**Convolution Neural Networks**

Convolution Neural Networks are defined as a class of deep learning neural networks primarily used in Image Recognition, Object Detection, and Image Classification. Basically, it functions as a filter applying filters to the input image. An image is convolution when a set of filters is slid over the image, resulting in several feature maps showing the image's various aspects.

What I analyzed:

A data set containing 25000 images of cats and dogs can be used for training. Using convolution networks, I trained a model on this subset of data to predict cats and dogs. The first model was trained on 1000 images, validated on 500 images, and tested on 500 images. The picture shows the architecture of the model:

A screenshot of a computer program

Description automatically generated

Observations:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Training from scratch | | |
| Performance Metric | Accuracy | Validation Loss | No. of epochs |
| Initial model  (Training=1000, Validation=500, Test=1000) | 69.3 | 0.573 | 30 |
| Data Augmentation  (Training=2000, Validation=500, Test=1000) | 81.4 | 0.457 | 100 |
| Increased Training Data  (Training=2500, Validation=500, Test=1000) | 68.6 | 0.632 | 30 |
| Optimal Training Data  (Training=3500, Validation=500, Test=1000) | 90.2 | 0.249 | 30 |

Observations for Pretrained Model:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Pretrained Model | | |
| Performance Metric | Accuracy | Validation Loss | No. of epochs |
| Initial model  (Training=1000, Validation=500, Test=1000) | 99.7 | 0.318 | 20 |
| Data Augmentation  (Training=2000, Validation=500, Test=1000) | 99.4 | 0.1532 | 60 |
| Increased Training Data  (Training=2500, Validation=500, Test=1000) | 99.4 | 0.30 | 30 |
| Optimal Training Data  (Training=3500, Validation=500, Test=1000) | 1 | 0 | 30 |

As the model is trained from scratch, random weights are assigned to it and gradually updated throughout. To achieve good performance, labelled data must be of good quality and large. Based on this dataset, the initial model built using Training from Scratch was 69.3% accurate and 0.573 validated.

A Pretrained neural network model trained on a large dataset is used as a starting point for a similar and smaller dataset. This improves performance, reduces overfitting, saves time and computational resources. The initial model built on this dataset using a pretrained neural network achieved 99.7% accuracy and 0.318 validation loss.

**Model adding Data Augmentation:**

When I add the model with data augmentation the test accuracy has increased from 69.3% to 81.4%, and the validation loss has decreased from 0.573 to 0.457. Meanwhile, in the pretrained model the accuracy also increases to 99.4% and validation loss has decreases to 0.153. This is because data augmentation is a technique used in model training to apply variations like rotating, flipping, and zooming to each image. This creates new forms of images and helps to learn more robust features and improve performance while reducing overfitting, causing an increase in validation loss in pre-trained networks.

**Sample Increment for Training model:**

A larger training sample in a CNN improves model performance by allowing it to learn more and capture more patterns. This results in higher test accuracy, up to 90.2% when training from scratch and 99.4% using a pretrained network. More samples also reduce overfitting, but low- quality data can increase it. In this scenario, validation loss is 0.26 when training from scratch and 0.30 when using a pretrained network.

**Sample Increment for Optimal Training:**

More training data improves the model’s performance, but there’s a limit. After a certain point, additional data doesn’t provide much useful information because the model has already learned all the important features, causing its performance to stabilize.

The best sample size for network model is 3500 having accuracy of 90.2 and validation loss of 0.26 for training from scratch network.

But whereas in pretrained trained model, accuracy is 1 and validation loss is 0, which is clear that the model is overfitting. When a pre-trained model is complex and deep, it can capture features in the data very well. However, this can cause overfitting when combined with a large sample size, as the model becomes too specialized for the training data. Additionally, if the distribution of new data is significantly different from the original pre-trained data, the model may not generalize well. It may focus too much on fitting the training data and fail to adapt to the differences in the new data.

**Conclusion:**

In conclusion, training a Convolutional Neural Network from scratch yielded satisfactory results. However, we found that using a pretrained model with data augmentation significantly improved the accuracy of the model, while also reducing the validation loss. I got that the optimal training model sample size with 3500 with the best accuracy.