Logo or No-go?

3263748

Software Technology 1 (4483) Tuesday 9:30am

Capstone Project

Introduction

- The problem: How do you tell if a logo is genuine or counterfeit?
- Is it possible to figure out the probability of a logo being genuine or counterfeit, instead of a simple true/false classification?
- Can the results of this check be packaged into a web interface for ease of use?

The Dataset

- As part of this project, the unit convenor assigned me a dataset consisting of corporate logos, with one class of genuine logos and one class of counterfeit logos.
- The dataset primarily consists of JPEG format images, 70 pixels wide and high.
- The dataset also contains a CSV file, with each row consisting of a path to a given logo file, an associated brand name for the logo, and a classification.

Note: "Counterfeit" Logos

- Upon inspection, the counterfeit logos, or "fake" logos as termed in the dataset, appear to be the genuine logos rotated by 45 degrees clockwise.
- I suspect this is because distributing counterfeit logos would violate copyright and/or trademark law.
- Regardless, the fact that there are two classes of images still presents an image classification problem with educational value.
  - As evidenced by the fact that the dataset was assigned to me by the unit convenor.

Exploratory Data Analysis

- Initial analysis of the image data alone, through data visualisation techniques, didn't reveal any features that could be used for classification on their own.
- Techniques used on the images included changing the input colour space (from RGB to greyscale), perspective transformation (flipping, etc), and applying artificial noise.
- Analysis of the .csv file present in the root of the dataset showed that a binary classification feature exists, meaning that this problem could be solved with machine learning image classification.

```
import pandas as pd
dataframe = pd.read_csv(base_path + "/file_mappin
dataframe.head()
```

	Filename	Brand Name	Label
0	output\Cowbell\000001.jpg	Cowbell	Genuine

Exploratory Data Analysis

- This binary classification feature is the "Label" column.
- As shown to the bottom right, there are two possible values, Genuine or Fake.
- As machine learning image classifiers rely on mappings of images to labels, the data is almost ready to train a model.

