An Event-Based Dialogue Model and its Implementation in MultiDial2

Olivier Grisvard, Bertrand Gaiffe

AN EVENT-BASED DIALOGUE MODEL AND ITS IMPLEMENTATION IN MULTIDIAL2

Olivier Grisvard and Bertrand Gaiffe
LORIA-INRIA Lorraine and LORIA-CNRS
Campus Scientifique, B.P. 239, F-54506 Vandœuvre-lès-Nancy Cedex, France
Olivier.Grisvard@loria.fr and Bertrand.Gaiffe@loria.fr

ABSTRACT

In this paper we present and justify the design and implementation choices we have made in order to build our dialogue system **MultiDial2**. We propose an event-based representation that enables us to structure the dialogue data upon several levels but within a single representation space. We show how this provides us with the necessary flexibility for a proper management of the dialogue. We describe the implementation of our model which uses the theory of Mental Representations, a formalism developed for referential resolution purposes.

Keywords: dialogue structure management, referential resolution, events.

1. INTRODUCTION

In this paper, we present an event-based dialogue model we have designed in order to implement the dialogue management component of our dialogue platform **MultiDial2**. This dialogue system has been tested on a video recording and editing application, and all the dialogue examples throughout this paper are taken from this environment. Our model relies on a multiple level representation that operates in a single representation space. Levels are related through the natural link that exists between events and their participants. We argue that this offers the necessary flexibility for a proper dialogue management. The implementation of our model is done using the theory of Mental Representations, which provides us with the appropriate representation for concrete objects as well as more abstract ones such as events and state. We first comment the main existing approaches in what concerns dialogue structure management. We then describe our dialogue model, Mental Representation Theory and the implementation of our model in **MultiDial2**.

2. DIALOGUE MANAGEMENT

Many studies about dialogue have shown how the interpretation of utterances in a dialogue relies on the structuring of the dialogue context. Yet, the various existing approaches propose many different solutions for dialogue structure management. Therefore, the problem is that, although they may all prove useful for a particular aspect of dialogue management, the resulting structures are not necessarily compatible.

2.1 Various Structures

Models that deal with discourse/dialogue structuring can be separated into three categories:

- linguistic models, which focus mainly on reference resolution;
- more pragmatic approaches concerned with speech act sequences and their structuring on the basis of dialogue grammars;
- task or application oriented models, which deal with intentions and structure them in terms of plans.

Formal semantics of discourse such as Discourse Representation Theory (DRT) [7] and Segmented DRT [2], build the discourse structure on the basis of sets of discourse referents that form contexts more or less accessible to anaphora resolution. SDRT, as well as less formal approaches such as [6], [9] and [11] proposes that the appropriate discourse/dialogue structure is a tree in which antecedents to anaphoras are searched for in open contexts that form the right frontier of the tree.

The models of the second category, such as [10], [5] or extensions of SDRT to dialogue [3], focus on the functional aspect of utterances and define rules for speech act sequences. Such rules can be for example, that a question is followed by an answer, an order by its execution and the confirmation of it, etc. These approaches then build the dialogue structure on the basis of dialogue grammars, which link the sequences together to form sub-dialogues.

In the last group of approaches, dialogue is viewed in terms of goals or intentions and structured via planning means. Plans can be inferred on the basis of the task underlying the dialogue [1], [6], yielding a structure of action goals and enabling one to predict parts of the task on the basis of smaller sub-tasks. As speech acts may be considered as true actions, this may include utterance productions themselves [1]. Thus planning can also be applied to utterances and yield a structure of communicative goals [4], [8].

2.2 Useful But Potentially Conflicting Structures

Each of the structuring methods described above proves useful for a proper management of the dialogue, but they

are each focused on a particular aspect of dialogue management, concealing the others. Therefore, there is no reason that the various resulting structures necessarily match with each other. As example (1) shows, a sequence of utterances that form a relevant sub-dialogue from the speech act point of view may lead to the execution of a single action in the task or application model, something like `_loadSequence(_sequence05)` in this case.

(1) $\mathbf{U}_1$: Load a video-sequence.
$\mathbf{S}_1$: Which one?
$\mathbf{U}_2$: The one we recorded yesterday.

