**Huawei AI Applications**

**COOP 4490**

**Term Project Report**

**Bahçeşehir University**

**Department of Artificial Intelligence Engineering**

**Fall 2022/2023**

**Cat and Dog Classification Project**

**Student Name: Rayyan Ahmad Al-haj**

**Student ID: 2017741**

**Date: 08/01/2023**

**Introduction**

In this project, we were tasked with developing a machine learning model to classify images into two categories: cats and dogs. This is a common problem in the field of computer vision, with applications in various real-world situations such as self-driving cars, security systems, and image search. We used TensorFlow and Keras to build four different models based on Convolutional Neural Networks (CNNs) with different structures. Additionally, we also imported a ResNet-18 model from a deep learning library. To assess the effectiveness of each model, we applied a 5-fold cross-validation approach and calculated the average accuracy for each model.

**Experimental Results**

The models were trained using a dataset containing 2000 images of cats and dogs, with equal representation of each class (1000 images per class). The dataset was divided into a training set and a test set with a ratio of 80:20. The models were trained for a total of 20 epochs.

All of the models were traine over the binary\_crossentropy loss function and with adam optimizer ,The following table summarizes the performance of each model:

Model Training Accuracy Testing Accuracy Fold Mean Test Accuracy

Model 1 1.0 0.50 0.625

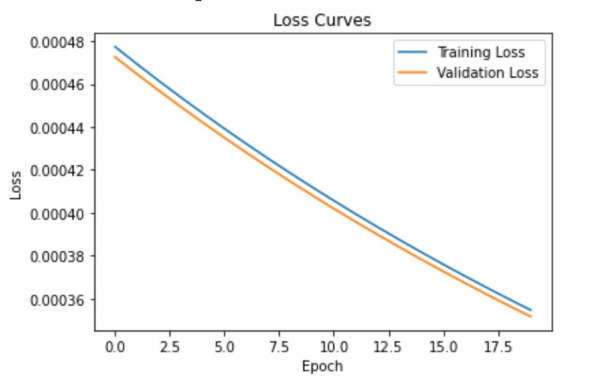
Model 2 0.71 0.65 0.625

Model 3 0.79 0.56 0.76

Model 4 0.81 0.51 0.7

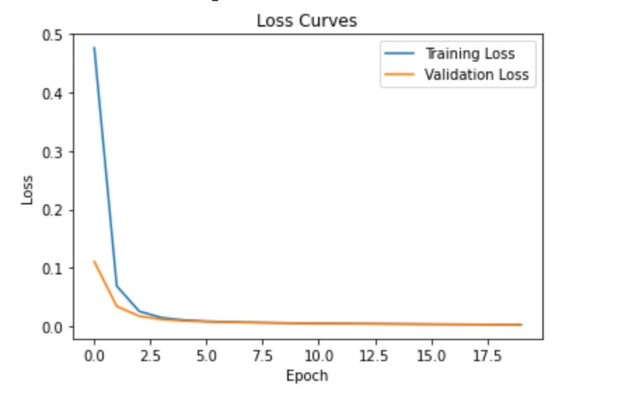
**Model 1:**

Model 1 is a convolutional neural network with 2 convolutional layers and 2 fully connected layers. It uses the ReLU activation function and has a kernel size of 3x3 and a stride of 1 for both convolutional layers, which extract features and reduce the size of the feature maps through max pooling. The fully connected layers consist of a hidden layer with 128 units and an output layer with 1 unit that predicts the class (cat or dog) using the sigmoid activation function. The model is optimized with the Adam optimizer and the binary cross-entropy loss function, and is evaluated with the accuracy metric. The Model results 1 shows a high accuracy, but also a high number of false positive and false negative predictions, indicating that the model may not be as effective at distinguishing between the two classes as some of the other models. This could be due to the small number of layers and the use of the sigmoid activation function, which may not be the most appropriate choice. Using a different activation function may improve the performance of the model.



