Reinforcement Learning of Two-Issue Negotiation Dialogue Policies

Kallirroi Georgila
Institute for Creative Technologies, University of Southern California
12015 Waterfront Drive, Playa Vista, CA 90094, USA
kgeorgila@ict.usc.edu

Abstract

We use hand-crafted simulated negotiators (SNs) to train and evaluate dialogue policies for two-issue negotiation between two agents. These SNs differ in their goals and in the use of strong and weak arguments to persuade their counterparts. They may also make irrational moves, i.e., moves not consistent with their goals, to generate a variety of negotiation patterns. Different versions of these SNs interact with each other to generate corpora for Reinforcement Learning (RL) of argumentation dialogue policies for each of the two agents. We evaluate the learned policies against hand-crafted SNs similar to the ones used for training but with the modification that these SNs no longer make irrational moves and thus are harder to beat. The learned policies generally do as well as, or better than the hand-crafted SNs showing that RL can be successfully used for learning argumentation dialogue policies in two-issue negotiation scenarios.

1 Introduction

The *dialogue policy* of a dialogue system decides on what dialogue move (also called *action*) the system should make given the dialogue context (also called *dialogue state*). Building hand-crafted policies is a hard task, and there is no guarantee that the resulting policies will be optimal. This issue has motivated the dialogue community to use statistical methods for automatically learning dialogue policies, the most popular of which is Reinforcement Learning (RL) (Szepesvári, 2010).

To date, RL has been used mainly for learning dialogue policies for slot-filling applications such as restaurant recommendations (Williams and Young, 2007; Chandramohan et al., 2010; Jurčíček et al., 2012; Gašić et al., 2012), flight reservations (Henderson et al., 2008), sightseeing recommendations (Misu et al., 2010), appointment scheduling (Georgila et al., 2010), technical support (Janarthanam and Lemon, 2010), etc., largely ignoring other types of dialogue. RL has also been applied to question-answering (Misu et al., 2012) and tutoring domains (Tetreault and Litman, 2008; Chi et al., 2011). There has also been some work on applying RL to the more difficult problem of learning negotiation policies (Heeman, 2009; Paruchuri et al., 2009; Georgila and Traum, 2011a; Georgila and Traum, 2011b; Nouri et al., 2012), which is the topic of this paper.

In negotiation dialogue the system and the user have opinions about the optimal outcomes and try to reach a joint decision. Dialogue policy decisions are typically whether to present, accept, or reject a proposal, whether to compromise, etc. Rewards may depend on the type of policy that we want to learn. For example, a cooperative policy should be rewarded for accepting proposals.

Recently, Georgila and Traum (2011a; 2011b) learned argumentation dialogue policies for negotiation against users of different cultural norms in a one-issue negotiation scenario. We extend this work by learning argumentation policies in a two-issue negotiation setting. We aim to learn system (or agent) policies that will persuade their interlocutor (a human user or another agent) to agree on the system's preferences.

Our research contribution is two-fold: First, to our knowledge this is the first study that uses RL for learning argumentation policies in a two-issue negotiation scenario and one of the few studies on using RL for negotiation. Second, for the first time, we learn policies for agents with different degrees of persuasion skills, i.e., agents that provide strong or weak arguments.

Section 2 introduces RL, and section 3 describes our two-issue negotiation domain and our learning methodology. Section 4 presents our evaluation results and section 5 concludes.

Proceedings of the SIGDIAL 2013 Conference, pages 112–116,
Metz, France, 22-24 August 2013.

2 Reinforcement Learning

Reinforcement Learning (RL) is a machine learning technique used to learn the policy of an agent (Szepesvári, 2010). RL is used in the framework of Markov Decision Processes (MDPs) (Szepesvári, 2010) or Partially Observable Markov Decision Processes (Williams and Young, 2007). In this paper we use MDPs.

