CS783 Assignment 2

Aman Tiwari(160094) Ankush Tiwari(160130)

February 2019

1 Problem Statement

The dataset consists of some coarse classes. Each coarse class has some fine classes. The problem is to predict coarse and fine class for a given test image.

2 Methods Used

2.1 Coarse grained Classification

For coarsed grained classification we used Resnet18 as feature extractor. We used the output of last convolutional block of the Resnet18 as features for input images. After that we added a fully connected layer,a dropout layer for regularization and a softmax layer to give the output classes.This method gave around 99 percent accuracy on our validation set.
We also tried to fine tune by training the last few layers of Resnet18 but since the accuracy was already very high it did not have a much effect.

2.2 Fine grained Classification

For fine grained classification, we used Bilinear CNN. We first used a standard ResNet18 architecture to extract image features and took the output of last convolutional block as features. We took an outer product of the layer with itself, then flattened it into a 1-D vector and then it was passed through signed square root step followed by l_2 normalization. We then added a Dense layer of shape equal to number of fine classes and then computed activations using softmax. The model was then trained and the loss function was sparse categorical crossentropy loss.To improve upon the baseline method, we tried several methods like:

2.2.1 Fine-tuning top layers of Resnet

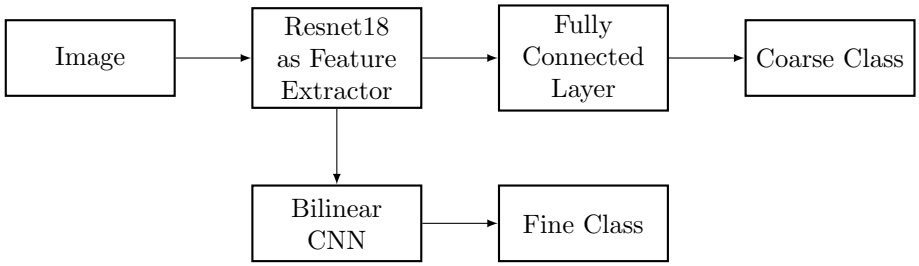
To do this we first instantiate the convolutional base of Resnet18 and load its weights. Then we add our Bilinear CNN model on top ans load its weights and then freeze the layers of Resnet18 model up to the last convolutional block.In our

case because of limitation on the number of parameters to be used we allowed fine tuning of the last convolutional block only.

2.2.2 Image Augmentation

One major problem that we faced during training was the problem of overfitting. To tackle this problem we used image augmentation which helps increase the variations of images in our data set by using horizontal/vertical flips,rotations,variations in brightness of the image,horizontal,vertical shifts etc.This effectively increases the size of training set and hence reduces chance of overfitting.
Also the number of training examples were not same for all categories of a class. For example - The aircraft class had 266 training examples for category 1 and other categories had about 65 examples only. This led to a bias towards category 1 while training. Using image augmentation we generated images for other categories and tried to equalize the training examples in all categories.

3 Algorithm architecture



4 Challenges Faced

For the fine grained classification one challenge that we faced was that of overfitting. The training accuracy was much higher but the validation accuracy was not that high. We used Image Augmentation and regularization to solve this problem.

5 Saved Models

The md5 hashes of saved models are:

aircrafts_weight.h5	00e12626599e7c16deb072367d2fb81e
birds_weight.h5	abf7f7bfab08b0072d81483d00fb9225
cars_weight.h5	793cc02e849ee4c09fddcc41dd75ab34
flowers_weight.h5	18f305f06428fe92b140e7ce921f3774
dogs_weight.h5	5b3999d52c60e43743bba08b399c0f24
coarse_model.h5	27f7cd3b2e69b44c21381b22d6daab8d