Heuristic Analysis

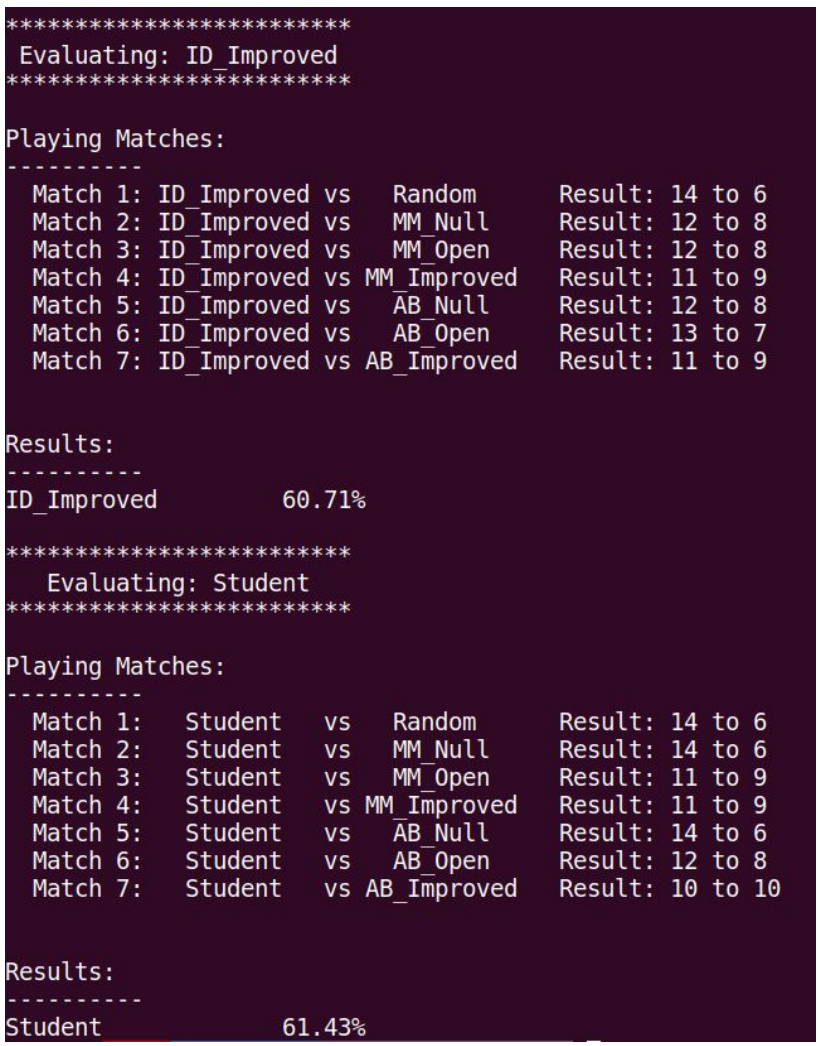
Local machine information:

Notebook Model	Processor	Memory	Operating System
Lenovo T440s	Intel i7-4600U @ 2.10GHz x 4	8Gb	Ubuntu 16.1

There are three different custom_score function in order to score different branch in the game three. We can therefore choose the highest score one to win the game finally.

All these three score functions are based on the remain legal moves of both side to calculate.

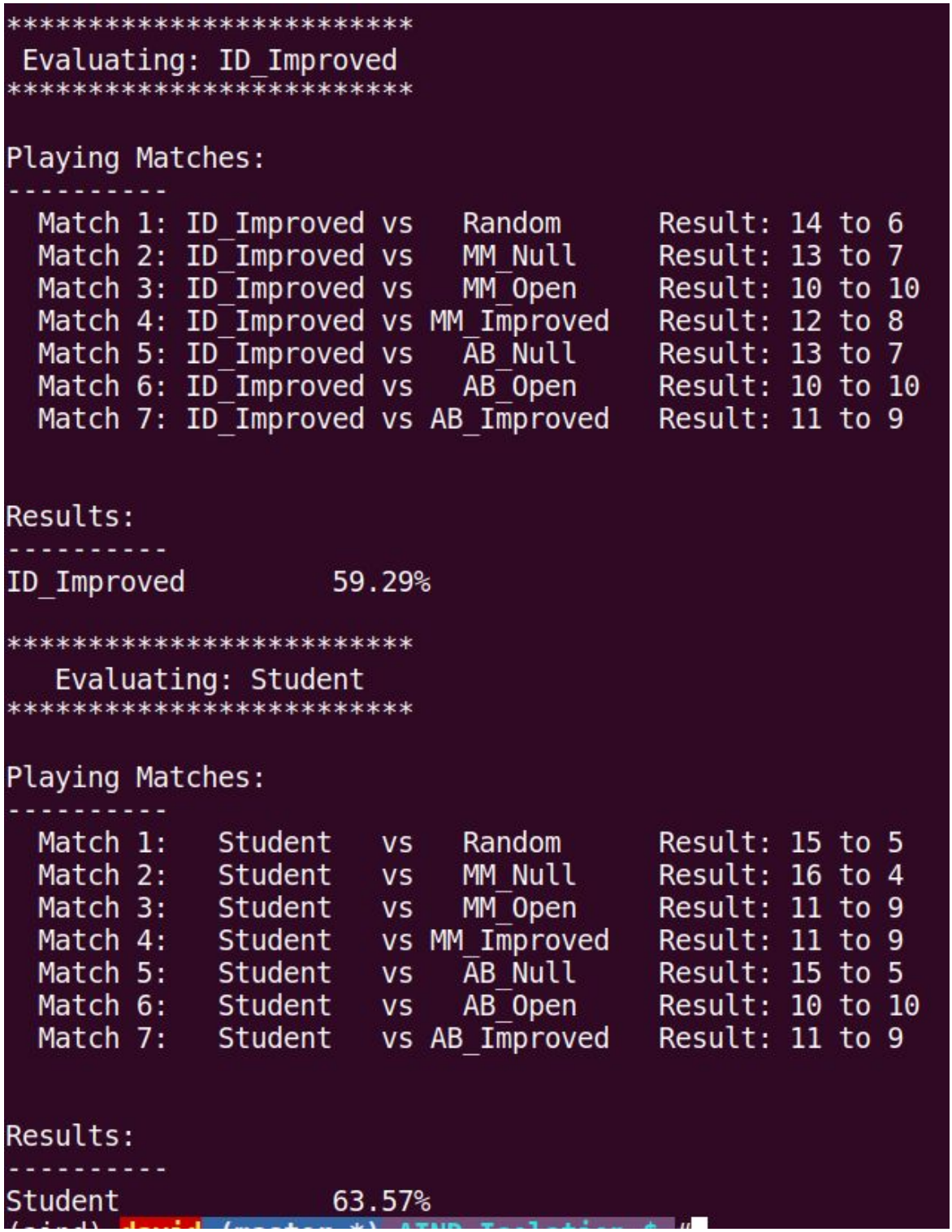
1. Host's legal moves - opposites' legal moves:



