Reinforcement learning of dialogue coherence and relevance

Sultan Alahmari, Tommy Yuan, and Daniel Kudenko

University of York, Department of Computer Science, Deramore Lane York YO10 5GH, UK
smsa500, tommy.yuan and daniel.kudenko@york.ac.uk

Abstract. In multi-agent systems, agents communicate with each other, using argumentation as one type of communication. Agents argue to resolve conflicts between them. In our previous work, an agent learnt how to argue in an abstract argumentation system and a dialogue game based argumentation. This research looks at improving the agent's performance as well as the coherence and relevance of the dialogue. We use a reinforcement learning method to encourage our agent to improve its performance and the coherence of the dialogue. We propose a new formula that motivated the agent to achieve a higher reward based on three different attributes: number of moves, number of contradictions and number of focus switches. The results were promising and the agent is able to learn how to argue and reach a good level of performance with regard to winning the argumentation and generating high quality dialogue contribution.

Keywords: Multi-agent systems · Argumentation · Dialogue game · Persuasion dialogue · Quality measures · Reinforcement learning

1 Introduction

In the past few decades, argumentation has played an important role and has been widely studied in artificial intelligence (AI) [7]. A review of further significant research in the field can be seen from Prakken [17]. One significant development was Dung's [9] introduction of the abstract argumentation framework which assumes a directed graph to represent arguments as nodes and attack relations as arcs.

In our previous work [1–3] we used Dung's framework and allowed different agents to play an argument game [23]. One of the agents was based on reinforcement learning (RL) [18], and the aim of our research was to allow agents to learn how to argue against different baseline agents. Some limitations were revealed in our earlier work [1–3]. One of which was not being able to generalise policy for different abstract argumentation graphs. The reason for this is difficult to discern. Some patterns that could help our learning agent transfer experience from one domain to another are hard to learn without reference to the internal structure of the arguments.

Therefore, this motivated us to move to propositional-logic based representation and a richer dialogue model [3, 4]. We assert that argument patterns, i.e. argument schemes and sources of evidence, could encourage our RL agent to learn [4] and in our work [4] we used an influential logic-based dialogue model the "DE" model [25, 26, 29]. There are some advantages in adopting the DE model rather than another model, for example

it allows enough room for strategic formations and a strategy is essential for an agent to make high quality dialogue contributions. DE computational agents have been built with hard-coded heuristic strategies [28], so they can be directly used as baseline agents. In addition, the DE has simple dialogue rules that control the evolving dialogue [25,26].

In [4], the performance of our RL agent showed promising result, improving over hard-coded agents. We also demonstrated that our RL agent was able to learn to win an argument game against baseline agents with the minimum number of moves. It would be worthwhile investigating whether the dialogue contributions made by the RL agents are of high quality in terms of coherence and relevance. Indeed, this will contribute to improving the agent's performance by learning how to win with the minimum number of moves in a fluent and coherent manner.

The rest of this paper is organised as follows: Section 2 introduces reinforcement learning for the DE dialogue game, Section 3 discusses the measuring of dialogue coherence and relevance, Section 4 introduces the experiment and discusses the results, Section 5 gives conclusions and our planned future work.

2 Reinforcement learning for a DE dialogue game

The DE dialogue model [24, 26, 29] was developed by Yuan [29] based on Mackenzie's DC system [10, 14]. The DE dialogue game defines the rules for participants making moves [29]. There are five move types in the DE game, namely *Assertion*, *Questions*, *Challenges*, *Withdrawals* and *Resolution demands* [26]. The DE model allows each agent to have its own public commitment store which contains statements that have been stated or accepted by a speaker [4, 26]. The commitment store has two lists, *an assertion list*, which contains the statements that have been explicitly asserted by the speaker and *a concession list*, which contains the statements that have been implicitly accepted by the speaker [26]. There are commitment rules that are used to update the commitment store, they are quoted from [24, 26, 29] as follows:

1. *Initial commitment,*CR_0: The initial commitment of each participant is null.
2. *Withdrawals,*CR_W: After the withdrawal of P, the statement P is not included in the move.
3. *Statements,*CR_S:After a statement P, unless the preceding event was a challenge, P is included in the move maker's assertion list and the dialogue partner's concession list, and $\neg P$ will be removed from the move maker's concession list if it is there.
4. *Defence,*CR_{YS}:After a statement P, if the preceding event was Why Q?, P and If P then Q are included in the move maker's assertion list and the dialogue partner's concession list, and $\neg P$ and $\neg$(If P then Q) are removed from the move maker's concession list if they are there.
5. *Challenges,*CR_Y: A challenge of P results in P being removed from the store of the move maker's if it is there.

Dialogue rules that an agent must follow during the dialogue are taken from [24, 26, 27, 29] as follows:

1. R_{FROM}: Each participant or agent can make one of the permitted types of move in turn.
2. $R_{REPSTAT}$: Mutual commitment may not be asserted until answering the question or challenge.
3. R_{QUEST}: The possible answers to question P can be "P", "¬P" or "No commitment".
4. R_{CHALL}: "Why P?" can be answered by withdrawal of P, a statement to the challenger or resolution demand for any commitments of the challenger which imply P.
5. $R_{RESOLVE}$: A resolution demand can happen only if the opponent has inconsistent statements in the commitment store.
6. $R_{RESOLUTION}$: A resolution demand has to be followed by withdrawal of one of the offending conjuncts or affirmation of the disputed consequent.
7. $R_{LEGALCHAL}$: The agent can challenge the opponent "Why P?", unless P is on the assertion list of the opponent's dialogue.

There are different reasons for adopting the DE dialogue model in this paper. One of the reasons is that the model leaves enough room for the agent to do strategy formation [27]. The strategy is the main core for the agent to make high quality dialogue contributions. In addition, the computational agents used in the DE dialogue model were built with hard-coded heuristic strategies, hence the model has shown benefits over other models because of its computational tractability and simple dialogue rules [25–27, 29]. The DE model was also built with propositional logic, which we consider to be a move forward from the abstract level of argumentation to the internal level of the argument [3, 4]. Thus, the DE model is more sophisticated and richer, since the dialogue state can be represented using different aspects, such as the commitment store, and different move types, for example, questions. We would expect the DE game to facilitate an effective learning experience for a computational agent, improving both dialogue coherence and relevance.

Before engaging our RL agent in the DE dialogue model, we will briefly review reinforcement learning and the structure of our agent. Reinforcement learning is one of the most common areas of machine learning [18, 22]. Agents interact with an environment in order to map states with a particular action and receive rewards, as illustrated in Figure 1 [19]. The agent explores a policy which involves mapping a state with an action by using trial and error [4]. In the DE dialogue model, our RL agent needs to engage with the model in order to play the dialogue game with other computational agents. As a result, agents need to persuade each other what they believe.

To design the RL agent it is necessary to identify the state action and reward for our agent. In [4], we identify these properties in detail. The state will be $(previousmove \cup CS1 \cup CS2)$ (where CS1 is the commitment store for the proponent and CS2 is the commitment store for the opponent) [4]. Actions are defined as the available move types in the DE model [25, 29] which are *assert*, *question*, *challenge*, *withdraw* and *resolution demand* as well as move contents, this means actions are also defined as the move content which is a proposition or conjunct of propositions [4]. Therefore, the RL agent aims to map a state with a particular action to identify the policy [18]. The RL agent will receive a reward based on the action that it takes, so that positive actions

