Assignment 2 - CS747

Hoang Dung Bui
NetID: G01301478
Team name on Kaggle Leaderboard: Hoang-Dung Bui

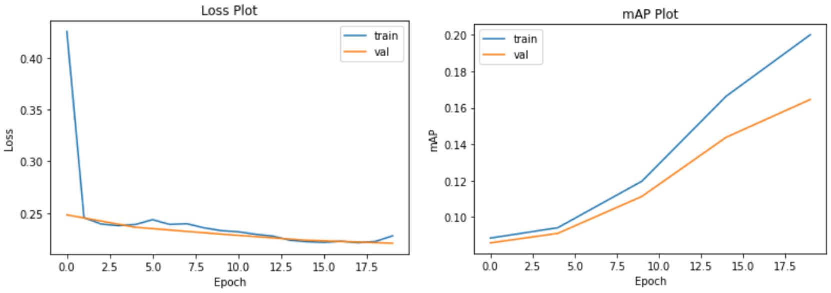
October 6, 2021

1 Pre-designed network for multi-label classification

The result of pre-designed network for multi-label classification:

1. Simple classifier

   - Report test mAP for simple classifier: 0.159
   - Visualize loss and mAP plots:



(a) Loss of the simple classifier (b) mAP of the simple classifier

Figure 1: Loss and mAP of Simple Classifiers

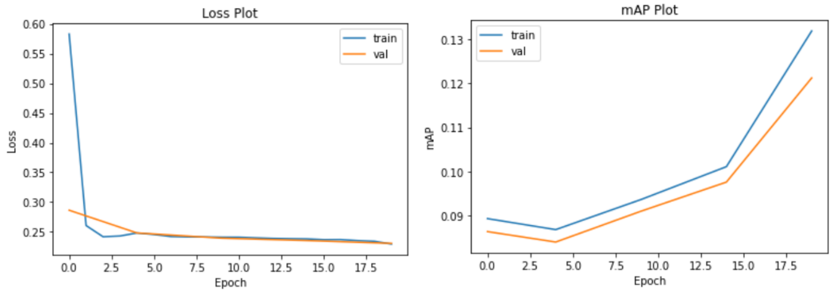
   - Analysis of the plots: From the Loss plots, after training the loss value reduces sharply at the several first epoch, and reduce gradually to the last epoch. It is the normal behavior of Neural Network after training. For the mAP plots, the mAP values increase gradually for both training data and evaluate data. However, at some last training epoch, the difference between training data and evaluating data is wider, and the mAP for training data goes up much faster. I think it is the sign of overfit.

2. AlexNet from Scratch

   - Report test mAP for alexnet: 0.1175 (with SGD optimizer)
   - Visualize loss and mAP plots
   - Comments for Alexnet Training from scratch: The behaviors of both loss and mAP functions for both training and evaluating data are normal (losses decrease and mAP increase gradually after training). However, the training process are slow the because the mAP increases too slow. I think that is because of the SGD optimizer. Adam optimizer is a better choice to speed up the learning process.

3. Pretrained AlexNet

   - Report test mAP for pretrained alexnet: 0.678
   - Visualize loss and mAP plots:
   - Provide analysis on differences to training from scratch (at least 3 sentences):



(a) Loss of the AlexNet from Scratch (b) mAP of the AlexNet from Scratch

Figure 2: Loss and mAP of AlexNet from Scratch

2 Self designed network for multi-label classification

Did you upload final CSV file on Kaggle: Yes

1. My best mAP on Kaggle: 0.5782
2. Factors which helped improve my model: block layers with residual layer, Adam Optimizer, kernel size

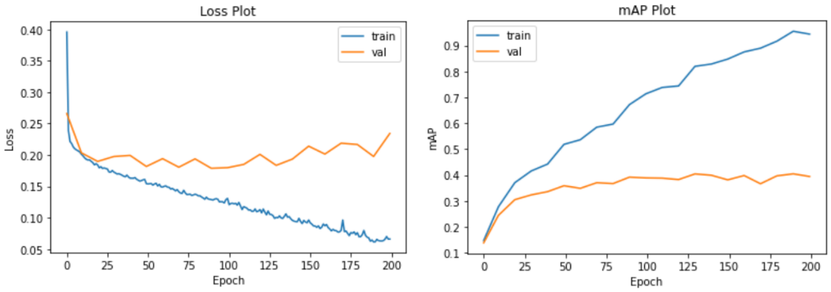
Table of final architecture

Layer N.	Layer Type	Kernel size	Input|Output Dimension	Input|Output Channels
1	conv2d	9	227 | 114	3|64
2	BatchNorm2d	-	114 | 114	64 | 64
3	relu	-	-	-
4	maxpool2d	2	114 | 57	-
5	conv2d	7	57 | 57	64|128
6	BatchNorm2d	-	57 | 57	128 | 128
7	relu	-	-	-
8	conv2d	5	57 | 57	64|128
9	BatchNorm2d	-	57 | 57	128 | 128
10	relu	-	-	-
11	conv2d	7	57 | 57	128 | 128
12	BatchNorm2d	-	113 | 113	128 | 128
13	relu	-	-	-
14	conv2d	5	57 | 57	128 | 128
15	BatchNorm2d	-	57 | 57	128 | 128
16	relu			
17	conv2d	5	57 | 57	3|64
18	BatchNorm2d	-	57 | 57	128 | 128
19	relu	-	-	-
20	Adaptive AvgPool2D			128 | 1
21	linear	-	64 | 21	-

Table 1: Neural Network Structure

There are some problems with my network:

- The maximum mAP it reached is only 0.405. If the network complexity is increased, the the overfit phenomenon happen.
- I have tried multiple way from Batch normalization for conv layers, dropout for linear layers, changing the kernel size, stride, but the overfit still occurs. The loss and mAP are plot as following:



(a) Loss of the designed Neural Network

(b) mAP of of the designed Neural Network

Figure 3: Loss and mAP of of the designed Neural Network