

CS5260 Assignment 6

-- Colossalai and LR Range Test

I. Introduction

Colossalai is a distributed training framework developed by Prof Yang You's team. We will take use of it in this assignment, but code frame is given so you don't have to dive deep. It can enable different kind of distributed training easily (including data parallel, tensor parallel, pipeline parallel and so on) and provides efficient memory handling.

Learning rate range test is a method proposed by Leslie N. Smith in paper *Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates*. The idea is: before normally training your model to convergence, first train your model with exponentially increasing learning rate (increase with batch step) for several epochs, and observe the loss curve. Usually the loss will first keep unchanged, and then go down, and finally explode. The learning rate corresponding to loss going down will be usable LR, and explosion part is not usable. Depending on your chosen optimizer and learning rate scheduling, this method can give you reference for LR choice region, and for some of the settings this method can serve as a super fast tuning method with high performance.

In this assignment, a file named **Colossalai_lr_range_test.ipynb** will be given, and it contains the code frame for LR range test part.

II. Tasks

In this assignment, we will study the use of LR range test, using Colossalai framework. The chosen model and dataset are LeNet5 and MNIST. You will be asked to:

1. Choose one **optimizer** from SGD, ADAM, ADAMW, RADAM, LARS, LAMB or other optimizer you are interested in. (Note: AdaGrad is not supposed to work with this method).
2. Choose two **learning rate scheduling method** from Pytorch library. (Including no scheduling) Some possible choice: Multistep, OneCycle
3. Take use of **Colossalai_lr_range_test.ipynb** to conduct LR range test for optimizer you chose. Propose several learning rates for real training.
4. Write your code (either adapt provided code or start form scratch) to train LeNet5 on MNIST with two learning rate scheduling methods you choose and proposed learning rate. A suggested epoch number is 30.
5. Observe the result, and write a brief report about what you find (optimizer and scheduling method you choose, corresponding LR region on LR range test plot. docx, pdf, ipynb are all acceptable). Remember to save necessary images or data during experiments for report writing.
6. Upload your work including **requirement.txt** to your github. Add the **github link** to your **report**.

III. Submission

Please submit the report to Luminus -> files -> assignments -> submission -> assignment 6 by 23:59 April 1st. Late submission will have penalty.

You will be given bonus (capped at full score of all 6 assignments) if you do extra work (test more optimizers and scheduling) and find interesting conclusion / give good explanation.