Who’s That Pokémon?

Andrew Sokol
CS4743
11/25/25

Introduction

The goal of this project was to build a machine learning model capable of identifying a Pokémon from an input image.

Image classification is widely used in:

- Automated identification systems
- Content tagging and recommendation
- Real-time mobile and AR applications

Pokémon provides a fun but challenging benchmark because:

- Classes vary widely in shape and color
- Many Pokémon look visually similar
- Dataset quality varies by source

Data & Preprocessing

Dataset:

https://www.kaggle.com/datasets/noodulz/pokemon-dataset-1000

Total Classes: 1000

All images are stored in a single folder automatically split using validation splitting.

Data Characteristics:

Colored images, no background noise.

Older classes also include older sprites which are lower resolution. May confuse the model.

Data & Preprocessing

Image Preprocessing:

- Images have been resized to 160 x 160 pixels
- Normalized pixel values (TensorFlow)
- One-Hot encoding class labels
- Batched data using the TensorFlow image_dataset_from_directory API
- Prefetched data for GPU efficiency

Data Augmentation:

- Random horizontal flip
- Random rotation
- Random zoom

Methodology - Model Architecture

Used MobileNetV2 as the base CNN because:

- It is pre-trained on ImageNet
- Lightweight and fast (optimized for mobile)
- Performs very well even with smaller datasets
- Helps prevent overfitting

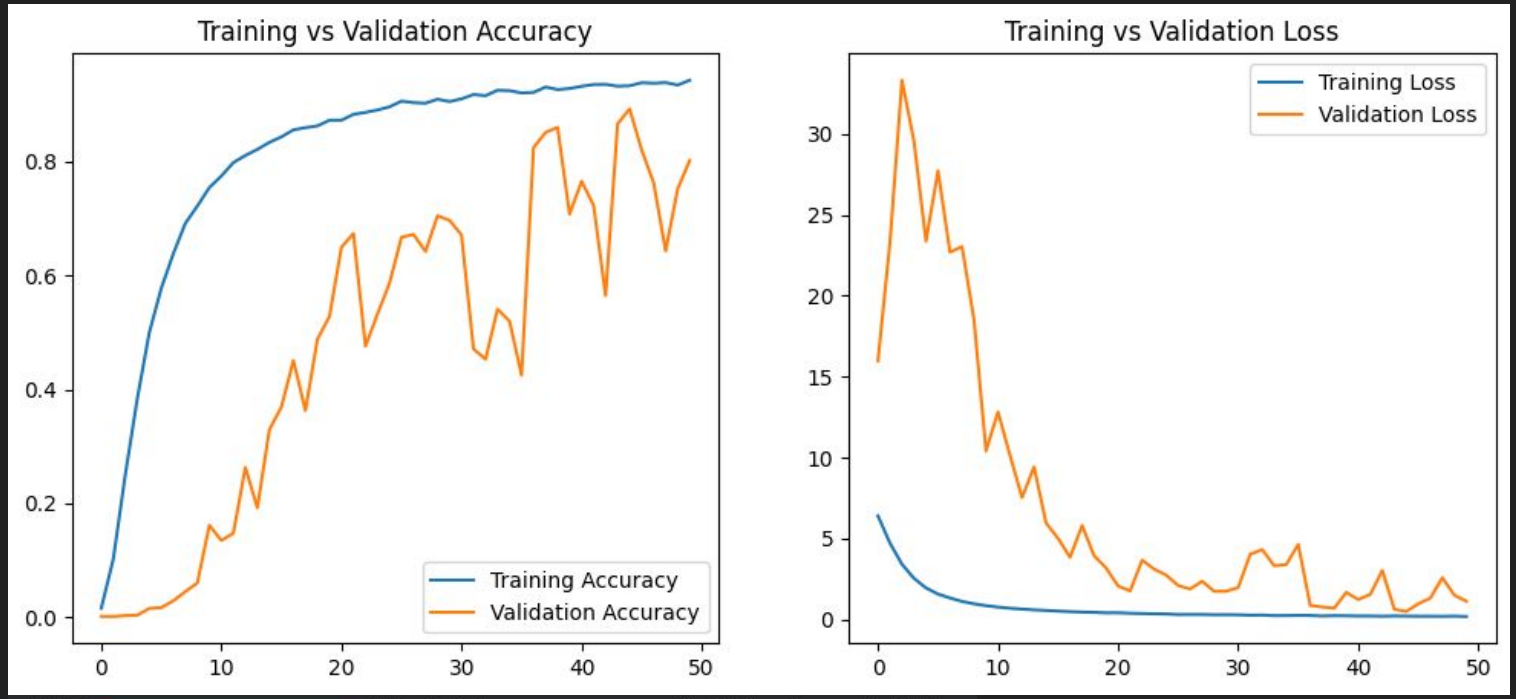
Network Structure:

- MobileNetV2 backbone
- Global average pooling layer
- Dense(128, ReLU)
- Dropout(0.3)
- Final Dense layer with softmax activation

