

# AI 3603 ARTIFICIAL INTELLIGENCE: PRINCIPLES AND TECHNIQUES

---

By: Yanzhi Huang, Tong Wu, Ji Ai

HW#: Interim Report

November 28, 2023

## I. INTRODUCTION

### A. Description

In this project we are involved in implementing a agent playing the game of Gomoku by implementing a search algorithm.

We plan to make our intelligent system strong enough to defeat pure-MCTS opponents. This method is a "pure" reinforcement learning method which need no human knowledges about Gomoku game. This method was introduced by DeepMind and was used in the famous Go game engine AlphaGo-Zero.

### B. Environment and Structure

Our project will work in Python 3.

We design a plan using the following structure.

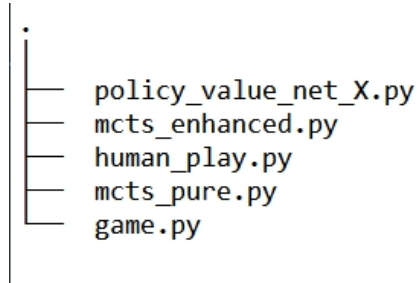


FIG. 1: Structure of our fold

- To play with provided models, run **human\_play.py**.
- To train the AI model, run **train.py**.

More functions haven't been introduced.

## II. FUTURE WORK

```
1
3 def graphic(self, board, player1, player2):
4     """Draw the board and show game info"""
5     width = board.width
6     height = board.height
7
8     print("Player", player1, "with-X".rjust(3))
9     print("Player", player2, "with-O".rjust(3))
10    print()
11    for x in range(width):
12        print("{0:8}".format(x), end=' ')
13    print('\r\n')
14    for i in range(height - 1, -1, -1):
15        print("{0:4d}".format(i), end=' ')
16        for j in range(width):
17            loc = i * width + j
18            p = board.states.get(loc, -1)
19            if p == player1:
20                print('X'.center(8), end=' ')
21            elif p == player2:
22                print('O'.center(8), end=' ')
23            else:
24                print('-', end=' ')
25        print('\r\n\r\n')
```

Above is a core code block in **game.py**.

Here are some of our plans to improve the performance. We will combine the Monte Carlo Tree Search together with Deep Neural Networks.

## III. REFERENCES

As for improving the agent, we referred to [this blog](#) and its [github code](#).

An important paper [UCT-ADP Progressive Bias Algorithm for Solving Gomoku](#) combines Adaptive Dynamic Programming (ADP), and UCB applied to trees (UCT) algorithm with a more powerful heuristic function based on Progressive Bias method and two pruning strategies for a traditional board game Gomoku.

Other than this, we have read and discussed [this paper](#).