COMP3314 Machine Learning

Programming Assignment 2: Convolutional Neural Networks

Release date: April 12, 2021
Due date: 11:59pm, May 9, 2021
(20% deduction for every 24 hours after the due date)

Task:

This assignment is about the design and implementation of a convolutional neural network for the task of digit recognition using the PyTorch framework. Your network design should at least include convolutional layers, fully-connected layers, activation layersand max-pooling layers. A code template is provided to facilitate the implementation.

Datasets:

This is a 10-category classification problem which classifies digit images into 0-9. The training set contains 3,000 samples for each class and the testing set contains 500 samples for each class. Each input image has three channels and each channel contains 32*32 pixels.

Guidelines:

[1] The core idea of this assignment is to give you an overall picture of convolutional neural networks. You are not required to implement the details of forward and backward passes of different types of neural network layers (such as matrix operations). Instead, you can use the existing interface in PyTorch (a popular framework for deep learning) to help build a simple image classifier.

[2] Students should first learn how to use PyTorch to implement a neural network. An official 60 minute blitz (https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html) will greatly help you to understand the basic components of PyTorch. Python is the default programming language and other frameworks besides PyTorch are prohibited in this assignment.

[3] In the provided template, you need to finish 3 tasks: building a convolutional neural network, implementing training and testing procedures and adjusting hyper-parameters such as initial learning rate, decay strategy and the number of training steps. Besides, you can choose to change any parts in the template (such as the optimizer, data augmentation and normalization strategies) for better training and testing results.

[4] Report writing: The report should contain the detailed results of your implementation: overall testing accuracy, accuracy of each class (0-9) and results for different experimental settings. Students are encouraged to assess the model of its over-fitting condition, its convergence of training, error rate, etc. Note that your program should be optimized to achieve an overall accuracy higher than 70% (the higher the better).

[5] Design a novel network architecture with network depth <= 20. A very deep network will inevitably increase the running time. You should record your computer configuration and the running time of your program in the report. The overall running time should be less than 45 minutes (including training and testing stages). GPU is not recommended in this assignment as the model already runs quite fast on CPUs. Existing network architectures, including LeNet and AlexNet, can NOT be used.

Submission Instructions:

[1] One report in **pdf** including the design, implementation and discussion about the model.

[2] Source codes packed in **zip** format that can be unzipped and compiled. Attach trained modelfile linkage and remove trained models in case the zipped file exceeds the file size limit.