**Dog Breeds Classification with CNN Transfer Learning**

**Installation**

Besides the Anaconda distribution of Python, the following packages are required to be installed:

opencv-python==4.2.0

h5py==2.9.0

matplotlib==3.1.1

numpy==1.18.2

scipy==1.4.1

tqdm==4.36.1

scikit-learn==0.21.3

keras==2.3.1

tensorflow==2.0.0`

**Project Overview**

The purpose of this project is to build a CNN model and train it to achieve a certain accuracy of predicting the breed of a dog when a picture is supplied.

CNN is a a deep neural networks, which is widely used for image classification. To improve the accuracy of the model, transfer learning is also used.

Transfer learning is a technique that allows a model developed for a task to be reused as the starting point for another task. The pre-trained models used were VGG19 and VGG16. Images from ImageNet were also used to acquire better weights for the model.

**File Descriptions**

Below you can find the files together with a description that were used for this project:

* **haarcascades**: haarcascade\_frontalface\_alt.xml: a pre-trained face detector provided by OpenCV
* **bottleneck\_features**: Please note that this folder is not provided as the file size was huge and unable to upload.
  + DogVGG19Data.npz: pre-computed the bottleneck features for VGG-19 using dog image data including training, validation, and test. You should download from the source indicated in the code.
  + DogVGG16Data.npz: pre-computed the bottleneck features for VGG-16 using dog image data including training, validation, and test. Also required to download by the user.
* **dog\_app.ipynb**: Jupyter notebook used as editor to write the code for the model.
* **extract\_bottleneck\_features.py**: functions to compute bottleneck features given a tensor converted from an image.
* **images**: images to test the model manually.

***Note:*** The datasets for this project can be download from: https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip for the dog images and: https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/lfw.zip for human images.

**Results**

The final model's accuracy is 73% on test data. If a dog image is supplied, the model gives a prediction of the dog breed. If a human image is supplied, the model attempts to match the human's face with the most resembling dog breed. Finally, if a random picture is supplied and the model detects neither a human nor a dog, it return an error.

**Licensing, Authors, Acknowledgements**

Thanks to Udacity for the data images used by this project.