Conversely a single utterance can be a request for the execution of several actions. In example (2) for instance, the utterance may lead to `_createWindow()` followed by `_changeColor(_window01)`.

(2) $\mathbf{U}_1$: Create a red window.

As another example of discrepancy between the possible structures, it may be the case that the appropriate structure for anaphora resolution is not relevant in the application model. For instance, in examples (3) and (4) below, the antecedent to "them" in the third utterance is the group formed by the camera and the window and by the camera and the sequence respectively.

(3) $\mathbf{U}_1$: Switch on the first camera.
$\mathbf{U}_2$: Open the control window.
$\mathbf{U}_3$: Connect Them.

(4) $\mathbf{U}_1$: Load the third sequence.
$\mathbf{U}_2$: Open the control window.
$\mathbf{U}_3$: (?) Connect Them.

Therefore, in each case, the pronoun can only be solved in a context formed by the two first utterances. But with regards to the application model this grouping operation is relevant in example (3) only, since the connecting operation applies to a camera and a window but does not apply to a sequence and a window. As these examples show, it appears that the building of the dialogue structure may rely on different aspects of dialogue management. Therefore, we argue that an effective dialogue model must be conceived as to integrate the different structuring solutions into a single representation space.

3. THE DIALOGUE MODEL

Since the various data structures useful to dialogue management do not necessarily match, we distribute the data upon several levels having each its own structure but using the same representation and operating within a single representation space. As we will see, this allows us to connect these different levels with a simple mechanism.

3.1 Three Levels of Representation

In our model, the dialogue data is distributed upon three levels of representation, the utterance level, the semantic level and the application level. These three levels of representation form the context in which subsequent utterances are interpreted.

The utterance level contains information specific to utterances as opposed to sentences, that is, their illocutionary force or speech act type, their temporal anchoring and ordering, and information that identifies their speaker, hearer and propositional content. Speech act structuring and dialogue grammars typically apply at this level of representation.

The semantic level contains purely semantic information, that is, a representation of the logical form of the propositional content of utterances. The propositional content is composed of discourse referents that can be representations of quite concrete objects such as cameras and windows or more abstract ones such as events and states. The structuring for reference resolution takes place at this level. For instance, having a separate level for discourse referents enables us to apply at this the solutions for referential treatment proposed by formal semantics such as DRT [7].

The application level contains the effective referents in the application of the discourse referents at the semantic level. Therefore, the application model or task description apply at this level. The need for this third level comes from the fact that the context formed by the set of discourse referents at the semantic level is not sufficient to fully interpret the utterances and execute the requested actions. Indeed, as examples such as (2) and (4) illustrate, discourse referents at the semantic level do not necessarily match with effective referents in the application.

Obviously, the three levels of representation described above must be connected. The utterance level must be linked to the semantic level in a way that represents the fact that a sub-space of the semantic level forms the propositional content of a given utterance. The semantic level must be linked to the application level in order to associate discourse referents with their effective referents in the application. Therefore, the problem is to find a unified representation for the three levels.

3.2 An Event-Based Model

In the command dialogue framework, we deal mostly with orders that lead to the execution of some action or sequence of actions on some object or set of objects. Discourse referents and effective referents corresponding to the execution of an action or a sequence of actions are events. Utterance productions may also be represented as events, and this matches our request for a unified representation. Therefore, it has seemed natural to us to chose events as the main component of our dialogue model.

Events prove to be an appropriate representation for the utterance level. Indeed, events enable us to represent all the information we need at this level in a simple fashion. The illocutionary force becomes the category of the speech act event. Temporal ordering of utterances is

simply based on the usual temporal ordering of events. Finally, the speaker, hearer and propositional content become participants of the speech act event. The speaker is the agent, the hearer the addressee and the propositional content an argument of the event. In the case of command utterances, the propositional content denotes another event, whose effective referent is the execution of some action or sequence of actions.

