Persuasive Dialogues in an Intelligent Tutoring System for Medical Diagnosis

Amin Rahati and Froduald Kabanza

Department of Computer Science,
University of Sherbrooke
Sherbrooke, Qubec J1K 2R1
Canada
{amin.rahati,kabanza}@usherbrooke.ca

Abstract. Being able to argue with a student to convince her or him about the rationale of tutoring hints is an important component of pedagogy. In this paper we present an argumentation framework for implementing persuasive tutoring dialogues. The entire interaction between the student and the tutoring system is seen as an argumentation. The tutoring system and the student can settle conflicts arising during their argumentation by accepting, challenging, or questioning each other's arguments or withdrawing their own arguments. Pedagogic strategies guide the tutoring system selecting arguments aimed at convincing the student. We illustrate this framework with a tutoring system for medical diagnosis using a normative expert model.

Key words: intelligent tutoring system, argumentation, persuasive dialogue, medical diagnosis

1 Introduction

One of the key problems in the development of an intelligent tutoring system (ITS) concerns the implementation of the verbal exchange (i.e., a dialogue) that takes place between a student and the ITS. A dialogue determines what the ITS tells the student, when and how, to support her learning process in the most effective way. Some approaches implement ITS dialogues using finite state machines (FSM) [1, 2]. Other approaches use dynamically generated dialogue structures, by using automated planning techniques [3, 4]. In these approaches, the requirement for the ITS to persuade the student is not formally acknowledged. Such a requirement can be implemented by adopting a formal framework of argumentation for the implementation of dialogues between the ITS and the students. A number of such frameworks has been developed, including applications to decision support systems [5–9].

Some ITS involve argumentation as the content of the learning material, for instance to learn skills of argument reasoning by analyzing arguments. In particular, LARGO is a system used to train students on acquiring argumentation skills [10]. LARGO has a graphic interface through which students can represent or visualize arguments they make and their relations. It can also provide feedback to students on their construction of arguments. However, in those ITS, which teach argumentation, argumentation is not involved as a pedagogical tool aiming at persuading the student on the rationale of the interventions made by the ITS to support her in her learning process.

In this paper we present a general approach for implementing persuasive tutoring dialogues. In our approach, every action performed by a student trying to solve a problem is considered as an argument. The ITS intervenes to help the student also by making arguments. Errors made by the student are considered as a disagreement and the ITS tries to help the student remedy them through an argumentation.

The framework is composed of three key components. The first component is a language for defining dialogue moves between the ITS and the student. A typical dialogue move specifies the content of an argument or a propositional attitude in the exchange of arguments (e.g., accepting the interlocutor's argument or withdrawing owns argument). The second component is a protocol regulating the moves and conveyed constraints on allowed move sequences in an argumentation dialogue. The third component is an argument generator used by the ITS to decide arguments to use which are persuasive for the student. We use Walton's argumentation theory ([8]) to model arguments, challenges to arguments and acceptance of arguments. We integrate this theory with the notion of preference among arguments [11], making it possible for the ITS to make decisions on the most convincing arguments. Higher level strategic rules ([12]) are also involved to select arguments based on a pedagogic goal.

The remainder of this paper is organized as follows. In the next section we start by introducing TeachMed [1], a medical diagnosis ITS, which we use as a testbed to illustrate our argumentation framework. This is followed by description of the argumentation framework. We then give an illustration using TeachMed testbed. We conclude with a discussion on related and future work.

2 Argumentative TeachMed

In TeachMed, a student starts by selecting a virtual patient having a particular disease with the objective to generate a correct diagnosis. The student makes a diagnosis by performing an investigation. To formulate some initial hypotheses she starts asking queries to the virtual patient about the different symptoms, life style and family background. She can also make queries in terms of a physical exam on a 3D model of the patient (e.g., reflexes) or in terms of lab tests (e.g., blood samples). Queries and tests are selected from a list including noise queries. Each query has an answer specified in the virtual-patient model, which includes his vital signs, symptoms and results of lab tests or physical exam. As more queries are asked, she will eliminate some hypotheses, strengthen others and generate new ones. This process continues until she can narrow the list of hypothesis down to one or two –that is, the final diagnosis. The challenges in solving this type of problems involve deciding what evidences to observe and how observe it, and determining the list of hypotheses that best explain the observed evidences. The queries and the differential diagnosis generation are the student's diagnosis (problem solving) actions. We also have student's utterance actions, in the form of requests for help or replies to utterances from TeachMed.