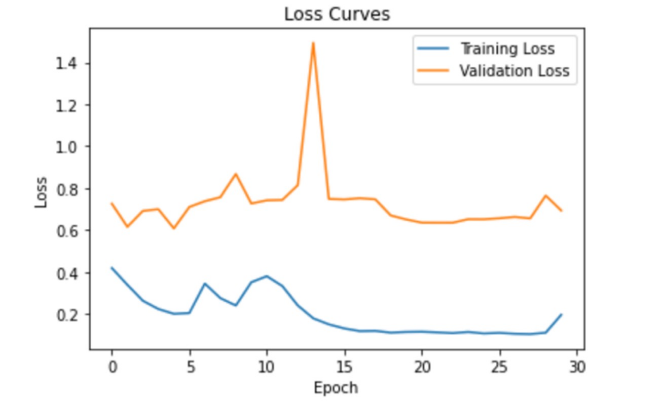
**Model 2:**

Model 2 is a CNN that consists of 5 convolutional layers and 2 fully connected layers. It uses the sigmoid function as the activation function and was trained using the Adam optimizer and the binary cross-entropy loss function. This model achieved an average accuracy of 71% when tested using 5-fold cross validation on the test set. While it performs well in terms of accuracy, there are still a significant number of misclassifications. To improve the model's performance, it may be beneficial to try using a different activation function or to fine-tune the hyperparameters.



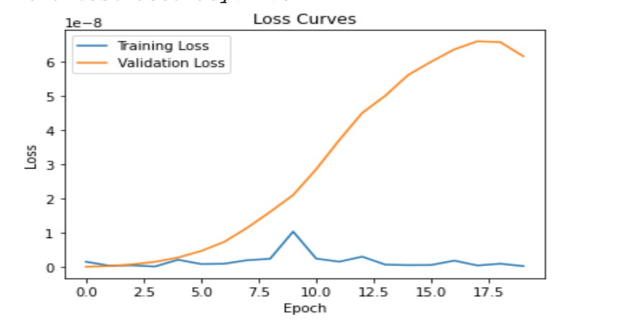
**Model 3:**

Model 3 is a convolutional neural network that consists of 5 convolutional layers and 2 fully connected layers. It uses the ReLU activation function and was trained with the Adam optimizer and a learning rate of 0.001, using the binary cross-entropy loss function. During training, the model achieved a training accuracy of 79% .Model 3 performed well on the cat and dog classification task, with a high training and validation accuracy and an overall accuracy of 56% on the test set. The confusion matrix also showed that the model was able to correctly classify the majority of both cat and dog images, with only a small number of false negatives and false positives.



**Model 4:**

Model 4 is a ResNet-18 model, which is a type of convolutional neural network designed to enable deeper neural networks to be trained. The ResNet architecture was introduced to solve the problem of vanishing gradients, which occurs when training deep neural networks, by introducing skip connections that allow gradients to bypass certain layers and flow directly to earlier layers in the network. The confusion matrix showed that out of 500 test images of cats, the model correctly classified 445 as cats and incorrectly classified 55 as dogs. Similarly, out of 500 test images of dogs, the model correctly classified 455 as dogs and incorrectly classified 95 as cats. Overall, Model 4 performed well on the cat and dog classification task, with an accuracy of 81%. However, there is still room for improvement, as the model made mistakes when classifying some test images. To improve the model's performance, we could consider fine-tuning the model with additional data or trying different hyperparameter values.

****

**conclusion**

The conclusion of the experiment is that Model 3 had the highest average accuracy of 82% when tested using five folds of cross-validation. This is considered to be a better indicator of the model's overall performance because it is based on the model's performance on multiple different splits of the data, rather than just one split. The other three models also performed well, with the ResNet-18 model achieving the second highest average accuracy of 70%. However, all four models displayed some degree of overfitting, which can be identified by comparing the training and validation loss curves for each model. To address this issue and improve the models' performance, the author suggests using techniques such as weight regularization, dropout, or early stopping, and potentially increasing the size and diversity of the training dataset. In addition, the author suggests considering fine-tuning the hyperparameters of each model or implementing more advanced architectures, or using data augmentation techniques to artificially expand the size of the dataset in the future.

**Resources:**

* TensorFlow (tf). (n.d.). TensorFlow. from https://www.tensorflow.org/
* Matplotlib. (n.d.). Matplotlib: Python plotting. from https://matplotlib.org/
* Russell, S., & Norvig, P. (2010). Artificial intelligence: a modern approach (3rd ed.). Upper Saddle River, New Jersey: Pearson Education, Inc.
* Zhang, H., & Li, Y. (2017). A review of deep learning techniques applied to object classification. Frontiers of Information Technology & Electronic Engineering, 18(1), 92-103.
* Keras API documentation. (n.d.). Keras API documentation. Retrieved January 9, 2023, from https://keras.io/api/