INSIGHT

Intelligent Negotiation Strategy for Information Games with Hidden Tactics

Authors: Justin Haddad, Maksym Bondarenko, Phebe Lew
Mentors: George Dragomir, Brace Hanyang Zhao

Introduction

From politics to business, economics and finance, the art of negotiation has always been seen as a crucial means of "dividing utility" between multiple parties. Our model "INSIGHT" (Intelligent Negotiation Strategy for Information Games with Hidden Tactics) employs machine learning and game theory techniques to develop a model able to make optimal negotiation bids in an environment with hidden information.

Our aim was to train INSIGHT to extract information from a foreign environment to come up with an optimal bidding strategy, as well as test its ability to compete with other strategic bidding models.

INSIGHT builds upon existing work [1] by creating a reinforcement learning model using modular Q-learning to adapt to and optimize performance in new multi-seller environments. In essence, we have simulated a generalized bargaining situation containing multiple agents and unknown environment, both of which our model learns and responds to.

Goals & Hypotheses

Our research aims to build on *Multiagent reinforcement learning in the Iterated Prisoner's Dilemma (Sandholm & Crites, 1996) [4]* , which found that Q-learning algorithms can successfully learn to outperform Heuristic algorithms like Tit-for-Tat, but found inconclusive evidence that Q-learning algorithms can perform when competing against each other.

We had two objectives for INSIGHT in both single and multi-seller environments:

1. Optimise performance (maximise profit) in an unknown environment.
2. Cooperate with other sellers when necessary to raise profits and enter a situation where pareto efficiency is higher, similar to how corporations collude to prevent excessive predatory pricing from undercutting their profits.

Methods

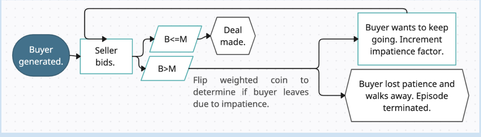
Model & Game Episode:

Players:

- Sellers:
  - Profit maximising goal. Profit is calculated as the difference between breakeven value *V* and the selling price of the product.
  - May make any bid B above or at their breakeven value V, and bids made to the same buyer must be monotonically decreasing (i.e. offer cheaper and cheaper).
- Buyers:
  - Randomly generated to have a maximum price M anywhere within a certain predetermined range.
  - Deterministic, accepting the first offer at or below this price, and rejecting all other offers. Sellers do not know this price.
  - Have an impatience increment, which determines the probability they walk away from negotiations and increases each round.

Game Episode:
The game is played over a large no. of episodes over many buyers but 1 seller. Each episode occurs as follows:

1. A buyer is randomly generated.
2. Sellers make a bid.
3. Buyer deterministically accepts bid if it is less than or equal to the price they are willing to buy at, i.e. $B <= M$.
   a. If buyer accepts, the episode is terminated and profits are calculated.
   b. If buyer does not accept, the episode continues in Step 4.
4. Flip a coin (weighted according to the impatience increment) to determine if the buyer leaves due to impatience.
   a. If buyer leaves, the entire EPISODE is terminated.
   b. If buyer does not leave, increase the impatience increment and continue from Step 2 (the seller makes another lower bid) with the higher impatience increment.



2-Seller Variant:
The two seller variant is almost identical to the one-seller variant, except:

1. Instead of a single seller making a bid, the 2 sellers make simultaneous bids.
2. Provided both bids are below the price the buyer is willing to buy at, the buyer accepts the lower bid, or flips a coin between both sellers if the bids are equal.

Key Concepts:

Reinforcement Learning: A machine learning training method based on rewarding desired behaviors while punishing undesired ones.

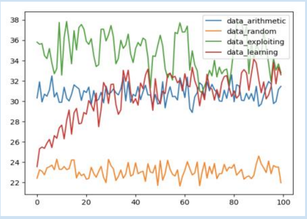
- Utilizes Markov Decision Processes (MDPs) - a mathematical framework used for sequential decision making under uncertainty where only the previous state affects the next.
- The Bellman Equation defines the relationship between the value of a state-action pair and values of its successor state-action pairs:
  $Q(s,a) = R(s,a) + \gamma \sum P(s'|s,a) * \max_{a'} Q(s', a')$
  where $Q(s,a)$ is the Q-value of taking action a in state s, $R(s,a)$ is the immediate reward of taking action a in state s, $P(s'|s,a)$ is the transition probability to the next state s' after taking action a in state s, and γ is the discount factor.

Tabular Q-Learning: Estimates the desirability (reward) of taking action a in state s by iteratively updating a table of Q-values based on experienced transitions and rewards.