To implement our approach, we modified TeachMed architecture [1], by extending the *user model* to store the student's order of preferences concerning the decision making parameters for making diagnostic actions and utterances. The original model only recorded the student's diagnosis actions. The student's actions are now recorded as arguments. We preserve the use of an influence diagram (ID) to model the *expert knowledge* - the ID represents the causal relationship between symptoms and diseases and the utility of queries. The *pedagogic model* is now replaced by the argumentation framework.

3 The Argumentation Framework

The entire session of a student learning to diagnose a case is considered as an argumentation between the student and the ITS. Whenever a student performs a diagnosis action, TeachMed interprets the action as an assertion which the ITS tries to reject if it can. An assertion will be rejected if the pedagogic model finds a convincing argument against it. This is done based on the information provided by expert model, the commitment store and the dialogue history. For instance, if a student asks a query which is irrelevant to the current differential diagnosis –for instance the value of information for the query is low, according to the ID expert model– the pedagogic model calculates a convincing argument. Then TeachMed tries to reject the query by initiating an argumentation phase during which TeachMed. During this phase, the student's actions will be utterances constituting replies to TeachMed's utterances. These utterances may be arguments, counter arguments, withdrawal of arguments, and acceptance of arguments. The student may also proactively requests help. This too triggers an argumentation phase during which the dialogue moves are utterances.

At any point during the interaction, each arguer is committed to some arguments. For the student, theses include the set of gathered evidence and the set of hypotheses. For TeachMed, they include the set of hypotheses explained by the ID expert model from the evidence gathered by the student. Commitments also include arguments asserted during verbal exchanges. They are updated depending on the performed actions. A structure called the "Commitment Store" keeps track of the current commitments. It is a list of pairs $(argument, arguer)$.

To have a formal argumentation framework modeling the interactions between the student and the ITS, we need a language for modeling the content of arguments and the exchange –or communication– of arguments.

3.1 Domain Definition and Communication Language

Following [13], we define the language at two levels, namely the *domain level* and the *communication level*. At the domain level, the language provides a syntax and semantics for arguments. At the communication level, the language defines primitives for move types - propositional attitudes– that are available for exchanging arguments.

An argument is a premise and a conclusion, where the premise is a conjunction of propositions and the conclusion is a proposition . A move type is a template operator described by a precondition (conjunction of predicates) specifying when the move is feasible –it specifies the conditions that the commitment store and/or the dialogue history must satisfy for the move to be applicable– and an effect specifying the update of the commitment store and the dialogue history. Figure 1(a) illustrates examples of move types: Open, Close, Assert, Accept, Withdraw, Reject, Challenge, and Question.

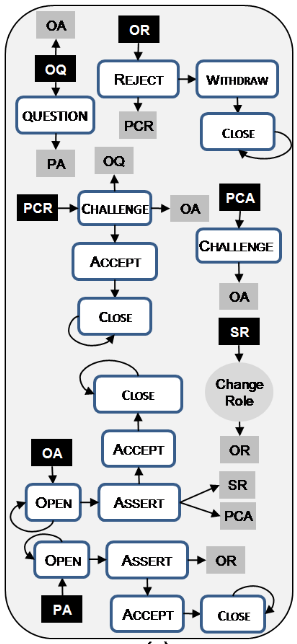
An arguer commits to an argument by asserting the argument or accepting it [14]. Arguers are not limited to committing to only what they believe or to believe what they commit to. Adopting the concept of a commitment store helps us avoiding the complexity and inefficiency regarding the use of a belief update framework in dialogue modeling [15]. As argued by [16], the commitment store concept also provides a means to settle conflicts between arguers by making the opponent commit to the proponent's assertion or the proponent withdraws his assertion. The dialogue-history keeps track of the history of moves made by the arguers. This is a path in dialogue tree. Figure 1(b) shows the details of the move Accept.

