## (0) Your own project from outside of EECS

Groups that choose this option will need to bring deep learning techniques to bear on new applications and data that are brought to the class by students from other departments. In particular, you will need to describe a problem in a new domain to which deep learning techniques from this class can be applied, and try doing it!

**Important Note** - This project option ***must*** be led by a graduate student outside of EECS who takes full responsibility for disciplinary expertise as well as providing both data and anything else required to work on it. The 182/282A course staff can give comments and point to deep learning directions, but have no domain expertise in your domain. We encourage graduate students with such projects to propose to use EdStem to help recruit interested students in the class to their project team, including undergraduates. Students from the EECS department are allowed to be on such teams as long as it is led by a graduate student outside of EECS.

Project Proposal

- Please write a **four** page proposal on your project idea; including a high-level overview of the problem domain, what are the existing/traditional non-deep-learning approaches to solving the specific problem, what your investigation will be, code bases you'll build upon, your training/test data, and your resource budget — an awareness of how much compute power you can muster and use is very important. Note, there is no limit here on compute and if the graduate student has access to additional compute resources you are free to use them.
- Please include all the group members in your proposal, but only one person needs to submit the proposal. Typically this will be the non-EECS graduate student leading the team.

Final Report Requirements:

- Must have a coherent story that clearly lays out the question being investigated, cites relevant literature while being a self-contained treatment, and shows clear and systematic analysis/exploration.
- For example, you really need to provide loss curves for training and validation set and give comparisons as needed to tell your story.
- Provide a link to your github repo with all the code used for this project as well as trained model checkpoints sufficient to efficiently replicate all plots and tables in your paper.
    i) If you are using any data that can't be shared, it is your responsibility to argue explicitly why what you are providing is enough for the peer reviewers to actually check and comment on your work.

## (1) Theoretical/empirical study of in-context learning

Read the late 2022 paper “What Can Transformers Learn In-Context? A Case Study of Simple Function Classes” by Garg, Tsipras, Liang, and Valiant, look at some follow-up work, and propose your own investigation that follows that theme and leverages things that you have learned. There are many different directions that are possible here and everything you propose will presumably involve at least a partial replication of what has already been done. For example, you can investigate other simple function classes that have not already been done (e.g. kernelized linear models), compare different architectural choices (e.g. transformer variants including those based on faster attention mechanisms, alternatives like state-space models, etc… Or take the architectural choices that ended up winning the nanoGPT speedrun and try them in this space), as well as questions related to the coverage of training vs test data (e.g. an appropriate distributional shift type of question) or even focus on questions of optimizers and hyper-parameter tuning (for example, exploring muP in this space).

We ran this project idea in Fall 2023 and it was a big hit. There’s been a lot of work since then in the literature, but there is still plenty of room here for exploration.

Project Proposal

- Please write a **one-two** page proposal on your project idea; including a high-level overview of your investigation, code bases you’ll build upon, how you will generate training/test data, and your resource budget — an awareness of how much compute power you can muster and use is very important.
- Please include all the group members in your proposal, but only one person needs to submit the proposal.

Final Report Requirements:

- Must have a coherent story that clearly lays out the question being investigated, cites relevant literature while being a self-contained treatment, and shows clear and systematic analysis/exploration.
- Must cite relevant work to put your work into an intellectual context.
- For example, you really need to provide loss curves for training and validation set and give comparisons as needed to tell your story. (Here, it can be fine to partner up with other project teams in this theme to share baselines, etc.)
- Provide a link to your github repo with all the code used for this project as well as trained model checkpoints sufficient to efficiently replicate all plots and tables in your paper.

## (2) Interpretability

While we expect most student groups taking option (1) above to be training your models from scratch, this option (unless combined with option (1)) is going to involve working with pretrained models in some way.

Interpretability is a broad area where what we're trying to do is understand in more detail how/why a trained neural net is behaving the way that it does. This is quite broad, and you can see https://www.neelnanda.io/mechanistic-interpretability/getting-started for a pointer to many resources. One way that one can test whether one has found an actual mechanism is to figure out an intervention based on that mechanism that would alter behavior. (For fun, take a look at https://www.anthropic.com/news/golden-gate-claude.) For another very recent example, take a look at https://arxiv.org/abs/2502.03708. (The "linear representation hypothesis" is what suggests this family of interventions and explorations — and so any work that further confirms, challenges, or extends this hypothesis is an example of what is welcome in a project.)

There are now multiple open-source models that have also released pretraining checkpoints. Those checkpoints can be used to explore many things, including the emergence/stability of mechanisms as well as the transferability of interventions across training.

Project Proposal:

- Please write a **one-two** page proposal on your project idea; including a high level overview of your investigation, your base pretrained models, code bases, datasets, and resource budget.
- Please include all the group members in your proposal, but only one person needs to submit the proposal.

Final Report Requirements:

- Must have a coherent story and be written as a more systematic investigation instead of a mere demo or hack.
- Must cite relevant work and put your own project into context.
- Provide a link to your github repo with all the code used for this project. Everything needs to be reproducible.