CONVOLUTION SUMMARY

Siva Sai Madhumitha Kotala

811257709

The objective of this research is to construct a convolutional neural network that, by identifying the distinctive characteristics of each animal, can accurately and successfully recognize photos of cats and dogs.A Kaggle dataset including 12,500 test and 25,000 training images, with an equal proportion of dogs and cats, is used in this experiment.

To be defined problem:

Determining whether an image is of a dog or a cat is the aim of the Cats-vs-Dogs dataset.

Data Methods:

There are 25,000 photos in the Cats-vs-Dogs dataset, which is equally divided between dogs and cats and has a compressed size of 543MB. I downloaded the dataset, unzipped it, and then made a new dataset with three subsets:

- A training dataset with 1000 samples from each type
- A validation dataset containing 500 samples;
- 500 samples in the test dataset

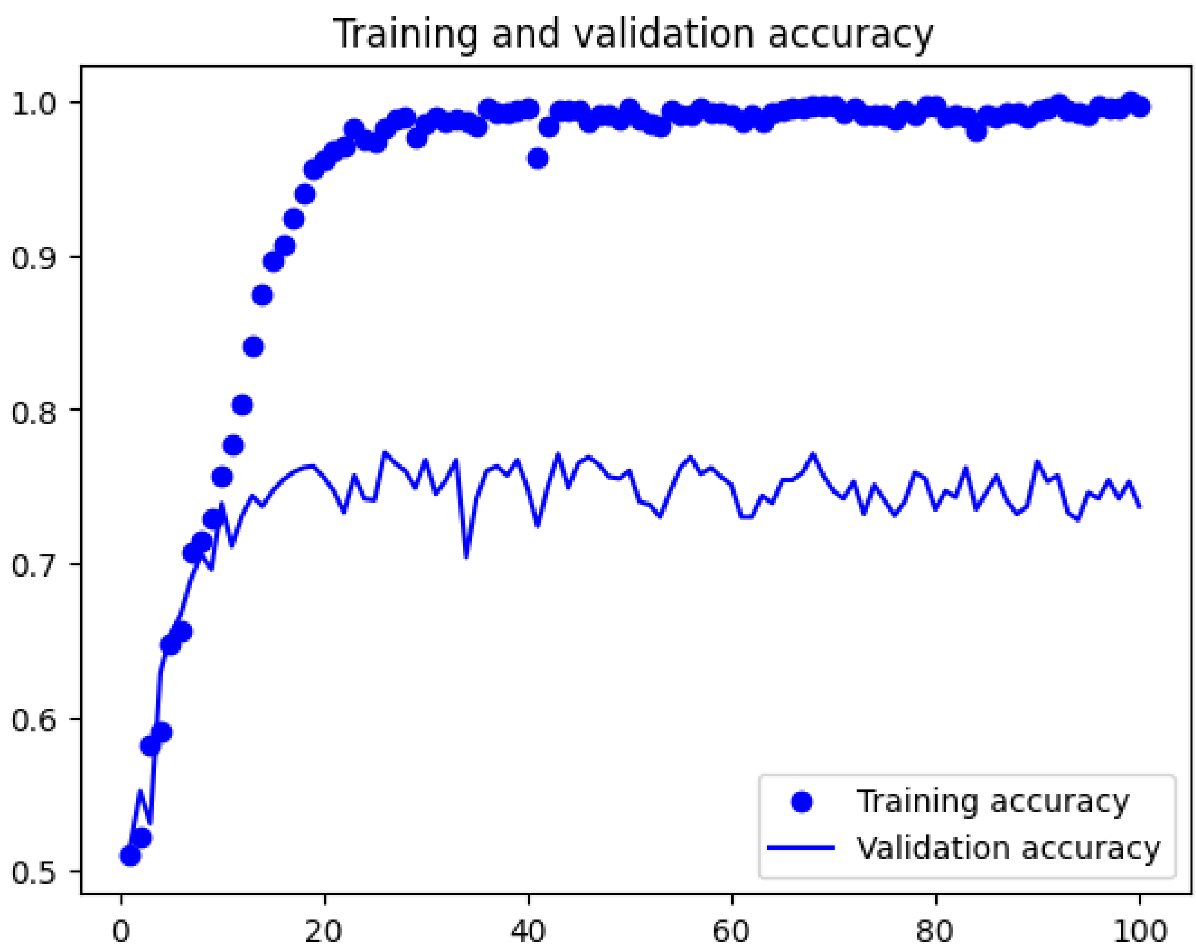
Our current task requires a broader image and is more complex, therefore we need to increase the size of our neural network. We will incorporate a step into our current Conv2D + MaxPooling2D design to accomplish this. This not only increases the network's capacity but also ensures that the feature maps won't be too large when we get to the Flatten layer. At first, our input photographs have a 150 x 150 resolution. The feature maps get smaller as we move up the network's stages, reaching 7x7 just before the Flatten layer. Despite appearing somewhat arbitrary, the input size selection is appropriate for the task at hand.