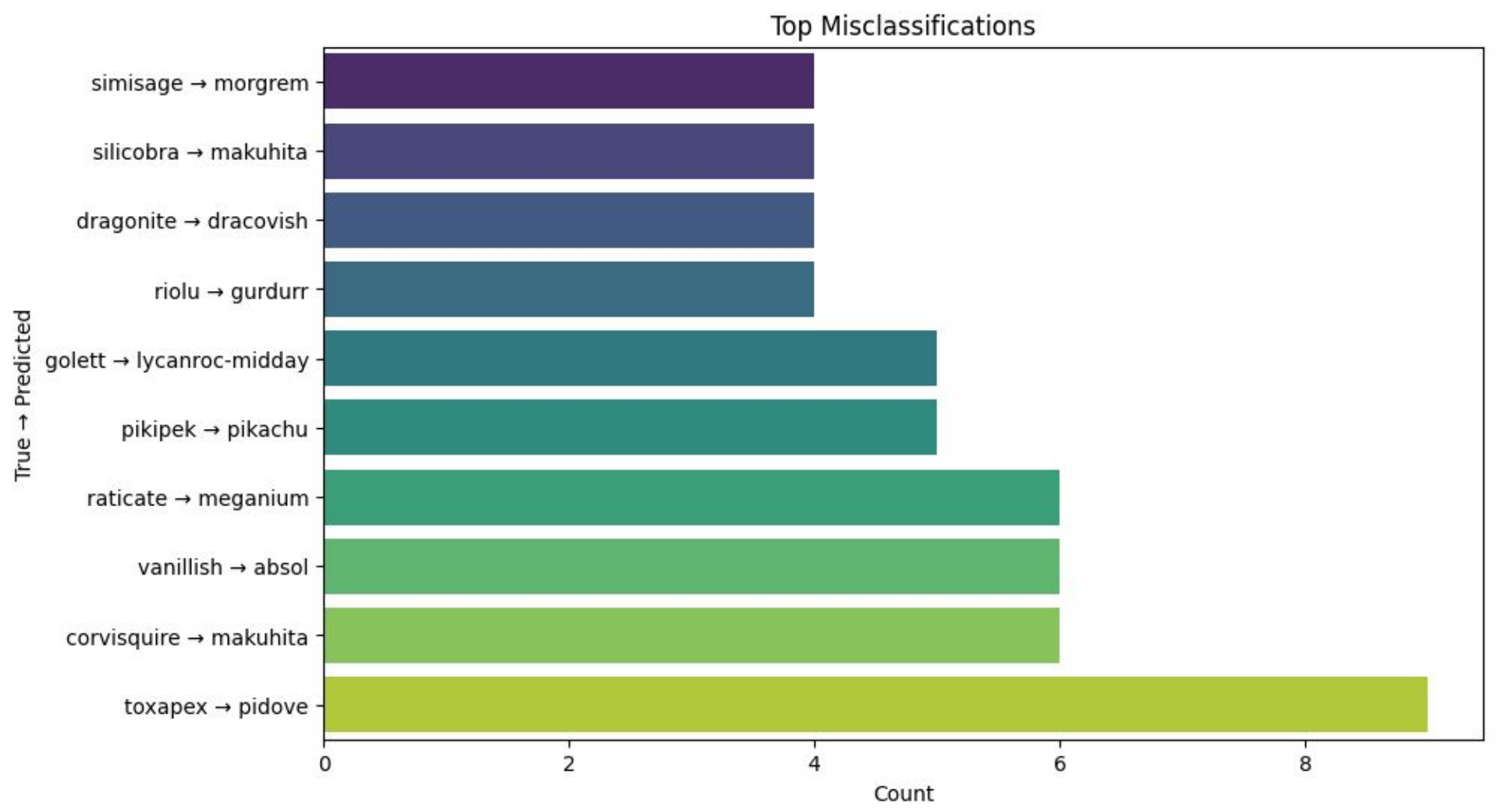
Methodology - Training Setup

Training Setup:

- Optimizer: Adam
- Loss: Categorical Crossentropy
- Batch size: 64
- Epochs: 50
- Validation split: 30%




```
51/51 ━━━━━━━━━━━━━━━━━━━━ 215s 4s/step - accuracy: 0.8603 - loss: 0.7430
Test Accuracy: 0.8511886596679688
Accuracy: 0.8511886384686632
F1 Macro: 0.8267142677412028
F1 Weighted: 0.8407911207891517
F1 Micro: 0.8511886384686632
```


Top Misclassifications
simisage → morgrem
silicobra → makuhita
dragonite → dracovish
riolu → gurdurr
golett → lycanroc-midday
pikipek → pikachu
raticate → meganium
vanillish → absol
corvisquire → makuhita
toxapex → pidove
True → Predicted
0
2
4
6
8
Count

Discussion

The goal was to build a Pokémon image classifier, and the model reached about 86% accuracy, which is strong given the large number of classes and visual similarity between many Pokémon.

Weaknesses include class imbalance and strange repeated discrepancies.

Future Work

- Add more images for low-count classes from other sources
- Increase input size to 224×224
- Remove shinies?
- Fix validation, reduce overfitting