Learning Better Trading Dialogue Policies by Inferring Opponent Preferences

(Extended Abstract)

Ioannis Efstathiou
Interaction Lab
Heriot-Watt University
ie24@hw.ac.uk

Oliver Lemon
Interaction Lab
Heriot-Watt University
o.lemon@hw.ac.uk

ABSTRACT

Negotiation dialogue capabilities have been identified as important in a variety of application areas. In prior work, it was shown how Reinforcement Learning (RL) agents can learn to use implicit and explicit manipulation moves in dialogue to manipulate their adversaries in non-cooperative trading games. We now show that trading dialogues are more successful when the RL agent builds an opponent model – an estimate of the (hidden) goals and preferences of the adversary – and learns how to exploit them. We explore a variety of state space representations for the preferences of trading adversaries, including one based on Conditional Preference Networks (CP-NETS), used for the first time in RL. We show that representing adversary preferences leads to significant improvements in trading success rates.

1. INTRODUCTION

Work on automated conversational systems has been focused on cooperative dialogue. However, non-cooperative dialogues, where an agent may act to satisfy its own goals rather than those of other participants, are also of practical and theoretical interest [5]. For example, it may be useful for a dialogue agent not to be fully cooperative when trying to gather information from a human, or when trying to persuade, argue, or debate, or when trying to sell something, or when trying to detect illegal activity, or in the area of believable characters in video games and educational simulations [5]. Another important area in which non-cooperative dialogue behaviour is desirable is in negotiation [8].

Recently it has been shown that when given the ability to perform both cooperative and non-cooperative (manipulative) dialogue moves, an agent can learn to bluff and to lie so as to win games more often, under various conditions such as risking penalties for being caught in deception – against a variety of adversaries [3]. Here we investigate the converse problem: can we develop a negotiator that models its adversaries' preferences and goals, and can learn how to use such models to improve its trading performance?

2. THE TRADING GAME "CATAN"

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

To investigate trading dialogues in a controlled setting we used a 2-player version of the board game "Catan", discussed in [4]. We call the 2 players the "adversary" and the "Reinforcement learning agent" (RLA). Trade occurs through proposals that may lead to acceptance or rejection from the adversary. In an agent's proposal (turn) only one 'give 1-for-1' or 'give 1-for-2' trading proposal may occur, or nothing.

State space for adversary preferences.

To overcome issues related to long training times and high memory demands, we implemented a state encoding mechanism that converts all of our trading game states to a significantly smaller number of states in a compressed representation, as discussed in [4]. The new state representation consists of 10 slots, and the first 5 (for resources) are encoded. The subsequent 5 state slots are used to refer to the adversary's inferred preferences. One of the two RLAs uses 2 different symbols to represent whether or not the adversary wants a particular resource (i.e. accepts it), and the other RLA uses 3 symbols because it also considers resources that the adversary does not want (i.e. it rejects that resource).

3. THE RLAS AND THE ADVERSARY

The **Adversary** sets a random goal. Hence its preferences (and therefore its responses to the RLA's proposals) will change according to the goal. We assume that its resources are infinite[1]. The RLAs are: **the Baseline RLA (BRLA),** which does not keep track of the adversarial preferences, **the Positive Preferences RLA (PPRLA),** which keeps track of the preferences and estimates the adversary's goal based on what resources the adversary has accepted in the trading dialogue so far, and **the Negative-Positive Preferences RLA (NPRLA),** which is as the previous but it represents rejected resources too.

4. EXPERIMENTS AND RESULTS

All agents are compared in respect of the percentage of trading games in which they achieve their goal (i.e. to get the resources required to build a city), within a sequence of 7 trading moves.

BRLA vs. Adversary: (Baseline).

The Baseline RLA (i.e. with no opponent modelling) played 250K games against the Adversary. The agent then played 20K testing games and scored a 28.1% success rate.

[1]Experiments conducted against the adversary with finite resources showed similar results but with lower scores.

PPRLA vs. Adversary .
We also trained the Positive Preferences RLA against the Adversary for 250K games, achieving a 44.2% success rate.

NPRLA vs. Adversary .
Likewise, the NPRLA achieved a success rate of 52.5%.

RLA Name	Success Rate
BRLA	28.1%
PPRLA	44.2%*
NPRLA	52.5%*

Table 1: ***Results (*= $p < 0.05$ over BRLA)***

5. CONDITIONAL PREFERENCE MODELS

We now extend the previous experiments to the cases where the RLA uses a CP-NET [1] to model preferences.