(a)

Move name and Usage	Informal Meaning
OPEN M1: OPEN(X,Y) M2: OPEN(Y,X)	**X** as proponent (or **Y** as opponent) indicates to **Y** (or to **X**) initiation of a dialogue with **M1** (or **M2**)
ASSERT M3: ASSERT(X, Y, s) M4: ASSERT(X, Y, ¬s)	**X** asserts premise s via **M3** or its negation via **M4** to **Y**.
ACCEPT M5: ACCEPT(X, Y, s)	**X** indicates to **Y** that it has accepted the premise s.
WITHDRAW M6: WITHDRAW(X,Y, s)	**X** indicates to **Y** that it has withdrawn its commitment to s.
REJECT M7: REJECT(X,Y, s)	**X** indicates its disagreement with the premise s claimed by **Y**.
CHALLENGE M8: CHALLENGE(X,Y, M3) M9: CHALLENGE(X,Y, M7)	**X** challenges **Y** for asserting (or rejecting) premise s via the move **M8** (or **M9**).
QUESTION M10: QUESTION(X, Y, s)	Participant **X** asks a question of **Y** about the validity of s.
CLOSE M11: CLOSE(X, Y) M12: CLOSE(Y, X)	**X** as proponent (or **Y** as opponent) indicates to **Y** termination of dialogue via **M11** (or **M12**).

(b)

ACCEPT (Move M5)
Usage: Accept(X, Y, s)
Meaning: X indicates to Y that it has accepted the premise s.
Precondition: Y must has already committed to s.
Effect: ACCEPT(X, Y, s) is added to DH and (Y,s) is added to CS.



(c)

Fig. 1. (a)Communication Moves; (b)Details of Accept;(c) The protocol

3.2 The Protocol

Each time the student makes an assertion, TeachMed checks whether it agrees with the assertion and whether it should reject it. The situation in which TeachMed intervenes is determined by the pedagogic strategic rules, which we discuss later. For the time being, let's assume the rule is to intervene on every error. For example, whenever the student generates a hypothesis not probabilistically related to current evidences given the ID expert model, TeachMed rejects the assertion by making an argument against it. The student may counter with her own argument against TeachMed's argument. And so on, the argumentation can continue until settling the initial disagreement.

Thus the settling of a disagreement could recursively spawn an argumentation dialogue within a current one. Accordingly we define the argumentation protocol using hierarchical state diagrams similar to statecharts [17, 18]. In the example of Figure 1(c), we have different diagrams, some representing superstates in other diagrams: *proponent assert*(PA), *opponent assert*(**OA**), *proponent challenge reject*(**PCR**), *proponent challenge assert*(**PCA**), *swap roles* (**SR**), *opponent rejection* (**OR**) and *opponent question*(**OQ**). The entry point of each diagram is shown by a black box. A superstate corresponding to a diagram is shown as gray box. A normal state corresponds to a dialogue move. A circle indicates a change in the roles of the arguers, switching from a proponent role (making an assertion) to an opponent one (challenging an assertion), or vice-versa.

3.3 Computing Convincing Arguments

Given an assertion made by the student, TeachMed must decide whether to reject or accept the student assertion. Here we follow Walton's argumentation theory [8] by specifying rules expressing how to respond to arguments made by the opposing party in a two-participant argumentation. Precisely, we want to model the rules for generating counterarguments by TeachMed to convince the student. These argument generation rules, called test questions by Walton, specify arguments that can challenge assertions made by the student - diagnostic actions as well as utterance actions during a conflict settling dialogue.

An argument generation rule (AGR) is a template rule for generating a counterargument to a given assertion, consisting of:

- **Parameters**: Variables used in the template.
- **Argument**: The challenged argument.
- **Context**: A conjunction of predicates over the commitment store and problem solving state
- **Premise**: Premise of the counterargument (conjunction of predicates).
- **Conclusion**: Conclusion of the counterargument (predicate).
- **Value**: Preference value of the counterargument

The variables in the predicates must be defined in the parameters. We associate preferences to the assertions made by the student and associate strengths to arguments generated by the AGRs. The preference value indicates the value of a specific decision making parameter in problem solving that is promoted by the counterargument.

