

Agent Name: Shamooshak

Team Members: Saber Khakpash and Amir Malekzadeh

Department of Computing Science, University of Alberta

Our main idea is to predict next actions of the opponents and choose the best action regarding to our predicting. So we implemented two separate modules: prediction and action. The performance of our agent highly depends on the performance of the prediction module, so we tried to make it as precise as possible. These two modules will be explained separately below.

Prediction Module

This module tries to predict the next action of the opponents according to their previous actions. To do this we use some different predictors and mix their outputs with some weightings and improve these weights due to the distance of their results with what actually happens.

The output of each predictor is a probability for each of the twelve possible actions for each of the two opponents. The implemented predictors are the followings:

- **Constant Predictor**

The purpose of this predictor is to find the agents whose actions almost remain constant during the game. So this predictor simply returns the last action of each player as its predicted action.

- **Against us Predictor**

This predictor catches the agents who try to sit against our last action. This may happen frequently, because it is a good action if you can sit against another agent, so other players may assume that our agent may stay in its last place and try to sit against it.

- **Against Them predictor**

This predictor is the same as the previous one except this one assumes that the agent which we are predicting its action, will sit against the last place of the other opponent.

- **Best Response Predictor**

This predictor assumes that the agent will choose one of its best responses for the last round, so it predicts the probability of its best responses equal and the sum of them will equal to one.

- **Middle Predictor**

The idea of this predictor is the same as the previous one, but this one thinks the agent will select its middle best response to last round.

- **Sequence Predictor**

This predictor works well if the agents play in a pattern. It checks the last two rounds of the game and see if these situations have happened consequently before, and predicts the next action of agents according to their actions in same situations.

Action Module

This module decides which action to take regarding the result of the prediction module. According to the result of the prediction module it checks which place has the most probability to be occupied at the next round and chooses exactly the opposite place. At the beginning of the game the prediction module cannot work properly so we start the game with two constant actions.

How To Run Our Agent

We have a jar file named “ca.ualberta.cs.shamooshak.jar” which you should include into the class path and our class which should be run is “Shamooshak” which is not in the ca.ualberta.cs.shamooshak package.