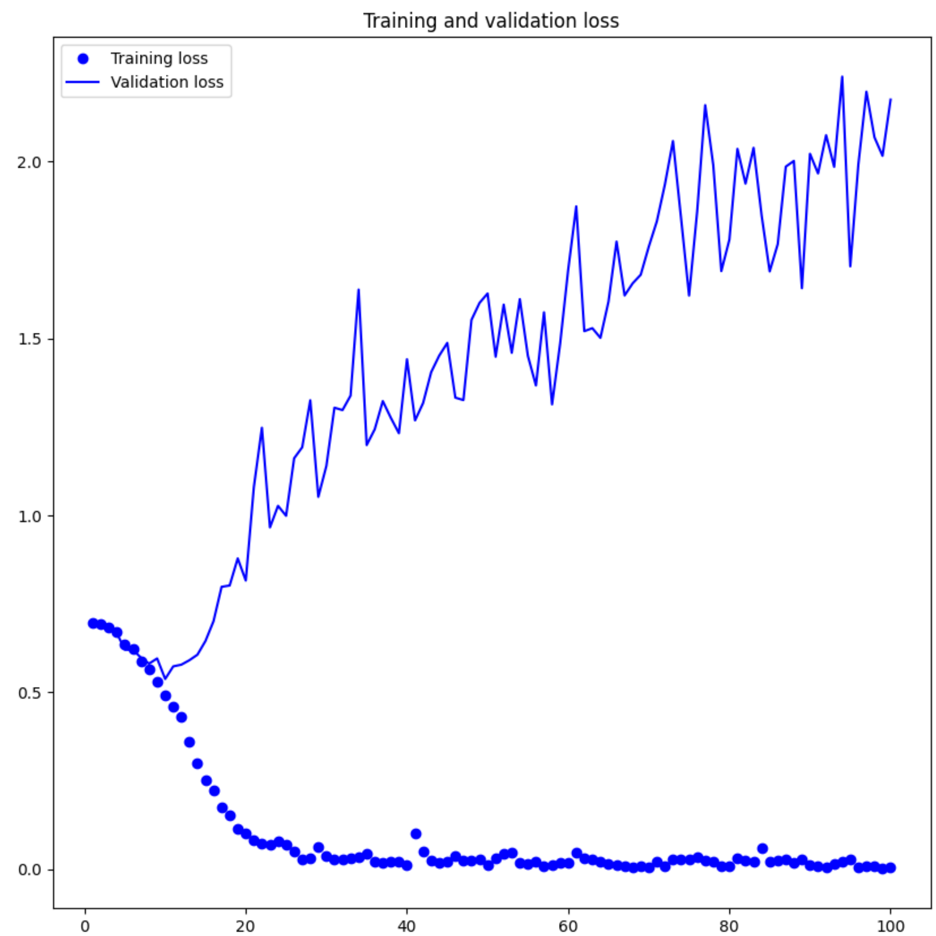
Preprocessing the data:

- Look over the picture files.
- To create RBG pixel grids, convert the JPEG data.
- Utilizing grids, construct floating point tensors.

  In order to optimize the performance of neural networks, the pixel values, which span from 0 to 255, should be rescaled to the [0, 1] interval.

I used the data flattening technique to convert the data transformation, selecting 255 as the batch size. With the use of 100 epochs, we were able to calculate the test accuracy to be 71.9% and the validation accuracy to be 73.7%.





We can infer from the preceding result that, whereas training accuracy is approximately 99.8%, test accuracy without data augmentation is approximately 71.9%.

Question 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?

Data Augmentation

This strategy can be used to increase the model's accuracy. Data augmentation is one method that enables reliable results to be obtained even with tiny datasets. This involves creating fresh data by introducing random modifications to the provided training samples. By ensuring the model sees a wide variety of images during training, this technique enhances the model's ability to successfully generalize.

All of the following results are based on the training sample of 1500 and the validation test of 500.