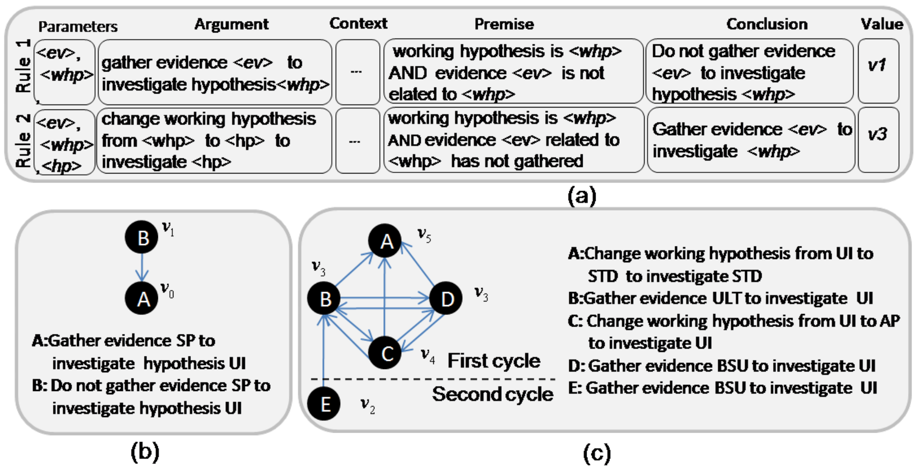


Fig. 2. Rules of test questions (a) and argument generation rules (b, c)

Given an assertion and a set of AGRs , we find a set of potential counterarguments by matching the assertion with the assertion component of the AGRs and the premise component with the context (problem solving state, commitment store). If a match is obtained, the resulting instantiation is used uniformly to replace the variables in the conclusion.

A convincing argument exists among this set of generated arguments if it defeats the student's argument but it is not defeated by any counterargument. To calculate such argument we adopted a decision-theoretic argumentation method from [11] – which is an extension of Dung's argumentation [19] .

Figure 2(a) shows examples of AGRs. Rule 1 is applicable to counter evidence gathered in order to justify a given a hypothesis. A pedagogic strategy for beginners is to forbid them zigzagging between hypotheses (e.g., asking questions to the patient related to one hypothesis, switching to another then coming back to the previous, and so on). To enforce such a strategy, the teacher may require the student to stick to one hypothesis as far as possible by exhausting the related evidence [20]. Rule 2 is meant to enforce such a pedagogic strategy by countering inappropriate changes of the current working hypothesis. This rule is matched when the student asks a query related to a new hypothesis while the ID suggests there are still evidences relevant to the current working hypothesis.

3.4 Move Selection

Given a convincing argument against the student's assertion, a dialogue strategy is needed to conveying the different parts of the arguments (premises and conclusion) in form of an argumentation dialogue. Such strategies are specified using an argumentation protocol, specified as a hierarchical transition diagram. In practice, we specify a dialogue strategy by following a three-level methodology, inspired from [12]:

1. maintain or alter focus of discussion
2. building own point of view or destroying the user's view
3. adopting method to achieve the objective set at level 1 and 2.

Level 1 is appropriate where concepts like relevance are important [21]. Following [22], at level 1, information of the student's profile regarding preferences among decision making parameters is taken into account. At level 2, build and destroy strategies are encoded in the protocol. Building means making the user to accept the proposition which shows TeachMed's point of view. Destroying means making the user withdraw of a proposition which indicates her point of view. The two first levels include some pedagogic goals. The third level refers to some domain dependent tactics which achieve those goals.

