

# Guía para el Reentrenamiento de Modelos

---

Esta guía paso a paso te ayudará a reentrenar modelos YOLOv5 y YOLOv8 utilizando tus propios datos. Asegúrate de seguir cada paso cuidadosamente para asegurar un reentrenamiento exitoso.

## Preparación Inicial

---

### Activación del Entorno Virtual

Antes de empezar con la preparación de datos y el reentrenamiento, es importante activar tu entorno virtual donde están todas las dependencias necesarias instaladas.

Para activar el entorno virtual, utiliza el siguiente comando dependiendo de tu sistema operativo:

- **Windows:** `.\venv\Scripts\activate`
- **Linux o macOS:** `source venv/bin/activate`

Este comando debe ejecutarse en la cmd parado en la carpeta `Re-training`. Una vez activado el entorno virtual, puedes proceder con los siguientes pasos.

## Preparación de Datos

---

### 1. Organización de Imágenes y Etiquetas:

- Coloca todas las imágenes que deseas utilizar para el reentrenamiento en la carpeta `H:\datasets\allData\images`.
- Coloca todas las etiquetas correspondientes en la carpeta `H:\datasets\allData\labels`.
- **IMPORTANTE:** Asegurate que las carpetas de `re-val`, `re-train` y `re-test` tengan sus respectivas carpetas `images` y `labels` vacías para no re-entrenar con imágenes ya usadas para entrenar y así no generar overfitting.

### 2. División de Datos:

- Ejecuta el script `splitData.py` para dividir los datos en train, val y test. Este script distribuirá automáticamente las imágenes y sus etiquetas correspondientes en subcarpetas adecuadas, con una distribución de:

`train: 0.7, val: 0.15, test: 0.15`

- Si quieres cambiar los porcentajes lo puedes hacer en la línea 11, preocúpate que la suma de estos sea igual a 1
- La línea que habría que cambiar es la siguiente:

```
def split_data(base_dir, data_path, train_size=0.7, val_size=0.15, test_size=0.15):
```

- 

## Pasos para Ejecutar `splitData.py`

- a. Abre una terminal o CMD en la ubicación donde hayas guardado el script `splitData.py`.
- b. Ejecuta el script con Python:

```
python splitData.py
```

- c. El script dividirá los datos en los conjuntos de entrenamiento, validación y prueba, y los moverá a las carpetas adecuadas dentro de `H:\datasets`.

## Reentrenamiento de Modelos YOLO

---

### YOLOv5

#### 1. Preparación:

- Actualiza el script `yolo5Retrain.py` con las rutas correctas según tu configuración:
  - `pathModel` : En esta variable deberás actualizar con el path de modelo que se quiere reentrenar

#### 2. Reentrenamiento:

- Desde una terminal, navega hasta la ubicación de tu script `yolo5Retrain.py`.
- Ejecuta el script con Python:

```
python yaml5Retrain.py
```

- El script realizará el reentrenamiento del modelo YOLOv5 con tus datos.

### YOLOv8

## 1. Preparación:

- Actualiza el script `yolo8Retrain.py` con las rutas correctas según tu configuración:
  - `pathModel` : En esta variable deberas actualizar con el path de modelo que se quiere reentrnar

## 2. Reentrenamiento:

- Navega hasta la ubicación de tu script `yolo8Retrain.py` en una terminal.
- Ejecuta el script con Python:

```
python yolo8Retrain.py
```

- El script comenzará el proceso de reentrenamiento para YOLOv8 con tus datos.

## Notas Finales

---

- Asegurate siempre de estar en el entorno virtual para correr los programas ya que acá estan las dependencias instaladas.
- Asegúrate de tener instaladas todas las dependencias requeridas para YOLOv5 y YOLOv8, incluidas las versiones específicas de PyTorch.
- Monitorea la salida en la terminal para detectar cualquier error y ajusta tu configuración según sea necesario.