MACHINE LEARNING ENGINEER NANODEGREE – CAPSTONE PROPOSAL

SANGAME KRISHNAMANI
DEC 31, 2017

DOMAIN BACKGROUND

The problem domain for this exercise is computer vision and deep learning. Image classification is one of the reasons why I got interested in machine learning course. Deep learning Dog Breed recognition project made me interested in pursuing further in improving the algorithm. But instead of staying in the realms of the same idea and improving on top of it, I have made it more challenging. The idea is to recognize missing pets and by security camera footage/images posted by people online of homeless pets.

The American Humane Association estimates

- over 10 million dogs and cats are lost or stolen in the U.S. every year. One in three pets will become lost at some point during their life.
- One in three pets will become lost at some point during their life
- Only 22 percent of lost dogs and 2 percent of lost cats that entered the animal shelters were reunited with their families

To make this more focused and achievable the aim of this project is to build a convolutional neural network that classifies whether image contains dogs vs cats.

It is easy for humans to recognize a dog vs cat, but is a challenge for computers because of the variety of breeds, large diversity of photo noise (wide variety of backgrounds, angles, poses, lighting, etc). This makes accurate classification quite difficult to achieve.

In an informal poll conducted many years ago, computer vision experts posited that a classifier with better than 60% accuracy would be difficult without a major advance in the state of the art. For reference, a 60% classifier improves the guessing probability of a 12-image HIP from 1/4096 to 1/459.

Advent in Machine learning has made this problem solvable with decent accuracy and what excites me the most is the fact that ML has opened the door to unlimited possibilities.

This base idea can be expanded to other species and even humans for face recognition, very similar to how Facebook provides suggestions for tagging a face.

Data Source: https://www.kaggle.com/c/dogs-vs-cats/data

Academic Paper:

https://www.robots.ox.ac.uk/~vgg/publications/2012/parkhi12a/parkhi12a.pdf

https://www.mathworks.com/company/newsletters/articles/deep-learning-for-computer-vision-with-matlab.html

PROBLEM STATEMENT

The goal is to build a face recognition system for Dogs and Cats and classify them appropriately.

The dataset images were manually classified and labeled by Asirra. We have a dataset of 25000 images for cats and dogs with labels to accomplish this. Once the computer is trained, we have a test dataset of 12,500 with no labels (cat or dog). Assumption would be that image would contain either a dog or cat but not both. The images might also contain other objects.

DATASETS AND INPUTS

A subset of **Asirra dataset** will be used for this project. CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) or HIP (Human Interactive Proof) are challenges that are easy for people to solve but not for computers. HIPs are used for many purposes, such as to reduce email and blog spam and prevent brute-force attacks on web site passwords.

Asirra (Animal Species Image Recognition for Restricting Access) is a HIP that works by asking users to identify photographs of cats and dogs. This task is difficult for computers, but studies have shown that people can accomplish it quickly and accurately. Many even think it's fun! Here is an example of the Asirra interface:

Asirra is unique because of its partnership with Petfinder.com, the world's largest site devoted to finding homes for homeless pets. They've provided Microsoft Research with over three million images of cats and dogs, manually classified by people at thousands of animal shelters across the United States.

Each image will have either a dog or cat in the image and can contain other objects and human. The train folder contains 25,000 images of dogs and cats. Each image in this folder has the label as part of the filename. The test folder contains 12,500 images, named according to a numeric id.

Few challenging images are shown here:

Some are blurry, other have more than one cat/dog, some have humans and other objects, the image sizes are different and the animals are facing in different angles.

Image format: .jpg file

Number of Images:

<u>Training</u>

Cat images: 12500
Dog images: 12500

Each image in this folder has the label as part of the filename

<u>Testing</u>

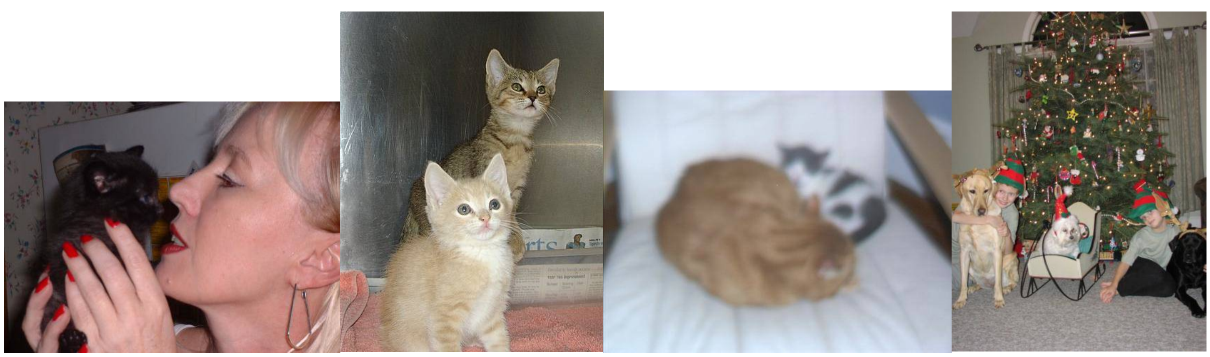
Cat/Dog images: 12500

<u>Size and description of images</u>

All images are different in sizes.

