

# Report

- Search strategy: Iterative deepening minimax with alpha-beta pruning
  1. Set time limit to 15 sec for a single move.
  2. Run iterative deepening minimax with alpha-beta pruning until TIMEOUT. (i.e. if TIMEOUT then break and return the best move of the deepest depth)
  3. Heuristic evaluation function

For cut-off states, evaluate a heuristic.

$$\underline{h(\text{state}) = 30 \times \Delta 13 + 9 \times \Delta 8 + 5 \times \Delta 5 + 3 \times \Delta 3 + 3 \times \Delta 2},$$

where

$$\Delta 13 = (\text{number of 13 in AI\_cards}) - (\text{number of 13 in User\_cards}) + 0.8 \times ((\text{number of 13 on board for AI}) - (\text{number of 13 on board for User}))$$

$$\Delta 8 = (\text{number of 8 in AI\_cards}) - (\text{number of 8 in User\_cards}) + 1 \times ((\text{number of 8 on board for AI}) - (\text{number of 8 on board for User}))$$

$$\Delta 5 = (\text{number of 5 in AI\_cards}) - (\text{number of 5 in User\_cards}) + 1 \times ((\text{number of 5 on board for AI}) - (\text{number of 5 on board for User}))$$

$$\Delta 3 = (\text{number of 3 in AI\_cards}) - (\text{number of 3 in User\_cards}) + 1 \times ((\text{number of 3 on board for AI}) - (\text{number of 3 on board for User}))$$

$$\Delta 2 = (\text{number of 2 in AI\_cards}) - (\text{number of 2 in User\_cards}) + 1 \times ((\text{number of 2 on board for AI}) - (\text{number of 2 on board for User}))$$

## 4. Utility function

For terminal states, check winner and return utility.

- (1) If AI wins, then return  $1000 + \text{score}(\text{AI}) - \text{score}(\text{User})$ .
- (2) If User wins, then return  $-1000 - (\text{score}(\text{AI}) - \text{score}(\text{User}))$ .
- (3) If tie is happened, then return 0. (score and the max of card of AI and User are the same)

## 5. Random order

Shuffle childnodes to improve the time complexity of alpha-beta pruning.