Final Project Proposal
Due October 22 11:59pm

Submit the following information in PDF form to Blackboard. Your answers can be concise as you like, as long as you give us a clear idea of what you're doing.

Team. List all members of your team. Your **teammate should submit an identical proposal** to Blackboard. (You must have a teammate unless you've been approved for an exemption, which is relatively rare.)

Elevator Pitch. Describe what you want to do in one sentence. Here are some examples from prior runs of the course:

* Train a convolutional neural network working with spectrograms to recognize whether the word "Jarvis" is present in some audio
* Collect new images of shoes, and train a neural network to recognize the shoe type
* Use text from news stories to make predictions about stock price in time series
* Train a neural network to generate new music from MIDI files
* Compare the performance of Monte Carlo Tree Search to minimax with alpha-beta pruning, for a new game not covered in class
* Train a Q-learning reinforcement learning agent with a neural network backend to perform a novel task, varying the state and reward structure to try to get better results

Choose a project that you think will take roughly as much work as these.

Context. If you are interacting with a technology we may not have heard of, a game we may not have played, or any other kind of jargon or specialized knowledge relevant to your project, provide links or descriptions here. (If you think the rest of your proposal is self-explanatory, you don't need to say anything here.)

Methods. Explain what AI/ML methods you will try to tackle the problem. As opposed to the elevator pitch, here you should be specific about what you will try.

Data Source. Will you collect your own data, or do you have a source in mind? If the latter, give the URL. You can skip this step if your AI will not be trained from data (like maybe an MCTS game player).

Code resources. What webpages or generative AIs will you rely for code? *Please be very frank with us so that we can work to make this a meaningful and interesting project. It is fine to use such resources as long as we're clear on your own contributions (see "What's New" below).*

What's new. Explain very clearly what part of the code will come from you instead of the code resources. In addition, mention any additional planned effort on your part that we should consider when deciding whether the project does enough that's new - like gathering a new dataset, for example. *If there isn't enough work mentioned here, your proposal feedback may include additional requirements for an A.*

Plan. Describe a reasonable plan to finish your project by the deadline (Dec 9 for F '24). Include at least 2 milestones (dates & deliverables), and take into account that you will need to write a short summary paper and do a "lightning talk" in one of the last two days of class.

Milestones will not be turned in, but we recommend trying to hit your own milestones, and we will suggest revisions if they don't seem reasonable.

Proposed demonstration or evaluation. How will we know which aspects of your project work well? Describe at least one experiment that would evaluate performance.

Experiments. Identify at least **two** major things about your setup that you can vary and measure with your evaluation. Each of these variations should entail a few different settings - for example, don't just promise to include or exclude one data augmentation type, but instead include or exclude several kinds of augmentation and see what happens. This is a major part of your final grade, so give it some thought.

Have fun brainstorming!

Appendix - Notes on common project types (not an exhaustive list)

Deep reinforcement learning - This is very sensitive to how the rewards work, so lay out at least a couple of different strategies in your proposal.

Game players - There's a tendency to get carried away with the game implementation here, but my interest is more in trying different evaluation functions, even if the game isn't 100% faithful to the original. You can use either classic minimax or Monte Carlo Tree Search.

GANs - I advise against GANs - students seem to rarely have success with them. They're just too difficult to train - finicky about parameters and bad about showing progress.

Neural network for images - Try not to make this too "safe" and close to the homework - do some research, and try something you've never tried before.

Scikit-learn - Don't propose a project that could be accomplished with scikit-learn alone, but proposing a comparison to a scikit-learner could count as one of your experiments.

Shiny black boxes - In principle, you can work with arbitrary code off the Internet and come up with good experiments to run with it. In practice, these projects can come across as a little low-effort. If you do want to do a project like this, be sure to come up with a good question to investigate that ideally involves writing nontrivial code.