

AI Competition

Context

Dans le casino du CS Paradise, vous vous approchez d'une table où plusieurs gardiens du sommet sont rassemblés. Vous les observez jouer à un jeu de carte dont vous n'avez jamais entendu auparavant. Vous demandez donc à un observateur de vous expliquer les règlements. Après cette explication, vous décidez d'impressionner les gardiens dans la quête de réunir les univers. Vous implémentez donc une intelligence artificielle capable de gagner à tous coup à ce jeu de carte mystérieux.

Instruction

- Nombre de participants: 2
- Durée de la compétition: 6 heures
- Vous pouvez utiliser votre propre laptop (Linux ou OSX) avec connexion internet.
- Language: Python 3.6

Vous devez développer un agent pouvant jouer à un jeu de carte. La façon dont vous développez votre agent vous revient. Les règles du jeu sont présentées à la prochaine section.

Règles du jeu

- Le jeu se joue à deux.
- Chaque joueur à un deck de 25 cartes
- Chaque joueur à un héro ayant 30 points de vie
- Un joueur perd lorsque le nombre de points de vie de son héro atteint 0
- Chaque carte coute un certain nombre de mana
- Le premier joueur commence avec trois cartes dans ses mains
- Le second joueur commence avec quatre cartes dans ses mains
- Chaque joueur commence avec 1 mana
- Au début de son tour, le joueur tire une carte de son paquet
- Au début de son tour, la mana du joueur est remplie
- Au début de son tour, le joueur obtient un mana supplémentaire

AI Competition

- Le nombre maximum de mana est 10
- Le nombre maximum de cartes qu'un joueur peut avoir dans ses main est 10.
- Lorsqu'un joueur à plus de 10 cartes, les cartes supplémentaires pigées sont détruites
- Certaines cartes sont des minions
- Les minions peuvent être mis sur le jeu
- Les minions sur le jeu peuvent attaquer les minions adverses ainsi que le héro adverse
- Les minions peuvent attaquer une seule fois par tour
- Vos minions doivent attendre un tour après avoir été invoqué avant de pouvoir attaquer
- Vous pouvez avoir un maximum de sept minions sur le jeu
- Les minions poisons tuent tout minion qu'ils touchent
- Certaine cartes sont des sorts
- Les sorts ont différents effets (voir table 1)
- Votre héro à un pouvoir pouvant infliger deux de dégats au héro adverse.
- Vous ne pouvez pas infliger de damages aux minions avec votre pouvoir
- Le pouvoir de votre héro coûte 2 manas
- Le pouvoir peut être utilisé une fois par tour
- Lorsque votre paquet est vide, vous recevez N dégâts, où N est le nombre de tour que votre paquet a été vide

Le Héro et les cartes que votre agent devra apprendre à utiliser sont décrits plus bas. Tous les agents utiliseront le même héro et les même cartes.

Le code de la compétition est disponible ici ¹. Quelques exemples d'agents sont disponible dans le dossier agent.

¹<https://github.com/lavoiems/AI2019.git>

AI Competition

Installation

Nous vous fournissons un environnement Conda ² avec toutes les dépendances nécessaires. Une fois conda installé, vous avez seulement besoin d'exécuter la commande suivante pour créer un environnement Conda avec les dépendances nécessaires

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

Lancer une partie

Voici un exemple de commande que vous pouvez utiliser pour lancer une partie:

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

Nous fournissons quelques exemples d'agents. Vous pouvez aussi jouer en utilisant l'agent human.py.

Par défaut, le script lance deux processus distincts pour isoler chacun des agents. Il est possible d'annuler ce mécanisme en utilisant le flag `--debug`. À l'évaluation, le programme ne sera pas lancé en mode debug.

Deck

Le deck est composé de 25 cartes. Celles-ci sont décrites à la table 1. Les cartes sont brassées aléatoirement au début de chaque partie. Ainsi, l'ordre des cartes pigés sera toujours différent.

Agent

Votre agent consiste principalement de trois fonctions (bien sûr vous être libre de créer autant de fonction que vous voulez).

Fonction 1: start

Cette fonction est appelée au début de la partie et sert simplement à signaler que la partie vient de commencer.

- Paramètre en entrée: Aucun

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

AI Competition

- Paramètre en sortie: Aucun

Fonction 2: play

Cette fonction est appelée à chaque tour. Elle signale à votre agent qu'il doit performer une action.

