Machine Learning Engineer Nanodegree
Capstone Proposal

Michael Low
11th February 2017

Domain Background

I have to chosen to work on the Dogs vs Cats Redux Kaggle competition (Kaggle, no date), which is the challenge of automatically distinguishing photos of dogs from photos of cats.

This has been a task that is traditionally very easy for humans, but difficult for computers due to large variety of shapes, breeds, colours, photo composition, lighting and so on in the photos. Ten years ago, it was used as a CAPTCHA challenge (Microsoft, 2007) to distinguish human users of a system from computers. In 2008, techniques in computer vision advanced to sufficiently attack the CAPTCHA with 82.7% accuracy with a SVM classifier (Golle, 2008) making it no longer viable. Modern computer vision techniques using convolutional neural networks should be able to improve on this substantially further still.

I think this is an exciting challenge to work on as, although it may have limited use itself, it encapsulates the fundamental techniques needed to solve a wide range of computer vision problems into a simple problem domain. Consequently, the methods used to achieve high accuracy on the Dogs vs Cats problem can then be applied to to a wide range of today's computer vision challenges.

Problem Statement

The problem to be solved is to automatically identify whether a given jpg image is of a cat or a dog, along with a probability reflecting the confidence in that prediction.

The correctness of the solution can be measured using accuracy, which is the percentage of of the predictions were correct out the total number. It can also be measured using log loss, which rewards correct predictions but weights the penalty applied to incorrect predictions according to confidence probability.

Datasets and Inputs

The dataset for the problem is provided by Kaggle and available on their site (Kaggle, no date). The training set contains 25,000 images, half of cats and half of dogs. The label for each image is given by the filename, such as *dog.1.jpg*. The test set contains 12,500 images, presumably also approximately evenly distributed. All images contain one, and only one, animal. The training set will be used to train the classifier, and its performance finally evaluated against the images in the test set.

The training and test set both originally come from the Asirra dataset (Microsoft, 2007), of which the full dataset contains more than 3 million images of cats and dogs from petfinder.com. This makes it an ideal dataset for this problem, although as the images are specifically of pet dogs and cats it may be skewed towards certain breeds that are most popular as pets.

Solution Statement

It should be possible to achieve a high accuracy rate in solving the problem by:

- Downloading an existing neural network pre-trained on ImageNet data.
- Splitting the training data into training and validation sets.
- Fine-tune it by training further of our training data.
- Adapting the network to return just 'cat' or 'dog' classes, rather than the 1000 image net classes.
- Testing and trying different parameters settings to get a high accuracy score on both the training and validation set.

Benchmark Model

A basic benchmark for this project would be whether it can give better results than could be obtained by chance. With the two output classes of either 'cat' or 'dog', random guessing would be expected to give an accuracy of 0.5. Therefore, we would expect this classifier to be significantly above 0.5 in order to prove its effectiveness. This is also the benchmark set by Kaggle, which currently lies around position ~750 of 990 in the leaderboard.

Evaluation Metrics

The two evaluation metrics proposed to evaluate the classifier performance are *accuracy* and *log loss.*

Accuracy is the number of correctly predicted class labels, divided by the total number of predictions made. This is a simple and easy-to-understand metric, and is appropriate due to the dataset being balanced and only binary classification required. In addition, a false positive is not considered either better or worse than a false negative. Therefore, other metrics suitable for imbalanced datasets, such as f1-score, precision and recall, do not offer any real advantage over accuracy in this case.

Log loss is the metric used by Kaggle to judge this competition (Kaggle, no date b).

$$LogLoss = -\frac{1}{n}\sum_{n}^{i=1}\left[y_i \log\left(\hat{y}_i\right) + \left(1 - y_i\right)\log\left(1 - \hat{y}_i\right)\right]$$

In general, a highly accurate model will also have a low log loss score. Log loss applies a higher penalty for incorrect predictions that had a high probability given to them, making it more appropriate for leaderboard ranking. However, it is less intuitively easy to understand.

As Kaggle uses log loss score, this is the only metric available for the test set. Accuracy can be measured by testing against the validation set.

Project Design

Ever since the AlexNet architecture was published (Krizhevsky et al, 2012), convolutional neural networks have dominated the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This is an annual competition assessing the state of the art in computer vision. This challenge attempts to classify images into one of a thousand common categories, and therefore techniques used here in should scale down well to classify images into just the two categories of cat or dog.

Pre-trained deep learning models of previous years winner of the ImageNet Challenge are available to download, along with the weights obtained by training them on the millions of images in ImageNet.

I have chosen to use the Keras deep learning framework for this project, due to its Python support, clean API and choice of of either Theano or TensorFlow backends. Keras makes available various "Keras Applications", which include previous winners of the ImageNet challenge such as VGG, ResNet and Inception (Keras, no date) and their ImageNet weights. I will use the VGG16 model, as it gives high accuracy without the much higher complexity of ResNet and Inception models.

The next step would be to use the model with the Kaggle data. For this, we can split the full training data (12,500 images each of cats and dogs) into a validation set (2000 images each) and remaining training data (10,500 images each). Having a separate validation set allows us to assess the performance of the model as it is being trained.

Then we can fine-tune the model by feeding in the training data in batches, and fitting the model to return just Dog or Cat classes, rather than the 1000 ImageNet classes. VGG16 was trained using 224x224 pixel images, so images would need to be resized to this dimension first.

After training for some number of epochs, this should be sufficient to achieve high accuracy of the image recognition. If not, other techniques may need to be explored. This could include data augmentation (changing the images subtly in different ways so the model has more variety of images to learn from), or using dropout or regularization to reduce overfitting if necessary.

References

Golle, P., 2008, October. Machine learning attacks against the Asirra CAPTCHA. In *Proceedings of the 15th ACM conference on Computer and communications security* (pp. 535-542). ACM.

Kaggle, no date. "Dogs vs. Cats Redux: Kernels Edition" available at https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition

Keras, no date. "Keras Applications" available at https://keras.io/applications/

Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

Microsoft, 2007. "Asirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization" available at https://www.microsoft.com/en-us/research/publication/asirra-a-captcha-that-exploits-interest-aligned-manual-image-categorization/