5.1 New extended state representation

The RLA has now a state representation which consists of 25 features: 5 for the resources (encoded) and 20 for the CP-NET adversarial preferences. An example of such a feature (preference) might be $s \rightarrow r$, that is the adversary's preference for sheep over rocks. We use again 3 characters to represent the RLA's knowledge about the adversary's preference for each of the CP-NET's 20 possibilities.

CPNET RLA vs. Adversary.
We first trained the CP-NET RLA against the adversary over 250K games. After 20K testing games this RLA achieved a success rate of 36.1%. The performance is better than that of the BRLA (28.1%), proving that the CP-NET improves the RL procedure, but it is worse than that of the PPRLA (44.2%) and that of the NPRLA (52.5%) The reason was the significantly larger state representation. Hence we ran another experiment for 1.5 million training games, and in the following 20k testing games the CP-NET RLA's performance increased to 46.9%. That result was better than that of the PPRLA in 250K training games but not as high as that of the NPRLA in 250k training games. Thus we ran another experiment for 2.5m training games. The performance of the CP-NET RLA was 50.3%. We ran a final experiment for 5 million training games, which resulted in a similar performance (52.4%) to that of the NPRLA.

6. MULTIPLAYER JSETTLERS

The experiments of this Section are all conducted using JSettlers [7] (full multi-player version of "Catan"). The goal is now to win the overall game (via building), rather than succeed in each trading dialogue. Here we tested the NPRLA policy against expert hand-crafted rule-based agents (which we call "Bots") which use complex heuristics to trade and to build pieces on the board. These Bots are the "benchmark" agents described in [6], and show strong performance, for example when compared to human play.

6.1 Our trained NPRLA Settlers agents

Both of our trained NPRLA agents are in fact a Bot modified so as to make offers based on the NPRLA learned policies, instead of using those of the Bot. The NPRLA maintains a history of all the opponents' (Bots) resource preferences during the game, treating them all as one adversary.

NPRLA vs. 3 Settlers Bots.
Our trained agent played 50K games versus the Bots and resulted in a win rate (in a 4-player game) of 25.19%, while those of the 3 Bots were 24.96%, 25.24% and 24.61% respectively. In 10K games the performances were 24.98% for the NPRLA, and 24.8%, 24.4% and 25.82% for the 3 Bots. The similarity of the results suggests that our agent knows how to trade towards a particular goal, like the Bots, even though it was not trained specifically to beat them.

7. CONCLUSIONS

We show that RL trading agents which track adversarial preferences outperform agents which don't. Furthermore, a RLA who considers the positive and negative preferences outperforms one which considers only the accepted trades by 8.3% success rate. Note that these preferences are inferred during learning since the beginning of each new game. We also extended this approach by using CP-NETs [1] showing that it results in a positive difference of 8% compared to the winning performance of the RLA who doesn't have preference information. We also evaluated the performance of our best trader (NPRLA) in the full multi-agent game of Settlers, compared to expert hand-crafted rule-based agents [6]. We found that our trained trading policies performed as well as these hand-crafted agents, suggesting that data-driven policy training can result in very competitive trading strategies, without expert hand-crafting. We also suggest that bilateral training environments may suffice for multilateral non-cooperative trading scenarios for effective RL, providing that efficient selection of the state representation and of the actions has been made. Deep Reinforcement Learning is also a promising research direction [2].

Acknowledgments

This work is funded by the ERC grant no. 269427 (STAC).

REFERENCES

[1] C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. CP-nets : A Tool for Representing and Reasoning with Conditional *Ceteris Paribus* Preference Statements. *Journal of AI Research*, 21:135–191, 2004.

[2] H. Cuayahuitl, S. Keizer, and O. Lemon. Strategic Dialogue Management via Deep Reinforcement Learning. In *Proc. NIPS*, 2015.

[3] I. Efstathiou and O. Lemon. Learning to manage risks in non-cooperative dialogues. In *Proc. SemDial*, 2014.

[4] I. Efstathiou and O. Lemon. Learning non-cooperative dialogue policies to beat opponent models: "the good, the bad and the ugly". In *Proc. SemDial*, 2015.

[5] K. Georgila and D. Traum. Reinforcement learning of argumentation dialogue policies in negotiation. In *Proc. INTERSPEECH*, 2011.

[6] M. Guhe and A. Lascarides. Game Strategies in *The Settlers of Catan*, booktitle = Proc. IEEE Conference on Computational Intelligence in Games. 2014.

[7] R. Thomas and K. Hammond. Java settlers: a research environment for studying multi-agent negotiation. In *Proc. of IUI '02*, pages 240–240, 2002.

[8] D. Traum. Extended abstract: Computational models of non-cooperative dialogue. In *Proc. of SIGdial Workshop on Discourse and Dialogue*, 2008.