HW4report

r10922061 資工所 王竑睿

Problem 1: Prototypical Network (70%)

1. (20%) Describe the architecture & implementation details of your model. (Include but not limited to the number of training episodes, distance function, learning rate schedule, data augmentation, optimizer, and N-way K-shot setting for meta-train and meta-test phase)
Please report the accuracy on validation set under 5-way 1-shot setting (during inference).

1. training episodes: 600
   distance function: Euclidean Distance
   learning rate schedule: lr=1e-3
   optimizer: Adam
   meta-train & meta-test setting: 5 ways 1 shot
   data augmentation:

```
self.transform = transforms.Compose([
    lambda x: Image.open(x),
    transforms.ToTensor(),
    transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
])
```

2. model architecture

```
class Convnet(nn.Module):
    def __init__(self, in_channels=3, hid_channels=64, out_channels=64):
        super().__init__()
        self.encoder = nn.Sequential(
            conv_block(in_channels, hid_channels),
            conv_block(hid_channels, hid_channels),
            conv_block(hid_channels, hid_channels),
            conv_block(hid_channels, out_channels)
        )
```

```
    def forward(self, x):
        x = self.encoder(x)
        return x.view(x.shape[0], -1)

def conv_block(in_channels, out_channels):
    bn = nn.BatchNorm2d(out_channels)
    return nn.Sequential(
        nn.Conv2d(in_channels, out_channels, kernel_size=3, padding=1),
        bn,
        nn.ReLU(),
        nn.MaxPool2d(2)
    )
```

3. Accuracy on validation set under 5-way 1-shot setting
    a. **accuracy 47.14 +- 0.83 %**

2. (20%) When meta-train and meta-test under the same 5-way 1-shot setting, 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.

Distance Function	Accuracy
Euclidean Distance	0.4496
Cosine Similarity	0.3654
Linear Network	0.3591

- design your parametric function
    a. parametric function 是將 Q 個 1600 維的 query samples 與 P 個 1600 維的 prototypes concatenate 成 $Q \times P$ 個 3200 維的 pair 之後，以四層 linear 將 pair project 成 $Q \times P$ 個1 維的 distance。
    b. 以下為 parametric function 的架構

```
RelationNet(
  (encoder): Sequential(
    (0): Sequential(
      (0): Conv2d(3200, 2560, kernel_size=(1, 1), stride=(1, 1))
      (1): BatchNorm2d(2560, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (2): ReLU()
```

```
    )
    (1): Sequential(
      (0): Conv2d(2560, 1280, kernel_size=(1, 1), stride=(1, 1))
      (1): BatchNorm2d(1280, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (2): ReLU()
    )
    (2): Sequential(
      (0): Conv2d(1280, 640, kernel_size=(1, 1), stride=(1, 1))
      (1): BatchNorm2d(640, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (2): ReLU()
    )
    (3): Sequential(
      (0): Conv2d(640, 1, kernel_size=(1, 1), stride=(1, 1))
      (1): BatchNorm2d(1, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (2): ReLU()
    )
  )
)
```

c. 由結果可以發現 parametric function 的 performance 不如 cosine similarity 以及 Euclidean distance 好，很可能是因為 parametric function 有較多的參數，需要較久的 training 才能夠收斂。上述結果為 40 epoches 得到的。

3. (10%) When meta-train and meta-test under the same 5-way K-shot setting, please report and compare the accuracy with different shots. (K=1, 5, 10)

- 以下結果為訓練 40 epoches 得到

K-shot	Accuracy
1	0.4496
5	0.6162
10	0.6656

Problem 2: Self-Supervised Pre-training for Image Classification (50%)

1. (10%) Describe the implementation details of your SSL method for pre-training the ResNet50 backbone. (Include but not limited to the name of the SSL method you used, data augmentation for SSL, learning rate schedule,

optimizer, and batch size setting for this pre-training phase)

1. 本次作業使用 BYOL Self-Supervised Learning method
2. data augmentation

```
self.transform = transforms.Compose([
    lambda x: Image.open(x),
    transforms.Resize(size=128),
    transforms.RandomRotation(degrees=15),
    transforms.RandomHorizontalFlip(),
    transforms.ToTensor(),
    transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
])
```

3. learning rate schedule: lr = 3e-4
4. optimizer: Adam
5. batch size: 32

2. (10%) Following Problem 2-1, please conduct the Image classification on Office-Home dataset as the downstream task for your SSL method. Also, please complete the following Table, which contains different image classification setting, and compare the results.

Setting	Pre-training (Mini-ImageNet)	Fine-tuning (Office-Home dataset)	Classification accuracy on valid set (Office-Home dataset)
A	-	Train full model (backbone + classifier)	156/406=0.3842
B	w/ label	Train full model (backbone + classifier)	196/406=0.4828
C	w/o label	Train full model (backbone + classifier)	**174/406=0.4286**
D	w/ label	Fix the backbone. Train classifier only	122/406=0.3005
E	w/o label	Fix the backbone. Train classifier only	70/406=0.1724

3. (10%) Discuss or analyze the results in Problem 2-2

1. train full model 的表現普遍比 fix backbone，只 train classifier 來得好。
2. 同樣都是 train full model 下，pretrain with label > pretrain without label > no pretrain
3. 同樣都是 train classifier only 下，pretrain with label > pretrain without label
4. 即使是在不同的 dataset 上作 pretrain，self-supervised pretrain 或是 supervised pretrain 依然能夠提升 finetune 後的表現。