ESE-5390 Final Project

Before the end of this semester, students will work in groups to complete a project in hardware/software co-design for deep learning. Students can choose to complete a project in one of the following two categories:

- **Category 1:** Original research in hardware/software co-design for deep neural networks
- **Category 2:** Reproduce results found in one or more published paper discussed in this course (See selection guidelines below).

You will be graded on how well you define your problem, survey previous work, (re)produce the engineering artifacts, conduct experiments, and present your results. The goal to aim for is either a workshop paper (for **Category 1**) or a comprehensive technical report (for **Category 2**).

Example(s) for Category 1 Project

- Evaluate different CPU/GPU BLAS library backends for PyTorch Conv2D computation. You may write a C++/Rust extension for `torch.nn.functional.conv2d` that utilizes the SIMD Intrinsics (such as AVX2) that accelerates Conv2D, or uses C/CUDA libraries such as GraphBLAS and CuBLAS.
- Evaluating TensorRT and `torch.quantization` for quantization of DNNs.

Example(s) for Category 2 Project

- Reproduce different fast algorithms for convolution (Winograd, FFT) as shown in papers "*Fast Algorithms for Convolutional Neural Networks*" and "*Fast Training of Convolutional Networks through FFTs*", integrate them with your PyTorch models, and evaluate them in training/inference.

I. General Guidelines

Regardless of original research or reproducing results from previous paper, there are several criteria that all projects must follow:

- The final project should be done in groups of up to 3 students. Projects done with fewer members will NOT receive additional credit. Please distribute the work equally among teammates, as all members of a group will receive the same grade.
- Experiments should be done primarily on neural network models written in PyTorch framework.
- For **Category 1** projects, you may brainstorm topics in:
  a. Hardware techniques for accelerating neural network computations.
     e.g. SIMD, numerical precision,
  b. Software integration of existing hardware (e.g. GPUs/TPUs) for deep learning.
- For **Category 2** projects, you may choose among these following papers:
  a. Deep learning with limited numerical precision

b. BinaryConnect: Training Deep Neural Networks with binary weights during propagations
c. Fast Algorithms for Convolutional Neural Networks
d. Fast Training of Convolutional Networks through FFTs
e. Channel Pruning for Accelerating Very Deep Neural Networks
f. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks
g. Speeding up Convolutional Neural Networks with Low Rank Expansions
h. Learning Structured Sparsity in Deep Neural Networks
i. Other relevant papers approved by course staff (before proposal due date)

- Due to the diversity of the project ideas, the course staff will only be able to help with high-level discussions about the project, such as general engineering directions.
- Although we still recommend using the Google Colab environment for the final project, you are not restricted to conduct experiments/development on the Colab platform. Please discuss with the TA before Nov.21st if you plan use your own environments.
- You must ensure that course staff are able to reproduce your work within reasonable effort. Using Colab to generate an ipynb file is highly encouraged, as this is the most convenient way for the course staff to evaluate and reproduce your work.
- Please follow the academic integrity guidelines outlined in Lecture 1. We do not tolerate plagiarism both from online resources (including direct copy from open-source repositories) and between groups.

II. Grading

The final project will consist of the following gradable items:

Project Proposal (10%)

Each group must submit their project proposal on **Nov. 9th**. Proposals should be one to two pages long plus references. They should include:

- A description of your topic
- A statement of why you think the topic is interesting or important
- A description of the methods you will use to evaluate your ideas (for Category 1)
- References to at least three papers you have obtained and read.

Project Report (40%)

Your final report should contain no more than 10 pages, excluding references and optional appendix. The report should have the following sections:

- **Abstract:** In less than 200 words
- **Introduction:** Include background and motivation
- **Methodology:** Describe your engineering effort here
- **Evaluation:** Include necessary figures and comparison
- **Conclusion and Discussion**: Discuss lessons learnt
- **Optional Appendix** (no more than 4 pages)

You must use the ACM Master Article Template.
We suggest using LATEX and collaborate on Overleaf.

Artifacts and Engineering Sophistication (30%)

You will submit your project source code or link to your repository on canvas before the due date. If you submit a link to your repository, please make sure that the course staff have access to the repository and tag the commit intended for submission. By default, we will consider the last commit before the due time for evaluation.

In your submission, you should include README.md to help TAs run your code and reproduce your results. You must ensure that the results claimed in your project report are reproducible by the course staff.

Final Presentation (20%)

Groups will take turns to present their project in the last two lectures of the course. Each group will have 10-12 minutes to present.

All group members should deliver part of the talk. The talk should give highlights of the final report, including the problem, motivation, results, conclusions, and possible future work.

The course staff will apply the same grading rubric as the paper presentations (excluding the Q&A requirement). Time limits will be strictly enforced to let everyone present. Please practice your talk to see how long it is. We also suggest that you have a plan for what slides to skip if you get behind.

III. Important Dates

Date	Item Due
Nov.9th	Group formation and Project Proposal (1-2 pages)*
Nov.21st	Project meeting with TA to check for progress
Dec.7th	Project Presentation Slides
Dec.7th & Dec 12th	Project Presentation
Dec 17th	Project Report and Code*
* Late days applicable	

Late Day Policy for Final Project

Each student has 5 free “late days” to use on labs and projects during the semester.

For the final projects, the number of late days allowed to a group is calculated by the **maximum** of the number of unused late days owned by each group member. For example, if student A has 3 late days, and student B has 4 late days, the group formed by A and B has 4 late days.

Late days are applicable to the Project Proposal due (Nov.9th) and the Code/Project report due (Dec.7th), but NOT for project presentations and slides.