Modeling a collaborative task with social commitments

Jean-Baptiste Louvet, Guillaume Dubuisson Duplessis, Nathalie Chaignaud, Laurent Vercouter, Jean-Philippe Kotowicz

21st International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

Modeling a collaborative task with social commitments

Jean-Baptiste Louvet[a,*], Guillaume Dubuisson Duplessis[b], Nathalie Chaignaud[a], Laurent Vercouter[a], Jean-Philippe Kotowicz[a]

[a]*Normandie Univ, INSA Rouen, UNIHAVRE, UNIROUEN, LITIS, 76000 Rouen, France*
[b]*Sorbonne Universités, UPMC Univ Paris 06, CNRS, ISIR, 75005 Paris, France*

Abstract

Our goal is to design software agents able to collaborate with a user on a document retrieval task. To this end, we studied a corpus of human-human collaborative document retrieval task involving a user and an expert. Starting with a scenario built from the analysis of this corpus, we adapt it for a human-machine collaborative task. We propose a model based on social commitments to link the task itself (collaborative document retrieval) and the interaction with the user that our assistant agent has to manage. Then, we specify some steps of the scenario with our model. The notion of triggers in our model implements the deliberative process of the assistant agent.
.

Keywords: human-machine interaction, collaborative document retrieval, social commitments

1. Introduction

Document retrieval (DR), which takes place in a closed database indexing pre-selected documents from reliable information resources, is a complex task for non expert users. To find relevant documents, interfaces allowing formulating more specific queries are hardly used because an expertise about the domain terminology is needed. It may require an external assistance to carry out this task according to the users information need. Thus, we propose to design software agents able to collaborate with a user on a document retrieval task.

To this end, we adopt a cognitive approach by studying a corpus of human-human (h-h) collaborative document retrieval task in the quality-controlled health portal CISMeF (www.cismef.org) [1], which involves a user and an expert. In previous work [2,3], extraction of dialogue patterns from the corpus has been done with their formalization into dialogue games [4], which can be fruitfully exploited during the dialogue management process [3]. This formalization uses the notion of *social commitments* introduced by Singh [5].

In this article, we are interested in linking the task itself (collaborative DR) and the interaction with the user that our assistant agent has to manage. We show that the formalism [3] used to model the dialogue games through social commitments can be enhanced to describe the task. Our model makes the link between a high level structure (the

* Corresponding author.
E-mail address: jeanbaptiste.louvet@insa-rouen.fr

task) and low level interaction (dialogue games). Starting with a scenario built from the analysis of the corpus of h-h interaction, we adapt it for a human-machine (h-m) interaction. Then, we specify each step of this scenario in terms of social commitments.

This article consists of 5 parts: Section 2 gives a short state of the art on dialogue models. Section 3 describes the model we used to specify a collaborative task. Section 4 presents the scenario modeling the h-h collaborative document retrieval process and a discussion on its transposition in a h-m context. In Section 5, some steps of this scenario are detailed in terms of commitments. Finally, Section 6 gives some conclusions and future work.

2. Related work on reactive/deliberative dialogue model

To model dialogue, plan-based approaches and conventional approaches are often viewed as opposite, although some researchers argue that they are complementary[6,7,3]: Communication processes are joint actions between participants that require coordination. Nevertheless, coordination must stand on conventions reflected by interaction patterns. Thus, dialogue can be considered as a shared and dynamic activity that requires both high-level deliberative reasoning processes and low-level reactive responses.

Dubuisson Duplessis[3] proposes to use a hybrid reactive/deliberative architecture where a theory of joint actions can be a "semantics" to the interaction patterns described as dialogue games. These dialogue games are modeled through the notions of social commitment and commitment store described below.

2.1. Social Commitments

Social commitments are commitments that bind a speaker to a community[5]. They are public (unlike mental states such as belief, desire, intention), and are stored in a commitment store. Our formalization classically distinguishes a propositional commitment from an action commitment.

Propositional commitment. A propositional commitment involves that an emitter (x) commits itself at the present on a proposition towards a receiver (y). Such a commitment is written C(x, y, p, s), meaning "x is committed towards y on the proposition p" is in state s. We only consider propositions describing present, which leads us to consider only two states for a propositional commitment: a propositional commitment is initially inactive (**Ina**). After its creation, it enters the state created (**Crt**). A created commitment can be canceled by its emitter. In this case it goes back in an inactive state.

Action commitment. An action commitment involves that an emitter (x) commits itself at the present on the happening of an action in the future, towards a receiver (y). Such a commitment is written C(x, y, α, s), meaning "x is committed towards y on the happening of the action α" is in state s. An action commitment is initially inactive (**Ina**). In this state, it can be created. The creation attempt can fail (**Fal**) or succeed (**Crt**). An action commitment in **Crt** state is active. An active commitment can be violated, leading it to the **Vio** state. It corresponds to a situation in which the satisfaction conditions of the content of the commitment can not be fulfilled anymore. An active commitment can be fulfilled, leading it to the **Ful** state. An action commitment is satisfied if its content has been completed.