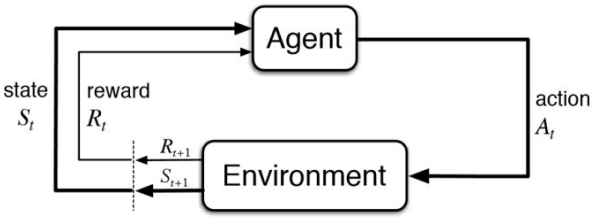

Fig. 1: RL agent interacts with an environment

receive positive reward and vice versa. The RL agent focuses on maximising its utility based on the long-term reward gained through repeated episodes during the game. The reward function in [4] is defined to allow the RL agent to seek to win with the minimum number of moves as in Equation (1):

$$R = 100 + \frac{W}{L} \tag{1}$$

such that W is the number of moves in a first winning episode (the benchmark), L is the number of moves in the current episode. Hence when L is at the minimum value, the reward will tend to increase. In addition, [4] used the Q-learning algorithm (Equation 2):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_t, a_{t+1}) - Q(s_t, a_t)] \tag{2}$$

3 Measuring dialogue coherence and relevance

This is our previous work in this area. In [4], we showed that the reinforcement learning (RL) agent played against different baseline agents, based on the DE dialogue game. The results were promising and the performance of the RL agent rapidly increased against the baseline agent in figure 2 and 3 respectively.

In addition, the reward shaping let the agent win the game with the minimum number of moves. It was then thought worthwhile to test whether the agent maintained coherence and relevance in a dialogue. The literature has a number of different approaches to measure the dialogue, for example persuasion dialogue [5], measuring an agent's uncertainty negotiation [11], measuring the argument strength through applying the concept of value of a game, defined in game theory [13, 15] and measuring dialogue games based on the external agent's point of view [12].

In particular, Amgound et al. [5] proposed three different measures: the quality of the exchanged arguments, the agent's behaviour, such as coherence and aggressiveness, and measuring the quality of the dialogue itself, for example for relevance. Amgoud et al. [5] argue that these measures are of great importance, because they can be used as guidelines for protocols between participants in order to make high quality dialogue. Weide [21] supports these measures being used as benchmarks for the agent to decide which dialogue move they should choose. Based on our work [4] we consider two criteria from [5]: coherence and relevance. We believe that coherence and relevance can

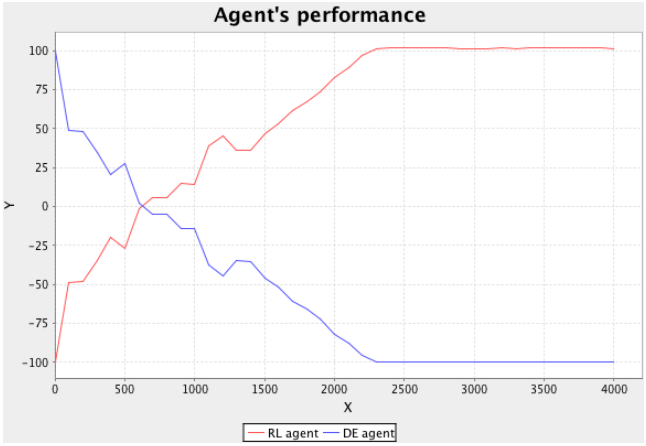

Fig. 2: Performance of RL agent against heuristic strategy agent

be used to assess the RL agent learning to argue in the minimum number of moves, and to contribute high quality dialogue against different baseline agents. In addition, our system [4] is based on persuasion dialogue and requires measuring and evaluating the quality of the dialogue in different aspects. Coherence [5,8,16] is based on a persuasion dialogue where an agent attempts to defend what it believes and does not contradict itself. So, in this paper we introduce a formula to measure the percentage of incoherence [5] to evaluate how the agent was incoherent in a dialogue. This formula (Equation 3) depends on how many times the agent has contradicted itself during the dialogue with respect to the number of moves for the agent.