An MDP is defined as a tuple (S, A, P, R, γ) where S is the set of states that the agent may be in, A is the set of actions of the agent, $P : S \times A \rightarrow P(S, A)$ is the set of transition probabilities between states after taking an action, $R : S \times A \rightarrow \Re$ is the reward function, and $\gamma \in [0, 1]$ a discount factor weighting long-term rewards. At any given time step i the agent is in a state $s_i \in S$. When the agent performs an action $\alpha_i \in A$ following a policy $\pi : S \rightarrow A$, it receives a reward $r_i(s_i, \alpha_i) \in \Re$ and transitions to state $s_i^{'}$ according to $P(s_i^{'}|s_i, \alpha_i) \in P$. The quality of the policy π followed by the agent is measured by the *expected future reward* also called Q-function, $Q^{\pi} : S \times A \rightarrow \Re$.

To estimate the Q-function we use Least-Squares Policy Iteration (LSPI) (Lagoudakis and Parr, 2003; Li et al., 2009). LSPI can learn directly from a corpus of dialogues and is sample efficient. We use linear function approximation of the Q-function. Thus $Q(s, \alpha) = \sum_{i=1}^{k} w_i \phi_i(s, \alpha)$ where s is the state that the agent is in and α the action that it performs in this state, and $\hat{w}$ is a vector of weights w_i for the feature functions $\phi_i(s, \alpha)$. The magnitude of a weight w_i shows the contribution of the feature $\phi_i(s, \alpha)$ to the $Q(s, \alpha)$ value.

3 Learning Argumentation Policies

In our experiments, two agents negotiate on two issues that are independent of each other. Each issue may have three possible outcomes. Our approach can be applied to any such issues. For the sake of readability, from now on we will use a negotiation scenario in which Agents 1 and 2 are having a party and need to agree on the type of food that will be served (Thai, Italian, Mexican) and the day of the week that the party will be held (Friday, Saturday, Sunday). Agents 1 and 2 have different goals. Table 1 shows the points that Agents 1 and 2 earn for each negotiation outcome.

We build hand-crafted simulated negotiators (SNs) for the two agents that interact with each other to generate simulated corpora. The SNs differ not only in their goals but also in whether they use strong or weak arguments to persuade their counterparts, and sometimes make irrational moves, i.e., moves not consistent with their goals. For example, Agent 1 may reject an offer for "Thai" food, and Agent 2 may offer or accept "Friday". This is to generate a variety of negotiation patterns. There is also some randomness regarding whether the SN will start the conversation by a direct offer or by providing an argument.

	Agent 1	**Agent 2**
Food type		
Thai	200	0
Italian	100	40
Mexican	0	80
Day of the week		
Friday	80	0
Saturday	40	100
Sunday	0	200

Table 1: Rewards for Agents 1 and 2.

The SNs for Agents 1 and 2 can choose among 13 actions: "offer-Thai", "offer-Italian", "offer-Mexican", "offer-Friday", "offer-Saturday", "offer-Sunday", "provide-argument-Thai", "provide-argument-Mexican", "provide-argument-Friday", "provide-argument-Sunday", "accept", "reject", "release-turn". In our setup Agents 1 and 2 do not provide arguments for "Italian" or "Saturday" since these are acceptable options for both agents. Because Agent 1 cares more about the food type and Agent 2 cares more about the day there is potential for trade-offs, i.e., "I'll give you the food type that you want if you agree on the day that I want". So we have one more action "trade-off" which is basically a combined action "offer-Thai, offer-Sunday". The two agents have to agree on both issues for the dialogue to end. If there is no agreement in 40 turns then the dialogue stops.

Note that for testing our learned policies (see section 4) we use a rationalized version of these SNs. For example, Agent 1 never offers "Sunday" and never accepts "Mexican". We will refer to the SNs that exhibit some degree of randomness and irrationality as "semi-rational" and the SNs that always behave rationally as "rational".

For training, 4 corpora are generated (50,000 dialogues each) using different SNs, each of which is limited to using either strong or weak arguments: SN for Agent 1 with strong arguments vs. SN for Agent 2 with strong arguments, SN for Agent 1 with strong arguments vs. SN for Agent 2

with weak arguments, SN for Agent 1 with weak arguments vs. SN for Agent 2 with strong arguments, and SN for Agent 1 with weak arguments vs. SN for Agent 2 with weak arguments.

