

FORMATO DE PROPUESTA DE PROYECTO
Clasificación de imágenes con redes neuronales

Semestre: _____ 2025-1 _____

Fecha de entrega: ____ 16/10/2024 _____

Nombre de los Integrantes:

Integrante 1: García Sánchez Daniel Alfredo

Integrante 2: Guerrero López Jesús Antonio

1. Título del Proyecto.

Clasificación de Diagnósticos Radiológicos de Tórax

2. Objetivo

Las radiografías son uno de los estudios más comunes realizados a pacientes debido a la facilidad y el bajo costo, el *dataset* de CXR-8 contiene más de 120,000 imágenes de radiografías de tórax de más de 30,800 pacientes únicos obtenidas del *National Institutes of Health Clinical Center*.

Objetivos:

- 1.- Crear un *pipeline* de datos derivado en la base de datos obtenida de: <https://paperswithcode.com/dataset/chestx-ray8>
- 2.- Crear un clasificador multi etiqueta utilizando redes neuronales convolucionales.
- 3.- Crear una API que permita consumir el modelo entrenado en el paso anterior para el apoyo de diagnósticos.

3. Herramientas para la implementación del proyecto

Este proyecto se realizará en Python y las principales librerías y herramientas en este contexto serán:

1. Pandas
2. Numpy
3. Matplotlib
4. OpenCV
5. Scikitlearn
6. Tensor Flow
7. Keras

4. Orígenes de datos.

Los datos se obtuvieron de la página <https://paperswithcode.com/dataset/chestx-ray14> quienes los obtuvieron del *National Institutes of Health Clinical Center*

5. Principales actividades a realizar.

Considerando que la base de datos inicial es enorme en cantidad de imágenes y que verificamos que no tenemos datos que agreguen ruido al modelo, la planificación del proyecto es la siguiente:

1. Partición de cada conjunto de la base de datos (Entrenamiento, Validación, Prueba).
2. Análisis y Visualización de cada Conjunto del modelo.
3. Procesamiento de Imágenes (Posiblemente realizaremos un Downsampling).
4. Iniciaremos la implementación del modelo con alguna de las librerías descritas y analizaremos el performance del modelo contra la magnitud del Downsampling u otras modificaciones para encontrar los parámetros óptimos.
5. Una vez terminado un modelo con buen porcentaje de precisión y verificado que no tengamos errores de

sobreajuste o del mismo tipo buscaremos emplear esos parámetros sobre la mayor cantidad de datos del dataset original que nos permita entregar el trabajo en tiempo y forma.

6. Terminado el modelo empezaremos el desarrollo del API para hacer uso del modelo entrenado.

6. Observaciones y comentarios generales del Alumno.

1.- El costo de preprocesar 120,000 imágenes y luego entrenar el modelo con esa cantidad es demasiado elevado para llevarse a cabo con un solo PC, de hecho, los autores del paper que entrenaron su modelo, utilizaron varios GPUs que les donó Nvidia.

7. Observaciones y resultado de la revisión del profesor.