CS 583A: Course Project

Ali Erkilic

05/19/19

1 Summary

I pariticipated in an inactive competition with late submission of classifying humpback whales based on photos of their tails. The final model I choose is a convolutional neural network architecture, which takes 100×100 images as input and outputs the class labels. I was asked to predict my top 5 guesses on the data set. I implemented the convolutional neural network using Keras and run the code on a PC with the AMD Ryzen 5 2600 CPU and 16 GB memory. I had a GPU but I couldn't get it working with Tensorflow and my code was running on my CPU. Performance is evaluated on mean average precision of 5 predictions. In the private leaderboard, my score is 0.35095. On public leaderboard, my score is 0.33846. I can't see my ranking since the competition is complete and the leaderboard is finalized.

2 Problem Description

Problem. The problem is to classify humpback whales based on photos of their tails. This is a multi-class classification and image recognition problem. The competition is at `https://www.kaggle.com/c/humpback-whale-identification/`. The goal of the problem is to help scientists understand whale population dynamics around the world.

Data. The data are JPEG images. Their sizes vary. The number of training samples is $n = 25,361$. The number of classes is 5005. The training set is not well-balanced. The top class is a class called new whale which is basically an unidentified whale. There are 9664 images of new whale.

Challenges. The major challenge of the problem is the data being imbalanced. Almost half of the data is labeled as new whale and there are many classes which only has 1 image in the training set. My biggest challenge was not being able to use my GPU. I couldn't use it which made my training and tuning take so much time.

3 Solution

Model. The model I finally choose is a convolutional neural network. I used 3 conv2D layers which has filter sizes of 16x16, 32x32 and 64x64, respectively. Kernel size for all of them is 3x3. Padding is the same throughout the model. I used relu activation after conv layes followed by a batch normalization and max pooling with pool size of 3x3. This is connected to a fully connected layer. There are 2 dense layers, first one has 1024 neurons and the final one has 5005. I used dropout with 0.25 percentage before the dense layers.

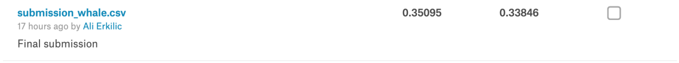
Implementation. I implemented my CNN model using Keras with TensorFlow as the backend. My code is available at `https://github.com/alierkilic/583-Project/blob/master/whale_final.ipynb`. I run the code on a PC with one AMD Ryzen2600 CPU and 16 GB memory It took around 6.5 hours to train the model.

Settings. The loss function is categorical cross-entropy. The optimizer is Adam with a learning rate of 0.001. I used several dropouts with 0.25 percentage for regularization. I used 80 epochs and a batch size of 100.

Cross-validation. I did a validation split of 20% validation and 80% training. After a couple of epochs I realized that my code was overfitting. I wanted to use VGG16 and ResNet on my data but I couldn't. My computer froze and crashed visual studio right after it went to the first epoch. So I tried to solve overfitting by increasing the data set by using data augmentation, adding dropouts and using batch normalization. My validation accuracy was constantly stuck around 0.38 On my final submission I didn't use data augmentation because it didn't help overfitting and increased my run time.

4 Outcome

I participated in an inactive competition. My score is 0.35123 in the public leaderboard and 0.35095 in the private leaderboard. My rank doesn't show on the leaderboard.



submission_whale.csv 0.35095 0.33846 ☐
17 hours ago by Ali Erkilic
Final submission

Figure 1: Score.