**Assignment 2: Convolution**

-Tejaswini Kolli

The basic models demonstrated increasing accuracies with increasing training sample numbers in training samples of 1000, 2000, and 5000 while test size and validation size are kept constant at 500. To construct the model, all the above models were run with the following parameters of metrics as accuracy, optimizer as adam and loss function as binary\_crossentropy. Additionally, the pretrained model of training size 3000 uses the optimizer rmseprop to gain more insight into the model's performance.

**Summary:**

* Training size of 1000, Test sample size 500 and validation sample size 500 showed loss value as 0.6202 and Accuracy as 0.6420. Throughout the course of epochs, training accuracy improved with each epoch.
* Training size of 3000, Test sample size 500 and validation sample size 500 showed loss value as 0.5253 and Accuracy as 0.7740. Throughout the course of epochs, training accuracy improved with each epoch.
* When training samples grew to 1000, 2000, and 3000, the base models' accuracies improved as well.
* As we have the highest accuracy training size of 3000, I have chosen the same training size to pretrain the model and run the model with data augmentation and without data augmentation.
* Among all the model’s run, pretrained model with data augmentation and training size of 3000 has the highest accuracy.
* As we have highest accuracy for 3000 training size I have changed the optimizer to rmseprop to know better about the model’s performance.
* Adam the optimizer has the slightest accuracy gain with the least amount of accuracy loss as compared to rmseprop. As a result, we can employ any optimizer.

The model is run using various training sizes, both with and without data augmentation, and with a pretrained model (both with and without data augmentation) in the tabular format that is shown below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Optimizer: Adam | | | | | |
| S.no | Training Set | Validation and Test Size | Data augmentation | Pretrained Model | Loss and Accuracy on Test |
| 1 | 1000 | 500, 500 | No | No | loss: 0.5659 - accuracy: 0.6980 |
| 2 | 2000 | 500, 500 | No | No | loss: 0.5604 - accuracy: 0.7360 |
| 3 | 3000 | 500, 500 | No | No | loss: 0.5253 - accuracy: 0.7740 |
| 4 | 1000 | 500, 500 | Yes | No | loss: 0.5944 - accuracy: 0.6840 |
| 5 | 2000 | 500, 500 | Yes | No | loss: 0.5832 - accuracy: 0.7280 |
| 6 | 3000 | 500, 500 | Yes | No | loss: 0.5433 - accuracy: 0.7600 |
| 7 | 3000 | 500, 500 | No | Yes | loss: 7.3143 - accuracy: 0.9729 |
| 8 | 3000 | 500, 500 | Yes | Yes | loss: 3.4010 - accuracy: 0.9720 |
| Optimizer: rmseprop | | | | | |
| 9 | 3000 | 500, 500 | No | Yes | loss: 7.3945 - accuracy: 0.9660 |

**Conclusion:**

The effects of different training sample sizes, data augmentation, and optimizer selections on model performance were examined in a series of experiments. Notably, both loss and accuracy significantly improved when the training size was increased from 1000 to 3000 samples. This highlights how important enough training data is to improving model performance. Moreover, the models trained with augmented data performed better than the ones that weren't. A useful method for expanding the effective dataset size and enhancing the model's capacity to generalize from sparse data is data augmentation. An analysis was conducted on optimizer selection, comparing RMSprop and Adam. For this scenario, Adam proved to be a more durable and dependable option than RMSprop, exhibiting a minor increase in accuracy and a reduction in accuracy loss. Ultimately, the model that achieved the highest accuracy was pretrained, utilized data augmentation, and was trained with a sample size of 3000. This model was further improved by employing the Adam optimizer. The findings emphasize the importance of data size, data augmentation, and optimizer selection in model training and demonstrate that these factors collectively lead to superior model performance.

**Recommendations:**

* Larger training sample sizes allowed for the extraction of more features and helped reduce overfitting.
* The approach of pre-trained convolutional neural networks trained on the ImageNet dataset significantly improved model's performance.
* With pre-trained CNNs and data augmentation, the model’s ability significantly increased accuracy especially with smaller training sample sizes. This implies that even with a small amount of training data, using pre-trained models can be advantageous.
* The choice of optimizer and hyperparameters can significantly impact training.