

[Return to "Deep Learning" in the classroom](#)[DISCUSS ON STUDENT HUB](#)

# Dog Breed Classifier

## REVIEW

## HISTORY

### Meets Specifications

#### Files Submitted



The submission includes all required, complete notebook files.

all required files present 👍

#### Step 1: Detect Humans



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

Good. The percentage of the first 100 images in human\_filenames and in dog\_filenames were detected and correctly returned. Ideally, we would like to have 100% of human images with a detected face and 0% of dog images with a detected face but the algorithm falls short as shown in the answers with 98% of human and 17% of dog images.

#### Step 2: Detect Dogs



Use a pre-trained VGG16 Net to find the predicted class for a given image. Use this to complete a `dog_detector` function below that returns True if a dog is detected in an image (and False if not).

vgg16 was implemented appropriately



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

Awesome! The percentage of the first 100 images in human\_filenames and in dog\_filenames with a detected dog were detected and correctly returned. 0% of human\_filenames detected a dog and 96% of dog\_filenames detected a dog. From observations, we notice that the results get better as the algorithm is modified.

Good job with trying other network. It gave you better results

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



Write three separate data loaders for the training, validation, and test datasets of dog images. These images should be pre-processed to be of the correct size.

the data loaders are implemented correctly. good job with applying different transformation to different dataset



Answer describes how the images were pre-processed and/or augmented.

awesome, you leveraged augmentation. You could try more augmentation techniques it helps in better learning



The submission specifies a CNN architecture.

wow you went too deep. 5 layers would end up getting a lot of minute features.



Answer describes the reasoning behind the selection of layer types.

good explanation. I appreciate that you leveraged Batch normalization. It really helps a lot in achieving better accuracy



Choose appropriate loss and optimization functions for this classification task. Train the model for a

number of epochs and save the "best" result.

nice work. Adagrad is good optimizer. did you try others like adam, adaboost



The trained model attains at least 10% accuracy on the test set.

Awesome. 27% is great.

## Step 4: Create a CNN Using Transfer Learning



The submission specifies a model architecture that uses part of a pre-trained model.

good job with trying different architecture. it is a good practice to do that in real world because you will be able to compare the performances of different models and would know the shortcoming is each other.



The submission details why the chosen architecture is suitable for this classification task.

Nice work



Train your model for a number of epochs and save the result with the lowest validation loss.



Accuracy on the test set is 60% or greater.

89% is super good. Awesome job



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.

A function that takes a file path to an image as input and returns the dog breed that was predicted by the CNN has been included and well implemented in the submission.

## Step 5: Write Your Algorithm



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

output for each detected image type (dog, human, other) and provides either predicted actual (or resembling) dog breed.

## Step 6: Test Your Algorithm



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

good blend of images. you could have tried images out of the ones provided with project but its ok.



Submission provides at least three possible points of improvement for the classification algorithm.

nice points

 [DOWNLOAD PROJECT](#)

[RETURN TO PATH](#)