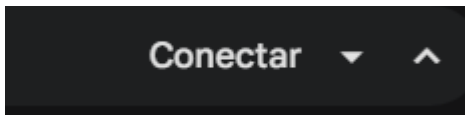


GUIA DE USO DE LA CLASE ENTRENAMIENTO EN GOOGLE COLLAB

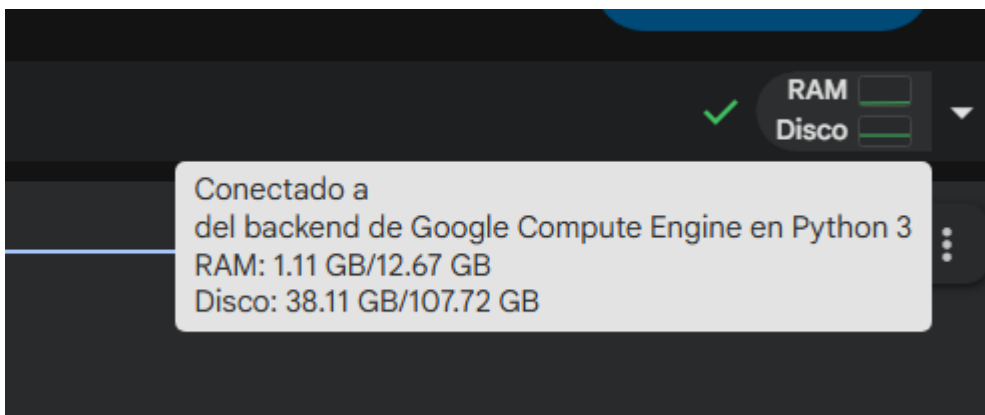
Para la creación de este modelo se uso el google collab para usar su funcion de ejecucion por medio de servidores remotos

1.- Hace log in y posteriormiente vamos a la pestaña de entorno y lo cambiamos al v5e-1 tpu:

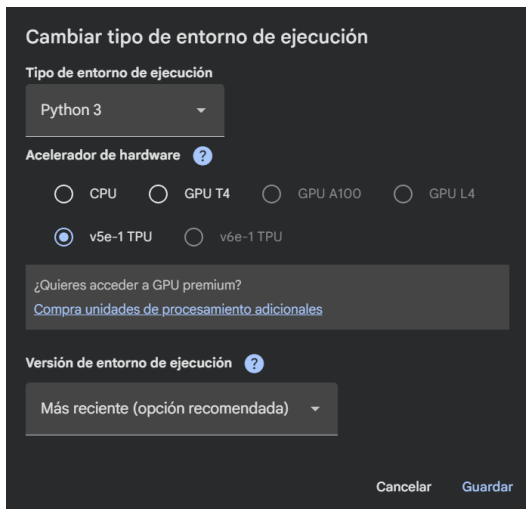
Conecta con los entornos



Das click en Ram y disco y busca “cambiar entorno”



Selecciona v5e-1 TPU para su ejecución



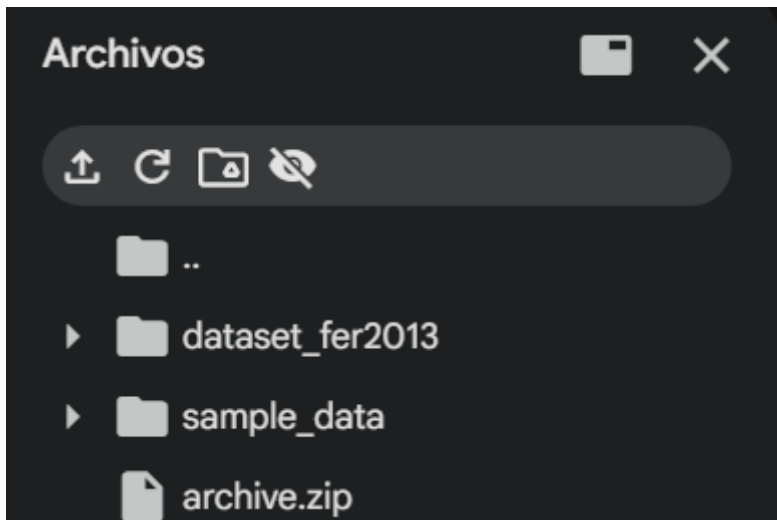
2.- Dentro de la notebook se introduce el código Entrenador.py del archivo que es este en cuestión

3.- En caso de ser necesario descomentar `!pip install tensorflow` para que se ejecute en el entorno y descargue la librería

y se vería así:

```
# =====
# ENTRENADOR DE EMOCIONES (4 CLASES)
# Dataset: FER-2013 (Kaggle)
# Emociones: sad, happy, angry, neutral
# =====|
!pip install tensorflow
import os
import zipfile
import numpy as np
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D, Dropout, Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau
import matplotlib.pyplot as plt
import shutil
```

4.- Se pone el nombre del dataset, en este caso al descargar el FER-2013 se nombra como "archive.zip" ajuste los epochs a lo que se desee:



```
# --- CONFIGURACIÓN ---
NOMBRE_DEL_ZIP = "archive.zip"
IMG_SIZE = 48 # FER-2013 usa 48x48
BATCH_SIZE = 32
EPOCHS = 40

# MAPEO FER-2013 A 4 CLASES
# FER-2013 tiene: angry(0), disgust(1), fear(2), happy(3), sad(4), surprise(5), neutral(6)
FER_TO_4CLASS = {
    'angry': 'angry',
    'disgust': 'angry', # Disgust se parece a angry
    'fear': 'sad',      # Fear se parece a sad
    'happy': 'happy',
    'sad': 'sad',
    'surprise': 'happy',
    'neutral': 'neutral'
}
```

Con estos ajustes el entrenador debería de ir de maravilla, ya si desea cambiar el entorno a uno mas potente es de su elección

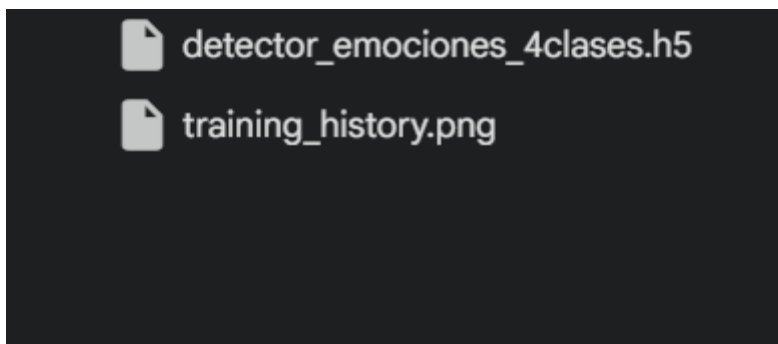
5.- Por ultimo se da click en ejecutar la notebook (el botón de play) y estos serian lo que se debería de ver en la ejecución (ejecución de ejemplo con 25 epochs):

```
Total params: 2,619,332 (9.99 MB)
Trainable params: 1,887,748 (7.20 MB)
Non-trainable params: 731,584 (2.79 MB)

🔧 Iniciando entrenamiento (25 épocas)...
⌚ Esto puede tomar varios minutos...

Epoch 1/25
718/718 ██████████ 57s 71ms/step - accuracy: 0.3003 - loss: 1.6343 - val_accuracy: 0.3181 - val_loss: 1.4135 - lear
Epoch 2/25
718/718 ██████████ 50s 69ms/step - accuracy: 0.3888 - loss: 1.3116 - val_accuracy: 0.3656 - val_loss: 1.3437 - lear
Epoch 3/25
718/718 ██████████ 50s 69ms/step - accuracy: 0.4408 - loss: 1.2537 - val_accuracy: 0.3613 - val_loss: 1.3900 - lear
Epoch 4/25
718/718 ██████████ 50s 70ms/step - accuracy: 0.4649 - loss: 1.2160 - val_accuracy: 0.4079 - val_loss: 1.2795 - lear
Epoch 5/25
718/718 ██████████ 50s 70ms/step - accuracy: 0.4747 - loss: 1.2015 - val_accuracy: 0.4137 - val_loss: 1.2665 - lear
Epoch 6/25
233/718 ██████████ 29s 61ms/step - accuracy: 0.4885 - loss: 1.1783
```

Una vez termine se vera así y se cargara un archivo de modelo tipo .h5 como se ve aquí

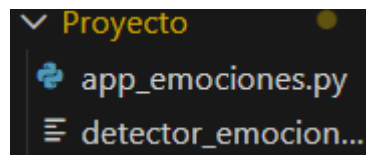


Y listo ya tiene su modelo de uso tipo .h5 para poder usar con el archivo “App_emociones.py” solo tiene que importar el modelo a la carpeta, especificar la ruta de la carpeta donde se ejecutan ambas como en las imágenes de abajo y listo

Ruta de la carpeta de ejecución

```
# CONFIGURAR RUTA DEL MODELO
# Opción 1: Modelo en carpeta específica
RUTA_CARPETA = r"C:\Users\torit\OneDrive\Escritorio\escuela\IA con Zuriel\Modulo4\Proyecto"
```

Como se debería de ver



Ya podría usar el detector de emociones por medio de la cámara digital como se mira en el video adjunto en el archivo “Video presentación”