Convolution Neural Network

Hai Nguyen Ngoc

May 2024

1 Introduction

The labwork is devoted for Convolution Neural Network, which use VGG16 to train the animals classification model. Dataset contains 9 different folders containing the images of the animals - dog, cat, horse, chicken, cow, sheep, butterfly, . Google Colab is used for training and testing model.

2 Implementation

2.1 Pre-process data

Folder ”Raw data” is the folder which contains folders of each animals . An ImageDataGenerator is configured to perform data augmentation, which helps to improve the model's generalization by applying random transformations like rotation, shifts, shear, zoom, and flips to the images. This generator splits the data into training and validation sets, with 20% for validation.

2.2 Compile model and train

Two data generators are then created: ”train generator” for the training data and ”validation generator” for the validation data, both resizing images to 224x224 pixels and organizing them in batches of 32. Next, a VGG16 model without the top classification layer is loaded, with its layers set to non-trainable to leverage its pre-trained weights. A new classification head is added, consisting of a flatten layer, a dense layer with 512 neurons and ReLU activation, and a final dense layer with softmax activation for classifying the images into one of the 9 animal classes.

The model is compiled with the Adam optimizer (learning rate set to 0.0001), categorical cross-entropy loss, and accuracy as the performance metric. Two callbacks are defined: ModelCheckpoint to save the best model based on validation loss, and EarlyStopping to halt training if the validation loss does not improve for 10 consecutive epochs. The model is trained for up to 50 epochs, and the best model is loaded for evaluation. Finally, the model's performance is assessed on the validation set, printing the validation loss and accuracy.

3 Evaluation

- After half of number of epochs, I terminated training process because the GPU runtime of Colab reached to limited. After 10 epochs, I got the model with the accuracy 82% for validation and 80% for test data.

- Impact of number of layers: We can observe that the more layers, the more complicated model. I used VGG16 with 16 convolutional layers. So that after first epochs, we have a good accuracy. We can understand that my model is complicated enough to reflect the properties of the data. But, it still need optimize the weights to get the best one. -Impact of data: With this data, the accuracy is good from first epochs. Training process indicated that after only five first epoch, we have the accuracy is 74%, and it increase gradually after epochs. It means that VGG is working well with this data. - Impact of number of epochs: My program stopped at 15 epochs because of limitation of Google Colab. At that position, we have the accuracy 80%; as I analyzed above, VGG is working well with this data, with 80% accuracy we can observe that my model is under-fitting, It means that model need more epochs to reach to expectation.

4 Conclusion

In conclusion, I developed an image classification model using the VGG16 architecture to identify nine different animal species with an accuracy of 80%. We can observe that we can continue to train the model because It has not reached its best.