Proyecto I - Redes neuronales para el reconocimiento de voz a partir de espectrogramas

Curso de Inteligencia Artificial
Escuela de Ingeniería en Computación
Instituto Tecnológico de Costa Rica

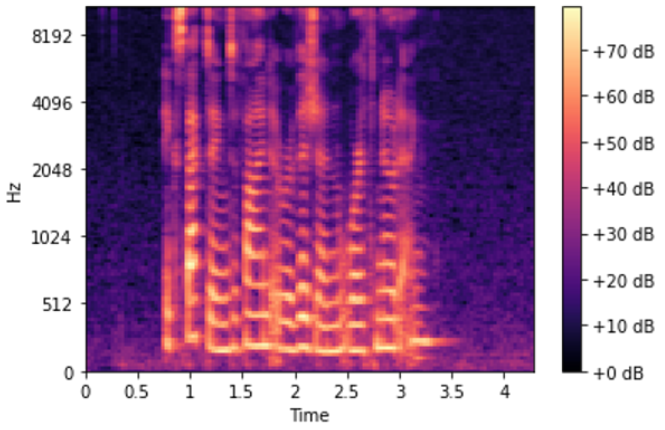

Fig. 1. Espectrograma de audio a modo de ejemplo

I. Objetivo

Este proyecto tiene como objetivo principal aplicar redes neuronales convolucionales (CNN) en la aplicación de clasificación multiclase basado en el procesamiento de señales en imágenes utilizando aprendizaje supervisado. Además de explorar herramientas para el desarrollo de modelos desde cero de redes neuronales, procesos de monitoreo de entrenamiento y bitácoras.

II. Descripcion del dataset

Para el desarrollo de este proyecto se utilizará un conjunto de datos público de clasificación de audio. Este dataset contiene grabaciones de corta duración con diferentes categorías sonoras, tales como sonidos ambientales, vocalizaciones de animales, instrumentos musicales o ruidos mecánicos [1]. Cada clip de audio se encuentra etiquetado con una clase, lo cual permite entrenar modelos supervisados para reconocer patrones acústicos, por lo tanto su objetivo es desarrollar modelos que permitan reconocer estos patrones para crear un modelo de clasificación multiclase.

Previo al entrenamiento, las grabaciones deben ser procesadas para obtener una representación en formato de imagen como se observa en la Figura 1. Estas representaciones servirán como entrada para los modelos convolucionales implementados [1]

[1]Enlace para descarga se encuentra disponible aquí

III. Diseño de arquitectura

Se deben construir manualmente dos modelos con **PyTorch**, sin utilizar librerías de alto nivel que abstraigan la definición de capas más allá de `torch.nn`. Es decir deben ser creadas por su equipo de trabajo.

Cada modelo debe incluir: definición de arquitectura (**y un respectivo diagrama**), función de pérdida, optimizador y rutinas de entrenamiento/validación. Además, se exige el control de aleatoriedad, registro de métricas y código reproducible.

Modelo A: LeNet-5 clásico

El **Modelo A** debe ser una variante clásica de *LeNet-5*, adaptada a clasificación de audio a partir de representaciones espectrales.

Se permite alterar el tamaño de entrada, el número de canales, y los hiperparámetros. Se deberá consultar al profesor cualquier otra modificación de la arquitectura original, y además debe de ser debidamente justificada en el informe.

Modelo B: Arquitectura alternativa basada en literatura

El **Modelo B** puede implementar cualquier arquitectura *distinta* basada en la literatura o un diseño propio **fundamentado académicamente**.

Entre las referencias sugeridas se encuentran *EfficientNet*, *MobileNet*, *Inception*, *DenseNet* o *ResNet*.

En caso de proponer un modelo propio, deberá explicarse claramente la motivación, las decisiones de diseño (profundidad, anchos de bloque, tipo de bloques, activaciones, etc) y cómo estas afectan la capacidad de generalización y la eficiencia computacional.

A. Preprocesamiento de los datos

Los datos son un conjunto de audios los cuales deben de ser transformados para poder utilizarse dentro de las redes neuronales (CNN) propuestas.

Además de esto, debe utilizar 2 conjuntos de datos diferentes

- El primer conjunto son los datos crudos luego de convertir los formatos de audio a imágenes (crudo).
- El segundo conjunto es igual que el primero, con la excepción de que debe de aplicar técnicas de data augmentation inspirada en audio [2]. **Debidamente justificada con literatura** (aumentado).

TABLE I
RÚBRICA DE PROYECTO

Criterio	Puntaje Máx.
Diseño y arquitectura definida Modelo A	5
Diseño y arquitectura definida Modelo B	15
Entrenamiento con datos crudos Modelo A	10
Entrenamiento con datos crudos Modelo B	10
Selección de aumentación	10
Entrenamiento con datos aumentados Modelo A	10
Entrenamiento con datos aumentados Modelo B	10
Presentación de métricas y selección del mejor modelo A (test)	10
Presentación de métricas y selección del mejor modelo B (test)	10
Comparación de modelos finales	10
Total	**100**

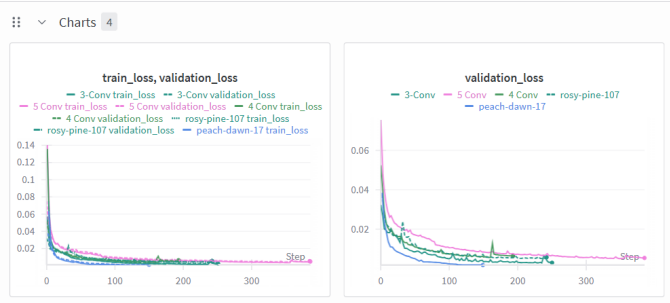

Fig. 2. Imagen de la plataforma interactiva de WandB

IV. ENTRENAMIENTO

Debe entrenar 4 distintos modelos base Modelo A/Base, Modelo A/Aumentado, Modelo B/Base, Modelo/Aumentado.

Para cada uno de ellos se deben de realizar al menos 5 entrenamientos con diferentes hiperparametros para seleccionar el mejor de cada combinación. Muestre los resultados de los entrenamientos y seleccione los mejores modelos y luego compare los mejores modelos base.

Recuerde que para cada dataset debe dividirlo en 3 partes (entrenamiento, validación y testing), durante el entrenamiento de su modelo debe verificar que no tenga problemas de overfitting o underfitting sus modelos, al igual que justificarlos por escrito en la investigación.

Recomendaciones: En caso de no contar con GPU en una unidad local, pueden utilizar Google Colab para realizar el proyecto, en este caso debe subir el dataset dentro de Google Drive para poder accederlo guía.

V. EVALUACIÓN DE MODELOS

Debe comparar los resultados de los modelos para indicar cúal es el mejor, para realizar esta comparación debe utilizar la herramienta de Weights and Biases, esta herramienta tracea y visualiza varios aspectos del proceso de entrenamiento del modelo en tiempo real como el overfitting, loss, accuracy, F1-Score, matriz de confusión y otras métricas de evaluación como se observa en la Figura 2.

VI. ENTREGA

El informe deberá realizarse en LaTeX (Overleaf) utilizando la plantilla IEEE para artículos científicos. El documento debe contener visualizaciones, y la interpretación de los resultados. Además, se debe adjuntar un **Jupyter Notebook** con el código implementado. La entrega final consistirá en un archivo comprimido (`.zip`) que contenga:

- Código fuente en LaTeX.
- El PDF del informe.
- El Notebook con el código fuente.

RÚBRICA

Si el trabajo no se encuentra debidamente ordenado y presentado siguiendo una adecuada estructura para el informe, puede ser considerado como incompleto y cualquiera de las rúbricas se puede ver afectada ver Tabla I. Esto quiere decir que se evaluará el contenido del informe y se verificará contra los notebooks.

REFERENCES

[1] K. J. Piczak, "ESC: Dataset for Environmental Sound Classification," in *Proceedings of the 23rd ACM international conference on Multimedia*. Brisbane Australia: ACM, Oct. 2015, pp. 1015–1018. [Online]. Available: https://dl.acm.org/doi/10.1145/2733373.2806390

[2] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.