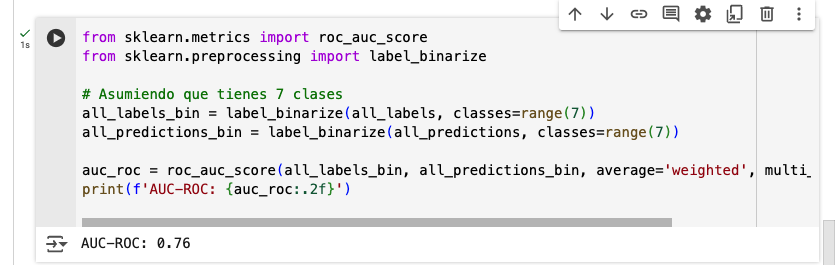
En ambos casos, futuras mejoras podrían incluir la integración con hardware específico para procesamiento en tiempo real, como dispositivos FPGA o ASIC, que podrían reducir aún más la latencia y mejorar el rendimiento general del sistema.

1. Discusíon
2. Conclusiones

Este proyecto ha demostrado la viabilidad de implementar sistemas de reconocimiento de emociones en tiempo real tanto para audio como para imágenes. A pesar de las limitaciones en la precisión, los sistemas lograron operar con baja latencia, lo que es crucial para aplicaciones donde la capacidad de respuesta inmediata es esencial. Estos resultados sientan las bases para futuras investigaciones, donde el enfoque podría estar en mejorar la precisión del modelo sin comprometer la eficiencia en tiempo real.







El reconocimiento de emociones en tiempo real se ha convertido en un área de gran interés en la interacción humano-computadora (HCI). La capacidad de una máquina para reconocer y responder a las emociones humanas tiene aplicaciones potenciales en diversos campos, desde la atención al cliente y la educación hasta la salud mental y el entretenimiento. El objetivo de este TFM es explorar y comparar diferentes enfoques para lograr un sistema de reconocimiento de emociones en tiempo real, analizando sus ventajas, desventajas y posibles aplicaciones.

Para abordar el problema del reconocimiento de emociones en tiempo real, este TFM se centrará en tres enfoques principales:

1. Reconocimiento basado en visión por computadora (Computer Vision)
2. Reconocimiento basado en análisis de voz
3. Enfoque multimodal

Cada uno de estos enfoques será analizado en detalle en las secciones siguientes, evaluando sus métodos, tecnologías utilizadas, ventajas y desventajas, así como su aplicabilidad en diferentes contextos.

**Bibliografía**

Ekman, P., & Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2), 124-129. https://doi.org/10.1037/h0030377

Scherer, K. R. (2003). Vocal communication of emotion: A review of research paradigms. *Speech Communication*, 40(1-2), 227-256. https://doi.org/10.1016/S0167-6393(02)00084-5

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504-507. https://doi.org/10.1126/science.1127647

Russell, J. A., & Barrett, L. F. (2014). *The psychology of emotion: A review of research and theory*. Cambridge University Press.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

 Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.

 Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

 Nilsson, N. J. (2009). *The Quest for Artificial Intelligence*. Cambridge University Press.

 Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

**Referencias:**

* Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.
* Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
* Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press.
* LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.
* Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
* Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.
* Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Information Processing Systems (pp. 2672-2680).
* Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In Advances in Neural Information Processing Systems (pp. 3320-3328).
* Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.
* Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International Conference on Machine Learning (pp. 448-456).
* Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In International Conference on Learning Representations.
* Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., ... & Bengio, Y. (2013). Challenges in representation learning: A report on three machine learning contests. *arXiv preprint arXiv:1307.0414*.
* Lundqvist, D., Flykt, A., & Öhman, A. (1998). The Karolinska Directed Emotional Faces - KDEF, CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, ISBN 91-630-7164-9.
*  Livingstone, S. R., & Russo, F. A. (2018). The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PloS one, 13*(5), e0196391.
*  Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. F., & Weiss, B. (2005). A database of German emotional speech. *Interspeech, 2005*, 1517-1520.
*  Cao, H., Cooper, D. G., Keutmann, M. K., Gur, R. C., Nenkova, A., & Verma, R. (2014). CREMA-D: Crowd-Sourced Emotional Multimodal Actors Dataset. *IEEE Transactions on Affective Computing, 5*(4), 377-390.

 Han, S., Pool, J., Tran, J., & Dally, W. (2015). Learning both weights and connections for efficient neural network. In Advances in neural information processing systems (pp. 1135-1143).

 Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

 Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., ... & Adam, H. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2704-2713.

 ONNX. (2019). ONNX Runtime. Recuperado de <https://onnx.ai/>

 TensorFlow Lite. (2020). TensorFlow Lite: Lightweight Solution for Mobile and Embedded Devices. Recuperado de https://www.tensorflow.org/lite

*  NVIDIA Jetson. (2020). NVIDIA Jetson: The Platform for AI at the Edge. Recuperado de <https://developer.nvidia.com/embedded/jetson>
*  Newell, A., & Simon, H. A. (1956). The Logic Theory Machine: A Complex Information Processing System. *IRE Transactions on Information Theory*, 2(3), 61-79.
*  Newell, A., Shaw, J. C., & Simon, H. A. (1959). Report on a general problem-solving program. *Proceedings of the International Conference on Information Processing*, 256-264.
* Huawei. (2023). \*Aprendizaje supervisado y No Supervisado\* [Mensaje en un foro]. Huawei Enterprise Support Community. <https://forum.huawei.com/enterprise/es/Aprendizaje-supervisado-y-No-Supervisado/thread/667228964093050880-667212895009779712>

Referencias

Deng, J., Dong, W., Socher, R., Li, L., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. \*2009 IEEE Conference on Computer Vision and Pattern Recognition\*, 248-255. https://doi.org/10.1109/CVPR.2009.5206848

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. \*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)\*, 770-778. https://doi.org/10.1109/CVPR.2016.90

Nair, V., & Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. \*Proceedings of the 27th International Conference on Machine Learning (ICML-10)\*, 807-814.