As we have mentioned above, at the semantic level, we represent the logical form of the propositional content of utterances. In the command dialogue framework, the propositional content can be represented as events and states. Indeed, an order that asks for the execution of some action is a request for an event that transforms the current state of some object or set of objects into a new state. States thus form the glue between events and vice versa. The referents of objects are simply participants of such events or states. Eventualities can be used to represent the semantic level as well as the application level. Linking this two levels means enriching the logical form of the propositional content, that is, finding the effective referents of the events, states and objects that form the semantic level. Since, as we have shown, there is a discrepancy between the structures of these two sets of referents, we need a connection mechanism. This is done via event sums.

3.3 Event Sums and Dialogue Structure

An interesting property of events is that they can be grouped together into larger events or conversely decomposed into sub-events. Therefore, the modeling of the dialogue structure at each level of representation simply relies on event summation principles.

At the utterance level, the event structure is built on the basis of a dialogue grammar or justified by principles such as those proposed by extensions of SDRT to dialogue. In [3] for example, the authors define the discourse relation *Question-Answer-Pair* to group together a pair of utterances corresponding to a question and its answer. This relation holds if and only if the answer is a relevant answer to the question. In terms of events, this means building an event sum on the basis of the two events corresponding to the two utterances. Such an event sum constitutes the context in which the second utterance can be categorized as an answer to the first.

At the semantic level, we need to build event sums in order to solve such referential expressions as “them” in examples (3) and (4) above. In these examples, the antecedent to the pronoun is one of the participants of a larger event corresponding to the sum of the two events that form the propositional contents of the two first utterances. Conversely, in example (5) below, we have to decompose the event corresponding to the propositional content of the first utterance into three sub-events in order to solve the referential expression “the first one”, since its antecedent is a participant of the first sub-event.

(5) $\mathbf{U}_1$: Open three windows.
$\mathbf{U}_2$: Iconify the first one.

Finally, at the application level, the task description or application model offer constraints to build event sums or decompose events into sub-events in order to assign effective referents to the events present at the semantic level. In example (3) for instance, there will be an effective referent to the event sum that enables us to solve the plural pronoun, justified by the fact that the connection is valid in the application model. This will not be the case in example (4), and the event sum will not have an effective referent.

Whatever the level, the structuring information provided by each knowledge source (dialogue grammar, task description, etc) is used to define summation principles. These principles are then used to justify or prohibit the building of an event sum or the decomposition of an event. For instance, an event sum or a sub-event will be relevant if and only if its category is more specific than the most general category “event” and the roles of its participant are clearly defined.

4. IMPLEMENTATION OF THE MODEL

In order to implement our model, we needed a formalism that enabled us to represent events, states, usual objects and the links between them within a single representation space. As we have mentioned above, speech acts are represented in that space as standard events and, for structuring purposes, the most important operation we had to implement was a grouping operation. We have chosen to use the theory of Mental Representations, a formalism that is being developed in our team for referential resolution purposes, precisely because it emphasizes on grouping operations.

4.1 Mental Representations

Mental Representations (MRs) are descriptions of abstract as well as physical objects. Each MR contains at least the category of the object it represents. Categories are hierarchically structured and determine in particular the possible subparts for the object. For instance, the “camera” category yields a partition that contains the sub-categories “on/off switch”, “zoom”, etc. An MR for an object categorized as a camera thus contains handles to its on/off switch and its zoom, which are themselves MRs. Events and states are a particular type of MR since they also contain handles to their participants: agent, patient, etc. MRs also contain a specific entry, which records the history of the object in terms of other MRs representing the events and states in which the object has been participating. Finally, each MR may contain a handle to the representation of the associated *physical* object in the application, typically a pointer. There are four operations on MRs.

1. **Creation**: typically, each referring expression in an utterance leads to the creation of a new MR.
2. **Merging**: this operation is mainly used when solving reference, the MR created on the basis of a referring expression is merged with the MR representing the effective object of the application.

3. **Grouping**: this operation yields a MR categorized as the plural for the first common category in the above hierarchy. For example, a group composed of a lion and a snail would be categorized as "animals". The only property of such plural groups is their partition. If the group may be categorized as a single object, it inherits other properties from its singular category.

4. **Extraction**: this operation consists in accessing a subpart. Two cases may appear, either the subpart is already represented as a MR, in which case the result is this precise MR, or it is not already represented and a new MR is created with the category associated with the handle.