- Paramètre en entrée: state
- paramètre en sortie: action

state

Le state est un dictionnaire. Ses items sont décrits dans la table [2](#)

action

L'action retournée par la fonction play est un tuple ³ contenant deux éléments. Le premier élément est l'action que votre agent prend. Le second contient les arguments en lien avec l'action prise. Votre agent retourne son choix d'action à l'aide de l'id qui lui est associé. La table [3](#) décrit les actions et leurs arguments. Si votre agent effectue une action invalide, par exemple jouer une carte qui demande plus de mana que possède votre héros, votre tour se termine automatiquement.

Fonction 3: end

Cette fonction signale la fin de la partie en signifiant à votre agent s'il a gagné.

- Paramètre en entrée: victory (boolean)
- Paramètre en sortie: Aucun

Remise

Vous devez remettre un dossier contenant votre agent ainsi que tous autres fichiers python – **zippés** – dans le dashboard. Votre dossier doit contenir minimalement un fichier nommé agent.py qui implémente les fonctions start, play et end. Il devrait aussi contenir un fichier __init__.py pouvant être vide.

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

AI Competition

Évaluation

Chaque agent jouera un maximum de 101 parties contre chaque agent adverse. Un agent gagne contre un adversaire lorsqu'il gagne 51 parties. Le classement est défini comme suit.

- Le nombre d'adversaires battus
- S'il y a égalité entre deux agents, l'agent ayant battu l'autre est supérieur
- S'il y a une triple égalité, l'ordre est défini par le nombre de parties gagnées

Nous allons rouler votre agent dans l'environnement conda que nous vous fournissons. Vous avez quelques packages pour vous aider à développer votre agent:

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

AI Competition

Nombre	id	Type	Attack	PV	Mana	Effet
2	0	Minion	1	3	1	Aucun
2	1	Minion	2	3	3	Poison
2	2	Sort	0	0	3	Détruit un minion adverse aléatoire
1	3	Sort	0	0	2	Inflige 2 de dégats
1	4	Minion	1	1	1	Aucun
1	5	Minion	2	5	5	Vos minions peuvent attaquer immédiatement après être invoqué
2	6	Minion	2	7	4	Aucun
2	7	Minion	1	1	1	Vos autres minions ont +1 atk
2	8	Minion	3	2	2	Aucun
1	9	Sort	0	0	4	Inflige 6 de dégats
2	10	Minion	7	14	10	Aucun
1	11	Minion	2	2	2	+2 atk +1 PV lorsqu'un minion allié meurt
2	12	Sort	0	0	3	Inflige 3 dommages. Inflige 5 dommages si vous avez au moins 1 minion
1	13	Sort	0	0	3	Pour chaque minion ennemi, invoquez 1 minion 1 atk 1 PV. Ces minions peuvent attaquer immédiatement après avoir été invoqué
2	14	Minion	6	5	6	Lors de sa mort, invoque 2 minions 2 atk 2 PV
2	15	Minion	2	2	2	Les monstres adjacents ont +1 atk

Table 1: Le deck que votre agent utilisera

key	value-type	Description
player_hand	list	Liste contenant les minions dans votre main
player_target	list	Liste contenant vos minions sur le jeu ainsi que votre hero
opponent_target	list	Liste contenant les minions ennemies sur le jeu et le héros de votre adversaire
player_health	int	Nombre de point de vie que vous avez
opponent_health	int	Nombre de point de vie de votre adversaire
player_mana	int	Nombre de mana que vous avez
opponent_mana	int	Nombre de mana de votre adversaire
n_opponent_hand	int	Nombre de cartes dans la main de votre adversaire

Table 2: Items du state

AI Competition

id	action	argument 1	argument 2	Description
0	Hero Power	aucun	Aucun	Utilise votre hero power: inflige deux damages à votre adversaire. Coûte 2 manas. Peut être utilisé une fois par tour
1	Jouer minion	index main	Aucun	Met le minion dans votre main (player_hand) à l'indice (index main) et le pose sur le jeu.
2	Jouer sort	index main	index cible	Joue le sort de votre main (player_hand) à l'indice (index main). Si la carte à une cible, applique l'effet de la carte à la cible (opponent_target) à l'indice (index cible)
3	Attaquer	index source	index cible	Attaque avec le minion sur votre jeu (player_target) à l'indice (index source) l'ennemie (opponent_target) à l'indice (index cible)
4	Terminer tour	Aucun	Aucun	Votre agent termine son tour

Table 3: Actions possibles