Heuristic Analysis

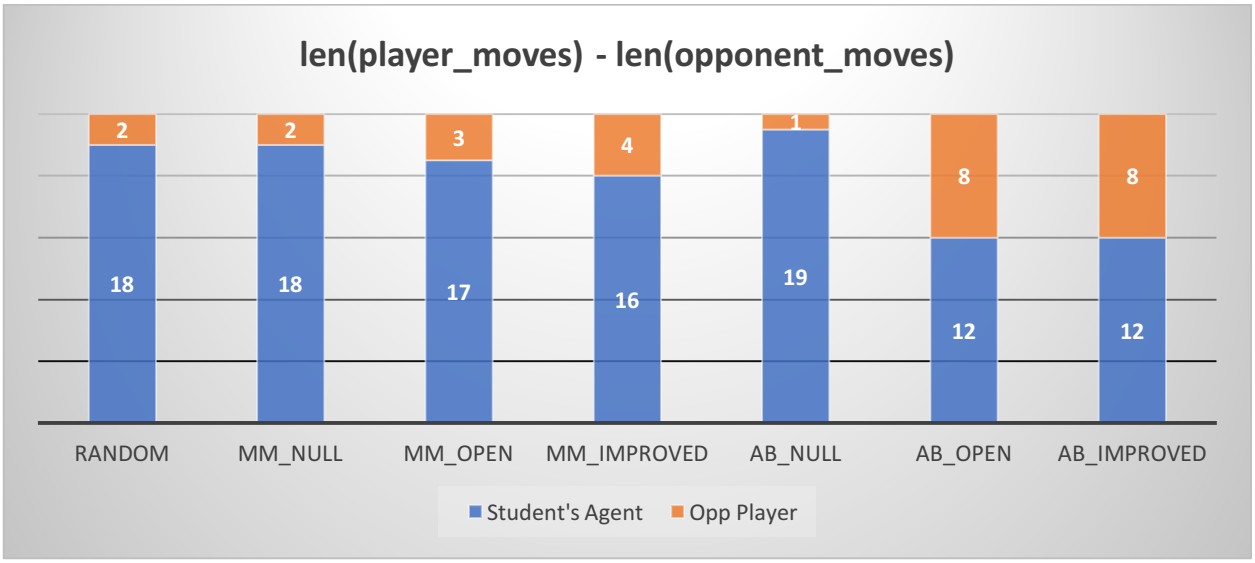
Best Value Function: 80% success rate

If we take the difference between our agent's remaining moves and opponent's remaining moves, then we get the best results for the agent

We see that 80% of the time our agent wins. Interestingly even when the opponent is playing randomly, our agent loses twice.

If we can research and create hash table of best first and ending moves, and use them before applying the iterative deepening, minimax and alpha beta pruning then I suppose we will be able to achieve higher scores.

Increasing of time out and/or running the algorithm on more powerful machine can give better results as iterative deepening will ensure that our agent performs better if given more time per move or more processing power per second.



Other Value Functions:

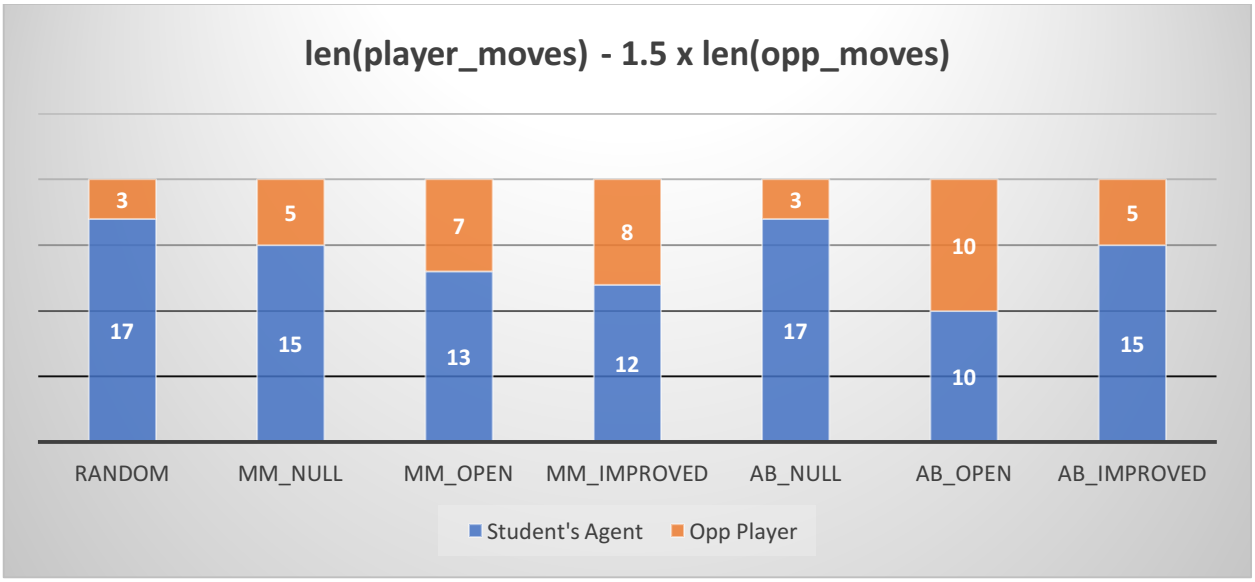
I made an attempt to change the value function by weighing the value of remaining moves for our agent and opposing players.

