Join the AI revolution !

Deep Learning Course – Capstone Project

Pet Classifier using CNN

Phone : +91 9742800566

Offices : Bengaluru

Email : vivek@cellstrat.com

Web : www.cellstrat.com

Problem Statement

- You are given two sets of images namely cats and dogs
- You need to build a CNN model that classifies the images correctly
- The images are of different size and have been taken in different lighting conditions

Pre-requisites

- You need to have the followings softwares installed
  - Python 3.5, 3.6 or 3.7 (as TensorFlow works with these Python versions only)
  - TensorFlow
  - Python opencv (cv2) compatible with Python 3.5 or higher
  - Jupyter notebook

CELLSTRAT

Program & data

- Extract the ipynb file and the data in the same folder

Note on data size and runs

- A production grade program has 10,000 training images
- But for this project, we will create a small program with 20 images of cats and 20 images of dogs (training data)
- The evaluation set has 10 images of cats and 10 images of dogs (evaluation data)
- The student is expected to run approximately 100-300 training steps (A production grade code would have about 20k-50k training steps, but here we will run 100-300 steps only)

Assignment overview

•You are GIVEN the following parts of the program already :-

- Import modules (Part 1)
- Set hyper parameters (Part 2)
- Read Image data set (Part 3)
- Run the TensorFlow model (Part 4)

•You are expected to write a CNN model (between Parts 3 and 4) using TensorFlow that trains on the data and calculates the accuracy score on the test data.

•The next slide has the details of the CNN model that you are to produce.

Task to be completed:

The CNN model (cnn_model_fn) should have the following layers

- Input layer
- conv layer 1 with 32 filters of kernel size[5,5],
- pooling layer 1 with pool size[2,2] and stride 2
- conv layer 2 with 64 filters of kernel size[5,5],
- pooling layer 2 with pool size[2,2] and stride 2
- dense layer whose output size is fixed in the hyper parameter:fc_size=32
- drop out layer with droput probability 0.4
- predict the class by doing a softmax on the output of the dropout layers

Training/ Evaluation

- For training step, define the loss function and minimize it
- For evaluation step, calculate the accuracy

Reading material

For ideas look at tensorflow layers tutorial and also the CNN tutorial (under CNN tutorial, particularly look for the MNIST image classification code)

Program Runs

Run the program for 100, 200 and 300 iterations

Report on the final accuracy and the loss on the evaluation data.

Thank you

Vivek Singhal
Co-Founder and AI Data Scientist, CellStrat
9742800566
vivek@cellstrat.com