- Is a specific temporal difference learning algorithm with an update rule of:
- $Q(s,a) \longleftarrow Q(s,a) + \alpha [R + \gamma \max_{a'}(Q(s',a')) - Q(s,a)]$
- Q-table is initialised to 0, but iteratively updating it derives the optimal policies even in environments with delayed rewards.
- α, the learning rate, determines the extent our model trades off exploring new strategies (exploration) over reusing the strategy that hitherto has generated the highest reward (exploitation). The learning rate can be adjusted over multiple rounds.
- By the law of large numbers, Q-learning converges to an optimal solution provided the reward function remains stationary.

Results & Discussion

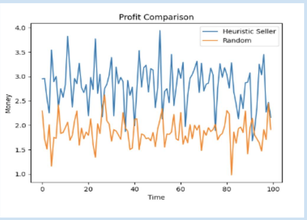
Single-Seller Approach:



For a single-seller approach, the Q-Learning algorithm eventually outperforms all other strategies. The red graph shows the progression of Q-learning algorithm as it "learns", and the green graphs shows the fully exploiting Q-learning algorithm, which slightly overperforms as compared to the learning algorithm. The yellow graph acts as a baseline to compare against, as it is simply a random strategy, and the blue is the heuristic strategy.

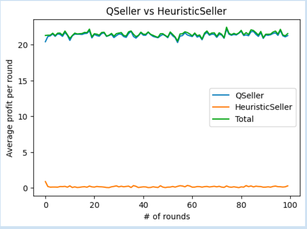
Multi-Seller Approach:

Heuristic vs Random



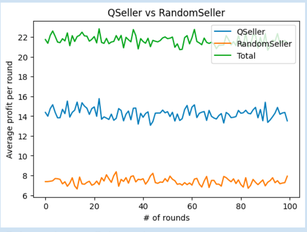
In the two-seller situation, we clearly see the Heuristic Seller outperforms the Random Seller in nearly every situation. This is much more interesting than in the single-seller situation, as a random seller has the potential to "steal" away any given buyer with its unpredictable moves, which thus renders credibility to the heuristic strategy.

Q-Learning vs Heuristic



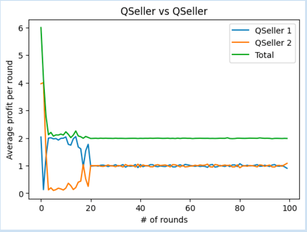
Here, the Q-Seller learns to out-compete the Heuristic Seller almost instantly (within the first round of 10,000 episodes). This shows how a set strategy will always fall prey to a Q-Seller, which will learn the strategy and learn exactly what moves it needs to make to beat it.

Q-Learning vs Random



In this situation however, the Q-Seller, represented by the blue line, does not out-perform the Random Seller quite as drastically as it had the Heuristic Seller. This is likely because the Q-Seller has a harder time predicting its opponents moves and thus knowing what the optimal move would be. The green line represents the total profit of the two sellers, or how well they exploit their given buyer environment. Of note in this experiment is that the Total profit made by the two combined sellers is lower than that of the Q-Seller against the Heuristic Seller

Q-Learning vs Q-Learning



When two Q-Sellers are pitted against each other, they eventually learn the only way they stand a chance to gain any sort of profit is to immediately offer the lowest possible price, of 1. This is why both their average profit is 1. This is extremely pareto-inefficient, as their total exploitation of the environment is close to 0.

Conclusions

- By implementing a combination of heuristical algorithms and Q-Learning/RL algorithms, INSIGHT learns to maximize their profit in any given situation, which can be directly beneficial to new sellers in unknown environments.
- Q-Learning Sellers do not compete well against each other, and need fine-tuning and other incentives to successfully maximize pareto efficiency.

Future Work

1. Deep reinforcement learning: our computational capacity to expand our state and consider more variables are limited by the tabular Q-table; using neural networks would increase the number of parameters we can take into account, including more buyers and sellers.
2. Natural Language Processing: Our work has focssed on the more quantitative aspects of negotiation, and we might expand on the qualitative aspect by using reinforcement learning and NLP to determine the appropriate tone of negotiations.
3. Colluding Sellers: Simulating cooperation between sellers to collectively raise prices will better reflect market conditions where dominant sellers hold a large sum of market share and have price-setting ability.

References

[1] Ali, S. N., Kartik, N., & Kleiner, A. (2023). Sequential veto bargaining with incomplete information [Preprint]. arXiv. http://arxiv.org/abs/2202.02462
[2] Kopalle, P. K., & Shumsky, R. A. (2012). Game theory models of pricing.
[3] Zou Y, Zhan W, Shao Y (2014) Evolution with Reinforcement Learning in Negotiation. PLoS ONE 9(7): e102840.https://doi.org/10.1371/journal.pone.0102840
[4] Sandholm, T. W., & Crites, R. H. (1996). Multiagent reinforcement learning in the Iterated Prisoner's Dilemma. Biosystems, 37(1-2), 147-166. https://doi.org/10.1016/0303-2647(95)01551-5

Our Github Repo: