**TECNM**

**Tecnológico Nacional de México**

**Campus Culiacán**

**Ingeniería en Sistemas computacionales**

**Inteligencia artificial**

**09:00 – 10:00**

**Tarea 1 Unidad 4 – Pre-procesado de imágenes**

**Integrantes:**

**Caro García Jorge Ariel**

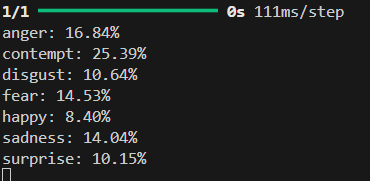
**Galván González Sebastián**

**Docente:**

**ZURIEL DATHAN MORA FELIX**

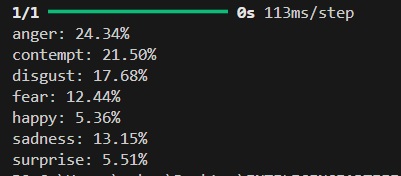
**24/05/2025**

Desprecio



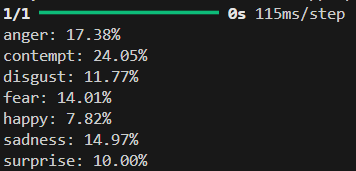


Enojo



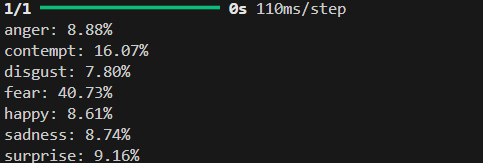


Disgusto



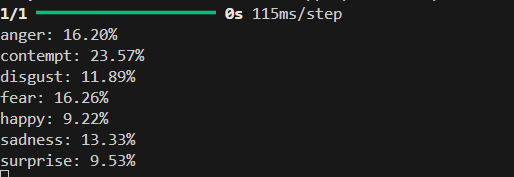


Miedo



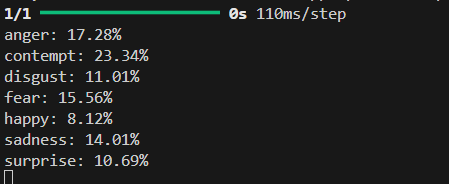


Felicidad



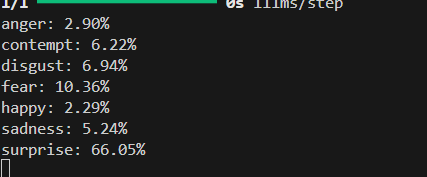


Tristeza



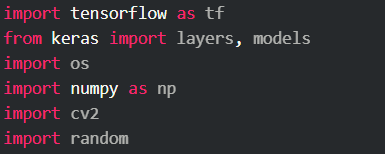


Sorpresa



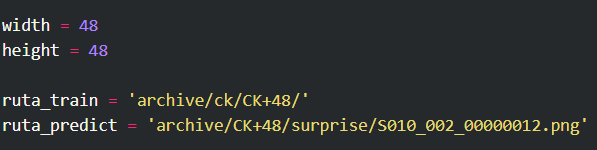


**Librerías**



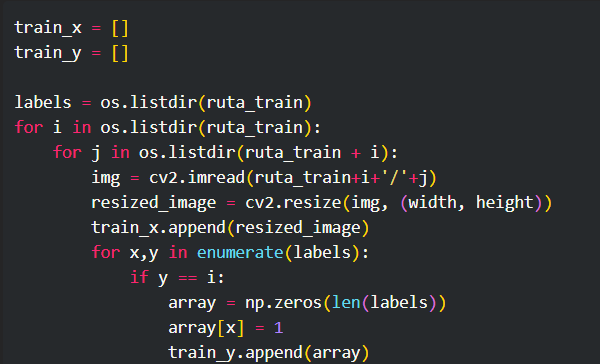
* **TensorFlow/Keras:** Para construir y entrenar la red neuronal.
* **os:** Para manejar archivos y carpetas.
* **numpy:** Para manipulación de arrays.
* **cv2:** OpenCV, para leer y procesar imágenes.

**Definición de parámetros y rutas**



* width, height: Tamaño al que se redimensionarán las imágenes.
* ruta\_train: Carpeta donde están las imágenes de entrenamiento, organizadas por carpetas de emociones.
* ruta\_predict: Ruta de una imagen para predecir su emoción.

**Preparando datos de entrenamiento**



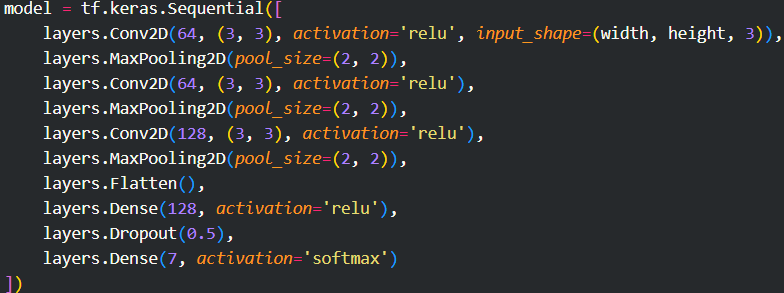
* **labels:** Lista de nombres de las carpetas (cada una representa una emoción).
* **Bucle:** Recorre cada carpeta y cada imagen dentro de ella.
  + Lee la imagen y la redimensiona.
  + La agrega a [train\_x](vscode-file://vscode-app/c:/Users/jacar/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ").
  + Crea un vector one-hot para la etiqueta correspondiente y lo agrega a [train\_y](vscode-file://vscode-app/c:/Users/jacar/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html).

**Conversión a arrays de numpy**



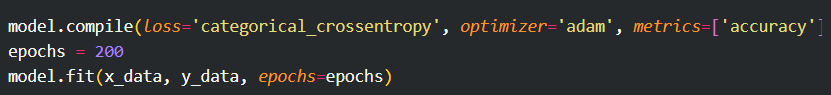
* Convierte las listas de imágenes y etiquetas a arrays de numpy, que son el formato que espera Keras.

**Definición del modelo CNN**

****

* Modelo secuencial: Tres capas convolucionales con max pooling, seguidas de una capa densa y una de salida softmax (para clasificación multiclase).
* Dropout: Ayuda a evitar el sobreajuste.
* Dense(7): Asume que hay 7 emociones/clases.

**Compilación y entrenamiento**

****

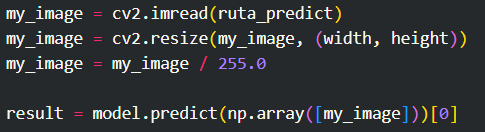
* Compilación: Usa categorical\_crossentropy para clasificación multiclase y el optimizador Adam.
* Entrenamiento: Ajusta el modelo a los datos durante 200 épocas.

**Guardar y cargar modelo**



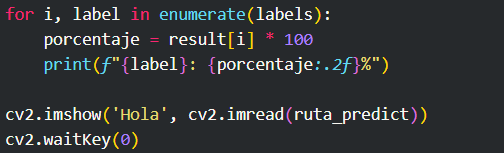
* Guarda el modelo entrenado y luego lo vuelve a cargar (para usar el modelo sin volver a entrenar).

**Predicción sobre una imagen nueva**



* Lee y redimensiona la imagen a predecir.
* Normaliza los valores de los píxeles a [0,1].
* Predice la emoción usando el modelo.

**Resultados**



* Muestra el porcentaje de probabilidad para cada emoción.
* Muestra la imagen predicha en una ventana.

**Dataset utilizado:**

<https://www.kaggle.com/datasets/shawon10/ckplus/data>