Go Board Game AI Engine Proposal
Kyu Ro and Sira Kim

The purpose of this project is to develop a model to play against a human player in the game of Go. Just as chess is a common board game that is widely played by many individuals in America, Go is a popular game played more commonly in Asian countries. Our motivation for creating an AI Go model simply comes from our enjoyment of playing the game; since we know how to play the game and are also studying about machine learning, we thought it would be interesting and provide a good opportunity to learn about ML by coding our own Go model.

There already exist many AI models for Go but understanding the algorithm behind the reinforcement learning of the pre-existing models and developing a new model can help improve the model's ability against human opponents. We will study various search tree algorithms and how to create a deep neural network to create our reinforcement learning model. The most well known model is AlphaGo, Google's AI which uses self-play and reinforcement learning. Developing all of the components of the game and model will be too complex for a semester project, so components of the board and other necessary features will be taken from a pre-existing open source Go AI model. This model is KataGo, an open-source model. This model uses a CNN as well but includes some advanced concepts of play such as ladders and liberty status. We will also use Jupyter and Colab notebooks to run and train our model, Tensorflow with Keras will be used to create the neural network, and finally the OpenAI Gym will be used for our main reinforcement learning environment.

https://github.com/lightvector/KataGo/tree/master/python.

For this project, a new model will be developed that will look to improve the performance compared to KataGo if possible. This model could be difficult to test with the complexity of the model so while the model can be created, training and testing of the model could be difficult due to the limitations of our machines, so we will also consider rescaling the board to smaller dimensions. The algorithm can still be studied to understand the methods of existing models. Our goal is that this project will result in a new completed model that plays against human opponents while gaining understanding of neural networks and the advanced search tree algorithms that factor into creating a self-playing Go AI model. This project will not be for the MS project requirement.

Progress Report
So far, we have worked on developing a miniaturized board for the model to use to make it easier to train. This board is a smaller 8x8 board where the human plays any move and the computer will respond with a randomized legal move. This randomized move is generated by using the valid_moves function in the GymGo environment which returns an array of all possible moves, and picks a spot from it to make a "random" move.

We are in the process of training the model to learn from those moves and this will eventually lead to an AI model that improves and hopefully beats a human player. Although we are still experimenting with the GymGo environment and how to build the best model the main steps to create the Go model is as below:

1) Generate the training set by making the random move AI play against another random move AI. We will make it play through several sets of games to gather the observation, reward, done, info data.
2) We choose a step goal which is the number of steps to run the moves and collect the score using the reward output. If this score matches the wanted score that we eventually want the model to achieve we can save the "game set" (observation, reward, done, info data).
3) Build a reinforcement learning model using Keras/Tensorflow with a decision tree and train this model using the generated training set.

4) Apply the trained model as the opposing player and have it predict the next best possible move by using the state of the board from the previous step (if it's the very first step, since there is board state to predict from, the random valid_move function will be used to generate this first step).
5) Try training the model on a normal 19x19 board and compare the results to the smaller board.

Research has shown that for AI Go models, a decision tree reinforcement learning algorithm would be the best for training these models. We will look into this and implement this. This algorithm can also be compared to other models for accuracy. If this is successful, we will also look to implement this training on a full 19x19 go board.

Kyu Ro - research and testing on OpenAI GymGo environment, research models + decision tree algorithm, generate testing dataset, test model on training set

Sira Kim - research and testing on OpenAI GymGo environment, research on building model with Keras/TensorFlow, generate testing dataset, test model on training set

Timetable

Due Date	Task	Team Member
9/16	Proposal Due	Kyu and Sira
9/23	Explore KataGo	Kyu and Sira
9/30	Research Models and Decide Model	Kyu and Sira (individual research then share together)
10/13 - 10/14	Create and have finished Model; Project Progress Report Due	Kyu and Sira
10/28	Combine and test Go Model on KataGo board	Kyu and Sira (test together modify different components individually)
11/18	Modify and Debug Go Model + Start writing report	Kyu and Sira (split and write individual parts to report)
12/2	Finish Report and Presentation, make final changes to Go Model	Kyu and Sira
12/6	Final Project Due	Kyu and Sira