$$AgentIncoherent = \frac{NumberOfContradiction}{NumberOfMoves} \tag{3}$$

Relevance in dialogue concerns an agent making a move that does not deviate from the subject of the dialogue [6]. However, in the DE dialogue game all moves are related to the subject [29]; therefore, it was necessary to find a way to measure the relevance of both agents in our system in another way. Since one of the strategies in the DE dialogue game [27, 29] allowed agents to change their current focus, it was considered of interest to minimise changing the current focus of both agents. It is argued that if the agent switches focus a large number of times, it would make the dialogue less fluent. Therefore, we used a new formula (Equation 4) to measure the relevance of each agent, based on how many times that agent switched focus during the dialogue, with respect to the number of moves:

$$AgentIrrelevance = \frac{NumberOfSwitchingFocus}{NumberOfMoves} \tag{4}$$

Therefore, we made the RL agent learn to argue by considering coherence and relevance, as well as winning in the minimum number of moves [4].

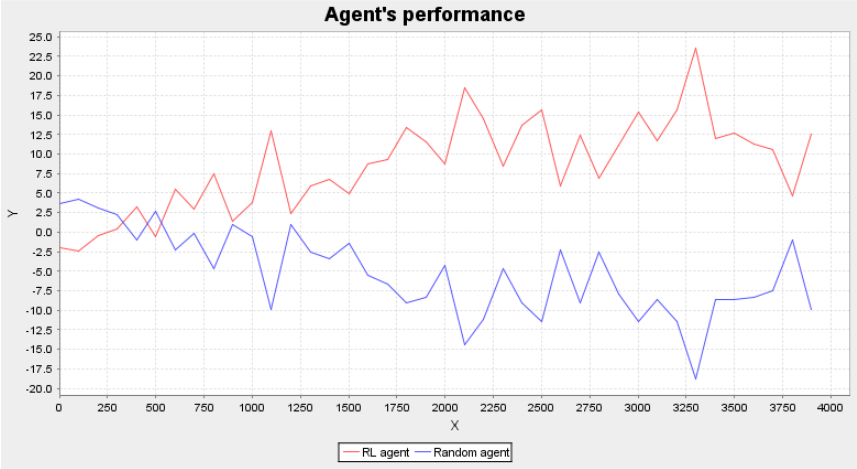

Fig. 3: Performance of RL agent against random agent

4 Experiment and discussion

In this section we discuss the experiments conducted between the RL agent and the baseline agent. The baseline agent that we considered was hard coded with strategic heuristics [29]. The agents adopting the DE dialogue game can use different strategies e.g. a heuristic or using random moves. [27]. The heuristic agent was developed based on heuristics in [27] and the random agent makes random moves with respect to the DE dialogue game rules. The RL agent plays the dialogue game against both heuristic and random agents. This allowed us to evaluate the coherence and relevance, as well as observing whether the RL agent could win with the minimum number of moves. It also allowed us to measure the quality of the dialogue with respect to coherence and relevance.

To measure the coherence of the agent we looked at the number of contradictions each agent had in their commitment store. Hence, Equation (3) measures coherence for the agent, which means the less contradictions made by the agent the more coherence in the agent's dialogue. On the other hand, relevance in Equation (4) measures the number of occasions that an agent did not address the previous move, which in effect is a focus switch. Therefore, the less number of focus switching means more focused the agent dialogue.

The RL agent first played against the heuristic agent. The reward function in Equation 1 was used initially. The game was played 4000 times, each time is considered as a debate episode between two agents. We allow RL agent to test the learned policy after every 100 episodes and the test will repeat 10 times to avoid randomness. The dialogue quality measures as defined in Equations 3 and 4 are used to visualise the results after taking an average every 500 episodes for representation purpose as in Figure 4 for incoherence and Figure 5 for irrelevance

