Assignment-2 Convolution

Objective: "Deep learning for computer vision" convents can successfully classify images. Specifically, consider the Cats & Dogs example. There were two broad approaches to organizing Cats & Dogs using convnets. You will examine the relationship between training samples and choosing to train your model from scratch versus using a pre-trained convnet.

**1. Consider the Cats & Dogs example. Start initially with a training sample of 1000, a validation sample of 500, and a test sample of 500 (like in the text). Use any technique to reduce overfitting and improve performance in developing a network you train from scratch. What performance did you achieve?**

In the unregularized model, the test accuracy is around: 75.2 with the ADAM optimizer.

The model is overfitting after a certain epoch which may not be generalized with the new data.

Regularized with dropout and augmentation:

In this model, there is better performance improvement with an accuracy of 79.2

In both models, 1000 samples are taken for training.

By fine-tuning the model, the test accuracy is at 82.0 and loss at 41.31.

The training sample of 1000 is insufficient for the model to reduce the loss and increase the accuracy.

**2. Increase your training sample size. You may pick any amount. Keep the validation and test.**

**samples the same as above. Optimize your network (again, training from scratch). What**

**performance did you achieve?**

For this model, I have increased the sample size to 5000.

For the unregularized model, It is observed that the model may be overfits after a certain epoch.

Test accuracy: 0.833, which is, in contrast, an improvement compared to the model with only 1000 training samples.

For regularized model:

For a regularized model, it is observed that the loss: is 0.2726 - accuracy: is 0.8880.

In contrast to the unregularized model regularized model seems to have a bit higher accuracy.

In comparison to the previous model, the accuracy seems to be improved while the loss is slightly reduced.

**3. Now, change your training sample so that you achieve better performance than those from Steps 1 and 2. This sample size may be larger or smaller than those in the previous steps. The objective is to find the ideal training sample size to get the best prediction results**.

In this model, the training size is 10,000 for the unregularized model. It is observed that the loss: is 0.4258 - the accuracy: is 0.872. For the regularized model, it is observed that the loss: is 0.2065 - accuracy: is 0.9140. In comparison, this model is better than the previous model.

**4. Repeat Steps 1-3 using a pre-trained network. The sample sizes you use in Steps 2 and 3 for the pre-trained network may be the same or different from those using the network where you trained from scratch. Again, use all optimization techniques to get the best performance.**

The pre-trained network may be the same or different from those using the network where you trained from scratch. Again, use any and all optimization techniques to get the best performance.

For training Sample Size of 1000 with regularization After fine-tuning the model, it is observed that the loss: is 0.2652 - Accuracy: is 0.8900, which is slightly better than the untuned regularized model.

For training Sample Size of 5000 with regularization After fine-tuning the model, it is observed that the loss: is 0.2429 - Accuracy: is 0.8976, which is slightly better than the untuned regularized model.

For training Sample Size of 10000 with regularization

After fine-tuning the model, it is observed that the loss: is 0.1656 - the accuracy: is 0.9380. which is so far the best model than the untuned regularized model. Time and space complexity

**Recommendations:**

From the different training sample sizes, every model did perform well, but the fine-tuned model with higher training samples with dropout and data augmentation is highly accurate with less loss rate.

However, there is a drawback to higher training samples. For a model with 5000 training samples, each epoch took around 59s, and for the model with 10000, it took almost 63s. There is an issue with time and space complexity with higher training sample models. If proper infrastructure is provided, then these issues can be handled.

But the best efficient model is the fine tune with a higher training sample (5000,10000) using the pre-trained network while test and validation remain of the same size of 500.

I considered only 35 epochs for each model and ran on GPU, but it still took a lot of time.

I did try to run for 100 epochs in the data augmentation and dropout model, but the collab timed out after a certain epoch and had to run for only 35.