Reinforcement Learning of Argumentation Dialogue Policies in Negotiation

Kallirroi Georgila and David Traum

Institute for Creative Technologies, University of Southern California, Playa Vista, CA, USA

{kgeorgila,traum}@ict.usc.edu

Abstract

We build dialogue system policies for negotiation, and in particular for argumentation. These dialogue policies are designed for negotiation against users of different cultural norms (individualists, collectivists, and altruists). In order to learn these policies we build simulated users (SUs), i.e. models that simulate the behavior of real users, and use Reinforcement Learning (RL). The SUs are trained on a spoken dialogue corpus in a negotiation domain, and then tweaked towards a particular cultural norm using hand-crafted rules. We evaluate the learned policies in a simulation setting. Our results are consistent with our SUs, in other words, the policies learn what they are designed to learn, which shows that RL is a promising technique for learning policies in domains, such as argumentation, that are more complex than standard slot-filling applications.

Index Terms: spoken dialogue systems, reinforcement learning, simulated users, argumentation, negotiation, culture.

1. Introduction

In the last ten years, the use of Reinforcement Learning (RL) for learning dialogue policies has received much attention in the literature (e.g. [1, 2]). In the RL paradigm, managing a dialogue can be seen as a Markov Decision Process (MDP) or a Partially Observable Markov Decision Process (POMDP) where dialogue moves transition between dialogue states and rewards are given at the end of a successful dialogue. The solution to the dialogue management problem is a policy specifying for each state the optimal action to take. Typically rewards depend on the domain and can include factors such as task completion, dialogue length, and user satisfaction.

Traditional RL algorithms require on the order of thousands of dialogues to achieve good performance. Therefore, it is not feasible to rely on data collected with real users. Instead, training data is generated through interactions of the system with simulated users (SUs), i.e. models that simulate the behavior of real users [3]. In order to learn good policies, the behavior of the SUs needs to cover the range of variation seen in real users.

To date, RL has mainly been used to learn dialogue policies for slot-filling applications, such as restaurant recommendation, largely ignoring other types of dialogue. RL has been employed to learn both how to collect information from the user [1, 2] as well as how to present information to the user [4]. Notable exceptions in this trend are the works of [5, 6] who learned negotiation policies for a furniture layout task. Also, [6] experimented with different representations of the RL state. Moreover, [7] studied how POMDPs can be applied to negotiation.

Negotiation dialogues are very different from slot-filling dialogues. In slot-filling dialogues the user presents a complex query or service request (e.g. a hotel booking), and the system iteratively asks for more information to fully specify and confirm a set of "slots" that are needed to generate a database query (e.g. location, price range, room type) and ultimately satisfy the user's request. Dialogue policy decisions are typically whether to ask for a slot value, confirm a slot value, query the database, or present an answer. A typical reward function is to multiply the number of slots that have been filled and confirmed by a weighting factor (e.g. 100 points) and subtract the number of system turns multiplied by a weighting factor (e.g. 5 points) [2]. In contrast to slot-filling dialogue, 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.

In this paper we focus on an important aspect of negotiation, namely, argumentation. Our goal is to learn system (or agent) policies that will persuade their interlocutor (a human user or another agent) to agree on the system's preferences by using the most appropriate types of arguments. In the real world, argumentation and negotiation are an integral part of everyday interactions and may also be part of information provision slot-filling tasks. Imagine an advanced spoken dialogue system, designed to book flights or recommend restaurants, that can argue with the user about what is best for her.

We consider three types of users that represent different cultural norms, i.e. individualists, collectivists, and altruists [8, 9]. Our task is to learn policies appropriate for interacting with these three cultural norms. To learn these policies we use SUs and RL. The SUs are trained on a spoken dialogue corpus in a negotiation domain, and then tweaked towards a particular cultural norm using hand-crafted rules. This is because our corpus does not contain culture-specific information. We evaluate the learned policies in a simulation setting. Our results are consistent with our SUs, in other words, the policies learn what they are designed to learn, which shows that RL is a promising technique for learning policies in domains, such as argumentation, that are more complex than slot-filling.

