Tarea 6. Análisis y clasificación de audio utilizando espectrogramas y redes neuronales convolucionales

Iván Gabriel Salinas Castillo

8 de marzo de 2025

1. Introducción

El análisis de señales de audio es un campo de estudio importante en la inteligencia artificial, con aplicaciones en reconocimiento de voz, identificación de hablantes y procesamiento de lenguaje natural [1]. En este trabajo, se aborda el problema de clasificación de hablantes utilizando espectrogramas y redes neuronales convolucionales (CNN). El objetivo es identificar a tres hablantes específicos (George, Lucas y Nicolas) a partir de sus grabaciones de audio, utilizando técnicas de procesamiento de señales y aprendizaje profundo [2] [3].

La identificación de hablantes es un problema desafiante debido a la variabilidad en las características de la voz, como el tono, la velocidad y el acento. Para abordar este problema, se utilizan espectrogramas, que son representaciones visuales de las frecuencias de una señal de audio a lo largo del tiempo. Estos espectrogramas se utilizan como entrada para una CNN, que aprende a clasificar a los hablantes basándose en patrones en los datos.

Este trabajo se basa en el dataset *Spoken MNIST* [4], que contiene grabaciones de dígitos hablados por diferentes individuos. Se seleccionaron tres hablantes específicos para este estudio, y se implementó un modelo de CNN para clasificar sus voces.

2. Metodología

2.1. Preprocesamiento de Datos

El dataset utilizado es *Spoken MNIST* [4], que contiene grabaciones de audio de dígitos del 0 al 9 pronunciados por diferentes hablantes. Para este estudio, se filtraron los datos para incluir solo las grabaciones de tres hablantes: George, Lucas y Nicolas. Los espectrogramas precalculados [1] se utilizaron como características principales para el modelo. El dataset cuenta con las columnas spectrograms, labels, audio y speakers [5] .

Se realizó un balanceo de clases para asegurar que cada hablante tuviera la misma cantidad de muestras en el conjunto de entrenamiento. Además, se redujeron los espectrogramas de 4 canales a 1 canal promediando los valores de los canales.

2.2. Modelo de Red Neuronal Convolucional

Se implementó una CNN con la siguiente arquitectura:

- Capa de entrada con la forma de los espectrogramas.
- Dos capas convolucionales con 32 y 64 filtros, respectivamente, seguidas de capas de *max pooling*.
- Una capa densa de 64 neuronas con activación ReLU.
- Capa de salida con 3 neuronas (una por cada hablante) y activación softmax.

El modelo se compiló utilizando el optimizador Adam con una tasa de aprendizaje de 0.0001 y se entrenó durante 10 épocas. La función de pérdida utilizada fue *sparse categorical crossentropy*.

2.3. Evaluación del Modelo

El modelo se evaluó utilizando el conjunto de prueba, calculando la precisión y el F1-score. Además, se generó un informe de clasificación que incluye métricas como precisión, recall y F1-score para cada hablante utilizando herramientas de *TensorFlow* [2] [6].

3. Resultados

3.1. Gráficas de Espectrogramas

En la Figura 1 se muestran los espectrogramas promedio para cada hablante. Estas gráficas permiten visualizar las diferencias en las características espectrales de las voces de George, Lucas y Nicolas.

3.2. Comparación de Audios

En la Figura 2 se comparan las formas de onda de los audios de los tres hablantes para el dígito 9. Se observan diferencias en la amplitud y la forma de las señales, lo que refleja las características únicas de cada hablante.

3.3. Frecuencias Representativas

En la Figura 3 se muestran las frecuencias más representativas para cada hablante, calculadas a partir de la transformada de Fourier de tiempo corto (STFT). Estas frecuencias son clave para distinguir entre los hablantes.

$$\begin{aligned} \text{Amplitudes George:} &\quad -133{,}51, -62{,}47 \\ \text{Amplitudes Lucas:} &\quad -348{,}55, -294{,}73 \\ \text{Amplitudes Nicolas:} &\quad -151{,}65, -74{,}61 \end{aligned}$$

3.4. Precisión del Modelo

El modelo alcanzó una precisión del 97 % en el conjunto de prueba. El F1-score ponderado fue de 96.88 %, lo que indica un buen rendimiento en la clasificación de los tres hablantes.

4. Conclusión

El modelo de CNN implementado demostró ser efectivo para la clasificación de hablantes, alcanzando una precisión del 97 % y un F1-score de 96.88 %. Las gráficas de espectrogramas y frecuencias representativas permitieron visualizar las diferencias entre las voces de los hablantes, lo que respalda la efectividad del enfoque basado en espectrogramas.

