CS 285 Final Project Proposal

1. **Problem Domain and Simulator**

   I propose to study the effectiveness of transfer learning in board games of varying dimensions. I plan to use Gym's environments to simulate the games of Go, Tic-tac-toe, Hex, and Othello in different board sizes. Below are the links to each of the simulators:

   https://github.com/aigagror/GymGo
   https://github.com/ClementRomac/gym-tictactoe
   https://github.com/FirefoxMetzger/minihex
   https://github.com/lerrytang/GymOthelloEnv

2. **Research Hypothesis**

   The computational resources needed to train deep reinforcement learning networks is a major challenge in many problem domains. This motivates the use of transfer learning, where the initial agent in a target domain uses the parameters of an agent that has been trained in a different but related source domain. Transfer learning seeks to generalize previously learned knowledge from the source domain so that the agent can more quickly learn to perform optimally in the target domain.

   This project seeks to understand how transfer learning from lower complexity environments can help agents learn better in similar higher complexity environments; specifically, this will be done through two player board games. In the games of Go, Tic-tac-toe, Hex, and Othello, complexity of the environment is directly associated with board size. One important fact to note is that in all of these games, the rules directly and naturally extend from the original games to arbitrary board sizes.

   Now I'll more formally define the proposed task:

- For a game G played on a board of size N_1, train an agent A_1 through DQN (or perhaps some other deep RL algorithm)
- Using some parameters of A_1, train another agent A_2 (using the same RL algorithm) to play the same game G on a board of size N_2 where $N_2 > N_1$.
- Train another agent A'_2 (with the same RL algorithm) to play game G on a board of size N_2, but this time using a completely randomized algorithm to start off with
- Compare the performances and training times of A_2 and A'_2 to see if A_2 can achieve the same (or better) levels of performance and A'_2 with less training

   There are now some interesting questions to investigate.

➢ What are different ways of transferring parameters from A_1 to A_2? Which network architectures tend to provide better support for transfer learning?

➢ What happens if A_2 is used to initialize RL training for agent A_3 to play game G on a larger board size N_3? Is repeating this process advantageous? Can the behaviors of such a sequence of agents be interpreted in some way to model how humans learn to understand more complex situations through simpler scenarios?

➢ Can agents trained to play one board game be used to help train agents for another board game? If so, can this be used to train a single agent to do well in all four games? Can such an agent be observed to apply "lessons" learned in one game to a different game?