The grouping operation has interesting properties when applied to events. It is always possible to build a plural group of events. This group, however, has no new property unless it gets categorized as a single event. In such a case, the new category provides access to the participants of the event. For instance, a speech act event such as **saying-that**(speaker:*A*, hearer:*B*, content:P_1) (*A* says to *B* that P_1) grouped together with **saying-that**(speaker:*B*, hearer:*A*, content:P_2) yields **saying-that**(participants:*A*+*B*, content:P_1+P_2). On the contrary, grouping an event such as **filming**(agent:*A*) with **opening**(agent:*B*, object:*C*) simply yields a MR with the very general category **event**(participants:*A*+*B*+*C*), which can be paraphrased by "something happened involving the group **objects**:*A*+*B*+*C*". Each inference rule dedicated to categorizing groups is written on the basis of the four operations mentioned above plus an operation that tests the category of a MR. For instance, the speech act example above uses the grouping rule:

```
saying-that-events:S
  X = S.extract("one");
  Y = S.extract("others");
  create(saying-that-event:Z) {
    Z.speaker = group(X.speaker, Y.speaker);
    Z.hearer = group(X.hearer, Y.hearer);
    Z.content = group(X.content, Y.content);
  }
  merge(S, Z);
```

When this rule is used, it triggers other grouping operations on the participants of *X* and *Y*.

4.2 MultiDial2

Mental Representations and our dialogue model have been implemented into the **MultiDial2** dialogue platform. **MultiDial2** is a generic oral dialogue system dedicated for now to command dialogue processing. It comprises a speech recognizer, a TAG analyzer and the dialogue component. This system can be connected to various controlling applications, the current test application being the video recording and editing environment. The dialogue component comprises a semantic analyzer, a referential solver and the dialogue manager.

5. CONCLUSION

In this paper we have advocated an event-based representation that integrates the context in which a dialogue occurs and the dialogue itself in a single representation space. In particular, speech acts are represented as events whose sole particularity is their category: saying-that, asking-if, etc. In this approach, the dialogue structure is not necessarily a tree, and different sub-structures may appear at different levels. For instance, even though two command utterances are not connected, the events they denote may compose a single event at the application level, typically a single action. As an extension of our model, we now investigate how deictics such as "now", "I", "you", etc, can be solved on the basis of speech act events.

6. REFERENCES

[1] Allen J. F. & Perrault C. R. (1980), Analyzing Intentions in Utterances. *Artificial Intelligence*, 15, pp. 143–178.

[2] Asher N. (1993), Reference to Abstract Objects in Discourse. Kluwer Academic Publishers.

[3] Asher N. & Lascarides A. (1998), Questions in Dialogue. *Linguistics and Philosophy*, 21, pp. 237–309.

[4] Cohen P. R. & Levesque H. J. (1990), Rational Interaction as the Basis for Communication. In: Cohen, Morgan & Pollack (eds.), *Intentions in Communication*. M.I.T. Press, pp. 221–255.

[5] Grau B., Sabah G. & Vilnat A. (1994), Pragmatique et dialogue homme-machine. *Technique et Science Informatique*, 13(1), pp. 9–30.

[6] Grosz B. J. & Sidner C. L. (1986), Attention, Intentions and the Structure of Discourse. *Computational Linguistics*, 12(3), pp. 175–204.

[7] Kamp H., & Reyle U. (1993), From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory. Kluwer Academic Publishers.

[8] Litman D. J. (1986), Linguistic Coherence: A Plan-Based Alternative. *Proceedings of the 24th meeting of the Association for Computational Linguistics*, pp. 215–223.

[9] Mann W. C. & Thompson S. A. (1987), Rhetorical Structure Theory: A Theory of Text Organization. Technical Report ISI/RS–87–190, Information Sciences Institute, University of Southern California.

[10] Moeschler J. (1989), Modélisation du dialogue : représentation de l'inférence argumentative. Hermès.

[11] Polanyi L. (1988), A Formal Model of the Structure of Discourse. *Journal of Pragmatics*, 12, pp. 601–638.