Our research contribution is four-fold: First, to our knowledge this is the first study that uses RL for learning argumentation policies and as discussed above one of the few studies on using RL for negotiation. Second, for the first time, we learn policies for three different types of SUs representing three cultural norms (individualists, collectivists, and altruists). Third, unlike [6] who built hand-crafted SUs for learning negotiation dialogue policies, our SUs are hybrid (partly learned, partly hand-crafted) in the same fashion as the SUs of [10]. Fourth, our hybrid SUs allow us to learn policies for different cultural norms from a corpus that contains no such information.

The experiments presented here are an extension of our work in [11]. The main differences are that here we use different and more realistic cultural norms, more actions for the policies to choose between, and a more complex state represen-

tation. More details will be provided below.

In section 2 we present the spoken dialogue corpus used in our experiments. In section 3 we describe our SUs. In section 4 we present how we learn argumentation policies for different cultural norms. In section 5, we describe our evaluation experiments. Finally in section 6 we present our conclusion.

2. Data

In our negotiation domain, the data consists of spoken dialogues between American undergraduates playing the roles of a florist and a grocer who share a retail space. The dialogues were collected by Laurie R. Weingart, Jeanne M. Brett, and Mary C. Kern at Northwestern University. The participants negotiate on four issues: the design of the space, the temperature, the rent, and their advertising policy. The florist and the grocer have different goals, preferences, and use different types of arguments. 21 dialogues were annotated using a cross-cultural argumentation and persuasion annotation scheme, described in [12].

Figure 1 depicts an example dialogue annotated with our coding scheme.[1] Given that the task of learning dialogue policies with RL can be very complex even for simple slot-filling applications, in this initial experiment we decided to simplify the problem and thus we focus on learning how to negotiate about only one of the issues, the temperature. The florist is in favor of lower temperatures to keep her flowers fresh whereas the grocer prefers higher temperatures so that her customers feel comfortable.

Florist: How does that work for you? **(request_info.preference)**

Grocer: Well, personally for the grocery I think it is better to have a higher temperature. **(provide_argument.logic.me.indirect)**

Grocer: Just because I want the customers to feel comfortable. **(elaborate)**

Florist: Okay. **(acknowledge)**

Grocer: And also if it is warm, people are more apt to buy cold drinks to keep themselves comfortable and cool. **(elaborate)**

Florist: That's true. **(accept)**

Florist: But what about your products staying fresh? Don't they have to stay fresh or otherwise? **(rebut_argument.logic.you.direct)**

Figure 1: Example annotated dialogue with speech acts.

We created a new smaller corpus by extracting the parts related to the temperature issue from the original corpus. We also excluded all dialogues with intertwined issues (3 dialogues) and dialogues where one party makes an offer in the first turn and the other party agrees immediately (3 dialogues). Thus we ended up with 15 shorter dialogues (87 florist and 101 grocer utterances). Furthermore, we simplified the speech acts as shown in Table 1 to deal with data sparsity issues.

Because the corpus is very small, splitting it for training and testing or performing cross-fold validation is not an option. Instead we took into account that the corpus is not symmetrical. In the experimental design upon which the data collection was based, the temperature issue was much more important for the florist than the grocer. Thus their behavior is not similar. We replaced the speaker part of every action in the original corpus with its opposite, so "florist" is replaced with "grocer" and vice versa. The rest remains the same. An example is given in the second column of Table 1. Thus in the reversed corpus the florist behaves like the grocer of the original corpus and likewise for the grocer. Since the behaviors of the florist and the grocer are not symmetrical, the two corpora are different. So one can be used for training and the other for testing (see section 3). Note that this reversal makes sense only because we use the corpus to learn a model of the sequence of speech acts that does not include temperature values (low, middle, high). The decisions on the temperature values that the florist and the grocer would argue for or against are hand-crafted (see section 3).

Table 1: *Example simplified dialogue used for training and testing the SUs (original and reversed corpus).*

Original Corpus Sequence of Speech Acts	Reversed Corpus Sequence of Speech Acts
grocer, provide_argument	florist, provide_argument
grocer, offer	florist, offer
grocer, release_turn	florist, release_turn
florist, reject	grocer, reject
florist, release_turn	grocer, release_turn
grocer, provide_argument	florist, provide_argument
grocer, elaborate	florist, elaborate
grocer, offer	florist, offer
grocer, release_turn	florist, release_turn

3. Simulated Users

Our SUs are built on the speech act level from dialogues in the format depicted in Table 1. Note that we have inserted one more action "release_turn", which was not part of the original corpus to mark the boundaries between turns. Our SUs are based on n-grams of speech acts [3]. For example, a valid 3-gram (Table 1, column 1) would be: [grocer,provide_argument] [grocer,elaborate] $\rightarrow$ [grocer,offer]. This 3-gram indicates that if the grocer provides an argument and then elaborates on this argument, then a possible action is for the grocer to make an offer. The probability of each action is computed from our corpus. In this experiment we used 3-grams. The SUs used for learning the policies are built from the original corpus whereas the SUs used for testing the policies are built from the reversed corpus.