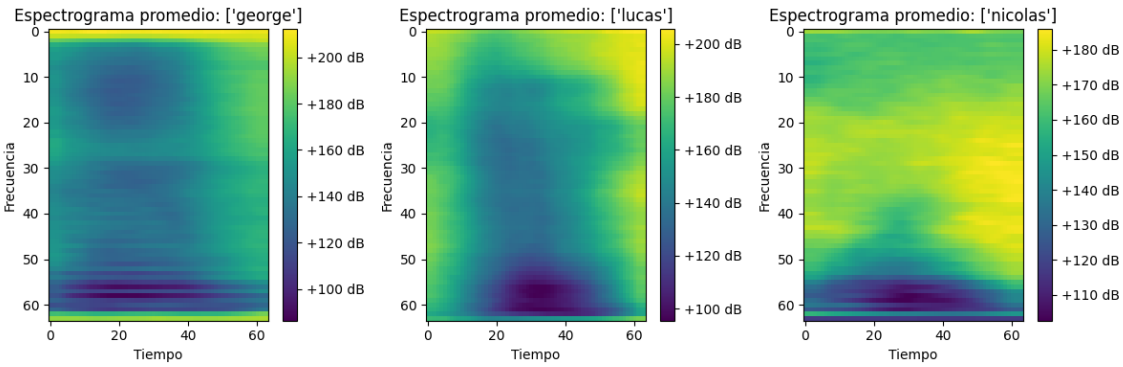

Figura 1: Espectrogramas promedio para cada hablante.

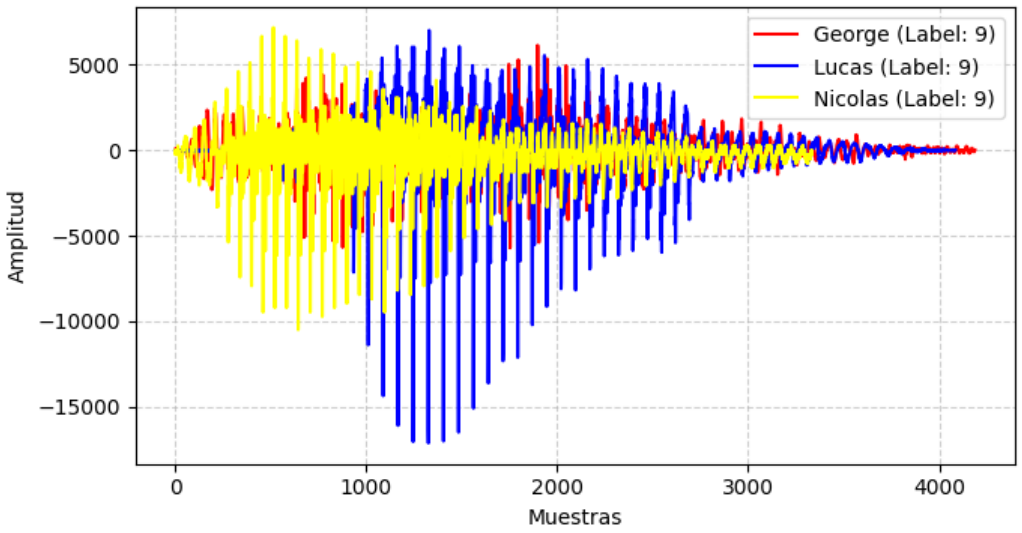

Figura 2: Comparación de audios para el dígito 9.

Este trabajo sugiere que las técnicas de aprendizaje profundo, combinadas con el procesamiento de señales de audio, son prometedoras para aplicaciones de identificación de hablantes. Futuras investigaciones podrían explorar el uso de modelos más complejos, como redes neuronales recurrentes (RNN) para mejorar aún más el rendimiento [7] [8].

Referencias

[1] Brian McFee et al. "librosa: Audio and music analysis in Python". En: *Proceedings of the 14th Python in Science Conference*. 2015, págs. 18-25. DOI: 10.25080/Majora-7b98e3ed-003.

[2] Martín Abadi et al. "TensorFlow: Large-scale machine learning on heterogeneous distributed systems". En: *arXiv preprint arXiv:1603.04467* (2016).

[3] Gregory R. Lee et al. *PyWavelets: A Python package for wavelet analysis*. 2019. URL: https://pywavelets.readthedocs.io/.

[4] Activeloop. *Spoken MNIST Dataset*. 2023. URL: https://activeloop.ai/.

[5] Mike Boers. *PyAV: Pythonic bindings for FFmpeg*. 2023. URL: https://pyav.org/.

[6] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". En: *Journal of Machine Learning Research* 12.Oct (2011), págs. 2825-2830.