We use LSPI to learn policies directly from the 4 corpora. Each agent is rewarded only at the end of the dialogue based on the agreement. So if the outcome is "Thai" and "Saturday" Agent 1 will earn 240 points and Agent 2 100 points. We set a small reward +1 point for each policy action taken. Table 2 shows our state representation.

The first 10 state variables are self-explanatory. Below we explain how the "counter" variables work. Initially the counter for "Thai" arguments is set to 0 and Agent 2 supports food type "Mexican". Every time the policy of Agent 1 provides an argument in favor of "Thai", the counter for "Thai" arguments is increased by 1 and the counter for "Mexican" arguments is decreased by 1 (like a penalty). Every time the policy of Agent 1 argues in favor of "Mexican" the counter for "Thai" arguments is decreased by 1 and the counter for "Mexican" arguments is increased by 1. When the counter for "Thai" arguments becomes 3, then the state variable "Thai-argument-counter-reached-threshold" becomes "yes" and Agent 2 is ready to yield to the demands of Agent 1. This threshold of 3 was set empirically after experimentation. Likewise for the rest of the "counter" variables. We also account for both strong and weak arguments. When the arguments of an agent are weak, even if the corresponding counters exceed the predefined threshold and the associated state variables change from "no" to "yes", the behavior of their interlocutor will not change. This is to simulate the fact that weak arguments cannot be persuasive. The release action counter works similarly. Initially it is 0 but after 4 consecutive actions of the same speaker it is set to 1 to ensure that the turns are not very long.

There are 786,432 possible states and 11,010,048 possible Q-values (state-action pairs). We use linear function approximation with 1,680 manually selected features. The rationale for selecting these features is as follows: We associate the action "offer-Thai" with the state variables "current-day-accepted", "Thai-rejected", "Italian-rejected", "Mexican-rejected", "Thai-argument-counter-reached-threshold", and "Mexican-argument-counter-reached-threshold". Thus we assume that the values of the other state variables are irrelevant. This is an approximation (to keep the number of features manageable) that has drawbacks, e.g., we may have an "offer-Thai" action even though the food type agreed so far is "Thai" (because there is no feature to associate the currently accepted food type value with a "Thai" offer). With this configuration we end up having $4 \times 2^5 = 128$ binary features just for the action "offer-Thai". Similarly, features are selected for the rest of the actions.

Current offer on the table (null/Thai/Italian/ Mexican/Friday/Saturday/Sunday/trade-off)
By whom is the current offer on the table (null/Agent1/Agent2)
Currently accepted food type (null/Thai/Italian/Mexican)
Currently accepted day (null/Friday/ Saturday/Sunday)
Has food type Thai been rejected? (no/yes)
Has food type Italian been rejected? (no/yes)
Has food type Mexican been rejected? (no/yes)
Has day Friday been rejected? (no/yes)
Has day Saturday been rejected? (no/yes)
Has day Sunday been rejected? (no/yes)
Has counter for food type Thai arguments reached threshold? (no/yes)
Has counter for food type Mexican arguments reached threshold? (no/yes)
Has counter for day Friday arguments reached threshold? (no/yes)
Has counter for day Sunday arguments reached threshold? (no/yes)
Has release action counter reached threshold (no/yes)

Table 2: State variables that we keep track of and all the possible values they can take.

We partition each one of our 4 simulated corpora into 5 subsets of 10,000 dialogues each. Each partition is processed independently and will be referred to as trial. We train policies for each trial of each corpus type (20 policies for each agent). Thus we end up with the following 4 types of policies for Agent 1 (and likewise for the policies of Agent 2): Agent 1 with strong arguments trained against Agent 2 with strong arguments (Agent 1 S(S)); Agent 1 with strong arguments trained against Agent 2 with weak arguments (Agent 1 S(W)); Agent 1 with weak arguments trained against Agent 2 with strong arguments (Agent 1 W(S)); and Agent 1 with weak arguments trained against Agent 2 with weak arguments (Agent 1 W(W)).

