

Une initiation concrète à l'intelligence artificielle

par Bastien Masse

Coordinateur de la chaire EO&IA
pour l'Education Ouverte
et l'Intelligence Artificielle,
CDP Université de Nantes.

Administrateur Association Class'Code

Le concept d'apprentissage automatique (machine learning) est au centre de l'intelligence artificielle. L'objectif étant de permettre à une machine « d'apprendre » à partir des données qui lui sont fournies ou quelle génère elle-même. C'est donc de la sélection et du traitement des informations que dépendent la qualité de l'apprentissage et l'efficacité de l'algorithme. Un des moyens de guider ce dernier à travers les données est de mettre en place un système de récompense/punition qui lui permette d'identifier les informations pertinentes afin de progresser. Ce type de traitement appartient au champ de « l'apprentissage par renforcement » qui consiste à permettre à un algorithme de se renforcer par l'expérience (les données) et de progresser au cours de son fonctionnement en cherchant à optimiser l'obtention de récompenses et à éviter les punitions. Cette présentation - qui se veut être accessible à tous les élèves à partir du collège (voire avant) - présente l'avantage d'être débranchée (c'est-à-dire sans ordinateur).

Le point de départ de cet article est un texte de Martin Gardner publié dans la revue Scientific American en 1962 et intitulé : Mathematical games *How to build a game-learning machine and then teach it to play and win*. Le lien suivant permet d'obtenir le texte auquel nous faisons référence : <https://www.cs.drexel.edu/~jpopack/Courses/AI/Wi12/assignments/HW2/Hexapawn.pdf>

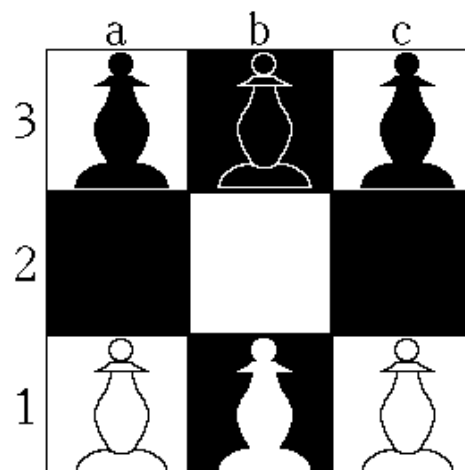
Il s'agit pour nous de transmettre les éléments de cet article et de fournir des éléments pratiques pour une mise en œuvre en classe. Le fil rouge de l'article de Martin Gardner est un jeu dont il est l'inventeur, ce jeu est nommé l'hexapawn.

L'hexapawn

Le jeu consiste à opposer un joueur humain à une machine composée de boîtes d'allumettes dont le rôle est d'indiquer à chaque tour le coup joué par l'Intelligence Artificielle.

Les règles du jeu de l'hexapawn sont les suivantes :

- 1) Les pions ne peuvent se déplacer que d'une case en ligne droite.
- 2) Les pions ne peuvent manger l'adversaire qu'en diagonale.
- 3) Il y a trois manières de gagner à l'Hexapawn :
 - Amener un de ses pion sur la ligne opposée dans le camp adverse
 - Prendre tous les pions de l'adversaire
 - Bloquer l'adversaire (qui ne peut plus prendre ou avancer à son tour)

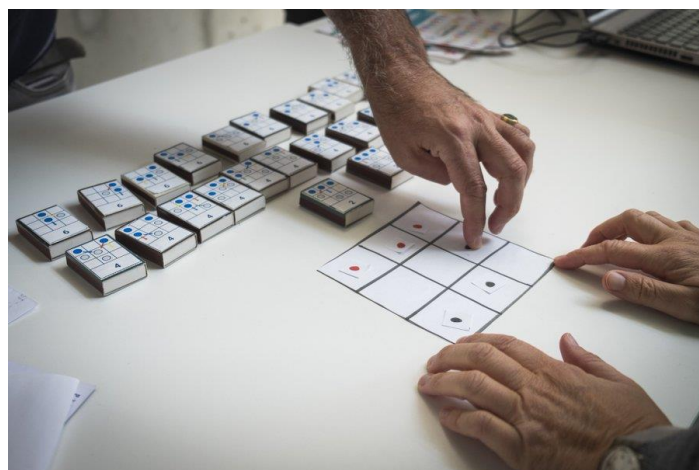


Le déroulé d'une partie

Par politesse, et puisque l'IA a besoin de données pour jouer, nous laisserons systématiquement le joueur humain commencer.

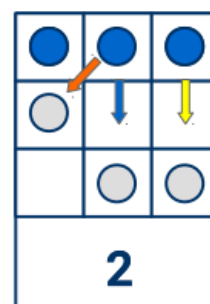
En ce qui nous concerne, nous (IA) allons systématiquement jouer en utilisant les boîtes d'allumettes.

En ce qui concerne notre adversaire (le joueur humain), nous le laissons jouer comme il l'entend.