In order to simplify the writing of the commitments, as in our case the interaction is between two interlocutors, we omit the receiver of the commitments. Consequently, a propositional commitment will be written C(x, p, s) and an action commitment will be written C(x, α, s).

2.2. Conversational gameboard

The *conversational gameboard* describes the state of the dialogue between the interlocutors at a given time. The conversational gameboard describes the public part of the dialogic context supposed *strictly shared*. T_i stands for the conversational gameboard at a time i (the current time). In the framework of this article, we use a simple theory of instants where "<" is the relationship of precedence. The occurrence of an *external event* increments the time and makes the table evolve. An *external event* can be dialogic (e.g. an event of enunciation of a dialog act) or extra-dialogic (e.g. an event like light_on showing the occurrence of the action of turning the light on).

The conversational gameboard includes a *commitment store*, which is a partially ordered set of commitments. It is possible to query the gameboard on the belonging (or non-belonging) of a commitment. This is formalized in equation 1a for belonging and 1b for non-belonging (c being a commitment).

$$\mathrm{T}_i \vDash c, \text{ true if } c \in \mathrm{T}_i, \text{ false otherwise} \tag{1a}$$

$$\mathrm{T}_i \nvDash c, \text{ equivalent to } \neg(\mathrm{T}_i \vDash c) \tag{1b}$$

2.3. Dialogue games

A dialogue game is a conventional bounded joint activity between an initiator and a partner. Rules of the dialogue game specify the expected moves for each participant, which are supposed to play their roles by making moves according to the current stage of the game. This activity is temporarily activated during the dialogue for a specific goal.

A dialogue game is a couple ⟨type, subject⟩, where *type* belongs to the set of existing dialogue games and *subject* is the goal of the game in the language of the subject of the game. We usually write a game under the form type(subject). A game is defined with social commitments. It's a quintuplet characterized for the initiator and the partner by[2,3] *entry conditions* describing the conditions the conversational gameboard must fulfill to enter the game, *termination conditions*, separated into two categories: *Success conditions* and *failure conditions*, *rules* expressed in terms of dialogic commitments, specifying the expected sequencing of expected or forbidden acts, and *effects* specifying the contextualized effects of dialogical actions in terms of generation of extra-dialogic commitments (i.e. related to the task).

A sub-dialogue game is a *child* dialogue game played in an opened *parent* game. The emitter of the sub-game can be different from the one of the parent game. Conditions for playing a sub-dialogue game can be hard to specify[4].

Dialogical action commitment. A dialogical action commitment is an action commitment contextualized in a dialogue game. It means that in a dialogue game, a participant is committed to produce dialogical actions conventionally expected relatively to an opened dialogue game. For example, in the context of the *offer* dialogue game, if x plays offer(x, α), the dialogical action commitment C(y, acceptOffer(y, α)|declineOffer(y, α), **Crt**) will be created, showing that the receiver of the dialogical action can accept or decline the offer.

3. Model to specify a collaborative task

This section describes the model we use to specify the task using commitments, conversational gameboard and dialogue games. First of all, we consider that a task can be split into subtasks that we call steps. Each step of the task is described by a table (named *step table*) divided in three parts: The **name of the step**, the **access conditions** to this step and a list of **expected behaviors** of each participant to the dialogue. Expected behaviors are alternatives and can be played in any order. An expected behavior is a description of:

- A conventionally **expected dialogue game**, with its emitter and content (action or proposition);
- The possible **outputs** of this dialogue game;
- **Trigger** (optional) that is conditions that must be fulfilled to play the expected game.

To define the trigger φ that emits the predicate E, we use the notation 2a, φ being a formula. To express that the conversational gameboard T_i fulfills the conditions for a trigger to emit E, we use the notation 2b.

$$\mathrm{E} : \mathrm{T}_i \mapsto \varphi \tag{2a}$$

$$\mathrm{T}_i \vdash \mathrm{E} \tag{2b}$$

Prior to play an expected game, the emitter must respect the entry conditions of this game. To shorten the writing of a step table, we do not repeat these conditions.