I found that best strategy is to stick with equal weightage to our agent's and opponent's remaining moves.

Value Function 2: 71% success rate

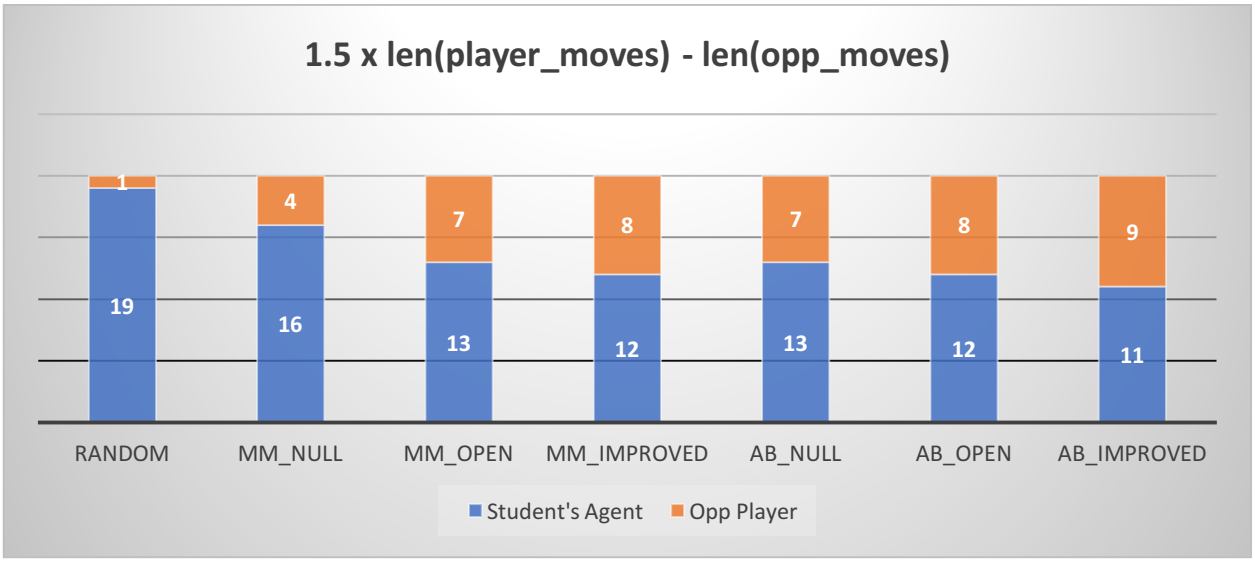
Slightly aggressive strategy, to limit opponent's remaining moves more than increasing player's remaining moves

One interesting result is that AB_Improved is being defeated slightly more than earlier with this strategy but overall the success rate has gotten 10% worse



Value Function 3: 69% success rate

What if we value our agent's remaining moves slightly more than limiting opponent's remaining moves? I found that the success rate is same as in value function 2 but overall the success rate is less than best value function.



Value Function 3: 63% success rate

What if we only value our agent's remaining moves and not care about opponents remaining moves?

This turns out to be a bad strategy compared to all the other strategies above.