

Mini project plan

Reporter: Zhengyu Wu

Report Time: 2018-08-20

Understanding of image classification task

The overview of a image classification task consists of two steps. First, feed your model with massive training data including the matrix expressions of images along with their labels to modify the parameters of your model. Second, input your test image data and take the most possible label as the result.

What I should do is to decide the architecture of the whole model and the function of each layer. Several methods are applied to optimize the performance of the training, for example, batch normalization, regularization, and dropout. The basic CNN is organized as the form like CONV-RELU-CONV-RELU-POOL-FC. The champions of ImageNet developed much more complex models such as AlexNet and GoogleNet, which proved useful.

My plan

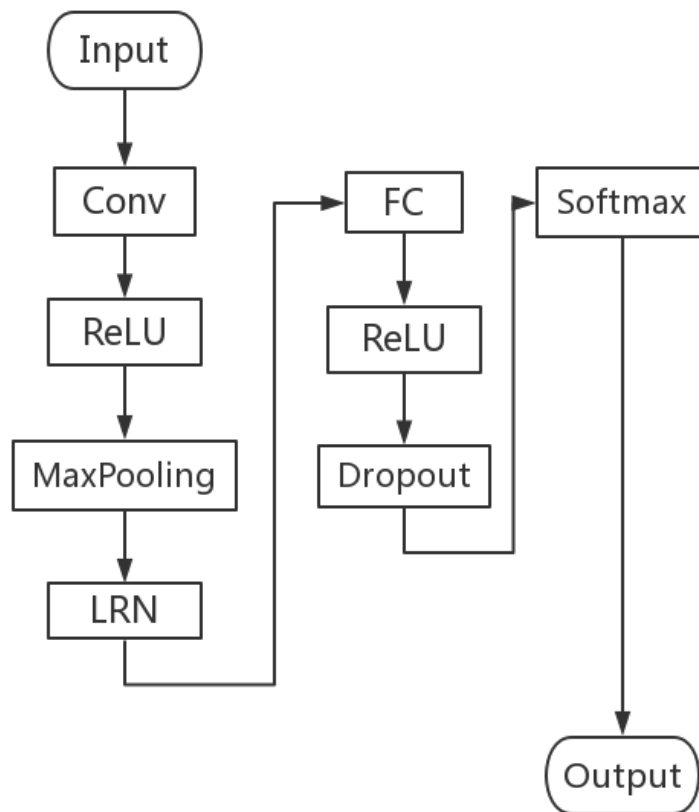
1. Input: 50000 or less training images from CIFAR.

The size of each image is 32 x 32. I'm familiar with the unpickle process to open the CIFAR image data batch.

2. Overall system flowchart

My demo model has 8 layers which includes convolution layer, activation layer, pooling layer, normalization layer, dropout layer, full connected layer, and loss layer.

The overall system flowchart is shown as the following.



Actually I don't know whether this model really works. I meant to arrange more layers like Conv-ReLU-MaxPooling-LRN, but a single image is only 32x32, which means a image's size will shrink to 8x8 after a pooling. So I just modify my design as you see now.

3. Outputs: corresponding possibilities of 10 image classes.

For example, an input image gets 30% for car, 20% for cat and so on for the rest classes. Class car has the highest possibility so this test picture is estimated as a car.

As far as I know, in some CV contests, the final answer will be regarded as true if the 3 most possible classes contains the true label.

My questions

1. Is my demo model feasible? Or this question only later experiments can tell?
2. Did the step 1 of the mini project require I to implement this model with TensorFlow? Or directly finish the AlexNet mentioned in ther step 2?