Our annotated dialogue data does not include information about cultural norms. Thus we cannot directly learn from the corpus a SU of a particular cultural norm. In our experiment we consider three different types of SUs, an individualist SU that if "pushed" with a sequence of arguments can be persuaded to agree on a middle-ground solution, a collectivist SU that agrees only on middle-ground solutions, and an altruist SU that if "pushed" with a sequence of arguments can be persuaded to agree on a solution in favor of her interlocutor.[2] More specifically, initially the individualist SU-grocer always supports high temperatures. Every time the florist policy provides an argument in favor of low or middle temperatures, the value of a

[1] The actual annotations are more detailed but here they are simplified for brevity.

[2] These definitions of cultural norms are quite rough. In reality behaviors can be more complex but these simplifications are required to make the learning problem more tractable. For more information on different cultural norms see [13].

counter is increased by 1. Every time the florist policy makes an argument in favor of a high temperature this counter is decreased by 4 (like a penalty). When the counter's value becomes 4 then the individualist SU-grocer starts supporting a middle-ground solution. This threshold of 4 was set empirically after experimentation. Now the policy has to learn that the only way to make the individualist SU compromise is to provide the appropriate sequence of arguments so that the value of the counter is set to 4 and the individualist SU changes its behavior, and as we will see in the evaluation section it succeeds in learning that. Note that the individualist SU-grocer will never agree on a low temperature so the best the policy can hope for is to reach a middle-ground solution. The collectivist SU-grocer always supports only middle temperatures so the counter is not used. Finally initially the altruist SU-grocer always supports middle temperatures. Every time the florist policy provides an argument in favor of low temperatures, the value of a counter is increased by 1. Every time the florist policy makes an argument in favor of a high temperature this counter is decreased by 4 (like a penalty). When the counter's value becomes 4 then the altruist SU-grocer starts supporting low temperatures. Likewise for the SU-florist and the grocer policies.

Note that these types of users are different from the ones used in [11]. In [11] we considered an individualist SU that never compromised, which is a bit unrealistic, and an altruist SU that always wants the best for her interlocutor and would never agree on a middle-ground solution, which is also unrealistic. Furthermore, in [11] we only considered two possible temperatures (low and high) whereas here we have a more realistic case of low, middle, and high temperatures. Also, in [11] the list of policy actions was simpler. Policies were either individualistic or altruistic (like the SUs) and could provide arguments or offers of one temperature only (low or high). In the current experiment both the florist and the grocer policies can choose between arguments and offers in favor of all three temperatures. This of course makes the learning problem much harder but this choice was deliberate because our goal here is to prove that RL is suitable for learning argumentation policies, thus we do not want to oversimplify the problem. In the real world you would probably not expect a florist policy to support high temperatures but it is interesting that we found such cases in our corpus, which further justifies our decision to have the policies choose between three temperatures. The SU actions (as well as system actions) used in our experiment are 12 in total: "provide_argument,low", "provide_argument,middle", "provide_argument,high", "elaborate", "rebut_argument", "acknowledge", "offer,low", "offer,middle", "offer,high", "accept", "reject", "release_turn".

4. Learning Argumentation Policies

After we have built our SUs, we have these SUs interact with our system using RL in order to learn different policies. In particular we use the SUs built from the original corpus. We use different reward functions for the florist and grocer policies. Thus the florist policy is rewarded when the outcome of the conversation is agreement on a low temperature (+800 points) or when there is a middle ground solution (+400 points), and penalized otherwise (-800 points). The grocer policy is rewarded when the outcome of the conversation is agreement on a high temperature (+800 points) or when there is a middle ground solution (+400 points), and penalized otherwise (-800 points).

In [11], to facilitate learning we also added one more penalty (-800 points) for some incoherent sequences of actions, e.g. accepting or rejecting a non-existent argument or offer, rebutting a non-existent argument, etc. In the current experiment, we did not use such penalties, which is also a big improvement. The system learns an optimal policy vs. the three SUs only using the rewards and penalties at the end of the dialogue.