- The images have varying levels of lightening
- there are other objects and humans in the dataset along with cats or dogs
- The face of the animal is angled at different positions and is not always front facing
- The animal is not always present in the center of the frame.

Data Source: https://www.kaggle.com/c/dogs-vs-cats/data

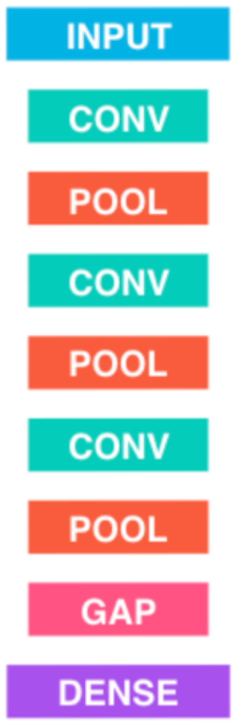


SOLUTION STATEMENT

Deep learning techniques are most effective on image classification and the same approach would be used to solve this problem. Data augmentation and transfer learning will be used to train a Convolutional Neural Network (CNN) to classify the images as dogs vs cats.

Option 1:

The CNN architecture is known for best image classification. As mentioned in the workflow section below, a CNN will be trained from scratch. Using an architecture similar to the one shown below, I would attempt to train a CNN from scratch. The architecture has 3 convolution layers with a ReLu activation function, followed by max-pooling layers. Before the fully-connected layer, because of the large amount of data a dropout layer can considered to reduce overfitting.



OR/AND

Option 2:

Transfer learning or inductive transfer is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. For example, knowledge gained while learning to recognize cars could apply when trying to recognize trucks. Many such networks pre-trained on imagenet challenge are available today – RESNET, Inception-V3, VGG-16, VGG-19, etc.

The CNN will be trained using transfer layer, one of the pre-trained network will be used for the same.

Comparison (optional)

If the previous option of training the CNN from scratch is attempted Option 1 can be compared with 2.

BENCHMARK MODEL

"The security of ASIRRA is based on the presumed difficulty of classifying images of cats and dogs automatically. As reported in [6], evidence from the 2006 PASCAL Visual Object Classes Challenge suggests that cats and dogs are particularly difficult to tell apart algorithmically. A classifier based on color features, described in [6], is only 56.9% accurate."
Ref: https://eprint.iacr.org/2008/126.pdf

EVALUATION METRICS

Either Log Loss or Accuracy Metrics:

Log Loss (categorical cross entropy) quantifies the accuracy of a classifier by penalizing false classifications. Minimizing the Log Loss is basically equivalent to maximizing the accuracy of the classifier, but there is a subtle twist which we'll get to in a moment.

In order to calculate Log Loss the classifier must assign a probability to each class rather than simply yielding the most likely class. Mathematically Log Loss is defined as

$$-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} p_{ij}$$

where N is the number of samples or instances, M is the number of possible labels, y_{ij} is a binary indicator of whether or not label j is the correct classification for instance i, and p_{ij} is the model probability of assigning label j to instance i. A perfect classifier would have a Log Loss of precisely zero.

Accuracy is the ratio of correct predictions to total number of events. In this case it would be the percentage labels correctly classified.

$$\frac{\mathrm{TN} + \mathrm{TP}}{\mathrm{TN} + \mathrm{FP} + \mathrm{FN} + \mathrm{TP}}$$

Where, TN – number of True Negative cases, FP – False Positives, FN – False Negatives, TP – True Positives.

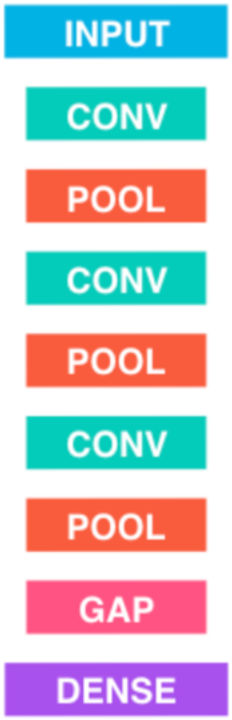
PROJECT DESIGN

Libraries: Scikit-learn, Keras, Tensorflow, NumPy, Pandas, matplotlib, OpenCv

Workflow:

Potentially these would be the steps I would follow -

- Resize the images, so they are of the same size
- Train a CNN from scratch for comparison with transfer learning models (optional)



- Train a CNN using transfer learning
  One of the pre-trained networks will be used RESNET, Inception-V3, VGG-16, VGG-19.
- Compare performance of multiple pre-trained networks before settling for the optimal one (optional)
- Do an objective comparison with the network trained from scratch (optional)
- Fine tuning the pre-trained network by choosing different optimizers.

REFERENCES

https://www.kaggle.com/c/dogs-vs-cats
https://www.petfinder.com/dogs/lost-and-found-dogs/why-microchip/
https://en.wikipedia.org/wiki/Transfer_learning
https://eprint.iacr.org/2008/126.pdf