MAS Assignment - 4

Automated Negotiations

Code Implementation of the Agent:

We made the agent with the reference of the paper GAIW_2021_paper_25. Here we skipped the user modelling because we already know the utility function for all the bids. We had to model the opponent and guess their preference profile and their utility function. We defined a class called opponents where we store the weights of the features and the items. We also had a map which contained the agentIds as the key value (this uniquely identifies opponents). The values in this map are the agent(opponents class) objects which will be changed based on the bid obtained from a given opponent.

The “opponents” class contains the iWeights, weights and frequency table for the given opponent and it also contains two functions where one is used to update the weights based on the given bid and the other is used to get the expected utility based on the given weights of the features and individual topic weights. Because of this opponents class, we get a new file after compiling the program which is the opponents.class file

$$\hat{f}'_k(j) = \begin{cases} 1 & \text{if } j = j_{\text{most}} \\ 0 & \text{else if } Z = 0 \\ \frac{h_k(j)}{Z} & \text{otherwise} \end{cases} .$$

.

$$\text{imp}_k = \frac{\sum_{i=1}^{J_k} \hat{f}'_k(i)\, h_k(i)}{\sum_{i=1}^{J_k} h_k(i)} = \frac{\sum_{i=1}^{J_k} \hat{f}'_k(i)\, h_k(i)}{Z + h_k(j_{\text{most}})} .$$

Now we go onto the negotiation strategy. We have implemented the strategy which is similar to the reference paper. We divide the negotiation into 4 parts.
First phase is the initial phase where we learn the opponents utility based on the frequency. We give the bid which gives us the highest utility in this phase and don’t accept any offers. In the second phase which is the discussion phase we give bids based on the time elapsed. We will give bids whose utility is just greater than ‘0.98 - 0.45*time*time’. This is a quadratic decrease (slow decrease). We don’t accept any offers in this phase. This ensures that we get a good utility in the end even if the opponent accepts the offer. We don’t accept any offer in this phase and learn the opponents utility based on the bids given by them. We steadily decrease the utility so

that the opponent learns our utility. The next phase is the consideration phase which is where we accept the reasonable bids. We have to compute the expected utility gain based on the nature of the opponent. If the opponent is a hardliner the expected utility gain would be equal to the reserve value. If the opponent is conceeder then the expected utility is 1. We compute the hardliner factor q which is a value between [0,1] where q =0 for an opponent indicates that the opponent is a hardliner and q= 0 indicates that the opponent is a conceeder. We calculate this value based on the highest bid offered by the opponent.

	initial phase	discussion phase	consideration phase	joint phase	final round
Bidding Strategy	ω_{best}	random from $\mathcal{S} = \{\omega \mid \hat{f}(\omega) > 0.95 - 0.5 \cdot time^2, \omega \in \Omega\}$	random from $\mathcal{T} = \{\omega \mid \hat{f}(\omega) > \max(0.95 - 0.5 \cdot time^2, \omega_{maxjoint}), \omega \in \mathcal{S}\}$	$\omega_{maxjoint}$	
Acceptance Strategy	doesn't accept any offers		over $\mathbb{E}[u_f] + (1 - \mathbb{E}[u_f])(1 - time)$		over reserve

The q value would be less if the highest bid reaches 1 and low if it is less than or equal to the reserve value. With this we get an expected value from the opponent till the final round. We then use a quadratic function which decreases from 1 to the highest bid offered's utility based on the time elapsed. Any bid value higher than this will be accepted. We give a bid which is the maximum value out of the two values 0.98-0.45*time*time and the max joint utility.

$$q = \frac{1}{1 + 2\left(\frac{\max(\hat{f}(\omega_{\text{reserve}}), \hat{f}(\omega_{\text{bestOffered}})) - \hat{f}(\omega_{\text{reserve}})}{1 - \max(\hat{f}(\omega_{\text{reserve}}), \hat{f}(\omega_{\text{bestOffered}}))}\right)}$$

Max joint utility is (2-0.2*time*time)* userutility(w) + summation of all the opponents' utility. We then give the bid which has the maximum joint utility. We give the user utility more weight compared to the other utilities. So we get an extra edge over other agents. We have the joint phase where we give the max joint utility bid as our bid and accept bids based on the expected utility value. Final round phase is where we accept a bid if it is over reserve if the last bid is not ours and provide the bid which gives the maxjoint when it is our turn to bid.

After Compiling add the Group5_Agent.class instead of Group5_Agent$1.class

Improvements from D3:

We modelled the opponents based on the frequency of each bid. We also changed the negotiation strategy as we divided the timeline into 4 parts as explained above. We considered every agent’s utility (joint utility) everytime we gave a bid from the bid space. This ensures the rationality of the bids.