Table 2: *Reward functions.*

Type of Policy	Outcome	Penalty per Action
Florist	low +800	-10
Florist	middle +400	-10
Florist	high -800	-10
Grocer	low -800	-10
Grocer	middle +400	-10
Grocer	high +800	-10

There is also a penalty of -10 points for each system and SU action. The fastest possible successful dialogue can be for one of the interlocutors to make an offer and the other to accept. Thus the highest possible reward in a dialogue can be 800 minus 4 actions = 760; the four actions are "offer", "release_turn", "accept", "release_turn". Table 2 shows the reward functions used in our experiment. The goal of RL is to learn the best action in each dialogue state so that the optimal outcome is achieved (e.g. a low temperature for the florist policy that interacts with an altruist SU-grocer, a middle temperature for the florist policy that interacts with an individualist SU-grocer, etc.).

Another important issue is how to represent the state so that the problem is tractable and at the same time good policies can be learned. After some experimentation, we used the state representation shown below, showing features with possible values, which leads to 4374 possible states. The policy actions are the same as the SU actions (see section 3). Note that in [11] we used a much simpler state representation leading to 864 possible states.

- Current speaker (florist/grocer)
- Most recent temperature supported by the florist (low/middle/high)
- Most recent temperature supported by the grocer (low/middle/high)
- Is there an argument on the table and by whom? (none/florist/grocer)
- Is there an offer on the table and by whom? (none/florist/grocer)
- If there is an offer, what is the temperature offered? (low/middle/high)
- Is there a rejected offer (the most recent rejection) and by whom? (none/florist/grocer)
- If there is a rejected offer, what is the rejected temperature? (low/middle/high)

For training we used the SARSA-λ algorithm with greedy exploration at 20% to explore the state-action pair space. We ran 20,000 iterations for learning the final policy for each condition. More specifically, we learned a florist policy trained against an individualist SU-grocer, a collectivist SU-grocer, and an altruist SU-grocer (3 florist policies). Then we also learned a grocer policy trained against an individualist SU-florist, a collectivist SU-florist, and an altruist SU-florist (3 grocer policies). All possible combinations are shown in Table 3 in the evaluation section (section 5).

5. Evaluation

We evaluate our learned policies against the SUs that were built from the reversed corpus (see section 2). We run each policy against all types of SUs (2000 simulated dialogues per case) and we report the outcome using different metrics. All scores are averaged over the 2000 simulated dialogues. The first metric is the total reward that we used for training (see Table 2). The second metric is the outcome reward, i.e. the reward used for training without taking into account the action penalty. We also report the total number of actions (both system and user), and the total number of incoherent system actions, e.g. accepting a non-existent offer, rebutting a non-existent argument, etc. On average the numbers of system and user actions per dialogue are equal. Results are given in Table 3. The notation is as follows: F(GA)-GI stands for a florist policy trained with an altruist SU-grocer and tested against an individualist SU-grocer, etc.

Table 3: *Evaluation results (average scores, 2000 dialogues).*

Type of Policy	Total Reward	Outcome	# Actions	# Incoher. Actions
F(GI)-GI	202.0	398.8	19.7	1.2
F(GI)-GC	228.9	394.4	16.6	2.0
F(GI)-GA	228.3	396.2	16.8	2.1
F(GC)-GI	-1006.9	-610.6	39.6	6.0
F(GC)-GC	284.5	399.8	11.5	2.1
F(GC)-GA	286.5	400	11.4	2.1
F(GA)-GI	-1001.5	-602.6	39.9	5.9
F(GA)-GC	298.4	399.8	10.1	1.3
F(GA)-GA	295.2	399.4	10.4	1.3
G(FI)-FI	110.5	366.6	25.6	4.5
G(FI)-FC	228.1	398.4	17.0	2.1
G(FI)-FA	222.1	399.2	17.7	2.3
G(FC)-FI	-1028.6	-603.4	42.5	6.6
G(FC)-FC	204.1	399.2	19.5	2.1
G(FC)-FA	194.4	398.2	20.4	2.2
G(FA)-FI	-1019.8	-598	42.2	6.5
G(FA)-FC	274.1	396.2	12.2	1.5
G(FA)-FA	277.5	397.8	12.0	1.5

