ADVANCED MACHINE LEARNING

(BA – 64061)

ASSIGNMENT – 2

CONVOLUTION – CATS VS DOGS DATASET

DIVYA CHANDRASEKARAN

ID: 811284790

EXECUTIVE SUMMARY

Our project aims to develop a novel convolutional neural network (CNN) specifically designed for computer vision tasks, utilizing the "Dogs-vs-Cats" dataset from Kaggle as our primary data source. The challenge lies in the relatively small size of our dataset, making the task of building an effective model daunting. Convolutional neural networks have emerged as a powerful tool in computer vision, renowned for their ability to detect and learn spatial hierarchies in images. This makes them exceptionally well-suited for applications such as image recognition, segmentation, and object detection.

We believe that despite the dataset's limitations, our CNN model has the potential to perform well. This optimism stems from the inherent ability of convolutional neural networks to learn and generalize from limited data by extracting pertinent features from the images. Our strategy involves training our model on this constrained dataset, enhancing its performance through transfer learning techniques, and evaluating its effectiveness using appropriate metrics. Our goal is to develop a CNN capable of accurately and efficiently classifying images from the "Dogs-vs-Cats" dataset, despite the limited data at our disposal.

PROBLEM STATEMENT

The objective of the Cats-vs-Dogs dataset binary classification task is to determine if an image is part of the dog or cat class.

INTRODUCTION

This report delves into how the size of the training dataset and the selection of neural network architecture impact performance, specifically within the context of the Cats & Dogs scenario. We will employ methods to mitigate overfitting and enhance model efficacy, assessing outcomes on both validation and test datasets. A comparison between a network trained entirely from the ground up and one that leverages a pretrained model will be presented.

• Training a Network from Scratch:

Beginning with a dataset comprising 1000 training samples, 500 validation samples, and 500 test samples, we constructed a CNN with a configuration of three convolutional layers, two max-pooling layers, and two fully connected (dense) layers. To combat overfitting, dropout was implemented following the second dense layer, resulting in an initial accuracy of 72.1%, indicative of overfitting issues. Subsequently, strategies such as data augmentation, early stopping, and additional dropout were introduced, which, however, led to a decrease in accuracy.

• Utilizing a Pretrained Network:

We adopted the VGG16 pretrained model, adjusting it for our specific dataset by freezing the convolutional layers' weights and incorporating a dense layer of 256 units, followed by retraining. The pretrained network model significantly outperformed the scratch-built model, achieving a validation accuracy of 97.6%. Enhancements to the model with data augmentation marginally improved validation accuracy to 97.7%. Further refinement through fine-tuning delivered a test accuracy of 97.9%, showcasing the superior efficacy of utilizing a pretrained model combined with strategic fine-tuning and augmentation.

Preprocessing:

- Load the image files.
- Convert the JPEG images into RGB pixel matrices.
- Transform these matrices into floating-point tensors.
- Normalize the pixel values from their original range (0 to 255) to a new range of [0, 1], as neural networks are more efficient with smaller input values.

Data Augmentation:

To improve the precision of our model, we intend to utilize data augmentation techniques. Data augmentation generates additional training data from our existing samples through random transformations, enabling us to achieve remarkable accuracy even with a limited amount of data. This approach ensures the model is exposed to varied versions of images during training, thereby improving its ability to generalize. Our strategy includes randomly applying transformations like flipping, rotating, and scaling to the images in our training dataset. This method allows us to create varied iterations of the existing images, increasing the diversity of our dataset and consequently enhancing the robustness of our model.

The Cats-vs-Dogs dataset consists of 25,000 images, evenly split with 12,500 images per category (dogs and cats), and has a compressed size of 543MB. Once downloaded and extracted, we plan to organize it into three distinct subsets: a training set with 1,000 images for each category, a validation set with 500 images per category, and a test set with 500 images per category. Given the increased resolution and complexity of the images we're dealing with, it's necessary to enhance our neural network's capacity. To achieve this, we'll introduce an additional stage to our current architecture, which combines Conv2D layers with MaxPooling2D layers. This addition aims to boost the network's capacity and simultaneously reduce the size of the feature maps to prevent them from becoming too large by the time they reach the Flatten layer. Initially, our images are 150x150 pixels, and as they pass through the network, the feature maps gradually decrease in size until they reach 7x7 pixels right before flattening. This initial choice of image size is somewhat arbitrary but suits our specific project needs well.

Table for Model from Scratch

Model no	Train Size	Validation and Test sample size	Data Augmentation	Test Accuracy%	Validation Accuracy%
Model 1	1000	500,500	NO	76.8	70.6
Model 1a	1000	500,500	YES	67.1	64.2
Model 2	1500	500,500	NO	83	71.9
Model 2a	1500	500,500	YES	70.37	70.3
Model 2b	1500	500,500	YES	81.7	73.2
Model 2c	1500	500,500	NO	72.7	73.8

Table for Pre-Trained Models

/

Data Augmentation	Train Accuracy %	Validation Accuracy%
NO	99.6	97
YES	95.8	97.2

CONCLUSION/RESULTS

Our findings reveal that both enlarging the training dataset and leveraging a pretrained network substantially enhance model performance. Additionally, we observed that fine-tuning a pretrained network with our specific dataset led to superior outcomes compared to developing a model from the ground up.

The connection between the size of the training dataset and the selection of a network architecture indicates that larger datasets typically boost model performance. However, the architecture chosen for the network is equally critical. With smaller datasets, less complex network structures might suffice, but as the dataset grows, more intricate architectures become necessary to discern the complex patterns within the data. Pretrained networks stand out for their effectiveness in scenarios with limited data, owing to their prior training on extensive datasets, enabling them to recognize a broad array of features.