ECE 379K: Machine Learning and Data Analytics for Edge AI
Project Description (Milestones 1, 2, and 3)

Released: Apr 8, 2021

Introduction

In this project, you will experiment with several *model compression* techniques to optimize various performance metrics for inference on the edge devices. You will also participate in a *model compression competition* that will bring bonus points to the winners. You are allowed to use all the software tools mentioned in previous homeworks.

Setup:

- Application: Image classification on CIFAR-10 dataset
- Neural network: MobileNet-v1 (pre-trained model given in HW3)
- Training Framework: PyTorch or TensorFlow (you need to use only one of them)
- Deployment framework: ONNX or TensorFlow Lite (you need to use only one of them)
- Training platform: TACC GTX GPUs
- Edge devices: Raspberry Pi 3B+ and Smart Power 2 device (one set per team)

M1 [35 points] Structural Pruning DUE: Apr 19, 12 noon (CST)

In Milestone 1 (M1), you are required to compress the model using the *structural pruning* technique introduced in lecture 10. More precisely, you are required to do the following:

- Implement the *L1-norm* based structural pruning technique based on your selected framework for MobileNet-v1. Specifically, you need to implement the following pruning function: `channel_fraction_pruning(model, fraction)`. This function prunes a certain fraction of output channels for each convolutional layer of the given model; e.g., if `fraction` = 0.2, for each convolutional layer, then 20% of output channels with the lowest L1 norm will be pruned.
- Explore different pruning parameters (e.g., pruning fraction, training epochs for fine-tuning) and report the obtained test accuracy. The following 5 parameters are required: *pruning fraction*=[0.05, 0.25, 0.5, 0.75, 0.9] with *fine-tuning epochs*=[0,3,5]. Besides these pruning parameters, you can explore as many combinations as you want (i.e., there is no upper limit).
- Use the given function `remove_channel(model)` to not only mask some weights to '0', but actually *remove* the pruned channels for MobileNet-v1 and change the structure of the model itself.
- Deploy the pruned network over the entire test set (available in ***HW3_files/test_deployment***) with different pruning parameters on Raspberry Pi and complete ***Table 1***:

Table 1

Pruning parameters	Maximum memory usage [MB]	Average latency for one image [ms]	Maximum power consumption [W]	Average energy consumption for one image [mJ]

Note: The Latency and Energy Consumption values need to be the average values over the *entire* test dataset. You should also refer to HW3 to measure the inference latency correctly. More

precisely, for ONNX you should only measure the time for ***sess.run()*** (the actual inference), while for TF Lite you should only measure the time for ***interpreter.invoke()***.

- Record a video clip presenting your experimental results and observations; upload the video to Canvas based on the Submission guidelines below.

Guidelines of the presentation and video recording:

- The M1 project presentation consists of **up to five slides** (this limit is strictly enforced). Your presentation should provide the following information:
    - Cover page: Project title and group members.
    - Approach (1-2 slides): Explain the main ideas/tasks of your approach.
    - Experimental results (1-2 slides): Show and discuss your results (e.g., model size/test accuracy tradeoffs, power/performance tradeoffs, etc.).
    - Conclusion (1 slide)
- Students should *pre-record* a **video presentation** in one of the following formats: AVI, MOV, mp4, MPEG, and WMV. We suggest this software to record the video, but other packages may also be used (you can even start a Zoom meeting and record it). Each video can last maximum **four minutes** (this limit is strictly enforced). Show the demo in full screen and the presenters in a small window on the screen.

Related paper: Structural pruning

Submission Instructions/Hints:

- Include the slides for your M1 presentation, your code (and all the relevant parameters you used), and the video presentation in a single zip file named <M1_TeamNumber>.zip and upload it on Canvas under the Project folder.
- Before you begin Milestone 1, read the entire description carefully to make sure you understand it and have all the tools you need available.
- **Start early!** This Milestone may take longer than you expect to complete.

M2 [35 points] Network Quantization DUE: Apr 26, 12 noon (CST)

In Milestone 2 (M2), you are required to implement the network *quantization* techniques introduced in lecture 11. More precisely, you are required to do the following:

- Implement the network dynamic quantization technique to convert the weights of the model from float point to ***unsigned*** 8-bit fixed number (uint8) based on your selected framework for MobileNet-v1. For details regarding deployment using ONNX check here and for deployment using TensorFlow Lite check here.
- Deploy the quantized network over the entire test set (available in ***HW3_files/test_deployment***) on Raspberry Pi devices and complete ***Table 2***:

Table 2

Pruning parameters	Maximum memory usage [MB]	Average latency for one image [ms]	Maximum power consumption [W]	Average energy consumption for one image [mJ]

Note: The Latency and Energy Consumption values need to be the average values over the *entire* test dataset.

- Combine the required structural pruning parameters from M1 and the quantization from M2 to compress the model even further and add your results to ***Table 2***.
- Record a video where you present your experimental results and observations; upload the video to Canvas. Follow the same guidelines as in M1.

Related paper: Pruning+Quantization

Submission Instructions/Hints:

- Include the slides for your M2 presentation, your code, and the video presentation in a single zip file named <M2_TeamNumber>.zip and upload it on Canvas.
- Before you begin M2, please read the entire description carefully to make sure you understand it and have all the tools you need available.
- **Start early! This Milestone may take longer than you expect to complete.**

M3 [30 points] Game of ~~Thrones~~ Compressions DUE: May 3, 12 noon (CST)

Milestone 3 (M3) is the real competition where each team uses their most optimized model to test various performance metrics. Based on M1 and M2, you can use *any* combination of known model compression techniques to obtain the best results for your model compression recipe. Based on your final design deployment results, complete ***Table 3***.

Table 3

Test accuracy [%]	Maximum memory usage [MB]	Average latency for one image [ms]	Maximum power consumption [W]	Average energy consumption for one image [mJ]	Framework used (TensorFlow or PyTorch/ ONNX or TensorFlow Lite)	FoM

Note: The Latency and Energy Consumption values need to be the average values over the *entire* test dataset. Make sure you run your model several times (at least three times) the test dataset so you can report consistent average values.

Road to the thrones:

1. Optimize the model for the *lowest latency*, while keeping the test accuracy ≥ 70%. For the same (or close) latency figures, the design with the highest accuracy will win.
2. Optimize the model for the *lowest energy* consumption, while keeping the test accuracy ≥ 70%. For the same (or close) energy figures, the design with the highest accuracy will win.
3. Optimize the model for the *highest score*:

   Figure of Merit (FoM) = $\frac{Accuracy}{Latency \times Energy\ Consumption}$

The winner of this competition will be decided based on your final (most optimized) design. The best project will receive 4 bonus points (added directly to the final score in this class).

You are required to do the following:

- Fill the data in Table 3.
- Present the details of your approach, experimental results, and your observations based on the Submission guidelines below.

Submission Instructions/Hints:

- Submit your code, as well as your best saved model (ONNX or TFLite, as appropriate), and only **one slide** containing *Table 3* with complete information in a single zip file named <M3_TeamNumber>.zip and upload it on Canvas. Highlight with bold fonts the most relevant result(s) based on the category[1] you would like to compete for the best project award. We will evaluate your saved model to validate your best reported results.
- You'll have 60sec to give a live pitch about the most notable performance metrics of your project, as well as what was the most important lesson you learned while working on it.
- Before you begin Milestone 3, you should read the entire description carefully to make sure you understand it and have all the tools you need available.
- **Start early!** This Milestone may take longer than you expect to complete.

Good luck!

[1] Of note, you can compete in only one category for the best project award. Make sure you choose the most representative category based on your preliminary results.