The step table is a complete description of what can conventionally happen. A generic example of step table is given in Table 1. Instances of this this model for our specific task can be found in Tables 3 and 4. It describes the access conditions and the expected dialogue games of the step and the modifications they bring to the conversational gameboard T_i. In our example, access conditions are that T_i contains C(z, $\{\alpha, p\}$, s) and triggers the predicate E_0 we give an example of expected behavior with DG(z, $\{\beta_1, p_1\}$) as dialogue game and $T_{i+1} \vDash$ C(z, $\{\beta_1, p_1\}$, s) as a modification to the conversational gameboard brought by DG. For games played by a software agent, the emission of a predicate (e.g. $T_i \vdash E_1$ for the first dialogue game of our example) has to be triggered in order to play the expected dialogue game. For games played by a human, there's no trigger row in the table as the decision to play a dialogue game only depends on his own cognitive process. Some expected dialogue games can be played several times, noted with $*$. This symbol is propagated in the output to show the commitments that can be generated several times. This is shown with the first dialogue game of our example. Sub-dialogue games (like SDG(z', $\{\gamma, p'_2\}$) in the Table 1) that can be played are indicated under their parent game (played by any of the participants), headed by a "↳".

This model gives a clear definition of what is conventionally expected from each participant in one step of the task. It is also possible to see triggers as defining the intentions of the agent. As a matter of fact, most of the agent's decisions are done thanks to the triggers, and the agent's behavior can be adapted by modifying these triggers.

Table 1. Generic step table. DG is a dialogue game and SDG is a sub-dialogue game. α, β_k, γ are actions, p, p_k and p_k' are propositions, DA1 and DA2 are dialogical actions and E_k are predicates. z and z' stand for the participants to the interaction.s, s_1 and s_2 are states.

⟨**Name of the step**⟩		
Access		$T_i \vDash$ C(z, $\{\alpha, p\}$, s) $T_i \vdash E_0$
Expected behaviors	**Expected game**	DG(z, $\{\beta_1, p_1\}$)*
	Trigger	$T_i \vdash E_1$ (only if z is a software agent)
	Output	$T_{i+1} \vDash$ C(z, $\{\beta_1, p_1\}$, s)*
	Expected game	DG(–, $\{\beta_2, p_2\}$) ↳ SDG(z, $\{\gamma, p'_2\}$)
	Trigger	$T_i \vdash E_2$ (only if z is a software agent)
	Output	$T_{i+1} \vDash$ C(–, $\{\beta_2, p_2\}$, **Crt**) ↳ $T_{i+1} \vDash$ C(z, $\{\gamma, p'_2\}$, s)
	Expected game	DG(z, δ)
	Trigger	$T_i \vdash E_3$ (only if z is a software agent)
	Output	$\vee$	DA1(z', δ) $\Rightarrow T_j \vDash$ C(z', δ, s_1) DA2(z', δ) $\Rightarrow T_j \vDash$ C(z', δ, s_2)

We use a specific syntax for dialogue games implying dialogical action commitments. To reduce the size of the tables, we directly map the expected dialogical action commitments to the expected dialogical action of the dialogue game. For example, the third part of Table 1 shows the writing for the dialogue game DG(z, δ): Playing the dialogue game DG(z, δ) creates the dialogical action commitment C(z', DA1(z', δ)|DA2(z', δ), **Crt**) (DA1 and DA2 being dialogical actions), playing the dialogical action DA1(z', δ) creates the commitment C(z', δ, s_1) and playing the dialogical action DA2(z', δ) creates the commitment C(z', δ, s_2).

4. Analysis of the h-h collaborative document retrieval process

4.1. Information Retrieval

This section introduces some models of information retrieval (IR) processed by an isolated person and in a collaborative framework. These models can be applied to DR.

IR is generally considered as a problem solving process[8] implying a *searcher* having an *identified information need*. The problem is then to fulfill this lack of information. Once the information need is specified, the searcher

chooses a *plan* he will execute during the search itself. He evaluates the results found to possibly repeat the whole process. IR is considered as an *iterative process* that can be split into a series of steps[9,10,11]: (i) information need identification, (ii) query specification (information need formulation and expression in the search engine, etc.), (iii) query launch, (iv) results evaluation, (v) if needed, query reformulation and repetition of the cycle until obtaining satisfying results or abandoning the search.

The standard model is limited by two aspects. On the one hand, the *information need* of this process is seen as *static*. On the other hand, the searcher refines repeatedly his query until finding a set of documents fitting his *initial* information need. Some studies showed that, on the contrary, the information need is not static and that the goal is not to determine a unique query returning a set of documents matching with the information need[12,13]. Bates proposes the model of "berrypicking"[12] which lays the emphasis on two points. The first one is that the information need of the searcher evolves thanks to the resources found during the search. Encountered information can lead the search in a new and *unforeseen* direction. The second one is that the information need is not satisfied by a *unique* set of documents obtained at the end of the search, but by a selection of resources collected *all along* the process. To sum up, the IR process is *opportunistic* and its *progression* influences the final result.

