

Efficient AI

Checkpoint 1: September 15th, 2026

Checkpoint 2: November 15th, 2026

1 Project Description

This project will be a course long project where you have to develop a recipe of model compression, i.e. given a model and a target domain, your recipe should be able to compress the model so as to maintain the performance of the model on the target domain. You can experiment and develop your own recipe (including reusing existing research / githubs / papers) as starting points) on different models / target datasets and keep improving the recipe over the course of the project. We will be learning about model compression techniques throughout the course and hopefully the discussion can help you think about how to improve.

Domain: We will be evaluating the models on "math" domain. However, the exact dataset for evaluation will be hidden and only be used for populating the leaderboard. In fact, it is quite likely that final evaluation will happen on dataset that is not available in public domain today.

1.1 Tracks

There are three tracks for the project. Each student will be required to submit for the first two tracks. The third track is bonus track and will be optional.

Track 1: Here the models will be evaluated on the "memory" alone, i.e. the size of the model checkpoint in memory. (Specifically, we will not worry about if this memory will lead to any speed benefits on CUDA or not.) This is the pure model compression track.

Track 2: Here the models will be evaluated on the "memory" but techniques used should be aligned with those that can be used to speedup the model on CUDA in principle. We will not ask you to implement the speedup part, but you should be able to explain how the techniques you use can be used to speedup the model on CUDA.

Track 2+: This is the bonus track where you can develop and submit kernels required for speedup the model on CUDA. You have to show speedups in specific submodules (ex. linear layers, attention layers, etc.) at some large scale regime. **This track will be open for top-performers in mid-term evaluation** and will be provided with A100/H100 GPU access. This track will be used to push up the grades.

1.2 Evaluation and Timelines

There will be two checkpoints where evaluation will happen. The first checkpoint happens at mid-term and second at end-term. Note: mid-term top-performers will be eligible for Track 2+. There will be two components of each evaluation

1. position on the leaderboard
2. Detailed report of the techniques tried – **Including what failed** and what worked.

The top-k performers in evaluation at checkpoint 1 will present their ideas in one of the sessions immediately following the checkpoint evaluation. Till this point, the code of every student will be hidden. After the presentations, the "snapshot" of all the codes will be made public for students to use among each other and students should "cite" what ideas they ended up using for their final evaluation.

You are expected to keep working on the project till the end-term evaluation with new ideas while reusing existing ideas from checkpoint 1 from the class and other existing research ideas. The final evaluation will follow the same format as the mid-term evaluation. And all the updated codes will be made public after the final evaluations.

1.3 Weightage

This project in itself is 60% of the course grade. People who submit for Track 2+ can gain upto 5% extra points (over the total grade) if they perform well in the bonus track. The breakdown of project grade (normalized to 100%) is as follows:

Eval 1		Eval 2	
50%		50%	
Position on Leaderboard	Report	Position on Leaderboard	Report
60%	40%	60%	40%

Remark: The reports have to be in a pdf format with a **maximum of 3 pages** of text (excluding figures, tables, etc.). (All artifacts should appear after page 3 and should be referenced in the text.) **Reports with more than 3 pages will not be considered for evaluation.** Also, reports have a strict "NO AI" policy. Any use of AI in the report will result in a grade of 0 for the entire project.

1.4 Resources

You can use Kaggle, Google Colab, Molab, etc. for development.

1.5 Submission details

- **What to submit for iterative submission and frequency of updates:** You can make a maximum of 1 submission per week. We will be updating the leaderboard every week with the latest submissions. You need to push your compressed model to your huggingface. And submit the model link to us (Exact details will be provided by the TAs)
- The starter codebase for the project will be provided to you. You should use this as a template (click on "use template" button on the top right corner of the repository) to create a "private" repository. You should not share this repository with any student. You should add TAs and instructor as collaborators to the repository.
- For every evaluation, we will be using the "roll_no/model_name on huggingface /code.py" to reproduce the model checkpoint and convert it to bf16 full model. Once you submit the model for evaluation, it is suggested to freeze this file and not update it.
- If using external librarians, you are expected to maintain a upto date pyproject.toml file in the repository. Please do not expect TAs to figure out your dependencies.
- The leaderboard will be anonymized – you will only be able to see the scores of the submissions (the submitters will be informed of their scores after the evaluation is done).

1.6 Honor Code

While doing every submission, you will be required to sign the honor code pledge. You can find the pledge at the end of the submission form. The honor code is as follows:

"I hereby declare that this submission is my own work and I have not shared my code or artifacts with any other student. I understand that any violation of the honor code will result in a grade of 0 for the project."

1.7 AI policy

The coding, ideation, etc has a Any-AI policy, which implies you can use any AI tools to help you with the project. In fact interesting AI usage is encouraged and will be rewarded when reported in the report.

However, the report writing has a strict NO-AI policy. Any use of AI in the report will result in a grade of 0 for the entire project. AI should not be used even to check grammar / spell check or otherwise. If found, the consequences are severe (0 grade for entire project).