Deep Learning for Computer Vision – HW#4

宋體淮, R09921135, Electrical Engineering

Problem 1: Prototypical Network

1. **Describe the architecture & implementation details of your model. And please report the accuracy on validation set under 5-way 1-shot setting.**

Accuracy
44.11%

- Describe the architecture & implementation details of your model:

# of training episodes	100
# of training epochs	200
Distance function	Euclidean
Learning rate scheduler	StepLR
Data augmentation	None
Optimizer	Adam
N-way K-shot setting for meta-train	10-way 1-shot
N-way K-shot setting for meta-test	5-way 1-shot

2. **Please report and discuss the accuracy of the prototypical network using 3 different distance function (i.e., Euclidean distance, cosine similarity and parametric function). You should also describe how you design your parametric function.**

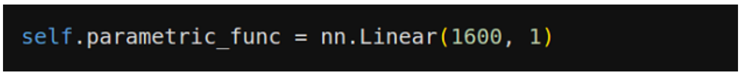
Euclidean	cosine similarity	parametric function
40.00%	23.22%	40.97%

- In this problem, all of the three scenarios are trained under 5-way 1-shot setting.

- Describe how you design your parametric function**:**

— The input of the function is $abs(\mathrm{a}-\mathrm{b})$.

— The function architecture is one linear layer.



```
self.parametric_func = nn.Linear(1600, 1)
```

- discuss the accuracy of the prototypical network using 3 different distance function:
  — In my opinion, why parametric function performs better is because the learnable network will decide a proper distance value on its own based on prediction results. In this way, it has more flexibility when deciding how close the two vectors are.

3. Please report and compare the accuracy with different shots. (K=1, 5, 10).

K=1	K=5	K=10
41.59%	60.62%	65.90%

- In this problem, all of the three scenarios are trained using parametric function as distance function.

- Compare the accuracy with different shots:
  — It meets my expectation that larger K has better results, because when using more support set, the prototype would be more reliable for the query set to compare.

Problem 2: Self-Supervised Pre-training for Image Classification

1. Describe the implementation details of your SSL method for pre-training the ResNet50 backbone.

SSL method	BYOL
Data augmentation	ColorJitter, RandomGrayscale, RandomHorizontalFlip, GaussianBlur, RandomResizedCrop
learning rate schedule	None
optimizer	Adam

batch size	128

- In my opinion, SSL is sensitive to learning rate. The results on the downstream task vary significantly with different leaning rates.

- I also notice that the pre-training loss does not reflect the pre-training effectiveness. If loss is surprisingly low, you should suspect that mode collapse occurs.

2. **Please conduct the Image classification on Office-Home dataset as the downstream task for your SSL method. Also, please complete the following Table.**

Setting	Classification accuracy on valid set
A	0.1962
B	0.3303
C	**0.4039**
D	0.3228
E	0.2031

- All of the settings are trained for the same number of epochs.

3. **Discuss or analyze the results in Problem 2-2.**

- The worst result happens in the setting A which is expectable, because it is trained from scratch and does not have any pre-knowledge.

- The settings other than using SSL are prone to over-fit on the training set, consequently leading to inferior results on the validation set. Because SSL does not learn absolute categories, it learns the reasoning process or the underlying knowledge instead.

- The settings of fixing the backbone have a little bit worse results than training full model, because they do not see the downstream dataset. However, they are less easy to overfitting.