UDACITY

Return to "Deep Learning" in the classroom

DISCUSS ON STUDENT HUB

Dog Breed Classifier

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Hi

Excellent work!! You passed the Dog Breed Classifier project.
Some additional AI resources:
My post about School of AI NDs
9 Deep Learning papers
MIT AGI: Building machines that see, learn, and think like people
YOLO Object Detection
⭐New Deep Reinforcement Learning Nanodegree⭐
The AlphaGo Movie

Keep up learning!
Sincerely
Leticia

Files Submitted

The submission includes all required files.

All the required files are in the submission.

Step 1: Detect Humans

The submission returns the percentage of the first 100 images in the dog and human face datasets with a detected human face.

You applied the face detector over dog and human images.

```
Humans in human files detected: 100.0%
Humans in dog files detected: 11.0%
```

The submission opines whether Haar cascades for face detection are an appropriate technique for human detection.

You answered the question 2. As you said, a CNN could be a good option.
More information:
Haar Cascades

Modern Face Recognition with Deep Learning

Facial Key Points Detection using Deep Convolutional Neural Network - NaimishNet

Step 2: Detect Dogs

The submission returns the percentage of the first 100 images in the dog and human face datasets with a detected dog.

You applied the dog detector over dog and human images.

```
In 100.0% of the dog images was a dog detected
In 0.0% of the human images was a dog detected
```

Step 3: Create a CNN to Classify Dog Breeds (from Scratch)

The submission specifies a CNN architecture.

You created a CNN for dog breed recognition.
You used Dropout layers for preventing overfitting.
Consider using Batch Normalization layers to normalize the input between hidden layers.
DropOut layers
BatchNormalization layers

The submission specifies the number of epochs used to train the algorithm.

You trained the model for 15 epochs.

The trained model attains at least 1% accuracy on the test set.

Test accuracy: 11.0048%

Step 5: Create a CNN to Classify Dog Breeds

The submission downloads the bottleneck features corresponding to one of the Keras pre-trained models (VGG-19, ResNet-50, Inception, or Xception).

You used VGG19 pre-trained model.
Consider tryign Resnet50, InceptionV3, or Xception that work better on this problem.
More information: ImageNet: VGGNet, ResNet, Inception, and Xception with Keras

The submission specifies a model architecture.

You created a simple model for tnrasfer leanring.

The submission details why the chosen architecture succeeded in the classification task and why earlier attempts were not as successful.

You answered the question 5.

The submission compiles the architecture by specifying the loss function and optimizer.

You used categorical_crossentropy loss and Adamax optimizer. Interesting choice.

The submission uses model checkpointing to train the model and saves the model weights with the best validation loss.

The submission loads the model weights that attained the least validation loss.

Accuracy on the test set is 60% or greater.

70.3349282297 great

The submission includes a function that takes a file path to an image as input and returns the dog breed that is predicted by the CNN.

You wrote dog_breed for meeting specifications.

Step 6: Write Your Algorithm

The submission uses the CNN from Step 5 to detect dog breed. The submission has different output for each detected image type (dog, human, other) and provides either predicted actual (or resembling) dog breed.

You wrote the required algorithm

Step 7: Test Your Algorithm

The submission tests at least 6 images, including at least two human and two dog images.

You answered the question 6. More improvement ideas: How To Improve Deep Learning Performance
You tested the algorithm over 6 images including two dog and two human images.

DOWNLOAD PROJECT

RETURN TO PATH