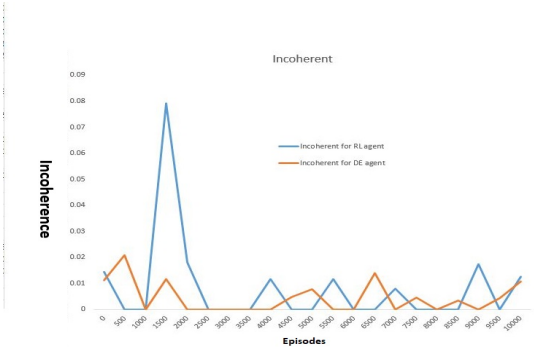

Fig. 4: Incoherent measuring between RL agent and heuristic agent

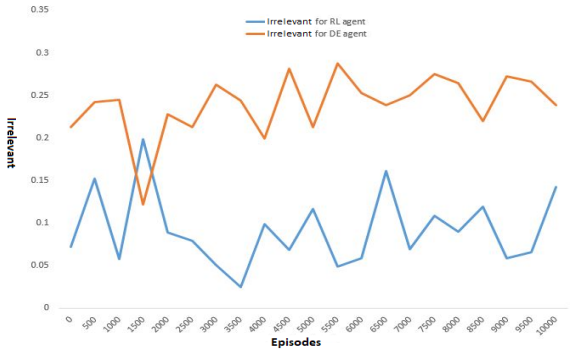

Fig. 5: Irrelevance measuring between RL agent and heuristic agent

In Figure 4, the x axis represents the number of episodes and the y axis represents the incoherence measure as specified in Equation 3. For the relevant measurement in Figure 5, the x axis represents the number of episodes and the y axis represents the irrelevance measure as specified in Equation 4. The results show that the RL agent did not make any performance improvement in either coherence or relevance. One conclusion, however, can be drawn from this, which is the proposed measures for coherence and relevance are independent of the number of moves, which means with the number of moves in an winning episode minimised, the coherence and relevance measures remain the same.

This encouraged us to incorporate coherence and relevance measures into the reward function in Equation 1 so that our RL agent was able to improve coherence and relevance as well as minimise the number of moves. Therefore, we have reshaped the reward function as shown in the following equation:

$$R = 100 + \frac{M1}{Mn} + \frac{C1}{Cn} + \frac{SF1}{SFn} \tag{5}$$

such that:

* R is the reward function.
* $M1$ is the number of moves in the first episode.
* Mn is the number of moves in the current episode.
* $C1$ is the number of contradictions in the first episode.
* Cn is the number of contradictions in the current episode.
* $SF1$ is the number of switching focus in the first episode.
* SFn is the number of switching focus in the current episode.

The design of the reward function was to motivate the RL agent to choose moves which minimises the number of moves, contradictions and focus switches. The agent was awarded 100 for winning the game. After running the experiment between the RL agent and the heuristic agent, the results can be seen in Figures 6 and 7.

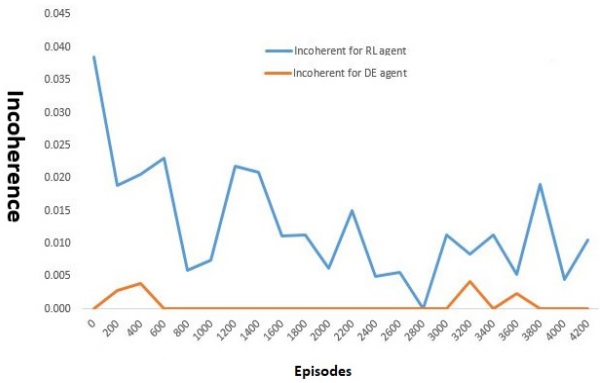

Fig. 6: Incoherent measuring between RL agent and heuristic agent

For the coherence measurement in Figure 6, the learning agent shows the learning curve in improving the coherence where the incoherence is decreased gradually. This means the new reward function, as in Equation 5, encourages the RL agent to maximise coherence in the dialogue. Whereas, the heuristic agent was able to maintain coherence in the dialogue. For the relevance measurement in Figure 7, it was surprising to see that the RL agent was able to maintain better relevance than the heuristic agent in the first appearance. By investigating the dialogue transcripts, it was found that the heuristic agent asked a large number of questions, therefore making the learning agent constantly stay on focus by passively responding to the questions. It is therefore worthwhile to experiment the RL agent with a random agent and then study the consequence.

