

Reconocimiento de Pokémon

Inteligencia Artificial

May 1, 2025

Objetivo del Proyecto

El objetivo de este proyecto fue desarrollar un sistema que reconozca Pokémon en imágenes usando técnicas de deep learning, en particular con una red neuronal convolucional. La solución debía ser capaz de entrenarse con datos descargados automáticamente y funcionar en tiempo real usando una cámara.

Arquitectura del Modelo

El modelo utilizado fue ResNet18, una red convolucional residual preentrenada en ImageNet. Se sustituyó su capa final para ajustarse al número de clases (Pokémon). Se emplearon tres técnicas fundamentales:

- **Transfer learning:** se reutilizan pesos preentrenados.
- **Fine-tuning:** se ajusta únicamente la capa final.
- **Augmentación:** se aplican transformaciones a las imágenes como rotaciones, brillo, volteo.

Esto permitió obtener buenos resultados con pocos datos y bajo poder computacional.

La implementación fue hecha en Python usando:

- **PyTorch:** para el modelo y entrenamiento.
- **OpenCV:** para capturar imágenes en tiempo real.
- **PokeAPI:** para descargar datos e imágenes de Pokémon.

Se desarrolló un sistema que permite entrenar el modelo, evaluarlo y predecir desde la cámara, todo desde un mismo archivo. También se guarda un mapeo de clases y se almacenan los modelos entrenados.

Partición del Dataset

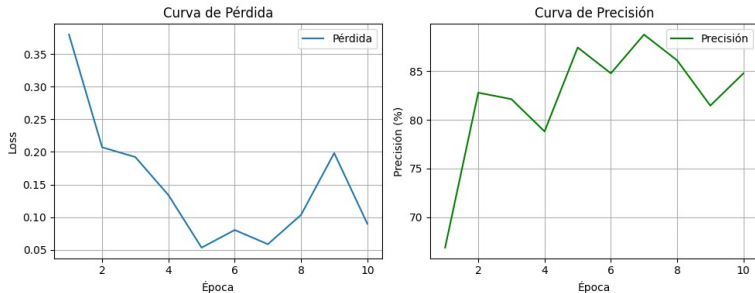
Para asegurar una buena evaluación del modelo, se dividió el dataset de imágenes en:

- 80% para entrenamiento.
- 20% para prueba.

Cada conjunto se guarda en un archivo CSV independiente: `train.csv` y `test.csv`. Esto ayuda a evitar sobreajuste y permite evaluar la capacidad de generalización del modelo en datos no vistos.

Curvas de Entrenamiento

A continuación se muestran las curvas de pérdida y precisión por época. Estas gráficas permiten monitorear el aprendizaje del modelo, detectar si hay sobreajuste (cuando mejora solo en entrenamiento) o subajuste (cuando no mejora en absoluto).



Resultados Obtenidos

El modelo alcanzó una precisión cercana al 85% en promedio en el conjunto de prueba. La detección en tiempo real funcionó correctamente con imágenes capturadas por cámara. A pesar del tamaño reducido del dataset, el sistema logró generalizar gracias al uso de técnicas como transfer learning y augmentación de datos.