4.2. Study of the h-h collaborative document retrieval process

To understand the collaborative aspect of the DR process of a user assisted by a human expert, we carried out a study on a h-h interaction. This study is based on the analysis of the corpus collected during the Cogni-CISMeF project[14,15].

4.2.1. The Cogni-CISMeF corpus

The corpus consists in assistance dialogues about the DR task between an expert and a user in a co-presence situation. The user expresses his information need. The expert has access to the CISMeF portal and has to lead the search cooperating with the user. The CISMeF portal has a graphical user interface and a query language enabling to decompose a query into MeSH ("Medical Subject Headings") lexicon elements. The CISMeF terminology contains keywords, qualifiers (symptoms, treatments…), meta-terms (medical specialties) and resources types (databases, periodicals, images…). The system also allows for extended queries, although many users are not comfortable with them.

The experiment was carried out with 21 participants (e.g., researchers, students, secretaries of the laboratory) submitting a query to one of the two CISMeF experts (researchers of the project who learned to use the CISMeF terminology). The corpus includes the transcript of the 21 dialogues (12 for the first expert and 9 for the second) and contains around 37 000 words.

Table 2 presents a translated extract of a dialogue (explained in Section 4.2.2).

A1	[…] Perhaps we can try to widen the search in our case if we consider the words we used
B2	We didn't use that much already
A3	Ummm, forget it then
B4	Why remove / we can remove "analysis"
A5	So let's remove "analysis"
B6	And "diagnostic"
A7	Yes
[…]	
A8	[…] I am almost tempted to put diagnostic anyway because / because we will see what it yields
B9	Yes normally it's a diagnostic / ok / let's try like this
A10	We will try like this otherwise we will remove extra things to have some / so I launch the search again with the "cancerology" thematic access, the CISMeF keyword "colon" and the qualifier "diagnostic" without specifying the type of searched resources

Table 2. Translated extract of a dialogue from the corpus (VD06). A is the expert and B the searcher.

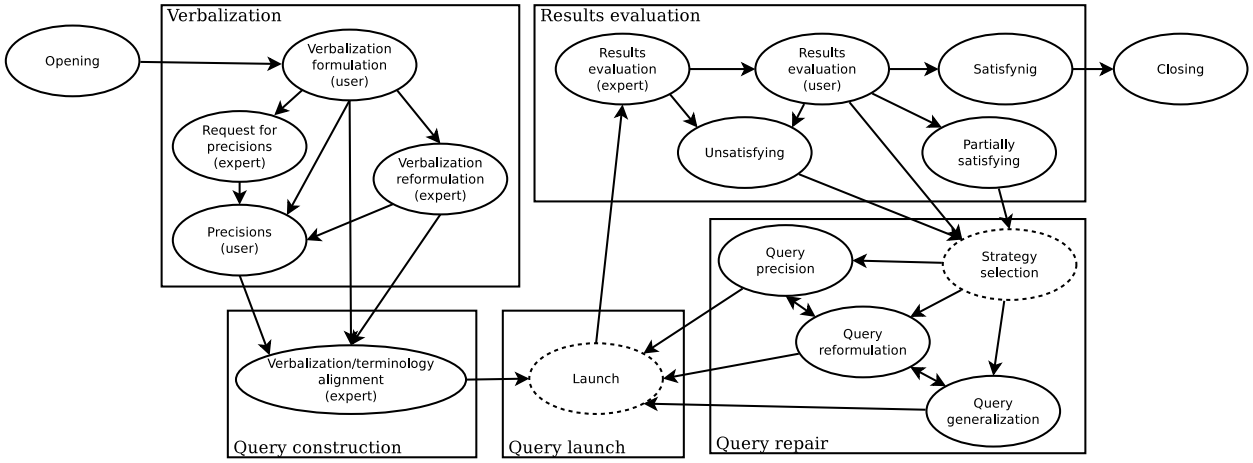

Figure 1. Scenario presenting the phases (squares) and the steps (ellipses) in a collaborative document retrieval task.

4.2.2. The collaborative document retrieval scenario

The analysis of the corpus enabled us to identify and characterize the different phases of the dialogues of the Cogni-CISMeF corpus playing a role in the task progress. Five phases were distinguished:

- Verbalization: It is the establishment of the search subject between both participants. It always starts by a request formulation from the user and can be followed by spontaneous precision. The expert can then start the query construction if he considers that the verbalization contains enough keywords, ask precision if not or try to reformulate the user's verbalization;
- Query construction: It is the alignment of the terms of the user's verbalization with the CISMeF terminology in order to fill in the query form;
- Query launch: It is the execution of the current query by the expert. This phase is often implicit;
- Results evaluation: The expert evaluates the results of the query. If they are not satisfying, he decides to directly repair the query. Otherwise he presents them to the user. If the latter finds them satisfying, the goal is reached and the search is over; If he finds them partially satisfying (not adapted to his profile, or not related totally to the information need) or not satisfying, the query must be repaired. If the results are rejected by the user, it is also possible to abandon the search;
- Query repair: The expert and the user try to use tactics to modify the query while respecting the information need. Three tactics were observed: Precision (to refine the query), reformulation (using synonyms for example) and generalization (to simplify the query). However, these tactics are not mutually exclusive: It is possible to combine precision or generalization with reformulation.

