

Bird Species Image Classification

CSE 6242 - Team 133



Motivation / Introduction

What is the problem:

Many existing bird species image identification applications are limited by factors such as bird species geography or being available exclusively on mobile app stores. The purpose of our team's project is to provide a website through which users can upload a bird image to determine the likely bird species of the image provided.

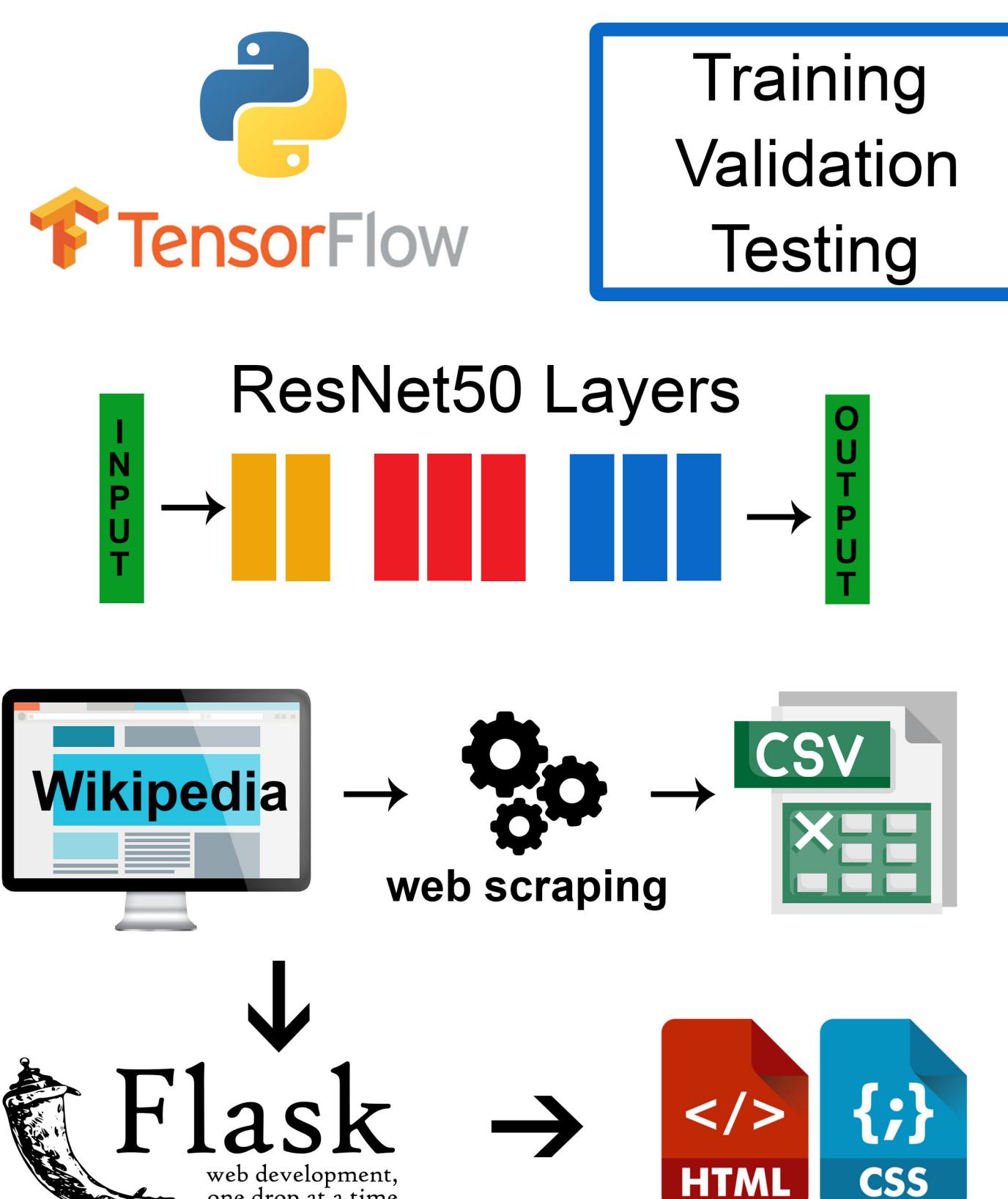
Why is it important:

- Free to use and "plug and play"
- Useful for bird photography enthusiasts/hobbyists, and researchers
- Practical for academia, non-profit/conservation organizations

Data

Kaggle Dataset - 2GB

76,262 train images of 450 bird species
2250 test images (5 images/species)
2250 validation images (5 images/species)
Images are 224 X 224 X 3 color images
in jpg format



Our Approach

What are they:

- Back-end: CNN model that takes a bird image and identifies the species.
- Front-end: User-friendly, interactive web page that shows identified bird species, brief information about the bird, and the confidence level.

