AI Competition

Context

In the CS Paradise casino, you walk closer to a table where some guardians are gathered. You notice that they are playing a card game that you have never seen before. Curious, you ask an external observer to explain the rules. After this introduction, you decide to impress the guardians in your quest to reunite the universes. Thus, you implement an artificial intelligence able to win at this mysterious card game every time.

Instructions

- Number of participants: 2
- Duration: 6 hours
- Bring your own laptop (Linux or OSX) with internet connection.
- Language: Python 3.6

You have to develop an agent that can play a card game. You are free to develop your agent however you want. The rules are presented in the next section.

Rules

- 2 player game
- Each player has a deck of 25 cards
- Each player has a hero with 30 hit points (HP)
- A player loses when their HP hits 0
- Each card costs a certain amount of mana
- The first player starts with three cards in their hand
- The second player starts with four cards in their hand
- Each player starts with one mana
- At the beginning of their turn, the player draws a card from their deck
- At the beginning of their turn, the player gains one mana
- The mana resets every turn

AI Competition

- The maximum amount of mana is 10
- The maximum amount of cards in a player's hand is 10
- When a player has more that 10 cards, every extra card drawn is destroyed
- Some cards are minions
- Minions can be put on the board
- Minions on the board can attack other minions or the opponent's hero
- Minions can attack once per turn
- Minions have to wait a turn after being summoned before attacking
- You can have up to seven minions on the board
- Poisonous minions kill any minion it touches
- Some cards are spells
- Spells have different effects (see table 1)
- Your hero has a power that deals two damages to the opponent's hero
- Your hero cannot deal damage to minions with its power
- The power of your hero cost two mana
- The power can be used once per turn
- When your deck is empty, you will receive N damages, where N is the amount of turns that your deck was empty

The cards that your agent will learn to play are described below. Every agent will use the same hero as well as the same cards.
The code is available here [1]. Some examples of agents are available in the folder agent.

[1] https://github.com/lavoiems/AI2019.git

AI Competition

Installation

We provide a Conda environment [2] with all the necessary dependencies. Once Conda is installed, you only have to execute the following commands in order to create a Conda environment:

```
$ cd AI
$ conda env create -f env.yml
$ source activate AI2019
```

Lancer une partie

Here is an example of command that you can run to launch a game:

```
$ python main.py --agent1 agent.agent1 --agent2 agent.agent1
```

We provide some example of agent. You can also try with the agent human.py.

By default, the script launches two processes in order to isolate your agents. It is possible to disable this feature with the flag --debug. At test time, the program will not be run in debug mode.

Deck

The deck is composed of 25 cards. They are explained in table 1. The cards are shuffled at the beginning of every game. As such the cards order will always be different.

Agent

Your agent essentially has three functions (of course you are free to create as many functions as you wish).

Function 1: start

This function is called at the start of the game and simply signal that the game starts.

- Input parameters: None
- Output parameters: None

[2] https://conda.io/projects/conda/en/latest/user-guide/install/index.html#regular-installation

Function 2: play

This function is called at every turn. It signals that your agent has to perform an action.

- Input parameters: state
- Output parameters: action

state

The state is a dictionary. Its items are described in the table 2

action

The actions return by your function is a tuple [3] containing two elements. The first element is the action your agent is taking. The second is a tuple contains the arguments related to the action (see table 3). Your agent returns its choice of actions using the id associated to the actions. Table 3 describe the actions with their arguments. Your turn will end if your agent performs an invalid action.

Fonction 3: end

This function signals the end of the game and notifies your agent if he won.

- Input parameter: victory (boolean)
- Output parameter: None

Evaluation

Each agent plays a maximum of 101 games against each agents. An agent wins against another opponent when it wins 51 games. The ranking is defined as follows.

- Amount of opponents defeated
- If there is a draw, the agent that defeated the other gets a better ranking
- If there is a triple draw, the order is defined by the total amount of games won

We will run your agent in the environment we provided. We provide you with the following packages:

[3] https://www.w3schools.com/python/python_tuples.asp

AI Competition

- numpy
- scikit-learn
- scipy
- pytorch
- tensorflow
- pandas

Number	id	Type	Attack	HP	mana	Effect
2	0	Minion	1	3	1	None
2	1	Minion	2	3	3	Poisonous
2	2	Sort	0	0	3	Destroy a random ennemy minion
1	3	Sort	0	0	2	Deal 2 damages
1	4	Minion	1	1	1	None
1	5	Minion	2	5	5	Your minions can attack immediately after being summoned
2	6	Minion	2	7	4	None
2	7	Minion	1	1	1	Your other minions have +1 atk
2	8	Minion	3	2	2	None
1	9	Sort	0	0	4	Deals 6 damages
2	10	Minion	7	14	10	None
1	11	Minion	2	2	2	+2 atk +1 HP when a friendly minion dies
1	12	Sort	0	0	3	Deals 3 damage. Deals 5 damage if you have at least one minion
1	13	Sort	0	0	3	For each enemy minion, summons 1 minion with 1 atk and 1 HP. These minions can attack immediately
2	14	Minion	6	5	6	At the moment of death, summons 2 minions with 2 atk and 2 HP
2	15	Minion	2	2	2	The adjacent minions have +1 atk

Table 1: Your deck

key	value-type	Description
player_hand	list	List containing the minions in your hand
player_target	list	List containing the minions on the board as well as your hero
opponent_target	list	List containing the enemy minions on the board as well as the enemy hero
hero_health	int	Hit point of your hero
opponent_health	int	HP of the enemy hero
player_mana	int	Your mana points
opponent_mana	int	Enemy mana points
n_opponent_hand	int	Amount of cards in the ennemy's hand

Table 2: Items in the state

id	action	argument 1	argument 2	Description
0	Hero Power	None	None	Use your hero power dealing two damages to the enemy hero. Costs two manas. Can be used once per turn
1	Place a minion	index hand	None	Put a minion on the board from (player_hand) at (index hand).
2	Play spell	index hand	index target	Play a spell card from your hand (player_hand) at index (index hand). If the card requires a target. Apply the effect to the target (oppenent_target) at index (index target)
3	Attack	index source	index target	Attack with your minion on the board (player_target) at index (index source) the ennemy (opponent_target) à index (index target)
4	End turn	None	None	End of your agent's turn

Table 3: Possible actions