The policies trained with individualist SUs perform well when tested against all types of SUs, which is very promising. The policies trained with a collectivist or altruist SU perform well when tested against a collectivist or altruist SU, but do not perform well when tested against an individualist SU, which is expected. In particular, the policies trained with an individualist SU learned what they were designed to learn; that the only way to reach an agreement, i.e. persuade the individualist SU to compromise, is to keep providing arguments in favor of a middle solution or a solution suitable for the policies. After 4 such arguments the individualist SU starts supporting middle temperatures (reward +400). This simulates what may happen in real life where a series of successful arguments could shift the behavior of our interlocutor in our favor. The policies trained with an altruist SU do not perform as well as intended. These policies did not learn to "push" the altruist SU towards agreeing on an optimal outcome (reward +800). However, they did learn to offer middle-ground solutions that the altruist SU accepts (reward +400), which is promising. We suspect that one reason that they did not reach an optimal behavior could be the action penalty. It could be the case that these policies learned that it is better to agree quickly on a sub-optimal (middle) solution than go for the optimal solution and in doing so keep being penalized with action penalties. However, this needs to be investigated further. So in the future, we will perform experiments applying the action penalty only when the dialogue exceeds a number of turns (e.g. 10 turns). To decrease the number of incoherent system actions we need a more informative state representation and/or to apply penalties for such actions during learning as we did in [11]. We also intend to use a combination of SUs to learn an optimal policy for all three cultural norms.

6. Conclusion

We learned argumentation policies for negotiation using SUs having different cultural norms (individualists, collectivists, and altruists). The SUs were trained on a spoken dialogue corpus in a negotiation domain, and then tweaked towards a particular cultural norm using hand-crafted rules. The evaluation of our learned policies in a simulation setting showed that RL is a promising technique for learning policies in domains, such as argumentation, that are more complex than standard slot-filling applications.

7. Acknowledgements

This research was funded by a MURI award through ARO grant number W911NF-08-1-0301. We are grateful to Laurie R. Weingart, Jeanne M. Brett, and Mary C. Kern who provided us with the florist-grocer dialogues. This work was also partially sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). The content does not necessarily reflect the position or the policy of the U.S. Government, and no official endorsement should be inferred.

8. References

[1] J. Williams and S. Young, "Scaling POMDPs for spoken dialog management," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2116–2129, 2007.

[2] K. Georgila, M. Wolters, and J. Moore, "Learning dialogue strategies from older and younger simulated users," in *SIGdial*, 2010.

[3] K. Georgila, J. Henderson, and O. Lemon, "User simulation for spoken dialogue systems: Learning and evaluation," in *Interspeech*, 2006.

[4] V. Rieser and O. Lemon, "Natural language generation as planning under uncertainty for spoken dialogue systems," in *EACL*, 2009.

[5] M. English and P. Heeman, "Learning mixed initiative dialogue strategies by using reinforcement learning on both conversants," in *HLT-EMNLP*, 2005.

[6] P. Heeman, "Representing the reinforcement learning state in a negotiation dialogue," in *ASRU*, 2009.

[7] P. Paruchuri, N. Chakraborty, R. Zivan, K. Sycara, M. Dudik, and G. Gordon, "POMDP based negotiation modeling," in *IJCAI MICON Workshop*, 2009.

[8] G. H. Hofstede, *Culture's consequences: Comparing values, behaviors, institutions, and organizations across nations*. Thousand Oaks, CA: SAGE, 2001.

[9] P. D. Allison, "How culture induces altruistic behavior," in *Annual Meetings of the American Sociological Association*, 1992.

[10] S. Jung, C. Lee, K. Kim, and G. Lee, "Hybrid approach to user intention modeling for dialog simulation," in *ACL*, 2009.

[11] K. Georgila and D. Traum, "Learning culture-specific dialogue models from non culture-specific data," in *HCI International*, 2011.

[12] K. Georgila, R. Artstein, A. Nazarian, M. Rushforth, D. Traum, and K. Sycara, "An annotation scheme for cross-cultural argumentation and persuasion dialogues," in *SIGdial*, 2011.

[13] J. Brett and M. Gelfand, "A cultural analysis of the underlying assumptions of negotiation theory," in *Frontiers of Negotiation Research, L. Thompson (Ed)*. Psychology Press, 2006, pp. 173–201.