Assignment (5)

24789 - Intermediate Deep Learning (Spring 2023)

Out Date: 2023/4/19 (Wed)
Due Date: 2023/4/27 (Thu) @ 11:59 pm EST

Submission file structure

/{andrewID}-HW5
report.pdf
/hw5
hw5_run.ipynb (all the Python script files)
simclr_lin_best.pt

Please convert the ipynb file as a pdf and submit **hw5_run.pdf** to **Homework 5** section, and the zip file with the contents mentioned above to **Homework 5 programming** section.

You can refer to Python3 tutorial, Numpy documentation and PyTorch documentation while working on this assignment. Any deviations from the submission structure shown below would attract penalty to the assignment score. Please use Piazza for any questions on the assignment.

Programming Exercises (50 points)

PROBLEM

Contrastive Learning Using SimCLR (50 points)

In this assignment, you are asked to train a SimCLR model for contrastive learning on the CIFAR-10 dataset using the provided starter code. The SimCLR model is a self-supervised learning framework that learns visual representations by contrasting positive and negative pairs of augmented images.

1a) Complete the implementation of the ***nt_xent*** loss function, which computes the normalized temperature-scaled cross entropy loss for SimCLR. Given below is the equation for the loss function:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_k\right)/\tau\right)} \tag{1}$$

1b) Complete the implementation of the ***barlow_twins*** loss function, which computes the normalized temperature-scaled cross entropy loss for SimCLR. Given below is the equation for the loss function:

Cross correlation matrix

$$\mathcal{C}_{ij} \triangleq \frac{\sum_b z^A_{b,i} z^B_{b,j}}{\sqrt{\sum_b \left(z^A_{b,i}\right)^2} \sqrt{\sum_b \left(z^B_{b,j}\right)^2}} \tag{2}$$

Barlow Twins loss

$$\mathcal{L}_{\mathcal{BT}} \triangleq \sum_i (1 - \mathcal{C}_{ii})^2 + \lambda \sum_i \sum_{j \neq i} \mathcal{C}_{ij}^2 \tag{3}$$

2) Define the forward function for the SimCLR model. It uses resnet18 as the base encoder. The forward function should return both the output from the encoder and the output from the projection head

3) Define the transformations, train dataset and dataloader, model, optimizer, scheduler, and parameters required for training

4) Fill in the missing parts in train function and pass in the appropriate hyperparameters. Train the model and save ***simclr_best_epoch.pt***. You will need it for the fine-tuning. Training each epoch takes about 90 seconds.

5) Define the train and test transformations, train and test dataset and dataloader, model, optimizer, scheduler, and parameters required for finetuning

6) Run the fine-tuning and save ***simclr_lin_best.pt*** and plot the train and test loss and accuracies wrt. epochs and add the plots in your report.

A template code and datasets are been given for you to start (you can download them from **Homework 5** section on **Canvas**).

In your submission to Gradescope, you should submit your code and your best model after finetuning and the report should be the pdf version of hw5_run.ipynb. Make sure you include the following details in your report:

- train and test plots for accuracies and loss for fine-tuning
- the hyperparameters for both training and fine-tuning
- data augmentations used for both training and fine-tuning

- the final test and train accuracies after fine-tuning

To get full credits, the final accuracy on **test set** should be more than 75%. You will get 5 bonus credits if the **accuracy** on the test set is more than 85%.