Displaying the enhanced images that were trained

83% is the Test accuracy. The validation accuracy of 85.2% shows better results than the preceding (Question 1), which can be attributed to the following factors:

The following factors have enhanced the model's performance:
In addition to adding 500 (1000–1500) training samples, we also introduced data augmentation to the convolution layer, which helped us enhance the featured extractions that resulted in superior performance. Together, these changes helped us raise test and validation accuracy by over 10%.

Question 3: Now change your training sample so that you achieve better performance than those from Steps1 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 best prediction results

Since using more and more data will help to improve the model's performance, we are unable to estimate the optimum sample size.

- Test sets including 500 samples and 2000 training samples with validation were used in this. I've discovered that test accuracy is higher with 1500 photos as opposed to training samples of 1000 and 2000 photos.
- Training accuracy increases with 1000 training samples.
- Increasing the training sample to 2000 while keeping the validation and test sets with 500 samples.

Results

Training samples	**Validation Accuracy**	**Test Accuracy**	**Data Augmentation**
1000	**73.7%.**	**71.9%**	NO
1500	**85.2%**	**83%**	YES
2000	**83.80%**	**80.5%**	YES

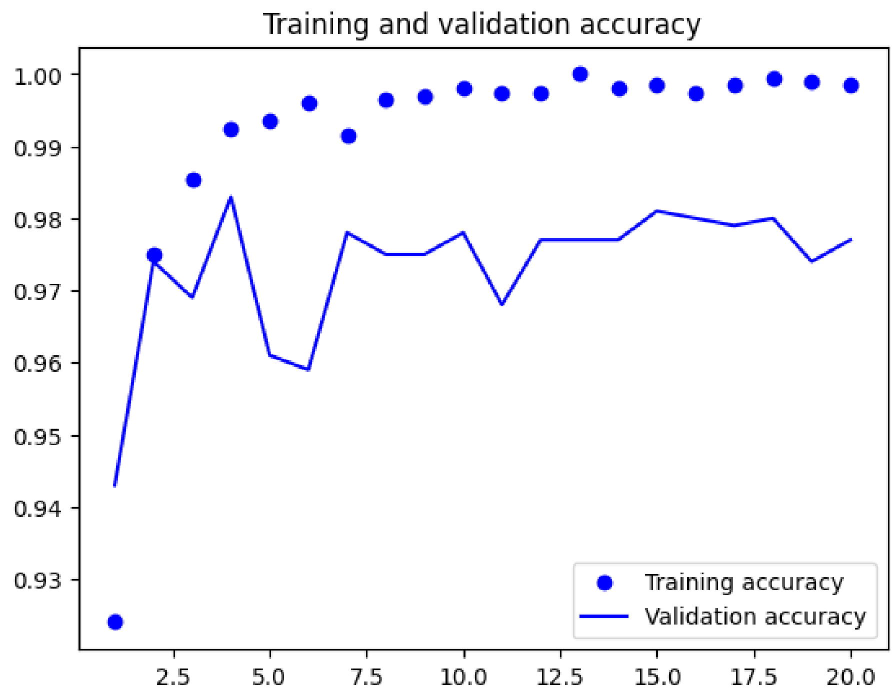
Question 4: Repeat Steps 1-3, but now using a pretrained network. The sample sizes you use in Steps 2 and 3 for the pretrained network may be the same or different from those using the network where you trained from scratch. Again, use any and all optimization techniques to get best performance.

