Assignment 2: Convolution

-Divya Gunde

- Veera Manikanta Vallabhaneni

The test and validation sizes were maintained at 500, but the basic models showed rising accuracies as the number of training samples increased to 1000, 2000, and 5000. Using the following parameters—metrics like accuracy, optimizer like Adam, and loss function like binary\_crossentropy—all of the aforementioned models were constructed. In addition, the optimizer rmseprop is used by the pretrained model with training size 3000 to help understand the model's output.

**Summary**:

• The accuracy was 0.6449 and the loss value was 0.7060 for the training sample size of 1000, the test sample of 500, and the validation sample of 500. Each period of training resulted in an improvement in accuracy.   
• The accuracy was 0.5253 and the loss value was 0.7440 for the training size of 3000, 500 for the test sample, and 500 for the validation sample. Each period of training resulted in an improvement in accuracy.   
• With training samples increasing to 1000, 2000, and 3000, the accuracy of the base models also increased.   
• I have used the same training size to pretrain the model and run it both with and without data augmentation because we have the highest accuracy training size of 3000. • The pretrained model with data augmentation and a training size of 3000 had the highest accuracy out of all the models run.   
• To have a better understanding of the model's performance, I switched the optimizer to rmseprop because we have the maximum accuracy for training sizes of 3000.   
• When compared to rmseprop, Adam the optimizer has the least accuracy loss and the smallest accuracy gain. We may thus use any optimizer.   
The model is executed in the tabular manner that is displayed below, with several training sizes, with and without data augmentation, as well as a pretrained model.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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: 0.6449 - accuracy: 0.7060 |
| 2 | 2000 | 500, 500 | No | No | loss: 0.5867 - accuracy: 0.7330 |
| 3 | 3000 | 500, 500 | No | No | loss: 0.5253 - accuracy: 0.7740 |
| 4 | 1000 | 500, 500 | Yes | No | loss: 0.5068 - accuracy: 0.7810 |
| 5 | 2000 | 500, 500 | Yes | No | loss: 6.2801 - accuracy: 0.9770 |
| 6 | 3000 | 500, 500 | Yes | No | loss: 0.5433 - accuracy: 0.7600 |
| 7 | 3000 | 500, 500 | No | Yes | loss: 6.1129 - accuracy: 0.9790 |
| 8 | 3000 | 500, 500 | Yes | Yes | loss: 0.5594 - accuracy: 0.9750 |
| Optimizer: rmseprop | | | | | |
| 9 | 3000 | 500, 500 | No | Yes | loss: 0.5516 - accuracy: 0.9730 |

Conclusion:

The study investigated the impact of varying training sample sizes, implementing data augmentation techniques, and selecting different optimizers on the performance of a model. Results indicated that increasing the training sample size from 1000 to 3000 led to significant improvements in both loss reduction and accuracy enhancement. This underscores the crucial role of having sufficient training data for improving model performance.

Furthermore, models trained with augmented data consistently outperformed those without augmentation. Data augmentation emerged as a valuable strategy for expanding the effective dataset size and improving the model's ability to generalize from limited data.

The study also compared the performance of two optimizers, RMSprop and Adam. It was found that Adam offered better performance compared to RMSprop, resulting in a slight increase in accuracy and a reduction in loss.

Ultimately, the model achieving the highest accuracy was pretrained, utilized data augmentation, and was trained with a sample size of 3000. Further optimization with the Adam optimizer contributed to its superior performance. These findings underscore the importance of factors such as data size, data augmentation, and optimizer selection in model training, highlighting their collective impact on achieving better model performance.

**Recommendations:**

• More features could be extracted and overfitting was minimized with larger training sample sizes.  
  
• Pre-trained convolutional neural networks trained on the ImageNet dataset greatly enhanced the performance of the model.  
  
• The model's accuracy rose dramatically, especially with smaller training sample sizes, when pre-trained CNNs and data augmentation were used. This suggests that employing pre-trained models can be helpful even when there isn't much training data.  
• Training can be greatly impacted by the optimizer and hyperparameter selection.