[7] Activeloop. *Deep Lake: Vector Database for AI*. 2023. URL: https://www.deeplake.ai/.

[8] Activeloop. *Hub: Fastest unstructured dataset management for TensorFlow/PyTorch*. 2023. URL: https://github.com/activeloopai/Hub.

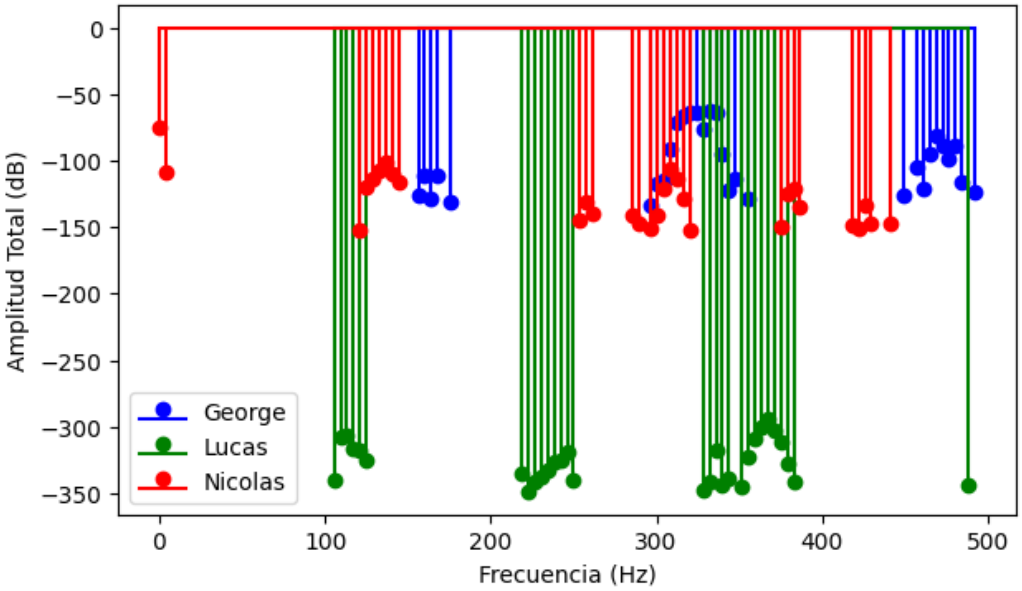

Figura 3: Frecuencias representativas para cada hablante.

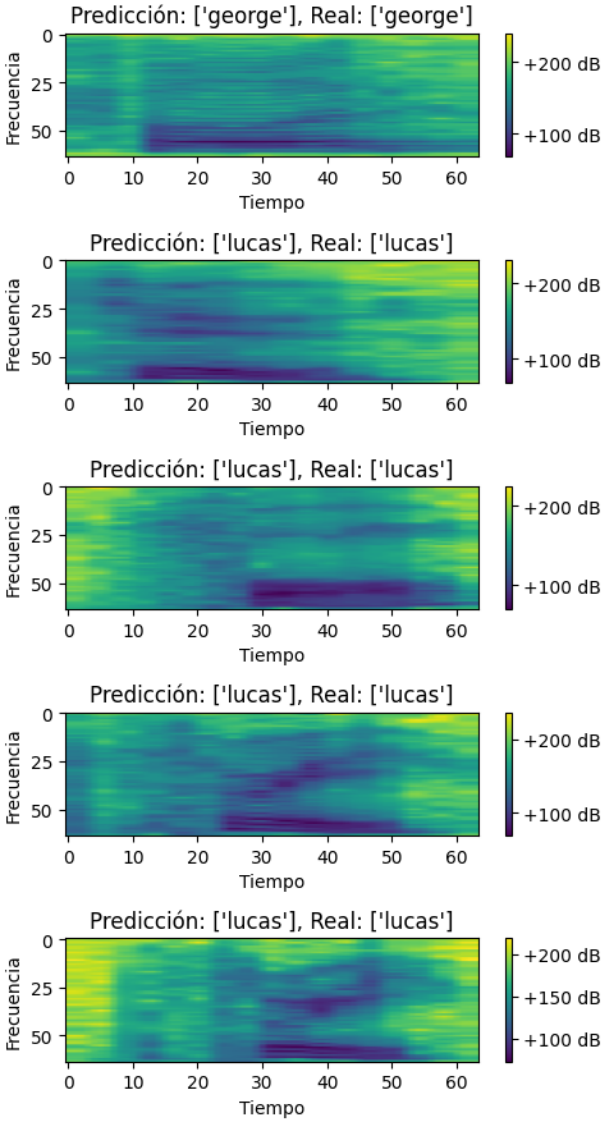

Figura 4: Comparación de predicción vs real.