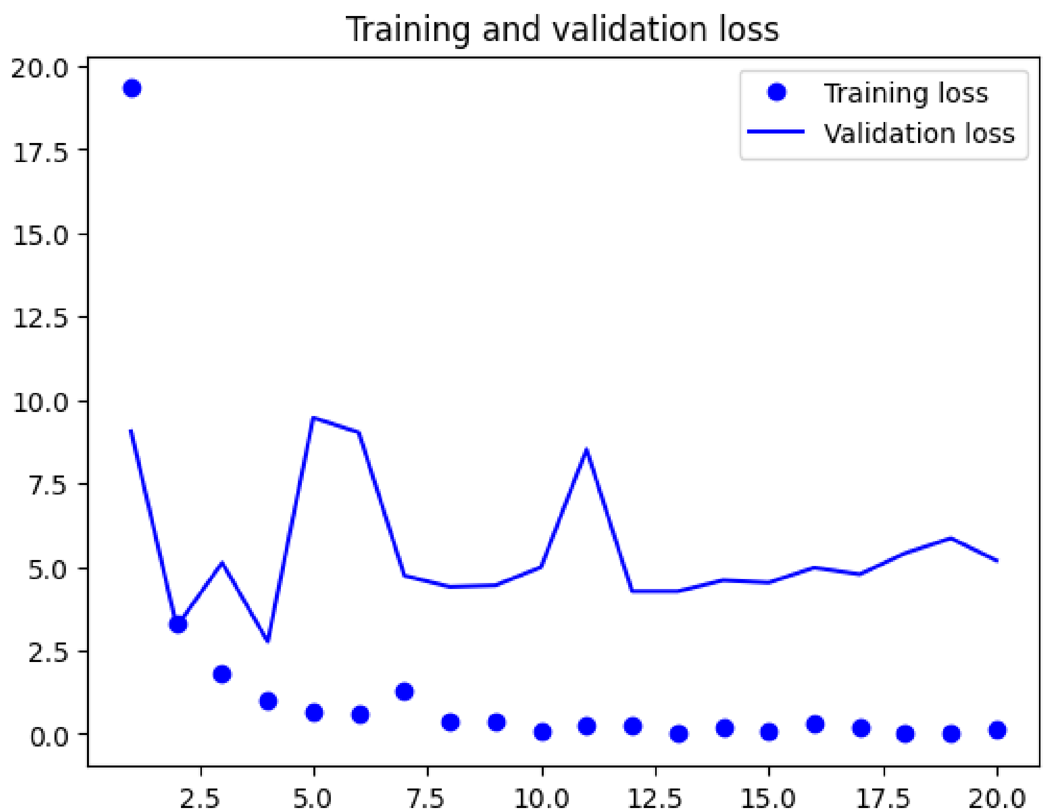
Pre-trained model:
For feature extraction and fine-tuning, pretrained networks are primarily employed. When a pretrained network has a large and varied baseline dataset, its features can be applied to a range of computer vision applications, serving as a generic model. The deep learning capabilities among the main advantages it has over other machine learning techniques is the capacity to apply learnt attributes to a range of tasks. The ImageNet dataset, which comprises 1.4 million annotated images and 1,000 distinct classifications, can be used as an example for studying a large trained convolutional neural network. Many breeds of dogs and cats are among the many animal categories that are represented in the collection. This network uses the popular and straightforward convnet design for ImageNet, known as VGG16 architecture.

In this case, feature extraction will be used to improve the outcomes, initially without data augmentation and then with data augmentation.

Pre-Trained model without Augmentation:

Our train accuracy is 99.85%, and our validation accuracy is 97.7%. Even though we are utilizing dropout at a rather high rate, the figures show that we are nearly instantly overfitting.


Training and validation accuracy
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Training accuracy
Validation accuracy


Training and validation loss
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Training loss
Validation loss

Pre-Trained model with Data Augmentation:
• validation accuracy is 98.2%
• train accuracy is 97.6%
• test accuracy: 97.2%.

Fine-tuning a pretrained model
• validation accuracy is 98.5%
• train accuracy is 99.5%
• test accuracy: 97.6%.

TABLE FOR PRE-TRAINED MODEL

Data Augmentation	Train Accuracy (%)	Validation Accuracy (%)
NO	99.85%	97.7%
YES	99.5%	98.5%

CONCLUSION:

With a brief training set of 1000 samples, we were able to achieve a 99.8% training accuracy.
Data augmentation is meant to lessen overfitting.

Overfitting can be avoided in the following ways:
- Expanding the training sample is not always an option. Utilizing data augmentation is one method to maximize the limited training data.
- What controls the amount of overfitting when the model's size is lowered is the number of learnable parameters in the model, or the number of layers and units in layers.
- One effective technique to prevent or minimize overfitting and reduce the complexity of a network is to zero off portions of the layer's weights during training.
- One effective method to reduce overfitting in training is to set some of the layer's output attributes to zero. The percentage of traits that are null is referred to as the "dropout rate".

The sample sizes and model settings for the train, test, and validation sets are shown in the above-drawn tables. For the trained models t with variable train and validation sizes or with an increase in train size, we present both the results with and without data augmentation for the original model. We compare the validation accuracy, accuracy, and data augmentation for the pre-trained model.

Both a larger training set and a different size validation set increase the model's accuracy.A comparison of the pre-trained model with and without data augmentation revealed no improvement in the model's accuracy or validation accuracy. Pre-trained models outperform completely created models when there is a shortage of training data.