We have done similar experiment between the RL agent and a random agent and the results are shown in Figure 8 and 9. Figure 8 confirms the result shown in Figure 6. Figure 9 shows the RL agent with improved performance (i.e. decrease of irrelevance) well above the random agent.

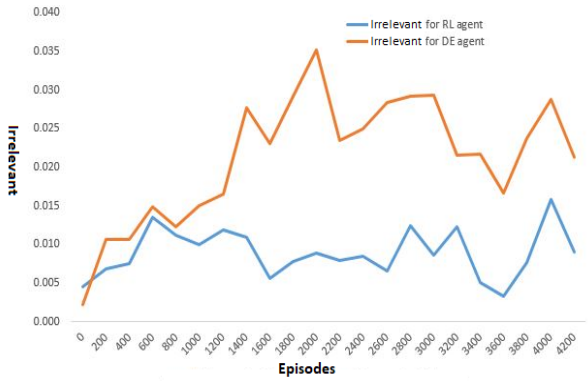

Fig. 7: Irrelevance measuring between RL agent and heuristic agent

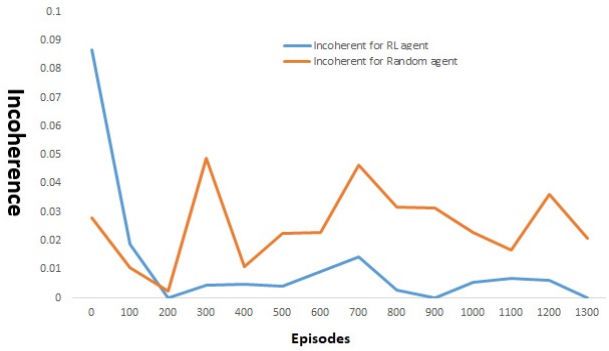

Fig. 8: Incoherent measuring between the RL agent and random agent

From these experiments we can conclude that the RL agent can learn to argue with more fluency and coherence.

5 Conclusion and future direction

We have proposed two quality measures for argumentative dialogue namely the fluency and coherence. We have incorporated the quality measures into the reward function of our RL agent and carried out a number of experiments. We conclude that the RL agent can learn to improve its performance with regard to coherence and fluency against both heuristic and the random agent. Different weights will be applied for experiment with the features in equation 5.

We are also planning to generalise our approach for different argument domains.We are building the new argument domain in *BREXIT* and investigating transfer learning

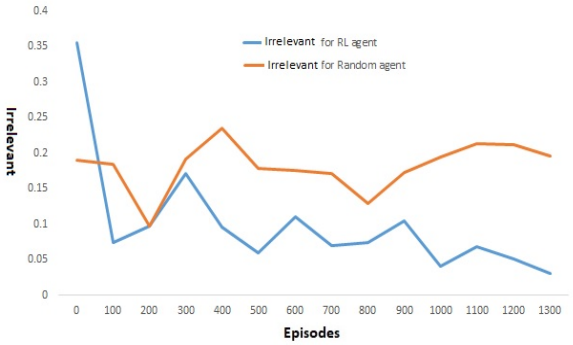

Fig. 9: Irrelevance measuring between RL agent and randomised based agent

techniques [20]. We will test whether our RL agent can apply what has been learned in one domain, e.g. *Capital punishment* to a new domain such as *BREXIT*.