How do they work:

- Pre-trained deep-learning ResNet50 model to classify bird species.
- Imported the ResNet50 base model, and built upon it with Tensorflow 2's Keras API.
- Scrapped Wikipedia for short summary of bird species and a path for example photos.
- Deployed the model with Python Flask environment.
- Loaded the trained tensorflow model and called HTML scripts

Why does it solve the problem:

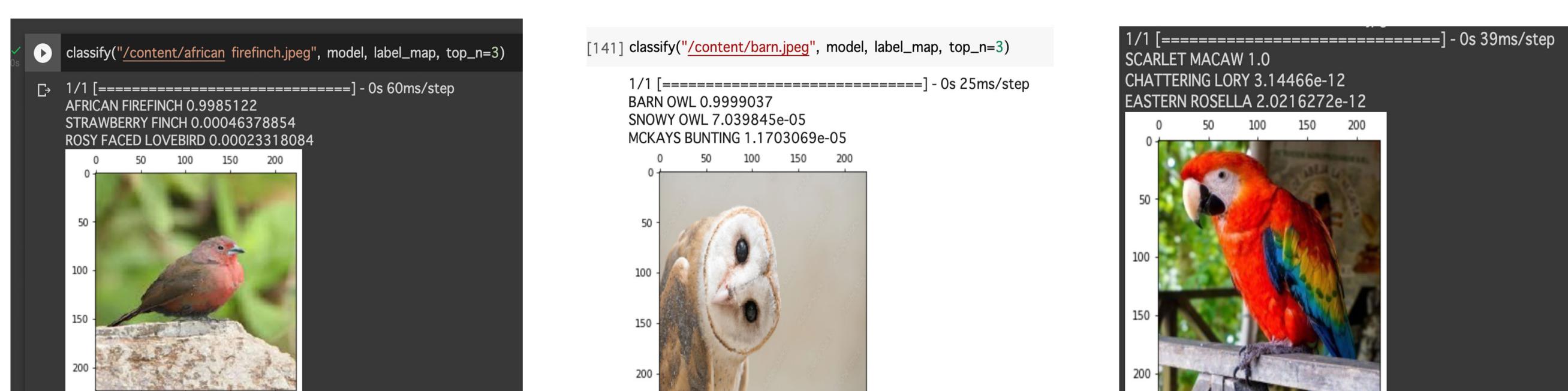
- Our model has higher accuracy
- Intuitive and user-friendly web interface
- No regional dataset download required

What is new:

- Utilizing transfer learning from a pre-trained base model
- Introducing noise into the training data to increase model generalization.
- Utilizing Google Colab's "GPU hardware accelerator" option to increase speed.

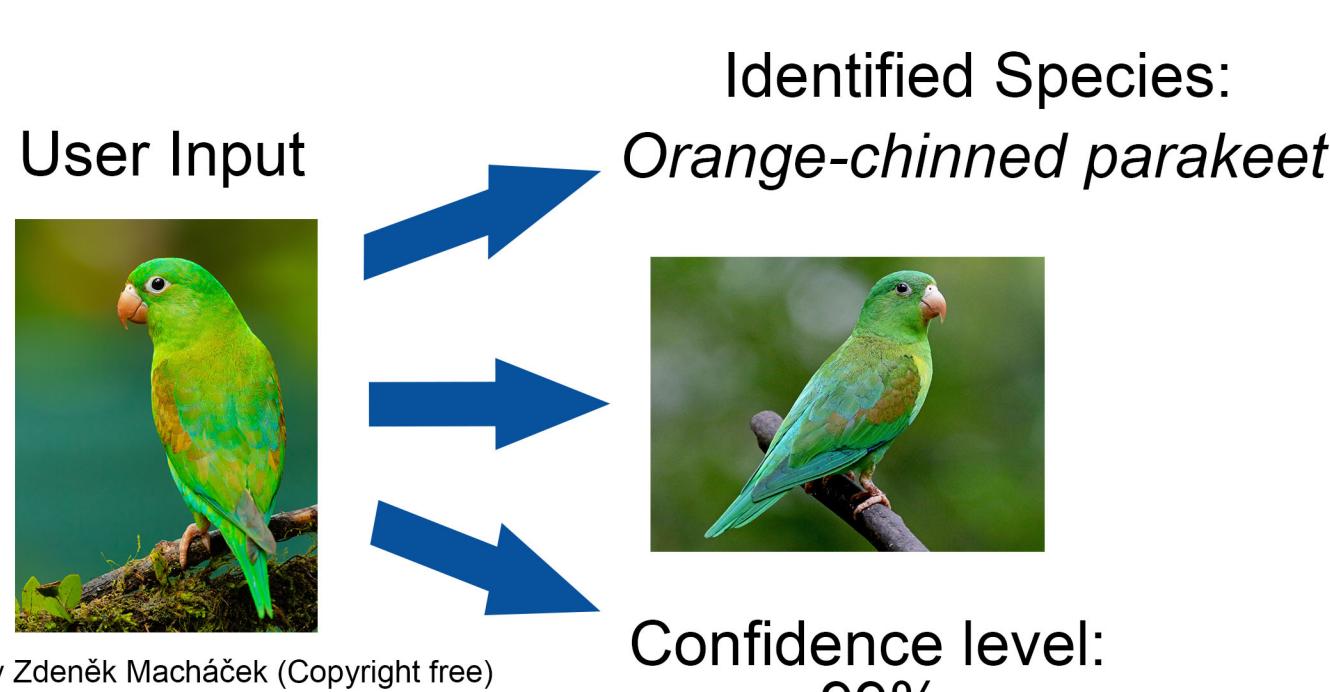
Experiments and Results

Evaluated its performance on novel bird images



Result:

- Achieved 96.8% accuracy on an unseen test set of images.
- The entire training process took 13 epochs (iterations).
- Total training time of about 3 hours.



How does it compare to other methods:

Our method is unique in its high accuracy prediction and in its interactive user interface that displays the top three predictions along with the description, confidence level, and a Wikipedia link for more information.

Final Visualization: