Towards a dialectical approach for conversational agents in selling situations

Maxime Morge, Sameh Abdel-Naby, and Bruno Beaufils

Laboratoire d'Informatique Fondamentale de Lille Université Lille 1 Bât M3 - F-59655 Villeneuve d'Ascq {maxime.morge,sameh.abdel-naby,bruno.beaufils}@lifl.fr

Abstract. The use of virtual agents to intelligently interface with online customers of e-commerce businesses is remarkably increasing. Most of these virtual agents are designed to assist online customers while searching for information related to a specific product or service, while few agents are intended for promoting and selling a product or a service. Within the later type, our aim is to provide proactive agents that recommend a specific item and justify this recommendation to a customer based on his purchases history and his needs. In this paper, we propose a dialectical argumentation approach that would allow virtual agents that have sales goals to trigger persuasions with e-commerce's customers. Then, we illustrate the proposed idea through its integration with an example from real-life.

Categories and Subject Descriptors I.2.11 [Artificial Intelligence]: Intelligent Agents.

General Terms Algorithms.

Keywords

Argumentation, E-commerce, Agents, Language Processors.

1 Introduction

Within the last twelve years, precisely from 1998 wherein the dot-coms' boom first made an impact, e-commerce has succeeded to pursue a massive number of shoppers to change their idea of buying [1]. Several existing businesses have taken an advantage of this boom by adding a virtual presence to their physical one by means of an e-commerce website, these companies are now called *brick and mortar* businesses (e.g., Barnes & Noble). Additionally, new companies that exist only through the web, called *bricks and clicks* businesses, have also appeared (e.g., Amazon). Although the online presence of companies is cost-efficient, yet the lack of a persuading salesman affects the transformation ratio (sales vs. visits). Apart from the *Business*'s reaction to the boom, in Computer Science, several research efforts were made to study, analyze, and better shape the processes of assisting customers while being present in an e-commerce space [2, 3]. In Artificial Intelligence, a considerable amount of the research conducted in the area of Software Agents [4] focus on the enhancement and the proper provision of online Embodied Conversational Agents (ECAs) [5].

Whether these agents sell, assist, or just recommend, it is now clear that such autonomous agents are capable of engaging in verbal and non-verbal dialogues with e-commerce's customers. However, the ability of these agents to transform an ordinary visitor of an e-commerce who needs assistance to an actual buyer is yet of no notable weight. For an overview of the issues encountering the development of virtual sales agents refer to [6].

In this paper, we propose the use of dialectical argumentation technologies as a step on the way to increase the sales-oriented negotiation skills of software agents in the business-to-consumer (B2C) segment of e-commerce. For this purpose, we suggest the exploitation of existing argumentation tools, such as those found in [7–9]. Using these tools we intend to build a sales-driven dialogue system that is capable of leading a virtual seller agent to influence the decision of a potential buyer in an e-commerce setting. Then, we illustrate the proposed idea through its integration with an example from real-life.

This paper is organized as follows. In section 2 we give an overview of the existing dialogue systems while pointing out their limitations. In section 3 we adopt a different approach for dialogue management based upon argumentation. Section 4 illustrates this approach using an intuitive scenario. Section 5 briefly describes the CSO language processor on which our dialogue system is based. The rest of the paper overviews the dialogue-game protocol we use. Section 7 presents our realization of the dialogue strategy. We then conclude this paper by discussing some of the related work and, providing a summary of our future work.

2 Dialogue systems

A dialogue system is a computer system that is capable of interacting with humans using the language they understand - *natural language*. Similar to that we can find TRAINS-93 [10], Collagen [11] and Artemis Agent Technology [12], which are mixed-initiative dialogue systems for collaborative problem solving. These dialogue systems can respond to initiatives made by users and, they also take initiatives themselves, which is required to support a selling process.

TRAINS-93 [10], Collagen [11] and Artemis Agent Technology [12] are adopting the same approach of focusing on the dialogue modelling itself besides the dialogue management that is based on intentions recognition. For example, out of the following utterance of a user, "I want to purchase a quilt", there can be three possible interpretations:

1. It can be a direct report of a need;

- 2. It can be a statement of a goal that a user is pursuing independently;
- 3. It can be a proposal to adopt this joint goal.

Particularly, the discourse structure considered by Collagen in [11] is based on a comprehensive axiomatization of SharedPlans [13], while TRAINS-93 and Artemis Agent Technology are based upon a BDI approach [14]. The semantics of utterances is specified with the help of a first order modal logic language using operators as Beliefs, Desires and Intentions. The notions of persistent goal is a composite mental attitude which is defined from the previous operators in order to formalize the intention expressed by utterances. According to the semantic language of FIPA-ACL [15] adopted by the Artemis Agent Technology, an agent i has p as a persistent goal, if i has p as a goal and is self-committed toward this goal until i comes to believe that the goal is achieved or, this goal is unachievable. Here, an intention is defined as a persistent goal imposing the agent to act, which accordingly generates a planning process.

The process of inferring intentions from actions is needed to constraint and reduce the amount of communications exchanged. Also, it is worth noticing here that it is hard to incorporate this process into practical computer systems due to the complexities encountered while facilitating natural intractability. Therefore, it is then required to develop a heuristic mechanism for software agents in a collaborative setting.

For this purpose, dialogue systems are required to recognize the intention of the user and reason about it. The implementation of this theory is problematic due to its computational complexity [16]. Moreover, the specification of the semantics for the speech acts in terms of mental states is not adapted for resolving the conflicts which can appear during a selling process. For instance, an information that is received by a virtual seller agent must be adopted even if this information is contradictory with its beliefs. Those are the reasons why we consider an alternative approach based upon dialectical argumentation.

3 Dialectical approach

Our approach for dialogue modelling considers the exchange of utterances as an argumentation process regulated by some normative rules that we call *dialogue*game protocol. Our approach is inspired by the notion of dialectical system that Charles L. Hamblin introduced in [17]. A **dialectical system** is a family of regulated dialogue, (i.e., a system through which a set of participants communicate in accordance with some rules).

From this perspective, Walton and Krabbe in [18] define a **dialogue** as a coherent and structured sequence of utterances aiming at moving from an initial state to reach the goals of the participants. These are the dialogue's *goals* that can be shared by the participants or they can be also each of the participants' individual goals. Based on this definition, Walton and Krabbe have distinguished between five main categories of dialogues depending on the initial situation and goals. These categories are: information seeking, persuasion, negotiation, enquiry and deliberation [18].

Initial situation \rightarrow	Conflict	Open problem	Ignorance of
$\operatorname{Goal}\downarrow$			a participant
Stable agreement	persuasion	enquiry	information
i.e., Resolution			seeking
Practical settlement	negotiation	deliberation	Ø
i.e., Decision			

Table 1: Systemic overview of dialogue categories

Table 1 represents the analysis grid for dialogues proposed by Walton and Krabbe. An **information seeking** appears when a participant aims at catching knowledge from its interlocutor. The goal is to spread knowledge. In a **persuasion** dialogue, the initial situation is disagreement, (i.e., a conflict of opinion). The goal consists of solving the conflict by verbal means. In a **negotiation** dialogue, the initial situation is a conflict of interest mixed with a need for collaboration. The goal consists of a deal, i.e. an agreement attracting all participants to maximizing their gains. An **enquiry** dialogue aims at establishing (or demonstrating) the truth of a predicate. This one must answer to an open question and a stable agreement emerges. Each participant aims at extending their knowledge. A **deliberation**, as an enquiry, begins with an open problem rather than a conflict. The discussion is about the means and ends of a future action. It is worth noticing that, in real world, the nature of dialogues can be mixed. A dialogue can be composed of different sub-dialogues with different natures as we will see in our scenario.

4 Dialogue: Phases & Purposes.

In this section, we explain the different phases of the overall online sales process that we are attempting to tackle in our research. Within these phases, we expect our virtual agent to rely on a specific language processor - *explained further ahead* - to handle online one-to-one conversations, related misspelling, and the use of diverse languages. Since the existing language processor is already capable of handling what is known to us as After-Sales, (i.e., assisting online users while searching for problems' answers), we then became extra interested to increase the salesability of this agent.

• **BEFORE-SALE**: in this phase we distinguish between two different processes that are possibly interleaved: a) the process of needs identification and, b) the process of product selection.

The **Needs Identification** can be performed with the help of an information seeking dialogue shifting from an initial asymmetric situation to a final one where both of the players share the user requirements.

The **Product Selection** allows the participants to constraint and to reduce the amount of communication by considering only relevant products later in the selling process. This task, in overall, also supports the information seeking dialogue where the virtual seller agent asks discriminatory questions in order to narrow its focus into a single product.

Both of these dialogues can be interleaved. The aim of the virtual seller agent here is to spread information about the products, while the aim of the user is expected to be the spreading of information about his needs.

• **SALE**: here, the aim for all dialogues' parties is to bargain over their interests and, eventually, "make a deal". For this purpose, the participants play a role in a **negotiation** dialogue. The simplest dialogue is: the virtual seller agent makes an offer and the user accepts or refuses this proposal.

If there is no single product corresponding to the user needs, then the participants attempt to maximize their benefits by conceding some aspects while insisting on others. If no product is matching user's needs, the user's high-ranked features of the products, (i.e., top priority conditions), are then altered to a lower ranked ones. On the other hand, if more than one product is corresponding to user's needs, the virtual seller agent picks the ones with the highest gross margin.

Later to that, the virtual seller agent can suggest - *after a single sale* - additional sub-items or an offer which invokes more of the user needs. These later approaches are well-known marketing techniques, (i.e., cross-selling and up-selling). In both cases, the new items aim at persuading the customer.

• AFTER-SALE: The initial condition of this phase is not coming out of a sales conflict, however, it comes out of the user's unawareness of the problem related to the concerned product. Therefore, participants here aim at extending their knowledge about each foreseen encounter and, eventually, decide about their subsequent action, (e.g., product replacement). As a result, users here play in an enquiry dialogue or a deliberation fashion.

The dialogue in Fig 1 consists of a specific case run which embraces the beforesale and sale phases between a user and a virtual seller agent (VSA). While the utterances #3 and #5 are related to the *Needs Identification*, the utterances #1,#6 and #8 are of the *Product Selection*. The rest of utterances are directly related to the sales process. The utterance #9 is a single-sale offer, the utterance #11 captures an up-selling offer and the utterance #13 is a cross-selling proposal.

It is also worth highlighting here that relying on argumentation will make the virtual seller agent more credible and, then, persuasive, since its appearance or involvement will yield from the fulfillment of a set of predefined conditions, (e.g., user X is wandering in the quilt section).

The After-Sale phase, (i.e., assistance), is out of the scope of this paper since the existing query answering systems, such as the CSO Language Processor we present in the next section, have been practically proven to be efficient for such tasks.

- 1. VSA: Your moves through our website indicates that you might be interested in Quilts, do you need help to select one?
- 2. User: Nice.
- 3. VSA: Are you allergic?
- 4. User: Yes.
- 5. VSA: Do you sweat?
- 6. User: Sure. Is there any organic quilts ?
- 7. VSA: Definitely, but which dimensions are you looking for?
- 8. User: 200x200. Is there any one within a budget of \in 350?
- VSA: Yes, it cost € 239 since the dimensions are 200x200, it is nonallergenic and organic.
- 10. User: I buy it.
- 11. VSA: Since it is still within your budget, I may also suggest you to buy a "four seasons" one that cost $\in 279$?
- 12. User: Good idea.
- 13. VSA: What about a quilt cover that $cost \in 99$ only? still in your budget.
- 14. User: No, thanks.

Fig. 1: Virtual Seller Agent Vs. User: Before-Sale & Sale scenarios

5 CSO Language Processor

The CSO Language Processor [9] is the technology upon which our dialogue system is built.

The CSO Language Processor provided by Artificial Solutions¹ allows to produce virtual dialogue assistants such as Anna² and numerous other systems in more than 20 different languages. Elbot, which has won the Loebner Prize³ in 2008, is built upon this technology.

This technology manages users' sessions, handles misspellings and it also contains a language dependent preprocessing feature. In accordance with the dialogue state, it selects and carries out the best dialogue move. Additionally, this technology is able to interact with a back-end system, (e.g., databases), to hand out answer document for requesting application/front end and to write log files for analysis.

The inputs of the language processor are the user queries, (i.e., the user's identity and his text inputs). After the identification of the session, the inputs are divided in sentences and words and the spelling is corrected. Another phase is carried out wherein an interpretation of the inputs is made: an answer retrieval for each sentences of the user's inputs based on some **interaction rules in a knowledge base**. Finally, the answer is selected and generated by replacing some template variables.

¹ http://www.artificial-solutions.com

² http://www.ikea.com

³ http://www.loebner.net/Prizef/loebner-prize.html

The interaction rules combine the meaning of the user's inputs and the dialogue state to define the conditions under which a dialogue move may be uttered. A given move can only be performed if the conditions are completely fulfilled.

However, the core of the CSO Language Processor is an inference engine that implements forward-chaining and so reactionary. Therefore, the language processor only makes it possible to respond to a user's queries and not to initiate or lead a sales-driven conversations. consequently, in order for us to make CSO proactive and enable it to go through sales-driven encounters, we introduce in the next section a formal framework for possible sale-driven dialogue management that can be adapted by virtual agents.

6 Dialectical system

A dialogue is a social interaction amongst self-interested parties intended to reach a common goal. In this section, we present how our game-based social model [8] handles the forseen conversation between a user and a virtual seller agent (VSA).

A dialectical system is a formal system that regulate persuasion dialogue, (See [19] for an overview). According to the game metaphor for social interactions, the parties are players which utter moves according to social rules.

Definition 1 (Dialectical system). Let us consider \mathcal{L} a common object language and \mathcal{ACL} a common agent communication language. A dialectical system is a tuple $DS = \langle P, \Omega_M, H, T, proto, Z \rangle$ where:

- P is a set of participants called players;
- $\Omega_M \subseteq \mathcal{ACL}$ is a set of well-formed moves;
- H is a set of histories, the sequences of well-formed moves s.t. the speaker of a move is determined at each stage by the turn-taking function T and the moves agree with the dialogue-game protocol proto;
- T: $H \rightarrow P$ is the turn-taking function;
- proto: $H \rightarrow 2^{\Omega_M}$ is the function determining the legal moves which are allowed to expand an history;
- -Z is the set of dialogues, i.e. the terminal histories.

Here, DS reflects the formalization of social interactions between players uttering moves during a dialogue. Each dialogue is a maximally long sequence of moves. Later to that, we specify informally the elements of DS for bilateral negotiation and information-seeking.

In our scenario, there are two players: the initiator init and, the responder **resp**, which utter moves each in turn. Since we address a proactive dialogue system, we consider the initiator to be a VSA. The **syntax** of moves is in conformance with a common **agent communication language**, \mathcal{ACL} . A move at time t: has an identifier, mv_t ; is uttered by a speaker ($sp_t \in P$) and the speech act is composed of a locution loc_t and a content $content_t$.

The possible locutions are: question, assert, unknow, introduce, request, accept and reject. The content consists of all instances of the following schemata

1.	VSA:	question(is(product,quilt) because search(user,product)).
2.	User:	assert(is(product, quilt)).
3.	VSA:	question(is(user, allergic)).
4.	User:	assert(is(user, allergic)).
5.	VSA:	question(is(user, sweat)).
6.	User:	assert(is(user, sweat)).
		<pre>question(is(product, organic)).</pre>
7.	VSA:	assert(is(product, organic)).
		question(dimension(product, 200, 200)).
8.	User:	assert(dimension(product, 200, 200)).
		question(budget(product, 350)).
9.	VSA:	introduce(is(product,quilt) because budget(product,239) and
		<pre>is(product, nonallergenic)) and is(product, organic)).</pre>
10.	User:	accept(is(product,quilt).
11.	VSA:	<pre>introduce(is(product, quilt) because budget(product, 279) and</pre>
		<pre>is(product, nonallergenic)) and is(product, fourseasons)).</pre>
12.	User:	accept(is(product,quilt).
13.	VSA:	<pre>introduce(is(product, quilt)) and is(product, quiltcovers) because</pre>
		<pre>budget(product, 333.90) and is(product, nonallergenic)) and</pre>
		is(product, fourseasons)).
14.	User:	reject(is(product, quiltcovers).

Fig. 2: A Possible Scenario Formalization

"S (because S')" where S (eventually S') is a set of sentences in the common object language, \mathcal{L} . Actually, natural language utterances are interpreted/generated by the language dependent preprocessing of CSO (See Section 5). A move is an abstract representation of natural language utterances.

The dialogue in Fig 2 depicts a possible formalization of the natural language dialogue of Fig 1. It is worth noticing here that each utterance can contain more than one move.

In Fig. 3, we present our dialogue-game protocols by means of a deterministic finite-state automaton. An information-seeking dialogue begins with a question. The legal responding speech acts are **assert** and **unknow**. Two possible cases can occur: i) the dialogue is a failure if it is closed by an **unknow**; ii) the dialogue is a success if it is closed by an **assert**. A negotiation dialogue either begins with an offer from the VSA through the speech act **introduce** or the offer is suggested by the user through the speech act **request**. The legal responding speech acts are **accept** and **reject**. Here, the possibly occurring cases are: i) the dialogue is a failure if it is closed by a **reject**; ii) the dialogue is a success if it is closed by an **accept**.

The strategy interfaces with the dialogue-game protocol through the condition mechanism of utterances for a move. For example, at a certain point in the dialogue the VSA is able to send introduce or question. The choice of which locution and which content to send is depending on the VSA's strategy. For instance, the VSA is **benevolent** in the dialogue represented in Fig 2 since he first



Fig. 3: Dialogue-game protocol for information-seeking (on the left), and negotiation (on the right)

attempts to identify the dialogue's party needs, he continues with the product selection phase and then it terminates with the sale dialogue. An **aggressive** agent would consider the sale prior to anything whether the before-sale tasks have been performed or not.

7 Arguing over utterances

In this section, we present how our computational model of argumentation for decision making [7] handles the dialogue strategy in order to generate and evaluate utterances.

In our framework, the knowledge is represented by a logical theory built upon an underlying logic-based language. In this language we distinguish between several different categories of predicate symbols. We use *goals* to represent the possible objectives of the decision making process (e.g. the dialogue to perform), *decisions* an agent can adopt (e.g. the move to utter) and a set of predicate symbols for *beliefs* (e.g. the previous utterance).

Assumptions here are required to carry on the reasoning process with incomplete knowledge, (e.g. some information about user's needs are missing), and we need to express *preferences* between different goals (e.g. some dialogues are prior depending on the agent's strategy). Finally, we allow the representation of explicit *incompatibilities* between goals, decisions and beliefs.

Definition 2 (Decision framework). A decision framework is a tuple $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$, where:

 $- \mathcal{DL} = \mathcal{G} \cup \mathcal{D} \cup \mathcal{B} \text{ is a set of predicate symbols called the$ **decision language**, $where we distinguish between goals (<math>\mathcal{G}$), decisions (\mathcal{D}) and beliefs (\mathcal{B});

- Asm is a set of atomic formulae built upon predicates in DL called assumptions;
- \mathcal{I} is the **incompatibility relation**, *i.e.* a binary relation over atomic formulae in \mathcal{G} , \mathcal{B} or \mathcal{D} . We require \mathcal{I} to be asymmetric;
- \mathcal{T} is a logic **theory** built upon \mathcal{DL} ; statements in \mathcal{T} are clauses, each of them has a distinguished name;
- $-\mathcal{P} \subseteq \mathcal{G} \times \mathcal{G}$ is the **priority** relation, namely a transitive, irreflexive and asymmetric relation over atomic formulae in \mathcal{G} .

In our framework, we consider multiple objectives which may or not be fulfilled by a set of decisions under certain circumstances. Additionally, we explicitly distinguish *assumable* (respectively *non-assumable*) literals which can (respectively cannot) be assumed to hold, as long as there is no evidence to the contrary. Decisions as well as some beliefs can be assumed. In this way, DF can model the incompleteness of knowledge.

The most natural way to represent conflicts in our object language is throughout some forms of logical negation. We consider two types of negation, as usual, (e.g., in extended logic programming), namely strong negation \neg (also called *explicit* or classical negation), and weak negation \sim , also called negation as failure. As a consequence we will distinguish between strong literals, i.e. atomic formula possibly preceded by strong negation, and weak literals, i.e. literals of the form $\sim L$, where L is a strong literal. The intuitive meaning of a strong literal $\neg L$ is "L is definitely not the case", while $\sim L$ intuitively means "There is no evidence that L is the case". The set \mathcal{I} of incompatibilities contains some default incompatibilities related to negation on the one hand, and to the nature of decision predicates on the other hand. Indeed, given an atom A, we have A $\mathcal{I} \neg A$ as well as $\neg A \mathcal{I} A$. Moreover, $L \mathcal{I} \sim L$, whatever L is, representing the intuition that L is evidence to the contrary of $\sim L$. Notice, however, that we do not have $\sim L \mathcal{I} L$, as in the spirit of weak negation.

Other default incompatibilities are related to decisions, since different alternatives for the same decision predicate are incompatible with one another. Hence, $D(a_1) \mathcal{I} D(a_2)$ and $D(a_2) \mathcal{I} D(a_1)$, D being a decision predicate in \mathcal{D} , and a_1 and a_2 being different constants representing different⁴ alternatives for D. Depending on the particular decision problem being represented by the framework, \mathcal{I} may contain further non-default incompatibilities. For instance, we may have $g \mathcal{I} g'$, where g, g' are different goals.

To summarize, the incompatibility relation captures the conflicts, either default or domain dependent, amongst decisions, beliefs and goals. The incompatibility relation can be easily lifted to set of sentences. We say that two sets of sentences Φ_1 and Φ_2 are *incompatible* (still denoted by $\Phi_1 \mathcal{I} \Phi_2$) if there is a sentence ϕ_1 in Φ_1 and a sentence ϕ_2 in Φ_2 such that $\phi_1 \mathcal{I} \phi_2$.

A theory gathers the statements about the decision problem.

Definition 3 (Theory). A theory \mathcal{T} is an extended logic program, i.e a finite set of rules $R: L_0 \leftarrow L_1, \ldots, L_j, \sim L_{j+1}, \ldots, \sim L_n$ with $n \ge 0$, each L_i (with

⁴ Notice that in general a decision can be addressed by more than two alternatives.

 $i \geq 0$) being a strong literal in \mathcal{L} . R, called the unique name of the rule, is an atomic formula of \mathcal{L} . All variables occurring in a rule are implicitly universally quantified over the whole rule. A rule with variables is a scheme standing for all its ground instances.

To simplify, we assume that names of rules are neither in the bodies nor in the head of the rules thus avoiding self-reference problems. We assume that the elements in the body of rules are independent. Besides, we suppose the decisions do not influence the beliefs and the decisions have no side effects.

In order to evaluate the relative importance of goals, we consider the *priority* relation \mathcal{P} over the goals in \mathcal{G} , which is transitive, irreflexive and asymmetric. $G_1\mathcal{P}G_2$ can be read " G_1 has priority over G_2 ". There is no priority between G_1 and G_2 , either because G_1 and G_2 are *ex æquo* (denoted $G_1 \simeq G_2$), or because G_1 and G_2 are not comparable.

We consider the dialogue formalized in Section 6. The generation and the evaluation of utterances by the VSA are captured by a *decision framework* $DF = \langle \mathcal{DL}, \mathcal{A}sm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ where:

- the decision language \mathcal{DL} distinguishes,

- a set of goals \mathcal{G} . This set of literals identifies various motivations for driving the possible dialogues, negotiation (negotiating(product)) and information-seeking ones for product selection (infoseeking(product)) or need identification (infoseeking(user)),
- a set of **decisions** D. This set of literals identifies the possible utterances (e.g. send(question(is(user, allergic)))),
- a set of beliefs, i.e. a set of literals identifying various situations identifying the possible queries of the user,
 (e.g. receive(question(is(product, nonallergenic))), behavior through the website (e.g. search(user, quilt)) or the knowledge about the product/needs information (e.g. is(user, allergic));
- the set of assumptions Asm contains the possible decisions and the missing information about the user, (e.g. ~ is(user,allergic)), or the product, (e.g. ~ is(product,nonallergenic));
- the incompatibility relation \mathcal{I} is trivially defined. For instance, send(x) \mathcal{I} send(y), with $x \neq y$ infoseeking(topic_1) \mathcal{I} infoseeking(topic_2), with $topic_1 \neq topic_2$ negotiating(topic_1) \mathcal{I} infoseeking(topic_2) whatever $topic_1$ and $topic_2$ are the therma \mathcal{I} contains the nuller in Table 2.
- the theory \mathcal{T} contains the rules in Table 2;
- If the VSA is benevolent, then the priority is defined such that: infoseeking(user)Pinfoseeking(product) and infoseeking(product)Pnegotiating(product).
 If the VSA is aggressive, then the priority is defined such that: negotiating(product)Pinfoseeking(product) and infoseeking(product)Pinfoseeking(user).

```
r_{11}: infoseeking(user)
                                           \leftarrow send(question(is(user, allergic))),
                                              \sim is(user, allergic), is(product, quilt), \sim receive(x)
r_{12}: infoseeking(user)
                                           \leftarrow send(question(is(user, sweat))),
                                              \sim is(user, sweat), is(product, quilt), \sim receive(x)
r<sub>21</sub>: infoseeking(product)
                                           \leftarrow send(question(is(product,quilt))),
                                              \texttt{search}(\texttt{user},\texttt{quilt}), \sim \texttt{is}(\texttt{product},\texttt{quilt})
r<sub>22</sub>: infoseeking(product)
                                           \gets \texttt{send}(\texttt{question}(\texttt{is}(\texttt{product},\texttt{nonallergenic}))),
                                              \sim is(product, nonallergenic), \sim receive(x)
r<sub>23</sub>: infoseeking(product)
                                           \leftarrow send(question(is(product, organic))),
                                              \sim is(product, organic), \sim receive(x)]
r<sub>24</sub>: infoseeking(product)
                                           \leftarrow send(question(dimension(product, x, y))),
                                              \sim \texttt{dimension}(\texttt{product}, x, y), \sim \texttt{receive}(z)
r_{25} : infoseeking(product)
                                           \leftarrow send(question(budget(product, x))),
                                              \sim budget(product, x), \sim receive(y)
r<sub>26</sub>: infoseeking(product)
                                           \leftarrow send(assert(is(x, y)), receive(question(is(x, y))), is(x, y))
r_{27}: infoseeking(product)
                                          \leftarrow send(assert(\negis(x, y)), receive(question(is(x, y))), \negis(x, y)
                                          \leftarrow send(unknow(is(x, y)), receive(question(is(x, y))), \sim is(x, y)
r_{28}: infoseeking(product)
r<sub>29</sub> : negotiating(product)
                                           \leftarrow send(introduce(product)), budget(product, y)
r_{31}: budget(product, 350)
                                           \leftarrow is(product, nonallergenic),
                                              is(product, organic), dimension(product, 200, 200)
r_{32}: is(product, nonallergenic) \leftarrow is(user, allergic)
r<sub>33</sub>: is(product, organic)
                                           \leftarrow \texttt{is}(\texttt{user},\texttt{sweat})
```

Table 2: The rules of a Virtual Seller Agent (VSA)

Our formalization allows to capture the incomplete representation of a decision problem with assumable beliefs. It provides a knowledge base on top of which arguments are built in order to reach decisions. We adopt here a tree-like structure for arguments.

Definition 4 (Argument). Let $DF = \langle DL, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework. An argument \bar{a} deducing the conclusion $c \in DL$ (denoted $conc(\bar{a})$) supported by a set of assumptions A in Asm (denoted $asm(\bar{a})$) is a tree where the root is c and each node is a sentence of DL. For each node :

- if the node is a leaf, then it is either an assumption in A or T^5 ;
- if the node is not a leaf and it is $\alpha \in D\mathcal{L}$, then there is an inference rule $\alpha \leftarrow \alpha_1, \ldots, \alpha_n$ in \mathcal{T} and,
 - either n = 0 and \top is its only child,
 - or n > 0 and the node has n children, $\alpha_1, \ldots, \alpha_n$.

The sentences of \bar{a} (denoted sent(\bar{a})) is the set of literals of \mathcal{DL} in the bodies/heads of the rules including the assumptions of \bar{a} . We write $\bar{a} : \mathbf{A} \vdash \alpha$ to denote an argument \bar{a} such that $conc(\bar{a}) = \alpha$ and $asm(\bar{a}) = \mathbf{A}$. The set of arguments built upon DF is denoted by $\mathcal{A}(DF)$.

 $^{^{5}}$ \top denotes the unconditionally true statement.

Arguments are built by reasoning backwards. Additionally, arguments interact with one another, and consequently, we reach to define the following attack relation.

Definition 5 (Attack relation). Let $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework, and $\bar{a}, \bar{b} \in \mathcal{A}(DF)$ be two arguments. \bar{a} attacks \bar{b} iff sent $(\bar{a}) \mathcal{I}$ sent (\bar{b}) .

This relation encompasses both the direct (often called *rebuttal*) attack due to the incompatibility of the conclusions, and the indirect (often called *under-mining*) attack, (i.e., directed to a "subconclusion").

Since the goals promoted by arguments have different priorities, the arguments interact with one another. For this purpose, we define the strength relation between concurrent arguments. Arguments are *concurrent* if their conclusions are identical or incompatible.

Definition 6 (Strength relation). Let $DF = \langle \mathcal{DL}, \mathcal{A}sm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework and $\bar{a}_1, \bar{a}_2 \in \mathcal{A}(DF)$ be two arguments which are concurrent. \bar{a}_1 is stronger than \bar{a}_2 (denoted $\bar{a}_1 \mathcal{P} \bar{a}_2$) iff $\operatorname{conc}(\bar{a}_1) = g_1 \in \mathcal{G}$, $\operatorname{conc}(\bar{a}_2) = g_2 \in \mathcal{G}$ and $g_1 \mathcal{P} g_2$.

Due to the definition of \mathcal{P} over \mathcal{T} , the relation \mathcal{P} is transitive, irreflexive and asymmetric over $\mathcal{A}(DF)$.

The attack relation and the strength relation can be combined to adopt Dung's calculus of opposition as in [20]. We distinguish between one argument attacking another, and that attack succeeding due to the strength of arguments.

Definition 7 (Defeat relation). Let $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework and \bar{a} and \bar{b} be two structured arguments. \bar{a} defeats \bar{b} iff:

- 1. \bar{a} attacks \bar{b} ;
- 2. and it is not the case that $\bar{bP}\bar{a}$.

Similarly, we say that a set S of structured arguments defeats a structured argument \bar{a} if \bar{a} is defeated by one argument in S.

Let us consider this example:

Example 1 (Defeat relation). Let us consider the situation after the second move in the dialogue represented in Fig. 1.

The arguments \bar{a} concludes infoseeking(user) since the VSA can ask to the user if he is allergic, (i.e. question(is(user, allergic))), the VSA is not yet aware about it, (i.e. ~ is(user, allergic)), the user is looking for a quilt,(i.e. is(product, quilt)), and the user did not query the VSA, (i.e. ~ receive(x)). The argument \bar{b} concludes infoseeking(product) since the VSA can ask to the user if the product must be nonallergenic,

(i.e. send(question(is(product, nonallergenic)))), the VSA is not yet aware about it (i.e. \sim is(product, nonallergenic)) and the user did not query the VSA (\sim receive(x)). While \bar{a} is built upon r_{11} , \bar{b} is built upon r_{22} . Since these

arguments suppose different decisions, they attack each others. If the VSA is benevolent, it is not the case that $infoseeking(product)\mathcal{P}infoseeking(user)$ and so \bar{a} defeats \bar{b} . If the VSA is aggressive, it is not the case that $infoseeking(user)\mathcal{P}infoseeking(product)$ and so \bar{a} defeats \bar{b} .

In our argumentation-based approach for dialogue strategy, arguments motivate decisions and they can also be defeated by other arguments. Formally, our argumentation framework (AF for short) is defined as follows.

Definition 8 (AF). Let $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework. The argumentation framework for decision making built upon DF is a pair $AF = \langle \mathcal{A}(DF), \text{ defeats } \rangle$ where $\mathcal{A}(DF)$ is the finite set of arguments built upon DF as defined in Definition 8, and defeats $\subseteq \mathcal{A}(DF) \times \mathcal{A}(DF)$ is the binary relation over $\mathcal{A}(DF)$ as defined in Definition 7.

We adapt Dung's extension-based semantics in order to analyze whenever a set of arguments can be considered as subjectively justified with respect to the agent's priority.

Definition 9 (Semantics). Let $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework and $AF = \langle \mathcal{A}(DF), \text{ defeats } \rangle$ be our argumentation framework for decision making. For $S \subseteq \mathcal{A}(DF)$ a set of arguments, we say that:

- S is conflict-free iff $\forall \bar{a}, \bar{b} \in S$ it is not the case that \bar{a} defeats \bar{b} ;
- S is admissible iff S is conflict-free and S defeats every argument \bar{a} such that \bar{a} defeats some argument in S;

Here, we only consider admissibility but other Dung's extension-based semantics [21] can easily be adapted.

Formally, given an argument \bar{a} , let

 $dec(\bar{a}) = \{ D(a) \in asm(\bar{a}) \mid D \text{ is a decision predicate} \}$

be the set of decisions supported by the argument \bar{a} .

The decisions are *suggested* to reach a goal if they are supported by an argument concluding this goal and this argument is a member of an admissible set of arguments.

Definition 10 (Credulous decisions). Let $DF = \langle D\mathcal{L}, Asm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $D \subseteq \mathcal{D}$ be a set of decisions. The decisions D credulously argue for g iff there exists an argument \bar{a} in an admissible set of arguments such that $conc(\bar{a}) = g$ and $dec(\bar{a}) = D$. We denote $val_c(D)$ the set of goals in \mathcal{G} for which the set of decisions D credulously argues.

It is worth noticing here that the decisions that credulously argue for a goal cannot contain mutual exclusive alternatives for the same decision predicate. This is due to the fact that an admissible set of arguments is conflict-free.

If we consider the arguments \bar{a} and \bar{b} supporting the decisions D(a) and D(b) respectively where a and b are mutually exclusive alternatives, we have $D(a) \mathcal{I} D(b)$ and $D(a) \mathcal{I} D(b)$ and so, either \bar{a} defeats \bar{b} or \bar{b} defeats \bar{a} or both of them depending on the strength of these arguments.

Proposition 1 (Mutual exclusive alternatives). Let

 $DF = \langle \mathcal{DL}, \mathcal{A}sm, \mathcal{I}, \mathcal{T}, \mathcal{P} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $AF = \langle \mathcal{A}(DF), \text{ defeats } \rangle$ be the argumentation framework for decision making built upon DF. If S be an admissible set of arguments such that, for some $\bar{a} \in S$, $g = \operatorname{conc}(\bar{a})$ and $D(a) \in \operatorname{asm}(\bar{a})$, then $D(b) \in \operatorname{asm}(\bar{a})$ iff a = b.

However, it is worth highlighting here the fact that mutual exclusive decisions can be suggested for the same goal through different admissible set of arguments. This case reflects the credulous nature of our semantics.

Definition 11 (Skeptical decisions). Let $DF = \langle D\mathcal{L}, Psm, \mathcal{I}, \mathcal{T}, \mathcal{P}, \mathcal{RV} \rangle$ be a decision framework, $g \in \mathcal{G}$ be a goal and $D \subseteq D$ be a set of decisions. The decisions D skeptically argue for g iff for all admissible set of arguments Ssuch that for some arguments \bar{a} in S conc $(\bar{a}) = g$, then $dec(\bar{a}) = D$. We denote $val_s(D)$ the set of goals in \mathcal{G} for which the set of decisions D skeptically argues.

Due to the uncertainties, some decisions satisfy goals for sure if they skeptically argue for them, or some decisions can possibly satisfy goals if they credulously argue for them. While the first case is required for convincing a risk-averse agent, the second case is enough to convince a risk-taking agent. Since some ultimate choices amongst various justified sets of alternatives are not always possible, we will consider in this paper only risk-taking agents.

Since agents can consider multiple objectives which may not be fulfilled all together by a set of non-conflicting decisions, high-ranked goals must be preferred to low-ranked goals.

Definition 12 (Preferences). Let $DF = \langle DL, Asm, I, T, P, RV \rangle$ be a decision framework. We consider G, G' two set of goals in G and D, D' two set of decisions in D. G is preferred to G (denoted GPG') iff

1. $G \supseteq G'$, and

2. $\forall g \in \mathbf{G} \setminus \mathbf{G}'$ there is no $g' \in \mathbf{G}'$ such that $g' \mathcal{P} g$.

D is preferred to D' (denoted DPD') iff $val_c(D)Pval_c(D')$.

Formally, let

$$\begin{split} \mathcal{A}\mathcal{D} &= \{ \mathtt{D} \mid \mathtt{D} \subseteq \mathcal{D} \text{ such that } \forall \mathtt{D}' \subseteq \mathcal{D} \text{ it is not the case that } \mathtt{val}_c(\mathtt{D}') \ \mathcal{P} \ \mathtt{val}_c(\mathtt{D}) \} \\ \text{be the decisions which can be accepted by the agent. Additionally, let} \\ \mathcal{A}\mathcal{G} &= \{ \mathtt{G} \mid \mathtt{G} \subseteq \mathcal{G} \text{ such that } \mathtt{G} = \mathtt{val}_c(\mathtt{D}) \} \end{split}$$

be the goals which can be reached by the agent.

Let us consider now the VSA's decision problem after the second move.

Example 2 (Semantics). The argument \bar{a} (respectively \bar{b}) (described in Example 1), concludes infoseeking(user) (respectively infoseeking(product)). Actually, the decisions {send(question(is(user, allergic)} credulously argue for infoseeking(user) and the decisions

 $\{\texttt{send}(\texttt{question}(\texttt{is}(\texttt{product},\texttt{nonallergenic})\} \text{ credulously argue for }$

infoseeking(product). If the VSA is benevolent, then

{send(question(is(user,allergic)} is an acceptable set of decisions. If the VSA is aggressive, then {send(question(is(product,nonallergenic)} is an acceptable set of decisions.

8 Related Works

Amgoud & Prade in [22] are presenting a general and abstract argumentation framework for multi criteria decision making. This framework captures the mental states (goals, beliefs and preferences) of the decision makers. Therefore, in their framework the arguments are prescribing actions to reach goals if theses actions are feasible while certain circumstances are true. These arguments that eventually conflict - are balanced according to their strengths. The argumentation framework we proposed earlier in this paper is conforming with this approach while being more specific and concrete.

The argumentation-based decision making process envisaged in [22] is divided into different steps where the arguments are successively constructed, weighted, confronted and evaluated. However, the computations we proposed earlier in this paper go through the construction of arguments, the construction of counterarguments, the evaluation of the generated arguments and the relaxation of preferences for making concessions. It is also worth noticing here that: a) the model we propose is unique in making it finally possible to concede, b) Our argumentation-based decision process suggest some decisions even if low-ranked goals cannot be reached.

Bench-Capon & Prakken formalize in [23] defeasible argumentation for practical reasoning. As in [22], they select the best course of actions by confronting and evaluating arguments. Bench-Capon & Prakken focus on the abductive nature of practical reasoning which is directly modelled within in our framework.

Kakas & Moraits propose in [24] an argumentation-based framework for decision making of autonomous agents. For this purpose, the knowledge of the agent is split and localized in different modules representing different capabilities. Whereas [24] is committed to one argumentation semantics, we can still deploy our framework for a number of such semantics by relying on assumptionbased argumentation.

Finally, to the best of our knowledge, few implementation of argumentation over actions exist. CaSAPI⁶ [25] and DeLP⁷ [26] are restricted to the theoretical reasoning. GORGIAS⁸ [27] implements an argumentation-based framework to support the decision making of an agent within a modular architecture.

9 Conclusions

In this paper, we have presented a dialogue management system that applies argumentation for generating and evaluating utterances. The agent start the conversation with the prior task which can consist of the need identification, the product selection or the negotiation depending on its strategy. During the dialogue, a proactive agent can query the user. Additionally, it can introduce a

⁶ http://www.doc.ic.ac.uk/~dg00/casapi.html

⁷ http://lidia.cs.uns.edu.ar/DeLP

⁸ http://www.cs.ucy.ac.cy/~nkd/gorgias/

product to sell and justify this choice depending on the information collected in the previous steps.

In order for us to implement an agent's reasoning method we are considering MARGO⁹ (A Multiattribute ARGumentation framework for Opinion explanation), which is an argumentation-based mechanism for decision-making [7]. We are currently rewritting MARGO in Java so that issues related to improving its performance, (i.e., the response time), and its scalability, (i.e., the number of rules which can be managed), are better tackled. This work is required to provide an industrial application rather than a research prototype. Besides, we need to interface this argumentation-based engine with the CSO Artificial Solutions' Language Processor in order to build conversational agents which are proactive in different selling situations.

Although the negotiation dialogue model we proposed allows single-sellings through the exchange of proposals and counter-proposals. However, we are currently working on an extension that will address cross-selling and up-selling.

Acknowledgements

This work is supported by the Ubiquitous Virtual Seller (VVU) project that was initiated by the Competitivity Institute on Trading Industries (PICOM).

References

- Hof, R., Green, H., Himmelstein, L.: Now it's YOUR WEB. BusinessWeek (October 1998) 68–75
- Poong, Y., Zaman, K.U., Talha, M.: E-commerce today and tomorrow: a truly generalized and active framework for the definition of electronic commerce. In: Proc. of the 8th international conference on Electronic commerce (ICEC), Fredericton, New Brunswick, Canada, ACM (2006) 553–557
- Palopoli, L., Rosaci, D., Ursino, D.: Agents' roles in B2C e-commerce. AI Communications 19(2) (2006) 95–126
- Wooldridge, M., Jennings, N.: Intelligent agents: Theory and practice. Knowledge Engineering Review 10 (1995) 115–52
- 5. Isbister, K., Doyle, P.: Design and evaluation of embodied conversational agents: A proposed taxonomy. In: Proc. of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), Budapest, Hungary (2002)
- Rist, T., Andr, E., Baldes, S., Gebhard, P., Klesen, M., Rist, P., Schmitt, M.: A review of the development of embodied presentation agents and their application fields. In: Life-Like Characters Tools, Affective Functions, and Applications, Springer (2003) 377–404
- Morge, M., Mancarella, P.: The hedgehog and the fox. An argumentation-based decision support system. In: Proc. of the Fourth International Workshop on Argumentation in Multi-Agent Systems (ArgMAS). (2007) 55–68
- 8. Morge, M., Mancarella, P.: Assumption-based argumentation for the minimal concession strategy. In: Proc. of the 6th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS), Budapest, Hungary (2009)

⁹ http://margo.sourceforge.net

- Roberts, F., Gülsdorff, B.: Techniques of dialogue simulation. In: Proc of the 7th International Conference on Intelligent Virtual Agents. Volume 4722 of Lecture Note in Computer Science., Paris, France, Springer (2007) 420–421
- George Ferguson, J.A.: Mixed-initiative systems for collaborative problem solving. AI Magazine 28(2) (2007) 23–32
- 11. Rich, C., Sidner, C.L., Lesh, N.: COLLAGEN applying collaborative discourse theory to human-computer interaction. AI Magazine **22**(4) (2001) 15–25
- Sadek, D.: Artemis Rational Dialogue Agent Technology: An Overview. In: Multi-Agent Programming. Springer-Verlag (2005) 217–225
- Grosz, B.J., Sidner, C.L.: Plans for Discourse. In: Intentions and Plans in Communication and Discourse. Cohen, Morgan and Pollack edn. MIT press (1990)
- Rao, A.S., Georgeff, M.P.: Modeling rational agents within a BDI-architecture. In: Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR), Morgan Kaufmann publishers Inc.: San Mateo, CA, USA (1991) 473–484
- C, F.T.: FIPA ACL Communicative Act Library Specification. Component, Foundation for Intelligent Physical Agents (6-12 2002) http://fipa.org/specs/fipa00037/.
- Breiter, P.: La communication orale coopérative : contribution à la modélisation et à la mise en œuvre d'un agent rationnel dialoguant. PhD thesis, Université de Paris Nord (1992)
- 17. Hamblin, C.L.: Fallacies. Methuen (1970)
- 18. Walton, D., Krabbe, E.: Commitment in Dialogue. SUNY Press (1995)
- Prakken, H.: Formal systems for persuasion dialogue. The Knowledge Engineering Review 21 (2006) 163–188
- Amgoud, L., Cayrol, C.: On the acceptability of arguments in preference-based argumentation. In: Proc. of UAI, Madison, Wisconsin, USA., Morgan Kaufmann (1998) 1–7
- Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. 77(2) (1995) 321–357
- Amgoud, L., Prade, H.: Using arguments for making and explaining decisions. Artificial Intelligence Journal 173(3-4) (2009) 413–436
- Bench-Capon, T., Prakken, H.: Justifying actions by accruing arguments. In: Proc. of the 1st International Conference on Computational Models of Argument, IOS Press (2006) 247–258
- 24. Kakas, A., Moraitis, P.: Argumentative-based decision-making for autonomous agents. In: Proc. of AAMAS, ACM Press (2003) 883–890
- Gartner, D., Toni, F.: CaSAPI: a system for credulous and sceptical argumentation. In: Proc. of ArgNMR. (2007) 80–95
- García, A.J., Simari, G.R.: Defeasible logic programming: an argumentative approach. Theory and Practice of Logic Programming 4(2) (2004) 95–138
- 27. Demetriou, N., Kakas, A.C.: Argumentation with abduction. In: Proc. of the 4th Panhellenic Symposium on Logic. (2003)