	Policy Score	Opponent Score	Policy #Actions	Opponent #Actions	Policy #Turns	Opponent #Turns
Agent 1 S(S) vs. Agent 2 S	214.3	164.3	7.6	6.2	2.0	1.6
Agent 1 S(S) vs. Agent 2 W	214.1	164.5	7.4	6.1	2.0	1.6
Agent 1 S(W) vs. Agent 2 S	213.9	165.1	7.6	6.2	2.0	1.6
Agent 1 S(W) vs. Agent 2 W	214.1	164.7	7.4	6.1	2.0	1.6
Agent 1 W(S) vs. Agent 2 S	192.4	196.5	9.1	8.5	2.5	2.4
Agent 1 W(S) vs. Agent 2 W	197.9	198.9	7.6	7.0	2.1	1.9
Agent 1 W(W) vs. Agent 2 S	195.0	197.9	8.8	8.5	2.5	2.4
Agent 1 W(W) vs. Agent 2 W	198.1	199.0	7.7	7.0	2.2	2.0

Table 3: Results of different training and testing combinations for learned policies of Agent 1 and rational SNs for Agent 2.

4 Evaluation

Each policy of Agent 1 resulting from a trial is evaluated against two hand-crafted SNs for Agent 2, one where Agent 2 provides strong arguments (Agent 2 S) and one where Agent 2 provides weak arguments (Agent 2 W). So for the condition "Agent 1 with strong arguments trained against Agent 2 with strong arguments (Agent 1 S(S))" we have 5 policies, each of which interacts with "Agent 2 S" (or "Agent 2 W"). We calculate the averages of the earned points for each of the agents, of the number of actions per dialogue of each agent, and of the number of turns per dialogue of each agent, over 10,000 dialogues per policy. Likewise for the policies of Agent 2. Note that the SNs used in the evaluation do not behave irrationally like the ones used for training, and thus are harder to beat.

In Table 3 we can see the results for the policy of Agent 1. Results for the policy of Agent 2 are similar given that the goals of Agent 2 mirror the goals of Agent 1. As we can see, the policy of Agent 1 with strong arguments learned to provide the appropriate arguments and make Agent 2 agree on "Thai" and "Friday" or "Saturday". When the policy of Agent 1 provides only weak arguments it cannot get day "Friday" but it can secure a trade-off. This is because both the learned policies and the SNs usually accept trade-off offers (due to the way the hand-crafted SNs were constructed). We also performed tests with SNs that did not propose or accept as many trade-offs. This arrangement favored the policy of Agent 1 with strong arguments, and hurt the performance of the policy of Agent 1 with weak arguments playing against Agent 2 with strong arguments. This shows that trade-offs help the weaker negotiators.

Furthermore, we experimented with testing on semi-rational SNs similar to the ones used for training and the results were better for the policy of Agent 1 with weak arguments and worse for the policy of Agent 1 with strong arguments. So like trade-offs a semi-rational SN favors the weaker negotiators.

5 Conclusion

We learned argumentation dialogue policies for two-issue negotiation, using simulated corpora generated from the interaction of two hand-crafted SNs that differed in their goals and in the use of strong and weak arguments to persuade their counterparts. These SNs sometimes made random or irrational moves to generate a variety of negotiation patterns.

We used these simulated corpora and RL to learn argumentation dialogue policies for each of the two agents. Each of the learned policies was evaluated against hand-crafted SNs similar to the ones used for training but with the modification that these SNs no longer made irrational moves and thus were harder to beat. The policies generally did as well as, or better than the hand-crafted SNs showing that RL can be successfully used for learning argumentation dialogue policies in two-issue negotiation scenarios.

For future work we would like to use automatic feature selection (Li et al., 2009; Misu and Kashioka, 2012) and learn policies for more than two issues and more than three outcomes per issue. Selecting features manually is a difficult process that requires a lot of experimentation and trial-and-error.

Acknowledgments

This work was funded by the NSF grant #1117313.

References

Senthilkumar Chandramohan, Matthieu Geist, and Olivier Pietquin. 2010. Sparse approximate dynamic programming for dialog management. In *Proc. of the Annual SIGdial Meeting on Discourse and Dialogue*, pages 107–115, Tokyo, Japan.

Min Chi, Kurt VanLehn, Diane Litman, and Pamela Jordan. 2011. Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. *User Modeling and User-Adapted Interaction*, 21(1-2):137–180.