In addition to these phases, an opening and a closing phases were observed. The opening phase is optional and consists simply in greetings (information demand about the user's name, age...). At last, the closing phase may give ideas for a new search.

The analysis of this corpus showed that the DR task fulfilled by the participants is iterative, opportunistic, strategic and interactive [16,17]. The iterative aspect of this process is illustrated by the systematic repetition of the pattern launch/evaluation/repair. On top of that, we remarked that it is clearly lead by the expert.

The dialogue in Table 2 is an example of a query repair showing the iterative, opportunistic, strategic and interactive aspects. The expert suggests to widen (generalization) the query (utterance A1). The partners elaborate jointly a plan to modify the query. In this case, it is mainly the user who suggests the moves to carry out (utterances B4 and B6) and the expert agrees (utterances A5 and A7). Then, the expert suggests to add (precision) the qualifier "diagnostic" (utterance A8). The user accepts and suggests the plan execution (utterance B9). The plan execution is accepted and done by the expert (utterance A10), who eventually launches the query.

The scenario presented in Figure 1 synthesizes the phases (squares) split into several steps (ellipses) and the possible runs. The dashed ellipses correspond to actions that can be carried out implicitly by the participants of the interaction.

4.3. Discussion

It is possible to take inspiration from the h-h interaction to the h-m interaction. Thus, the presented scenario in the context of the h-h interaction (Figure 1) can be transposed to a h-m context. However, it has to be adapted in order to fit the constrains of the h-m interaction.

The h-m interaction framework changes the collaboration situation as far as it gives to the user the ability to lead the search and to modify the query, without requiring the system's agreement. It is an important change which gives to the user new privileges and more control over the interaction. It implies a restriction of the software assistant's permissions when compared to the human expert. As a matter of fact, the system can take initiatives to modify the query by suggesting modifications that will be applied only if they are accepted by the user. However, this inversion of the query modification rights allows each participant to take part in the interaction: As far as the opportunistic aspect are concerned, the user and the system can participate to the interaction at any moment.

Despite the lack of cognitive abilities that has a human expert, the software agent has some edges that can be beneficial for a collaborative DR task. The system can access online dictionaries of synonyms, hyponyms, hypernyms and "see also" links, allowing it to find terms related to the user's verbalization. For the reformulation, the system can make use of the lexical resources on the query terms [18]. In the context of CISMeF, Soualmia [19] offers tools to correct, precise and enrich queries. On top of that, the assistant agent can store the previous DR sessions and take advantage of them (by linking information needs, queries and documents) to find terms related to the current search [20]. It also has the ability to launch queries "in background" (i.e. without notifying the user), beforehand suggesting any query modification or launch. It makes possible, before suggesting a modification to the user, to check if it brings interesting results.

5. Application to the Cogni-CISMeF project

We described a h-h collaboration for DR process from which we want to draw in a h-m framework. The model described in Section 3 can be used for a h-m collaboration. As a matter of fact, this model makes possible to express the different characteristics of a h-m collaborative DR:

- Iterative: It is possible to give a circular sequencing to step tables using their access conditions (to represent the launch/evaluation/repair loop, for example);
- Opportunistic: Depending on the current state of the conversational gameboard, the most relevant dialogue games can be played;
- Interactive: Each participant can take part to the interaction at any moment;
- Strategic: It is possible to describe dialogue games combinations in order to reach given states of the conversational gameboard.

An interesting aspect of our model is that the assistant agent can behave according different cognition levels. This can be done thanks to the triggers that capture the deliberative process of the agent. The agent's reasoning can be very reactive with simple rules or, on the contrary, it can entail a more high level decision process.

In this section, we describe the steps of the DR scenario (see Section 4) in terms of our model (see Section 3). Our goal is to show the expressiveness of our model applied to a h-m collaborative DR task drawn from an h-h one. Only some relevant step tables are presented to illustrate the interest of the model.

5.1. Verbalization precision step

This step takes place in the verbalization phase. The user has done a first verbalization of his information need but he has to give more details about it, so he is committed to perform the action of verbalization precision ($T_i \models C(x,$ preciseVerbalization, **Crt**) present in the "Access" part).

