Use a Pre-trained Image Classifier to Identify Dog Breeds

# Problem:

The city was hosting a citywide dog show and I volunteered to help the organizing committee with contestant registration. Every participant that registered submitted an image of their dog along with biographical information about their dog. The registration system tagged the images based upon the biographical information. Some people registered pets that aren’t actual dogs. I had to develop a program that will process the images submitted and detect the breeds of all dogs registered as well as identify registrations for pets that aren’t dogs.

# Solution

On this project, I used a pre-trained image classifier to identify dog breeds. The classifier is setup to use one of 3 algorithms depending on the user's input. I used an image classification application using a deep learning model called a convolutional neural network (often abbreviated as CNN). CNNs work particularly well for detecting features in images like colors, textures, and edges; then using these features to identify objects in the images. I'll use a CNN that has already learned the features from a giant data set of 1.2 million images called ImageNet. There are different types of CNNs that have different structures (architectures) that work better or worse depending on the criteria.

# Metrics

With this project I explored the three different architectures (AlexNet, VGG, and ResNet) and determined which is best for this application by tracking the following metrics:

1. Which image classification algorithm worked the "best" on classifying images as "dogs" or "not dogs".
2. How well the "best" classification algorithm worked on correctly identifying a dog's breed.
3. How long each algorithm took to solve the classification problem.

# Trial Run

# Final Results

Of the 40 registrations for the citywide dog show, 30 of them were for actual dogs and 10 for other pets. Here’s a summary of the results from the 3 different algorithms tested.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **% correctly classified as not-a-dog** | **% correctly classified as dog** | **% correctly classified dog breed** | **% label match** |
| **AlexNet** | 100 | 100 | 80 | 75 |
| **ResNet** | 90 | 100 | 90 | 82.5 |
| **VGG** | 100 | 100 | 93.3 | 87.5 |

For the first metric, it was observed that both VGG and AlexNet correctly identify images of "dogs" and "not-a-dog" 100% of the time.

For the second metric, VGG provides the best solution because it classifies the correct breed of dog over 90% of the time.

For the final metric, running the program using the VGG algorithm takes 3 seconds while the AlexNet and ResNet both take less than 1 second.

# Conclusion

Given these results, the "best" model architecture is VGG. It out performed both of the other architectures in achieving the desired end results albeit taking a slightly longer time to process the images. Although ResNet did classify dog breeds better than AlexNet, only VGG and AlexNet were able to classify "dogs" and "not-a-dog" at 100% accuracy. The model VGG was the one that was able to classify "dogs" and "not-a-dog" with 100% accuracy and had the best performance regarding breed classification with 93.3% accuracy.