DLCV HW4

R09521603

January 5, 2022

1 Problem 1

1.1 Prototypical Network Architecture and Accuracy

a Model Architecture

First of all, i think the most important part of prototypical network is how we implement the N-way K-shot training algorithm. For the Dataset class of mini data is just similar as dataset of normal classification task. The magic part is in the class of dataloader Sampler. In the initialization part of Data Sampler, i first sort all the image with same label. In the sampling time, i randomly choose N classes, and randomly choose K shots image. In the training part i choose 15 image as query data just as testing set. The training episodes is also same as testing with 600 episodes. As the distance function i only use Euclidean Function to get the baseline accuracy. And at the training phase, i use the tips TA told and train the model with 10 way 1 shot which is much difficult than 5 way 1 shot in testing time. As the data augmentation part i didn't do too much effort but just only normalized the image. And we set the learning rate as 0.0001 and use the Adam optimizer.

b Accuracy Report

Accuracy for validation data : 42.78 +- 0.79

1.2 Different Distance Function

a Graph and Chart

The following graph is the training and validation curve of three types of distance function

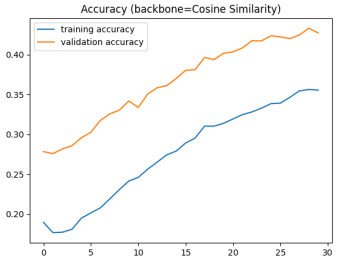

Figure 1: Euclidean

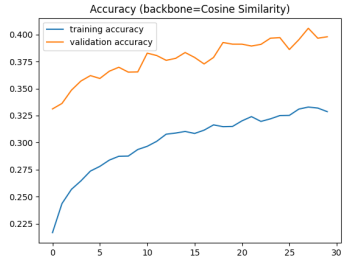

Figure 2: Cosine_Similarity

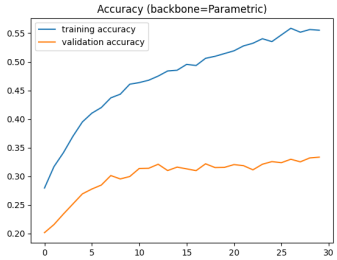

Figure 3: Parametric

The follow chart is the best Validation Accuracy of each distance function

Model Name	Validation Accuracy
Euclidean	0.4331
Cosine Similarity	0.4057
Parametric	0.3334

b Analysis

As the graph ans chart point out, the best model of mine is using Euclidean distance function. But there's a little different of Parametric model. As i didn't come up with better idea, the parametric model is trained using only 5 way yet the other two models are trained with more difficult 10 way data as TA hint in the ppt. Yet using this setup, as the chart point out, the accuracy is lower than two other types of distance function. So i think if i can come up with more robust way to train the parametric model with 10 way the accuracy should be close to other two. Though i didn't try many different parametric parameter, the graph has shown that parametric model is prone to overfitting compare to other two. I think maybe i should add more regularizer to overcome this problem. As the Euclidean distance and Cosine_Similarity the graph have shown that Euclidean distance model can still be trained to gain better result but the curve of cosine similarity has began to slow down. For my parametric model, i concatenate each query data feature with other 5 way data features, and using a MLP with input shape of (N_way+1)*feature_dimension and output a N_way class. I only use one hidden layer and with ReLU activation. I think i can finetune the parameter to receive more accurate result but i also think the euclidean can do well one this task.

1.3 Different K Size

a Graph

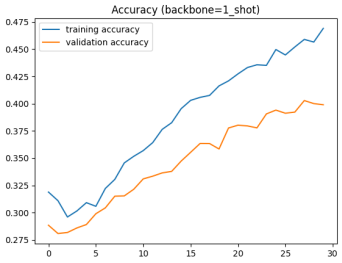

Figure 4: K = 1

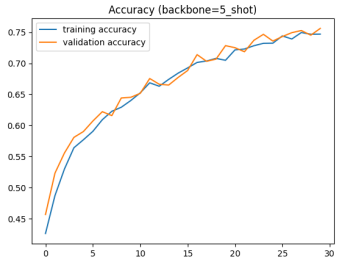

Figure 5: K = 5

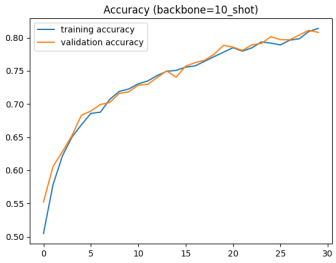

Figure 6: K = 10

b Analysis

As the curve has shown, when the K start to increase the accuracy of validation is getting much higher, which make sense since we get more training data. And as the graph shown when the k increase the train and validation curve is getting more and more closer.

2 Problem 2

2.1 Model Architecture

For SSL i directly use BYOL model. Normalize the image into 128*128. And i use the BYOL default setting to train the pretrained restnet. And i use the default learning rate used by BYOL which is 3e-4. And train it for 200 epoch. My batch size is 128 which is much higher than default. And i only used Adam optimizer without special learning rate scheduler. And i train my model without training last fc layer of restnet50 since i think it can be trained much faster.

2.2 Different Setting

Setting	Validation Accuracy
A	0.19996
B	0.25656
C	0.37041
D	0.29720
E	0.33785

2.3 Analysis

For all the five different setting, i trained all the model with batch size 32 and learning rate 0.001. For the downstream model, i only use two hidden layer. For my experiments, as the layer increase the accuracy will drop fast. So i constrains my layer with small number. As the table point out, the setting of A has lowest accuracy since the model haven't pretrained on any data. And other observation is that when frozen the backbone model, the accuracy will be lower than train the backbone together. I think the reason of this is that although our training data is small, training the backbone on data still can learn more adaptive feature on new data. And compared to TA's model, with or without frozen backbone give similar result, that is, using my w/o label pretrained model we usually get better result. I guess the reason might be that when training with label data, model tend to overfit onto the training data instead of extract more meaningful feature of image.

3 Collaborator

r09521607 r09521601

References