AI for 2048

Tianchen Zhong

Carnegie Mellon University, Information Networking Institute

Abstract

2048 is a very popular online game. It is very easy but hard to achieve its goal. Therefore we decided to develop an AI agent to solve the game. We explored two strategies in our project, one is ExpectiMax and the other is Deep Reinforcement Learning. In ExpectiMax strategy, we tried 4 different heuristic functions and combined them to improve the performance of this method. In deep reinforcement learning, we used sum of grid as reward and trained two hidden layers neural network. For ExpectiMax method, we could achieve 98% in 2048 with setting depth limit to 3. But we didn't achieve a good result in deep reinforcement learning method, the max tile we achieved is 512.

Game Rule

Player play this game on a 4 * 4 grid board, and each tile having an even number(Fig.1). Player can make four direction movement, which are up, down, left and right. Then all ties will move in that direction until it stopped by another tile or the border of the grid. If two tiles having the same number collide while moving, they will merge into a new grid with their sum on it. After that, this new tile will not merge with its neighbors again during this move. A new tile with number 2 or 4 will randomly appear on the empty tiles and player need to makes a new move. If there is no empty tiles on the grid, the game would end. If player could get 2048 before the game end, he wins. The score for each player starts at zero, and it adds by the value of the new tile when two tiles collide together.

Fig 1. Game 2048

Methods

In the first part of project, we implemented ExpectiMax search methods and improved them with human heuristics. Since the search tree would expand exponentially, we tried to use the depth-limited search to max performance in a limited computation resource. Later we implemented Deep Reinforcement Learning to solve the game.

Methods

ExpectiMax search is a classic algorithm in two-player game. In our Game 2048, we can view computer as an opponent and it randomly pick two and four with probability 9:1 ratio in the empty tile. We tried four different heuristic functions.

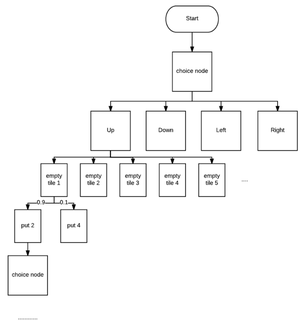

Fig 2. Flowchart for ExpectiMax

1. **Total sum of all number in the grid**: In this game, our goal is to achieve 2048, which means we need two 1024. Therefore, making sum of all grids as much as possible to help agent achieve higher tile value.
2. **Empty tile**: The empty space is the key to have high score. In many failure circumstance, all the grid was full of random number and they cannot merge together.
3. **Possible merges**: when we see same number in the adjacent position, it is very easy to merge them after single action.
4. **Monotonicity in a line**: For a monotonic line, it is the most possible way to achieve the high goal. Put tile in an ascending or descending order help tiles merge consecutively.

We can also view this problem as a MDP process, and we implement a reinforcement learning to let agent learning policy to play this game. We implement a neural network with 3 layers. The two hidden layers we chosen of size 256 and size 64. The input is a vector with size 16, which represents all tiles value, the output is a vector of size 4, and each is associated with Q value after taking 4 different actions on this state.

In the two hidden layers, we use ReLU activation function. Linear transformation is used in the output layer to get Q value. When we need to decide which action to take, we just compare all four Q value output and choose the maximum value. It is possible that the best move cannot change the state, therefore we will use the second max output value as our action to execute. If all four moves cannot change game state, game is over.

RESULTS

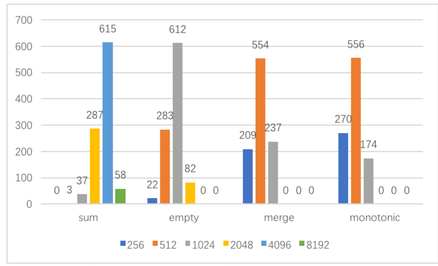

Fig3. Performance for different heuristic functions

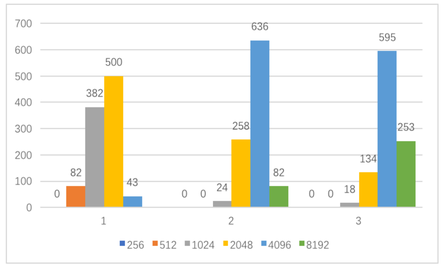

Fig 4. Performance with Different search depth

We tried 1000 trials for each heuristic methods and we set a depth limit to 2 for ExpectiMax search. The reason for that is because the limitation of computation resource. From the graph we could see that only using sum for all tiles as heuristic method is much better than other heuristic functions, which could achieve 287 in 2048, and get 615 times in 4096, which gets 90% to achieve goal. We think the sum of all tiles could really tell the state of the grid, while other heuristic methods could only tell you one part of grid state. And they could not tell you whether this action is better than others in the long run.

We also combine these heuristic methods with weight 5 for sum, 2 for empty tiles, 1 for merge, 1 for monotonic, which defined after getting the result of using single heuristic function. We set depth from 1 to 3 and run 1000 tries. We could see there is an increasing trend with depth-limit value and achieve 98% to get 2048 with depth 3. It is possible for us to get higher performance when we increase depth-limit. We also compared our program with the best ExpectiMax program on the Internet. The results are both based on the depth with 2. We could see that our result is close to the best one. The improvement could be from these two ways: 1. Change parameters for each heuristic methods. 2. Using more heuristic methods to achieve higher score.

For the reinforcement learning algorithm result, we didn't achieve 2048. The highest score we achieve is 512. The result is better than human player but not as good as search tree method. Compared to the Expectimax, this method use less time to make decision but it could not get the goal.

max tile	64	128	256	512
Counts	90	408	457	45

Future work

For ExpectiMax methods, The improvement could be from these two ways:

1. Change parameters for each heuristic methods.
2. Using more heuristic methods to achieve higher score.

For Deep Reinforcement Learning method, we could implement different neural network architecture like CNN, trying to learn the right Q value for each actions. From algorithm used by alpha go, we could use MCTS and neural network to implement on 2048 in the future to see whether it could help us to achieve the goal.