For example, the following rules are used at the third level to decide a move for TeachMed in reaction to the student's move:

- **PA** rules: If the student as proponent asserts an argument and TeachMed does not find a convincing argument against it then the argument is accepted and the dialogue ends (PA-1). Otherwise, TeachMed rejects the assertion through a transition to the OR diagram (PA-2).
- **OR** rules: if the student rejects an argument then TeachMed challenges the student through a transition to the PCR diagram to find the cause of conflict.
- **PCR** rules: if the student challenges the rejection of her argument by TeachMed, then the TeachMed tries to resolve the conflict by using a destroy strategy or a build strategy. If the destroy strategy has chosen, and a premise of the convincing argument exists to which the student has not committed so far, TeachMed begins asks a question on that premise through a transition to the OQ diagram (PCR-1). Otherwise, if such premise does not exist, or the build strategy is chosen, then TeachMed asserts the convincing argument through a transition to the OA diagram (PCR-2).
- **OA** rules: if the student as opponent has asserted an argument then TeachMed has to accept the assertion if it does not have a convincing counterargument (OA-1). Otherwise, TeachMed indicates its conflict with the assertion by rejecting the asserted argument through the choice of a transition to the SR diagram, which in turn leads to the OR diagram (OA-2).

We also have rules controlling backtracking after a child dialogue terminates. For instance, if the terminated child dialogue was created by the OA diagram invoked by the OQ diagram, and the proponent has committed to the assertion made by the opponent, then this means that the opponent has succeeded in applying a destroy strategy to justify the rejection. Therefore backtracking is necessary to REJECT state of the OR diagram of the parent. This gives the proponent the opportunity to make another choice as response to the opponent's rejection.

4 Example

Let's define the set of preferences $V = \{v_1, v_2, v_3, v_4, v_5\}$ with partial order $v_1 > v_2 > v_3 > v_4 > v_5$. Let's then associate these preferences to pedagogic goals as follows:

- v_1: states evidence gathering actions must be consistent with medical knowledge and the available patient information;
- v_2 : evidence gathering actions should be minimized (e.g., take into account the monetary costs of lab tests; delays; and intrusive physical exams). TeachMed evaluates this by taking into account the expect value of information for queries.
- v_3: the current working hypothesis should remain the focus until exhausting related evidence;
- v_4: most life-threatening hypothesis should be investigated first.
- v_5: most likely hypothesis should be selected first.

Initially, TeachMed presents a patient to the student with short description of the patient complain (e.g., "patient complaining of abdominal pain") together with the vital signs. The student can initialize a differential diagnosis right away based on the problem statement. In the scenario illustrated herein, a student was presented with a pelvic inflammation case, the patient complaining of abdominal pain. The ID is the same as in [1] and covers abdominal pains.

Figure 3 depicts an excerpt of the scenario with a trace of the internal inferences behind the argumentation process. Until Step 4, the student was querying the patient. The dialogue states (third column) and participant's role (fourth column) are those in Figure 1. The moves (fifth column) are those discussed in Section 3.1. The rules (last column) are those in Section 3.4.

The student began with the hypotheses of *Urinary Infection*(UI), *Sexual Transmitted Disease*(DST) and *Appendicitis*(AP). UI is the most probable hypothesis and AP is the less probable one but is threatening to the patient's life. Thereafter, the student has started investigation of the UI (current working hypothesis).

According to Step 1, the student's action (proponent's assertion M3), initiated the persuasive dialogue D1. TeachMed and the student synchronize the start of new dialogue by the move OPEN and end it by the move CLOSE. Based on the PA diagram, TeachMed has to choose among two permitted moves: M5 and M7. As it does not find any argument against the student move, the rule PA-1 is matched, and TeachMed agrees with the student's assertion by making the move M5. Note that, to provide a natural problem solving interaction for the student, TeachMed remains silent when it accepts the student's diagnostic action.

Collected Evidence: Acute Lower abdominal pain
Working Hypothesis: Urinary Infection (UI)