The idea here is very simple. The model is caring that the host should have more legal move then opposite.

We can see the result that ID_impoved has 60.71% winning rate and Student 61.43% winning rate.

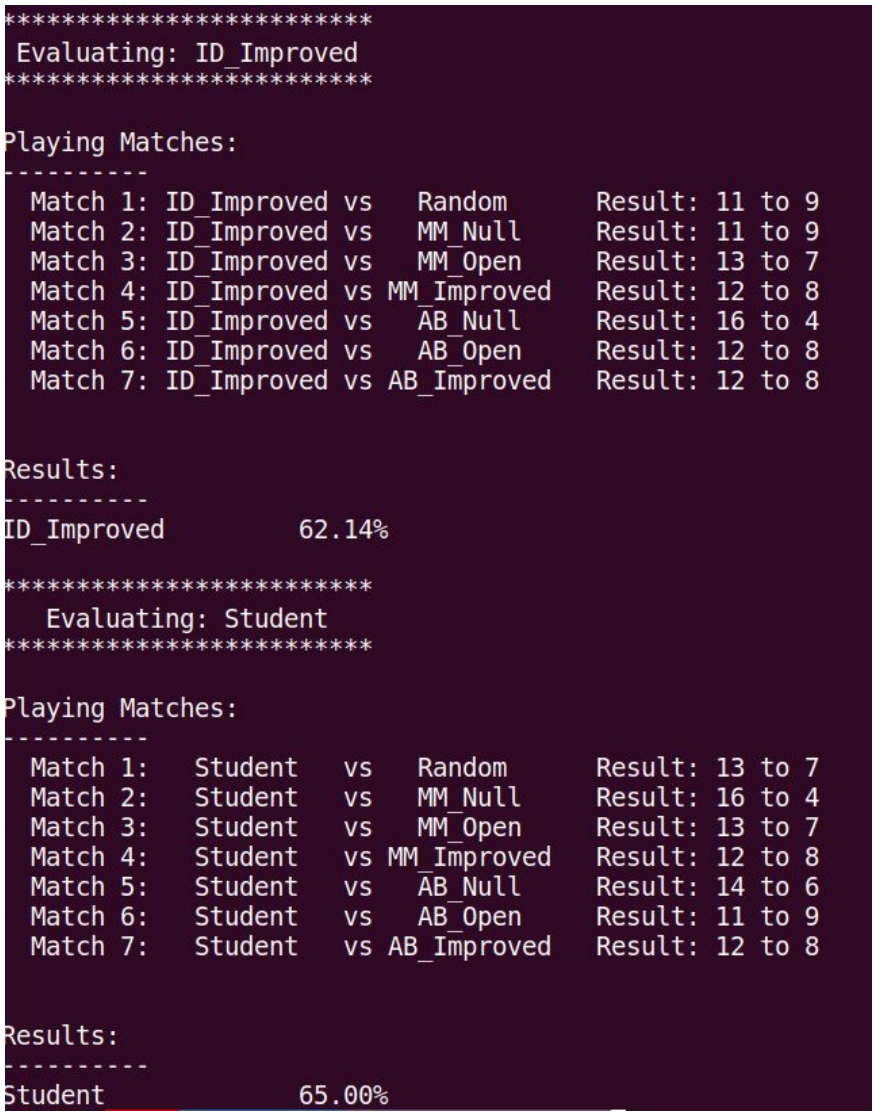
2. Host's legal moves/2 - opposites' legal moves:



Here the idea is trying to give a higher rate to opposites' legal moves. It means that it is trying to give a tougher calculation than the above one.

From the result, we can see that ID_improved has 59.29% winning rate and student has 63.57% winning rate.

3. Host's legal moves/Remain empty place - opposites' legal moves/ Remain empty place:



Here, the program is using the percentage of legal moves as the calculation rather than the actual number.

As we can see in the result that ID_improved has 62.14% winning rate and Student has 65% wining rate.

As a result, the second method, Host's legal moves/2 - opposites' legal moves, will be chosen as my final scoring function in my project, although the third one provides a high winning rate comparing to the rest.

In this project, my goal is to develop a heuristic such that Student outperforms ID_Improved. In this sense, we will find that the winning of student agent is higher than ID_Improved 0.72% in the first method, 4.28% higher in second method and 2.86% higher in the third one.

To conclude, the second method is the most suitable one based on the aims of our project.