Source : photographie personnelle de l'auteur

Chaque boîte d'allumette correspond à un état du jeu. A noter que certaines positions étant équivalentes du point de vue des actions potentielles pour l'IA, elles ne sont pas représentées. On dit que ces positions sont « en miroir ». Par exemple au premier tour, car les réponses possibles pour l'IA sont identiques, que le joueur humain avance le pion de droite ou de gauche.



Dans chaque boîte d'allumettes se trouve un nombre de perles de couleur correspondant au nombre de coups possibles à jouer pour l'IA (par exemple 3 coups possibles et donc 3 perles de couleurs différentes dans l'exemple ci-dessus). On secoue donc la boîte et on tire (innocemment c'est important) une perle de couleur. On joue ensuite le coup correspondant à cette couleur. On met alors la perle de côté car on ne sait pas encore si ce coup joué est "bon" ou "mauvais".

C'est ensuite au joueur humain de jouer. Si ce dernier gagne la partie à l'issue de son tour, on estime que le coup précédemment joué par l'IA était mauvais, on retire alors du jeu la perle que l'on avait mise de côté. Si le coup n'a rien produit, ou a pu faire gagner l'IA, on remet bien sûr la perle dans sa boîte. Il est à noter que le joueur humain ne jouera pas de manière optimale, et qu'il est possible de garder un « mauvais coup » joué sur un tour car l'humain aura laissé passer sa victoire. Ce n'est pas grave, l'algorithme mettra simplement plus de temps à progresser puisqu'il n'apprend que lorsqu'il perd.

En multipliant ainsi les parties, on retirera de plus en plus de "mauvaise perle" ce qui permettra à l'IA "d'apprendre" les réponses les plus favorables contre les joueurs humains. Bien sûr il s'agit ici de mettre de côté tous les choix défavorables. Le jeu Hexapawn est un jeu dit résolu, ce qui fait que les boîtes d'allumettes représentent la totalité de l'arbre des possibles de ce jeu. En enlevant les perles, on coupe les mauvaises branches, et on converge ainsi vers un algorithme optimal. Puisque dans le cas où il ne resterait plus qu'une perle dans chaque boîte et pour chaque branche, il deviendrait alors impossible de gagner contre l'IA.

Quelques conseils pour l'animation :

L'algorithme converge au bout d'une dizaine de parties environ. Le meilleur indice reste le nombre de perles retirées, puisque l'Hexapawn commence à devenir très fort au bout de 4 ou 6 défaite et quasi imbattable au bout de 8 ou 10. Le jeu peut se jouer seul (en gérant l'IA) mais fonctionne bien mieux en groupe. Il est possible de faire jouer les élèves chacun leur tour, et de donner à l'un d'eux la responsabilité de gérer l'IA. On tient un compte des victoires de manière à voir les progrès de l'IA en notant à la suite « IA » ou « Humain » en fonction de qui gagne. Lorsque l'algorithme aura convergé, il ne devrait plus rester que des victoires de l'IA. Il est également possible d'utiliser l'intelligence collective en prenant l'avis du groupe sur chaque coup.

De l'intelligence

L'intérêt de ce jeu est d'illustrer ce qu'est l'apprentissage par renforcement utilisé en IA, de voir émerger un algorithme optimal et de s'interroger sur la partie d'intelligence dans le système (surtout lorsque l'on perd contre des boîtes d'allumettes). C'est en effet le joueur humain qui crée la donnée nécessaire au système pour apprendre. Cet apprentissage est qui plus est supervisé, puisque c'est nous qui avons fixé les règles de ce qu'est un « bon coup » ou un « mauvais coup ». On constate ici l'avantage d'une mémoire parfaite et de la capacité de répétition spécifique à l'informatique. Contrairement à nous la machine ne fait jamais 2 fois la même erreur.

Conclusion

Pour finir, ce jeu peut également servir à faire émerger des stratégies face à la machine, comme prendre des risques qui paraîtraient illogiques contre un humain, afin d'explorer des scénarii dans lesquels l'IA n'a pas encore appris. On peut également s'interroger sur la méthode pédagogique ou d'apprentissage à adopter, puisqu'au lieu de "punir" en enlevant la perle indésirable, on pourrait aussi bien "récompenser" en ajoutant par exemple une perle de même couleur à chaque coup gagnant de l'IA. Ce procédé a d'ailleurs un avantage puisqu'il s'il permet d'améliorer les performances de l'IA d'un point de vue statistique, il laisse toujours une opportunité au joueur humain de gagner.

Pour aller plus loin

Dans le principe, tous les jeux résolus se prêtent bien aux démonstrations d'apprentissage par renforcement. Il existe des variantes pour le jeu de Nîm (qui vous permettra de recycler vos allumettes) ou encore du Morpion (mais il vous faudra alors 304 boîtes d'allumettes). L'implémentation dans un programme devient alors plus intéressante et vous pourrez trouver des exemples sous Scratch ou en Python. Dans ce dernier cas, quelques instructions conditionnelles permettront à l'algorithme d'explorer tout seul l'ensemble des options possibles en jouant contre lui-même. Ce qui permet aux élèves de coder eux-mêmes leur IA et de voir émerger la stratégie optimale au bout de quelques centaines de parties soit quelques secondes pour l'ordinateur.