Player	Utterance	Dialog#	Role	Moves	Rules
1. Student:	Any fever?	D1(Open)	P	M1, M2, M3	
Patient:	I don't know.				
TM Pedagogue:	Silence	D1(Close)	O	M5, M11, M12	PA-1
2. Student:	Is it worse on one side?	D2(Open)	P	M1, M2, M3	
Patient:	No				
TM Pedagogue:	Silence	D2(Close)	O	M5, M11, M12	PA-1
3. Student:	Do you urinate more since beginning of your pain?	D3(Open)	P	M1, M2, M3	
Patient:	I don't know. More often I think.				
TM Pedagogue:	Silence	D3(Close)	O	M5, M11, M12	PA-1
4. Student:	Do you have a sexual partner(SP)?	D4(Open)	P	M1, M2, M3	
5. TM Pedagogue:	You cannot ask such a question.		O	M7	PA-2
6. Student:	Why?			M9	
7. TM Pedagogue:	It is not related to the working hypothesis UI.	D5(Open)	P	M2, M1, M4	PCR-2
8. Student:	But I can.		O	M7	
9. TM Pedagogue:	Why?		P	M9	OR
10. Student:	Because I investigate Sexual Transmitted Disease (STD).	D6(Open)	P	M1, M2, M3	
11. TM Pedagogue:	You cannot investigate STD		O	M7	OA-2
12. Student:	Why?		P	M9	
13. TM Pedagogue:	Did you finish working with UI?		O	M10	PCR-1
14. Student:	Yes.	D7(Open)	P	M1, M2, M4	
15. TM Pedagogue:	But working on UI has not finished yet.		O	M7	PA-2
16. Student:	Why?		P	M9	
17. TM Pedagogue:	Evidence Burning Sensation of Urinating (BSU) has remained.	D8(Open)	P	M2, M1, M3	PCR-2
18. Student:	You are right.	D8(Close)	O	M5, M12, M11	
19. Student:	I withdraw my claim that working with UI has finished.	D7(Close)	P	M6, M12, M11	BT
20. TM Pedagogue:	So you should continue working with UI .	D9(Open)	P	M2, M1, M3	BT, PCR-2
21. Student:	Ok.	D9(Close)	O	M5, M12, M11	
22. Student:	I withdraw my claim for changing working hypothesis UI to STD.	D6(Close)	P	M6, M12, M11	BT
23. Student:	I accept that my question about SP is not related to the UI.	D5(Close)	O	M5, M11, M12	BT
24. Student:	I withdraw my claim for gathering evidence SP.	D4(Close)	P	M6, M12, M11	BT

Fig. 3. Medical Diagnostic Scenario. Shorthand notations: P (Proponent), O(Opponent), BT (Backtracking)

Until Step 4, TeachMed has not found any conflict with the student's actions (assertions). At Step 4, TeachMed notices inconsistencies between the evidences and the hypotheses formulated by the student. This matches the counter-argument B in Figure 2(b), which spawns further argumentation with the student to settle the disagreement. This time, the rule PA-2 matches, so that among two permitted moves (M5 and M7) TeachMed chooses M7, meaning it rejects the student's action (Step 5). In response, according to choices offered by the OR diagram, the student has to choose among two moves M6 and M9. Permitted moves are provided to the students through a menu selection.

All through steps 5 to 24 , the matched counter arguments and move selection rules drove TeachMed towards persuading the student of her mistake and remedy the situation. More specifically, from step 5 to 10 TeachMed uses only the persuasive argument B of figure 2.b but after Step 10 it uses also the persuasive argument E of figure 2.c since any assertion made by the student or TeachMed starts a new persuasive dialogue. In this example, the student ends up accepting that she had committed a mistake and switched back to the urinary infection.

5 Conclusion and Future work

In this paper we described an argumentation framework that can be integrated to any ITS for conducting a persuasive dialogue with the student. Our framework is still in a prototyping phase and still has some limitations. The above scenario works as indicated in the current implementation. Arguable, the dialogue with the student is still not yet realistic, mainly because the argument rule base still needs significant fine tunings. In particular, the utterances made by the students are actually text templates on move choices offered to him at the current step of the interaction. Refinement of the dialogue transitions and the utterance templates will contribute to making the dialogues more realistic.

The fact that an assertion made by a student can be challenged, from a pedagogic point of view, it does not mean that ITS should indeed challenge it. It can be very frustrating for a student to see ITS intervene on every error. Rather, depending on pedagogic goals and constraints set by a teacher, the ITS should intervene when a given number of errors with some level of severity have accumulated. This provides another area of improvement.

Although still quite preliminary, the current experiment demonstrates the potential of our approach in fostering learning by the student, by making her reveal her understanding of current problem solving step, and leading her to actively search in her knowledge to generate a convincing argument, reflect upon it, and remedy to a situation. Besides improving the current implementation as just mentioned, a crucial component of future works concerns an evaluation of the argumentation capability to determine to what extent it indeed facilitates diagnostic skill acquisition compared to some other types of training intervention.

6 Acknowledgement

We wish to thank the reviewers for insightful and constructive comments which helped us improving the writing of this paper.

References

1. Kabanza, F., Bisson, G., Charneau, A., Jang, T.S.: Implementing tutoring strategies into a patient simulator for clinical reasoning learning. Journal of Artificial Intelligence In Medicine (AIIM) **38** (2006) 79–96
2. Zhou, Y., Freedman, R., Glass, M., Michael, J.A., Rovick, A.A., Evens, M.W.: Delivering hints in a dialogue-based intelligent tutoring system. In: Proceedings of the Sixteenth National Conference on Artificial Intelligence. (1999)
3. Freedman, R.: Plan-based dialogue management in a physics tutor. In: Proceedings of the 6th Applied Natural Language Processing Conference, Seattle (2000)

4. Zinn, C., Moore, J.D., Core, M.G.: A 3-tier planning architecture for managing tutorial dialogue. In: The 6th International Conference on Intelligent Tutoring Systems, London, UK, Springer-Verlag (2002) 574–584
5. Nguyen, H.: Designing persuasive health behaviour change dialogs. In: Proceedings of the 11th international conference on User Modeling. (2007) 475–479
6. Andrews, P., De Boni, M., Manandhar, S.: Persuasive argumentation in human computer dialogue. In: Proceedings of the AAAI 2006 Spring Symposia. (2006)
7. Guerini, M., Stock, O., Zancanaro, M.: Persuasion models for intelligent interfaces. In: Proceedings of the IJCAI Workshop on Computational Models of Natural Argument. (2003)
8. Walton, D.: Argumentation Schemes for Presumptive Reasoning. Erbaum: Mahwah, NJ (1996a)
9. Prakken, H.: Formal systems for persuasion dialogue. Knowl. Eng. Rev. **21**(2) (2006) 163–188
10. Ashley, K., Pinkwart, N., Lynch, C., Aleven, V.: Learning by diagramming supreme court oral arguments. In: Proceedings of the 11th international conference on Artificial intelligence and law. (2007) 271–275
11. Bench-Capon, T.J.M.: Persuasion in practical argument using value-based argumentation frameworks. Journal of Logic and Computation **13**(3) (2003) 429–448
12. Moore, D.: Dialogue game theory for intelligent tutoring systems. Ph.d. dissertation, Leeds Metropolitan University, Leeds, UK (1993)
13. Sierra, C., Jennings, N.R., Noriega, P., Parsons, S.: A framework for argumentation-based negotiation. In: Proceedings of 4th International Workshop on Agent Theories Architectures and Languages. (1998) 167–182
14. Mackenzie, J.D.: Question-begging in non-cumulative systems. Journal of Philosophical Logic (1979) 117–133
15. Kibble, R.: Reasoning about propositional commitments in dialogue. Research on Language and Computation **4**(2-3) (2006) 179–202
16. Walton, D.: Argument Structure: A Pragmatic Theory. University of Toronto Press (1996b)
17. Harel, D.: Statecharts: A visual formalism for complex systems. Science of Computer Programming **8**(3) (1987) 231–274
18. Drusinsky, D.: Modeling and Verification Using UML Statecharts, First Edition : A Working Guide to Reactive System Design, Runtime Monitoring and Execution-based Model Checking. Newnes (2006)
19. Dung., P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence **77**(2) (1995) 321–358
20. Chamberland, M., Hivon, R., Tardif, J., Bédard, D.: Évolution du raisonnement clinique au cours d'un stage d'externat : une etude exploratoire. Pédagogie Médicale **2** (2001) 9–17
21. Prakken, H.: Relating protocols for dynamic dispute with logics for defeasible argumentation. Synthese **127** (2001) 187–219
22. Amgoud, L., Maudet, N.: Strategical considerations for argumentative agents (preliminary report). In: Proceeding of 9th International Workshop on Non-Monotonic Reasoning, Special session on Argument, Dialogue, and Decision. (2002) 399–407