UDACITY

Back to Deep Learning Nanodegree

Dog Breed Classifier

REVIEW

HISTORY

Meets Specifications

Dear Excellent Student,
This is great. Thank you for your submission as I enjoyed reviewing your work. It was well done and this submission meets all of our expectations and you should be proud of yourself. We have come to the end of this project and I can observe you have a good mastery of the domain. Keep it up. I encourage you to continue working hard in other projects and you will be the best in all you do.

Files Submitted

The submission includes all required files.

Great job compiling the necessary files!

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.

Fantastic job in getting the correct percentage of humans and dogs!

- The accuracy for human_files is 98 percent.
- The accuracy for dog_files is 11 percent.

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

> Yes and no. Yes if this is the headshot photo for official documents (passport, identity card, etc), since a clear face is necessary for these purposes. In general, we do not need a clear face to identify if it is a human face. The above showed the human images that were not identified as human face and dog images that were identified as human face. From the figures, it appears an exact front view of face is needed so that it can be detected as a human face. The model also makes mistakes (see above figures) Intuitively, we can use deep learning approach to get a face detector. By training the model with faces with different viewing angles, the algorithm should be able to tell if the picture has faces.

That is great knowledge and rationale. I believe that there should be more complex and flexible algorithms that would help us classify on images that are hard to describe especially the obscure and angled ones. It would be a more complicated version of the current one but it is hopefully ways better in classifying.

Tips

Here are some documents that talk about Haar cascades, take an inspiration from this and complete your answer using Haar cascades.

- Object detection using Haar-cascade Classifier
- OBJECT DETECTION : FACE DETECTION USING HAAR CASCADE CLASSFIERS

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.

Good job on this one!

- The accuracy for human_files is 1 percent.
- The accuracy for dog_files is 100 percent.

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

The submission specifies a CNN architecture.

This is remarkable work in specifying the architecture. You have good knowledge in model architecture and choosing only the ideal steps for optimal output.

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

An epoch number of 10 is interesting.

Comments

- Do you think a number greater than 10 would be better in assessing the accuracy?
- I highly suggest next time for you to try a bigger number of epoch, like 15. It may take longer but it would provide you better accuracy.
- Considering the speed and the capacity of the processor, a lower number is also acceptable.

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

3.1100% is a fine number for the test set! Anything more than 1% is an acceptable rate at this point.

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).

The bottleneck features are downloaded. Good job.

The submission specifies a model architecture.

Great architecture for your model. I hope your model outshines the base model and it seems that would be the outcome by the way it looks.

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

Remarkable work in finding ways to improve the model and finding a way to garner more output from the model.

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

Good use of the loss function and optimizer!

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

Nice work here.

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

Good job loading the model weights that attained the least validation loss.

Accuracy on the test set is 60% or greater.

Excellent. I see you tested out all the models suggested in the project. We can clearly see that Xception outshines all other models.

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.

Great work in implementing a function that takes the file path to an image as an input. This would make the function re-usable and would make the model's job easier as it just needs to recycle the function with a different path per image.

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.

The implementation is great and uses the CNN from the previous step to detect the dog breeds! It would be fun to predict what kind of dog breed some humans are using your model!

Step 7: Test Your Algorithm

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

Splendid work with testing your algorithm and getting superb results. You did a fine job in detecting and classifying the dog breeds and humans alike using the different architecture.

Comments

As you mentioned, one key thing we can do to improve is augment the data. I recommend you to implement the suggestions you mention and see how well the architecture classify the different images.

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

Student FAQ