References

1. Sultan Alahmari, Tommy Yuan, and Daniel Kudenko. Reinforcement learning for abstract argumentation: Q-learning approach. In *Adaptive and Learning Agents workshop (at AAMAS 2017)*, 2017.
2. Sultan Alahmari, Tommy Yuan, and Daniel Kudenko. Reinforcement learning for argumentation: Describing a phd research. In *Proceedings of the 17th Workshop on Computational Models of Natural Argument (CMNA17)*, 2017.
3. Sultan Alahmari, Tommy Yuan, and Daniel Kudenko. Policy generalisation in reinforcement learning for abstract argumentation. In *Proceedings of the 18th Workshop on Computational Models of Natural Argument (CMNA18)*, 2018.
4. Sultan Alahmari, Tommy Yuan, and Daniel Kudenko. Reinforcement learning for dialogue game based argumentation. In *Accepted of the 19th Workshop on Computational Models of Natural Argument (CMNA19)*, 2019.
5. Leila Amgoud and Florence Dupin De Saint Cyr. Measures for persuasion dialogs: A preliminary investigation. *Frontiers in Artificial Intelligence and Applications*, 172:13, 2008.
6. Leila Amgoud and Florence Dupin de Saint-Cyr. On the quality of persuasion dialogs. *Studies in Logic, Grammer and Rheroric*, 12(36):69–98, 2011.
7. Trevor JM Bench-Capon and Paul E Dunne. Argumentation in artificial intelligence. *Artificial intelligence*, 171(10-15):619–641, 2007.
8. Lauri Carlson. Dialogue games: An approach to discourse anaphora. *Dordrecht: Reidel*, 1983.
9. Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357, 1995.
10. Jim D Mackenzie. Question-begging in non-cumulative systems. *Journal of philosophical logic*, 8(1):117–133, 1979.
11. Omar Marey, Jamal Bentahar, Rachida Dssouli, and Mohamed Mbarki. Measuring and analyzing agents' uncertainty in argumentation-based negotiation dialogue games. *Expert Systems with Applications*, 41(2):306–320, 2014.

12. Omar Marey, Jamal Bentahar, and Abdeslam En-Nouaary. On the measurement of negotiation dialogue games. In *SoMeT*, pages 223–244, 2009.
13. Paul-Amaury Matt and Francesca Toni. A game-theoretic measure of argument strength for abstract argumentation. In *European Workshop on Logics in Artificial Intelligence*, pages 285–297. Springer, 2008.
14. David John Moore. *Dialogue game theory for intelligent tutoring systems.* PhD thesis, Leeds Metropolitan University, 1993.
15. J v Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.
16. Henry Prakken. Coherence and flexibility in dialogue games for argumentation. *J. Log. Comput.*, 15(6):1009–1040, 2005.
17. Henry Prakken. Historical overview of formal argumentation. *IfCoLog Journal of Logics and their Applications*, 4(8):2183–2262, 2017.
18. Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.
19. Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press Cambridge, 2012.
20. Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(Jul):1633–1685, 2009.
21. Thomas L van der Weide. *Arguing to motivate decisions*. PhD thesis, Utrecht University, 2011.
22. Marco Wiering and Martijn Van Otterlo. Reinforcement learning. *Adaptation, learning, and optimization*, 12:51, 2012.
23. Michael Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2002.
24. Tangming Yuan, David Moore, and Alec Grierson. Computational agents as a test-bed to study the philosophical dialogue model" de": A development of mackenzie's dc. *Informal Logic*, 23(3), 2003.
25. Tangming Yuan, David Moore, and Alec Grierson. A human–computer debating system prototype and its dialogue strategies. *International Journal of Intelligent Systems*, 22(1):133–156, 2007.
26. Tangming Yuan, David Moore, and Alec Grierson. A human-computer dialogue system for educational debate: A computational dialectics approach. *International Journal of Artificial Intelligence in Education*, 18(1):3–26, 2008.
27. Tangming Yuan, David Moore, and Alec Grierson. Assessing debate strategies via computational agents. *Argument and Computation*, 1(3):215–248, 2010.
28. Tangming Yuan, David Moore, Chris Reed, Andrew Ravenscroft, and Nicolas Maudet. Informal logic dialogue games in human–computer dialogue. *The Knowledge Engineering Review*, 26(2):159–174, 2011.
29. Tommy Yuan. *Human-Computer Debate, a Computational Dialectics Approach*. PhD thesis, Unpublished Doctoral Dissertation, Leeds Metropolitan University, 2004.