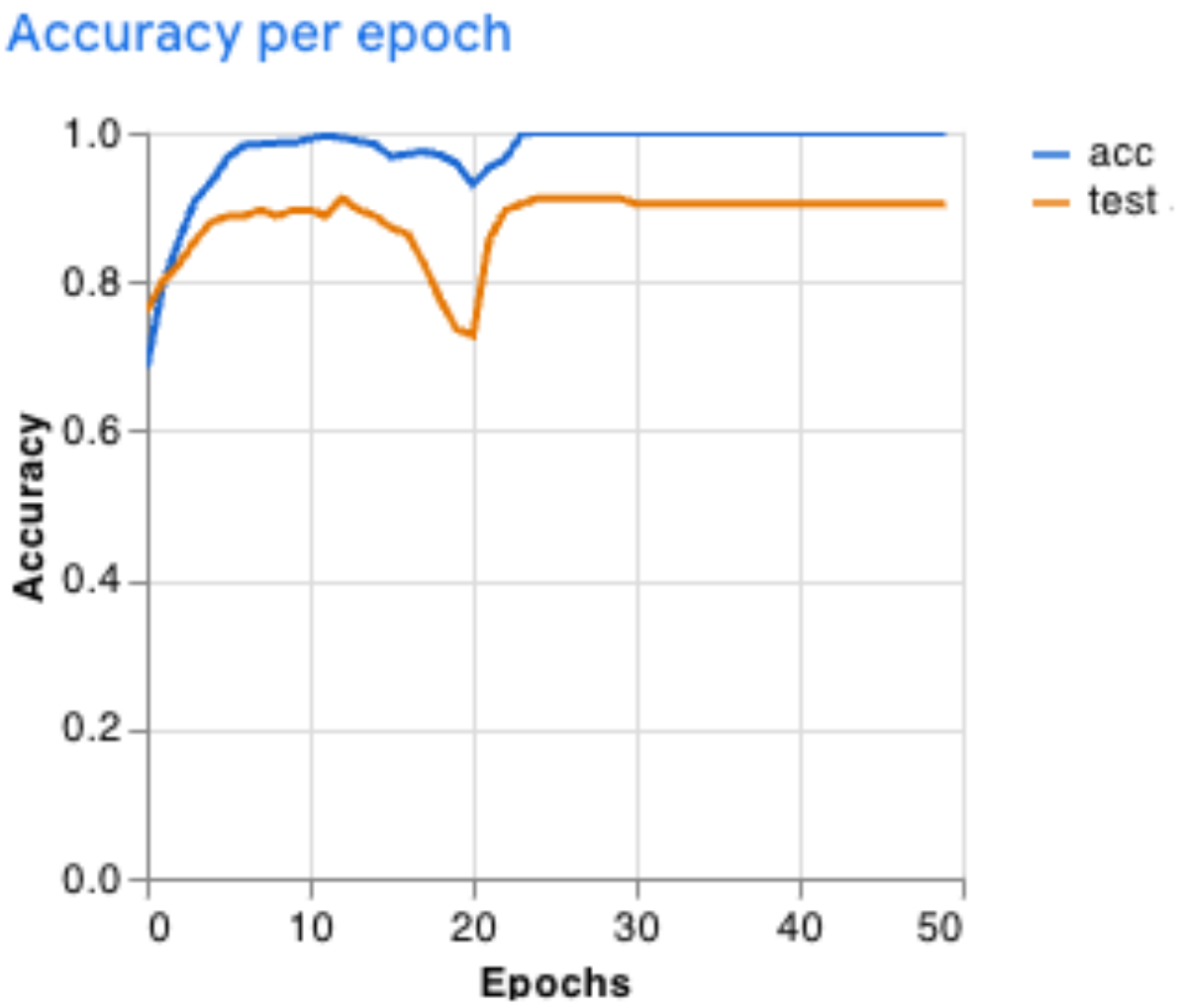
Label
Genuine
Genuine
Fake
Fake
Fake

Predictive Data Analysis

A machine learning model was trained using Teachable Machine's image classification mode to solve the problem.

This model showed an accuracy of approximately 90%.

This provides an important figure. The deployed version should only return a negative result if the probability is below 0.9.



PDA: Model Test

- I loaded the model into Keras, which showed that it could accurately classify its input, further demonstrating that this is a binary classification problem.
- Given that the model is now in a usable state, a solution to the problem can now be implemented.

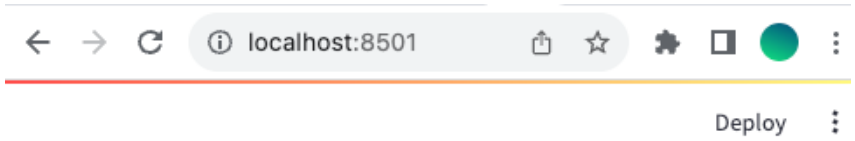
```
1/1 [==============================] - 1s 906ms/step
Class: Genuine
Confidence Score: 0.9994029
```

```
1/1 [==============================] - 1s 898ms/step
Class: Fake
Confidence Score: 1.0
```

Deployment with Streamlit

- At this point in time, I've provisionally chosen Streamlit as the framework I'll use to implement the solution.
- So far, the rendered page displays an indicator that the model (and Keras) loaded successfully.
- The next steps are to:
  - Implement image upload support (by Thursday or Friday).
  - Convert the provided image into a usable form for Keras to process.
  - Perform inference on the image against the model (i.e., the EDA product).
  - Display a message to the user depending on the results of the inference step.

Streamlit: Screenshot



Logo or No-Go

Provide a picture of a logo to determine whether or not it's genuine.

Model loaded and compiled with no errors.

```
model = load_model("keras_model.h5")
st.markdown("Model loaded and compiled with no errors.")
```

Questions?