In this situation, it is conventionally expected from the user to add verbalization expressions e to his verbalization with an "inform" dialogue game (inform(x, verbalizationExpression(e))). This action can be repeated as often as the user needs (expressed by the $*$ following the dialogue game). The consequences of this dialogue game on the conversational gameboard are: (i) Every time the dialogue game is played, a new commitment on a verbalization expression is created (**Crt**) and (ii) The commitment on the action of verbalization precision is fulfilled (**Ful**). The $*$ in the output is propagated for the creation of the commitment on a verbalization expression but not on the action fulfillment commitment (it is performed only the first time the dialogue game is played).

It is also expected that the user tells the agent when his precision is finished. This can be performed by playing the dialogue game inform(x, verbalizationComplete) that creates the commitment C(x, verbalizationComplete, **Crt**) in the conversational gameboard. The model of this step is shown in Table 3.

Table 3. Verbalization precision step, $i < j$.

Verbalization precision		
Access	$T_i \models$ C(x, preciseVerbalization, **Crt**)	
Expected game	inform(x, verbalizationExpression(e))*	
Output	$\wedge$	$T_j \models$ C(x, verbalizationExpression(e), **Crt**)* $T_j \models$ C(x, preciseVerbalization, **Ful**)
Expected game	inform(x, verbalizationComplete)	
Output	$T_j \models$ C(x, verbalizationComplete, **Crt**)	

5.2. *Query precision step*

This step takes place in the query repair phase. Query q were launched ($T_i \vdash$ lastQueryLaunched(q)) and its results evaluated by the user, who turned them down ($T_i \models$ C(x, ¬queryResultsSatisfying(q), **Crt**)). We place ourselves in the context where query is too general ($T_i \vdash$ queryTooGeneral(q)) and must be precised. This implies some conventionally expected behavior from each participants: The user is expected to add keywords to the query or request a query launch. The agent is expected to offer the user to add a keyword or to specify one, the goal being to reduce the number of results. It can inform the user that the keyword it is offering to specify the query is actually a specification of a keyword of the query. It can also offer the user to launch the query. All these expected behaviors are explained in the following of this section.

The addition of keywords to the query by the user corresponds to the inform(x, queryKeyWord(kw))*dialogue game. The user requests to the agent to launch the query with a request(x, launchQuery(q)) dialogue game. In this case, our collaborative agent will always accept the requests formulated by the user (that's why we only put the acceptRequest(y, launchQuery(q)) dialogical action commitment in the output row).

The agent offers to add keywords to the query with an offer(y, addKeyWord(kw))*and to specify the query with an offer(y, specifyKeyWord(kw, skw))*. In each case, the user can accept or decline the offer. If he accepts the offer, the conversational gameboard is updated with the commitments generated by execution of the action (keyword addition that generates C(y, queryKeyWord(kw), **Crt**)*or query specification that generates C(y, queryKeyWord(skw), **Crt**)*and C(y, queryKeyWord(kw), **Ina**)*, as skw replaces kw). If the user declines the offer, the action fails (the commitment on the action becomes **Fal**: C(y, addKeyWord(kw), **Fal**)*or C(y, specifyKeyWord(kw, skw), **Fal**)*).

In the case where the agent proposes to specify a keyword, it can play a sub-dialogue game (inform(y, isSpecification(kw, skw))) to inform the user that the new keyword is a specification of one keyword. The trigger of this sub-dialogue game is empty, because all the conditions needed to play this dialogue game (i.e. checking that skw is actually a specification of kw) are already reached in the parent dialogue game.

When the query has been specified enough, the agent can offer to launch it, if the user did not already decline the offer to launch it.

For illustrative purposes, we defined arbitrarily in Equations 3 the trigger predicates used in Table 4. These definitions depend on the expected behavior we want our agent to adopt. Trigger 3a makes sure that the query q is the last launched. Trigger 3b points that the query q is too general (i.e. the results are too numerous). The predicate queryResultsNb(q) gives the number of results returned by the search engine when launching query q. Trigger 3c

Table 4. Query precision step, $i < j$. x stands for the expert agent, y for the user and z either for the user or the expert agent

Query precision		
Access	$\wedge$	$T_i \models C(x, \neg\text{queryResultsSatisfying}(q), \textbf{Crt})$ $T_i \vdash \text{lastQueryLaunched}(q)$ $T_i \vdash \text{queryTooGeneral}(q)$
Expected game	$\text{inform}(x, \text{queryKeyWord}(kw))^*$	
Output	$T_j \models C(x, \text{queryKeyWord}(kw), \textbf{Crt})^*$	
Expected game	$\text{request}(x, \text{launchQuery}(q))$	
Output	$\text{acceptRequest}(y, \text{launchQuery}(q)) \Rightarrow T_j \models C(y, \text{launchQuery}(q), \textbf{Crt})$	
Expected game	$\text{offer}(y, \text{addKeyWord}(kw))^*$	
Trigger	$T_i \vdash \text{relevantForPrecision}(kw)$	
Output	$\vee$	$\text{acceptOffer}(x, \text{addKeyWord}(kw)) \Rightarrow T_j \models C(y, \text{queryKeyWord}(kw), \textbf{Crt})^*$ $\text{declineOffer}(x, \text{addKeyWord}(kw)) \Rightarrow T_j \models C(y, \text{addKeyWord}(kw), \textbf{Fal})^*$
Expected game	$\text{offer}(y, \text{specifyKeyWord}(kw, skw))^*$	
Trigger	$T_i \models C(z, \text{queryKeyWord}(kw), \textbf{Crt}) \wedge T_i \vdash \text{specification}(kw, skw)$	
Output	$\vee$	$\text{acceptOffer}(x, \text{specifyKeyWord}(kw, skw)) \Rightarrow \wedge$ { $T_j \models C(y, \text{queryKeyWord}(skw), \textbf{Crt})^*$; $T_j \models C(y, \text{queryKeyWord}(kw), \textbf{Ina})^*$ } $\text{declineOffer}(x, \text{specifyKeyWord}(kw, skw)) \Rightarrow T_j \models C(y, \text{specifyKeyWord}(kw, skw), \textbf{Fal})^*$
Expected game	$\text{offer}(y, \text{specifyKeyWord}(kw, skw))$ $\hookrightarrow \text{inform}(y, \text{isSpecification}(kw, skw))$	
Trigger	$\varnothing$	
Output	$T_j \models C(y, \text{isSpecification}(kw, s), \textbf{Crt})$	
Expected game	$\text{offer}(y, \text{launchQuery}(q))^*$	
Trigger	$T_j \not\models C(y, \text{launchQuery}(q), \textbf{Fal}) \wedge T_i \vdash \text{queryPreciseEnough}(q)$	
Output	$\vee$	$\text{acceptOffer}(x, \text{launchQuery}(q)) \Rightarrow T_j \models C(y, \text{launchQuery}(q), \textbf{Crt})$ $\text{declineOffer}(x, \text{launchQuery}(q)) \Rightarrow T_j \models C(y, \text{launchQuery}(q), \textbf{Fal})^*$

means that a keyword *kw*, related to the verbalization of the user (relatedToVerbalization(*kw*)), brings precision to the current query (currentQuery(q)). Trigger 3d gives a keyword (*skw*) more specific than a current one (*kw*). The predicate isMeSHHyponym(*kw*, *skw*) gives the information that *skw* is an hyponym (a more specific word) of *kw* according to the MeSH lexicon. Trigger 3e expresses that query q is precise enough to be proposed to the user for launch.

$$\text{lastQueryLaunched}(q) : T_i \mapsto \begin{array}{l} \exists n \leq i, \wedge \left| \begin{array}{l} T_n \models C(y, \text{queryLaunched}(q), \textbf{Crt}) \\ T_{n-1} \not\models C(y, \text{queryLaunched}(q), \textbf{Crt}) \end{array} \right. \\ \wedge \nexists m > n, \forall q' \neq q, \wedge \left| \begin{array}{l} T_m \models C(y, \text{queryLaunched}(q'), \textbf{Crt}) \\ T_{m-1} \not\models C(y, \text{queryLaunched}(q'), \textbf{Crt}) \end{array} \right. \end{array} \tag{3a}$$

$$\text{queryTooGeneral}(q) : T_i \mapsto \text{queryResultsNb}(q) \geq 75 \tag{3b}$$

$$\text{relevantForPrecision}(kw) : T_i \mapsto \begin{array}{l} \exists q, \text{currentQuery}(q), \text{queryResultsNb}(q) > \text{queryResultsNb}(q + kw), \\ \text{relatedToVerbalization}(kw) \end{array} \tag{3c}$$

$$\text{specification}(kw, skw) : T_i \mapsto \text{isMeSHHyponym}(kw, skw) \tag{3d}$$

$$\text{queryPreciseEnough}(q) : T_i \mapsto \exists q, \text{currentQuery}(q), \text{queryResultsNb}(q) < 75 \tag{3e}$$

6. Conclusion and future work

Our work is based on a cognitive study of a corpus of h-h collaborative DR task for the quality-controlled health portal CISMeF. Starting with a scenario of a collaborative DR task, built from the analysis of this corpus, we adapt it in a h-m context, where an assistant agent (software) helps a user in his task.

In this article, we described a model to specify a collaborative task in terms of social commitments. We shown how these social commitments link the task itself to the interaction with the user. This model has been applied to the CISMeF portal to specify each steps of the scenario in terms of social commitments. The notion of triggers in our model implements the deliberative process of the assistant agent. The agent's reasoning can be very reactive with simple rules or, on the contrary, it can entail a more high level decision process.

We currently work on these triggers to express the decision process of our assistant agent. As a matter of fact, it concerns the reasons of the agent both to enter a step of the scenario and to choose the dialogue game to play.

The validation of our system consists in evaluating the added value brought to CISMeF. The idea is to compare the queries made by the user with and without our assistant agent. This comparison would be made by calculating queries precision and recall.

Finally, to prove the genericity of our approach (the scenario, the model, the dialogue games, ...), we have started to investigate collaborative document retrieval on a transport law database (http://www.idit.asso.fr).

References

1. Darmoni, S., Leroy, J., Baudic, F., Douyere, M., Piot, J., Thirion, B.. CISMeF : a structured health resource guide. *Methods of Information in Medicine* 2000;**39**:30–35.
2. Dubuisson Duplessis, G., Chaignaud, N., Kotowicz, J.P., Pauchet, A., Pécuchet, J.P.. Empirical Specification of Dialogue Games for an Interactive Agent. In: *11th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS'13)*; vol. 7879 of *Lecture Notes in Computer Science*. Salamanca, Spain; 2013, p. 49–60.
3. Dubuisson Duplessis, G., Pauchet, A., Chaignaud, N., Kotowicz, J.P.. A Conventional Dialogue Model Based on Dialogue Patterns. *International Journal on Artificial Intelligence Tools* 2017;**26**(1):1–23.
4. Maudet, N.. *Modéliser l'aspect conventionnel des interactions langagières: la contribution des jeux de dialogue*. Thèse de doctorat en informatique; Université P. Sabatier; Toulouse; 2001.
5. Singh, M.P.. Social and psychological commitments in multiagent systems. In: *AAAI Fall Symposium on Knowledge and Action at Social and Organizational Levels*. 1991, p. 104–106.
6. Hulstijn, J.. Dialogue games are recipes for joint action. In: *Proceedings of the Forth Workshop on the Semantics and Pragmatics of Dialogue*. 2000, .
7. Yuan, T., Moore, D., Reed, C., Ravenscroft, A., Maudet, N.. Informal logic dialogue games in human-computer dialogue. *Knowledge Engineering Review* 2011;**26**(2):159–174.
8. Marchionini, G.. Information-seeking strategies of novices using a full-text electronic encyclopedia. *Journal of the American Society for Information Science* 1989;**40**(1):54.
9. Broder, A.. A taxonomy of Web search. *ACM SIGIR Forum* 2002;**36**(2):3–10.
10. Marchionini, G., White, R.. Find What You Need, Understand What You Find. *Int J Hum Comput Interaction* 2007;**23**(3):205–237.
11. Sutcliffe, A., Ennis, M.. Towards a cognitive theory of information retrieval. *Interacting with computers* 1998;**10**:321–351.
12. Bates, M.J.. The Design of Browsing and Berrypicking Techniques for the Online Search Interface. *Online Review* 1989;**13**(5):407–424.
13. O'Day, V.L., Jeffries, R.. Orienteering in an information landscape: how information seekers get from here to there. In: *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*. ACM; 1993, p. 438–445.
14. Loisel, A., Dubuisson Duplessis, G., Chaignaud, N., Kotowicz, J.P.. A conversational agent for information retrieval based on a study of human dialogues. In: *International Conference on Agent and Artificial Intelligence (ICAART)*. 2012, p. 312–317.
15. Loisel, A., Kotowicz, J.P., Chaignaud, N.. An Issue-Based Approach to Information Search Modelling: Analysis of a Human Dialog Corpus . In: *Conference on Text, Speech and Dialogue (TSD)*. 2008, p. 609–616.
16. Bates, M.J.. Information search tactics. *JASIS* 1979;**30**(4):205–214.
17. Bates, M.J.. Where should the person stop and the information search interface start? *Inf Process Manage* 1990;**26**(5):575–591.
18. Audeh, B.. Semantic query expansion for fuzzy proximity information retrieval model. In: *International Conference on Biomedical Ontology (ICBO 2012), KR-MED Series, Graz, Austria, July 21-25, 2012*. 2012, .
19. Soualmia, L.F., Dahamna, B., Thirion, B., Darmoni, S.J.. Strategies for Health Information Retrieval. In: *Ubiquity: Technologies for Better Health in Aging Societies - Proceedings of MIE2006,International Congress of the European Federation for Medical Informatics, Maastricht, The Netherlands*. 2006, p. 595–600.
20. Guedria, Z.E., Vercouter, L.. Customized Document Research by a Stigmergic Approach Using Agents and Artifacts. In: Rovatsos, M., Vouros, G.A., Julián, V., editors. *EUMAS/AT*; vol. 9571 of *Lecture Notes in Computer Science*. Springer; 2015, p. 50–64.