

Rapport du TP4 d'IA :

A. Decks type

Un itemset fréquent fermé représente ici des cartes souvent jouées par les joueurs du dataset. Ces itemset nous permettent donc de repérer quelles cartes sont souvent jouées par les joueurs. Ainsi nous pouvons définir une méta et chercher les contres pour les cartes souvent jouées ou simplement préparer nos decks avec les cartes et combos que nous avons repérés comme fonctionnels dans cette méta-game.

Certains de ces itemsets sont assez triviaux, par exemple le plus fréquent est l'itemset de la pièce. Cet itemset ne nous apprend rien puisque la pièce est toujours donnée au joueur qui joue en second dans une partie et donne 1 cristal de mana supplémentaire pour un tour.

Certains itemsets nous apprennent des choses intéressantes, par exemple, on peut voir que le Drake Azur est joué très souvent et que les joueurs ont tendance à le jouer le plus vite possible en jouant la pièce avec. Ça permet aux joueurs de prendre un avantage à la pioche sur leur adversaire et donc d'avoir plus de possibilités pour la suite de la partie.

B. Séquences et règles séquentielles

1. L'algorithme CloSpan

Une séquence représente ici un combo de cartes joué par un joueur ainsi que la réponse de son adversaire. Une séquence fréquente représente donc ici un combo de cartes fréquent et son contre. Avec ces données, nous pouvons mettre en avant les meilleurs combos et leurs potentiels contre. Cela nous permet, par exemple, de prévoir dans notre deck un moyen d'empêcher le contre de notre combo ou de prévoir les contres à d'autres combos. On peut aussi prévoir ce que l'adversaire va probablement jouer dans les prochains tours grâce à ce qu'il a déjà joué si cela correspond au début d'une combo que l'on aurait trouvé, et ainsi se préparer à contrer cette combo ou tout simplement l'empêcher de la finir.

Une grande partie de ces séquences sont triviales, par exemple la plus fréquente est la séquence constituée de la pièce seule. Cette séquence ne nous apprend rien puisque la pièce est donnée gratuitement au joueur qui joue en second dans la partie et elle ne présente pas d'autre carte que la pièce, et donc pas de combo intéressante avec cette dernière.

Nous n'avons pas pu trouver de séquences fréquentes intéressantes. Nous avons donc tenté de réduire le support demandé pour la recherche afin d'avoir plus de résultats. Nous avons ensuite appliqué nos connaissances pour supprimer le plus de résultats jugés inintéressants afin de pouvoir mieux lire le fichier de sortie. Pour ce faire nous avons supprimé tous les résultats contenant la pièce seule dans un des items de la séquence. Nous avons ensuite supprimé tous les résultats ne contenant qu'un item dans la séquence afin de ne pas faire de redondance avec l'étude de la partie A.

Malgré tous ces efforts, les séquences trouvées dans les résultats ne présentaient aucune information intéressante car les données sont trop imprécises (on ne sait pas où sont jouées les cartes présentes dans les séquences ni l'état du plateau de jeu à ce moment). On retrouve dans la plupart des cas une suite de pouvoir héroïque joué par un joueur puis par l'autre.

2. Les algorithmes ERMiner et RuleGrowth

De la même manière que pour l'algorithme CloSpan, les séquences représentent ici des combos de cartes jouées par un joueur suivis de la réponse du joueur adverse. En classant les résultats trouvés par ordre décroissant de confiance, on trouve assez facilement des combos élaborés et intéressants. Par exemple, on trouve le combo Dragonnet du Crépuscule, Sombre sectateur, Nova sacrée → Soins inférieurs. Ce combo permet de poser 2 serviteurs (une 2/6 et une 3/4) sur le plateau et d'infliger 2 points de dégâts à tous les serviteurs ennemis tout en soignant nos serviteurs déjà sur le plateau. L'adversaire, dans notre dataset, a toujours répondu à ce combo par son pouvoir héroïque de prêtre qui lui permet de soigner un serviteur de 2. On apprend donc que lorsqu'on joue ce combo, l'adversaire perdra forcément 2 points de mana en soignant un de ses serviteurs au tour suivant.

On ne trouve pas que des combos intéressants cependant. En effet, avec une confiance de 100%, on retrouve la séquence bloc de glace → Explosion de feu. Cette séquence ne nous apprend rien puisqu'elle manque d'informations. On ne connaît pas l'état du plateau ni la cible de l'explosion de feu. On ne peut donc pas savoir si l'explosion de feu visait le héros adverse ou un serviteur, ni si elle était destinée à activer le bloc de glace de l'adversaire.

Avec ces algorithmes de règles séquentielles, on trouve beaucoup plus de patterns intéressants qu'avec l'algorithme CloSpan. Les patterns inintéressants sont éliminés grâce à la présence de la confiance dans l'algorithme.

Les règles séquentielles permettent de mettre en avant des suites probables, et la probabilité qu'elles apparaissent, à un combo de cartes joué par l'un des joueurs. Ainsi un joueur peut prévoir la réponse de son adversaire à son tour grâce à ces résultats.