Milica Gašić, Matthew Henderson, Blaise Thomson, Pirros Tsiakoulis, and Steve Young. 2012. Policy optimisation of POMDP-based dialogue systems without state space compression. In *Proc. of the IEEE Workshop on Spoken Language Technology (SLT)*, pages 31–36, Miami, FL, USA.

Kallirroi Georgila and David Traum. 2011a. Learning culture-specific dialogue models from non culture-specific data. In *Proc. of HCI International, Lecture Notes in Computer Science Vol. 6766*, pages 440–449, Orlando, FL, USA.

Kallirroi Georgila and David Traum. 2011b. Reinforcement learning of argumentation dialogue policies in negotiation. In *Proc. of Interspeech*, pages 2073–2076, Florence, Italy.

Kallirroi Georgila, Maria K. Wolters, and Johanna D. Moore. 2010. Learning dialogue strategies from older and younger simulated users. In *Proc. of the Annual SIGdial Meeting on Discourse and Dialogue*, pages 103–106, Tokyo, Japan.

Peter A. Heeman. 2009. Representing the reinforcement learning state in a negotiation dialogue. In *Proc. of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Merano, Italy.

James Henderson, Oliver Lemon, and Kallirroi Georgila. 2008. Hybrid reinforcement/supervised learning of dialogue policies from fixed datasets. *Computational Linguistics*, 34(4):487–511.

S. Janarthanam and O. Lemon. 2010. Adaptive referring expression generation in spoken dialogue systems: Evaluation with real users. In *Proc. of the Annual SIGdial Meeting on Discourse and Dialogue*, pages 124–131, Tokyo, Japan.

Filip Jurčíček, Blaise Thomson, and Steve Young. 2012. Reinforcement learning for parameter estimation in statistical spoken dialogue systems. *Computer Speech and Language*, 26(3):168–192.

Michail G. Lagoudakis and Ronald Parr. 2003. Least-squares policy iteration. *Journal of Machine Learning Research*, 4:1107–1149.

Lihong Li, Jason D. Williams, and Suhrid Balakrishnan. 2009. Reinforcement learning for dialog management using least-squares policy iteration and fast feature selection. In *Proc. of Interspeech*, pages 2475–2478, Brighton, United Kingdom.

Teruhisa Misu and Hideki Kashioka. 2012. Simultaneous feature selection and parameter optimization for training of dialogue policy by reinforcement learning. In *Proc. of the IEEE Workshop on Spoken Language Technology (SLT)*, pages 1–6, Miami, FL, USA.

Teruhisa Misu, Komei Sugiura, Kiyonori Ohtake, Chiori Hori, Hideki Kashioka, Hisashi Kawai, and Satoshi Nakamura. 2010. Modeling spoken decision making dialogue and optimization of its dialogue strategy. In *Proc. of the Annual SIGdial Meeting on Discourse and Dialogue*, pages 221–224, Tokyo, Japan.

Teruhisa Misu, Kallirroi Georgila, Anton Leuski, and David Traum. 2012. Reinforcement learning of question-answering dialogue policies for virtual museum guides. In *Proc. of the Annual SIGdial Meeting on Discourse and Dialogue*, pages 84–93, Seoul, South Korea.

Elnaz Nouri, Kallirroi Georgila, and David Traum. 2012. A cultural decision-making model for negotiation based on inverse reinforcement learning. In *Proc. of the Annual Meeting of the Cognitive Science Society (CogSci)*, pages 2097–2102, Sapporo, Japan.

P. Paruchuri, N. Chakraborty, R. Zivan, K. Sycara, M. Dudik, and G. Gordon. 2009. POMDP based negotiation modeling. In *Proc. of the IJCAI Workshop on Modeling Intercultural Collaboration and Negotiation (MICON)*.

Csaba Szepesvári. 2010. *Algorithms for Reinforcement Learning*. Morgan & Claypool Publishers.

Joel R. Tetreault and Diane J. Litman. 2008. A reinforcement learning approach to evaluating state representations in spoken dialogue systems. *Speech Communication*, 50(8-9):683–696.

Jason D. Williams and Steve Young. 2007. Scaling POMDPs for spoken dialog management. *IEEE Trans. on Audio, Speech, and Language Processing*, 15(7):2116–2129.