Command Dialogue Management

Olivier Grisvard, Bertrand Gaiffe

Command Dialogue Management

Olivier Grisvard and Bertrand Gaiffe
LORIA-INRIA Lorraine / LORIA-CNRS

Introduction

In this paper we focus on dialogue management in the context of an application controlling system. The discrepancy between the representations of the utterances on the one hand and the associated sequences of actions on the other hand leads us to propose an event-based representation for both. We then show how event summation principles provide us with means to build the appropriate dialogue structure for anaphora resolution. Finally we briefly describe the implementation in our dialogue platform **MultiDial2** of the model we propose.

From utterances to actions

An application controlling dialogue system aims at executing the actions expressed by the user. An utterance such as (1) may thus lead to `_closeWindow(_w1)` where `_w1` is a handle to an object and `_closeWindow` a function.

(1) Close the control window.

However, mapping each verb to a function, each noun to an object type and each adjective to some property of objects proves to be too simplistic for it implies among other things that the number of arguments of a given verb be exactly the same as the number of parameters of the associated function. If the user may refer to some particular window saying "the green window" and if each predicate in the language is mapped to one and only one action in the application, enabling the system to understand utterances such as (2) means defining numerous similar actions, such as `_createAGreenWindow`, `_createABlueWindow`, etc.

(2) Create a green window.

Therefore, a system such as ours must be able to handle cases where a single utterance yields several actions and, conversely, cases where several utterances lead to the execution of a single action, as in (3) below.

(3) **U**: Switch on a camera.
S : Which one?
U: The second one.

Action reference and events

Since predicates in the utterances do not necessarily match single actions in the application, we must be able to represent actions at the utterance level independently of the representation of actions at the application level. Yet the descriptions of both the *linguistic* and the *effective* actions must be integrated into a single representation space since they clearly are connected. We argue that a simple way to meet these constraints is to view actions as events. Indeed, action predicates denote events at the utterance level and the referents of these events are events corresponding to action executions at the application level. In command dialogues, we mainly deal with transition events, that is transformations of the current state of some object into a new state. Action reference resolution, which can be seen as connecting events at the utterance level to events at the application level, then requires reasoning in terms of plans to find the appropriate sequence of events to transform the current state of an object into the requested final state, or conversely, to map a sequence of events to a single event.

Objects as participants in events

An interesting property of events is that they offer means to structure the sets of objects one can refer to with pronouns. Typically, when an argument of an event is a group of objects, it may be globally referred to by means of a plural pronoun even though its individual components cannot be accessed by singular pronouns, as example (4) shows.

(4) **U**: Create a green window and a blue one.
U: Iconify them.
? **U**: Iconify it.

This idea is the basis of *topic* calculus in Segmented Discourse Representation Theory (Asher 1993). Schematically, if some condition holds, namely a *Narration* relation stands between two utterances that denote events, SDRT forecasts that only those discourse referents that occur in the second event and in the *topic* (an event computed on the basis of the two events) are accessible to pronouns. Although

we are not in that precise *Narration* case, the very same phenomenon occurs in examples such as (5).

(5) **U**: Create a green window.
U: Create a blue one.
U: Iconify them.

The pronoun "them" gets solved on the patient of an event computed on the basis of the first two utterances. Therefore, our main problem in what concerns pronoun resolution is to build grouping events when and only when it is justified.

Event sums and dialogue structure

It is always possible to group several events into a larger one. The problem is that the category of the resulting event may be so general that it does not yield any interesting information. More precisely, it may be relevant to view some event sums as single events and others as groups of events. Some cases, such as (5), are obvious. A creation event being distributive over its objects, creating two windows yields creating each of them. Thus conversely, the two individual creations may be gathered into a global one and, as we have shown, this allows the use of a plural pronoun in order to refer to the two windows. Not all event sums are that easy to compute however. In (6) for instance, although we still need a global event in order to solve the plural pronoun, no such property as in (5) is at hand.

(6) **U**: Switch on the first camera.
U: Open the control window.
U: Connect them.

(Grosz and Sidner 1986) have shown that the dialogue structure often matches the associated task structure. Although we cannot always build the entire task structure, since the system does not necessarily have means to infer the task the user has in mind, the task structure may still yield constraints on event summation. More precisely, generic sub-task descriptions enable us to categorize a group of events as a single one. This is what happens in cases such as (6), where the global event is built on the basis of the conceptual model of the task. The problem of dialogue structure is not peculiar to task-based dialogues. More generally, it seems that the appropriate structure for anaphora resolution is a tree in which antecedents are searched for in the right frontier of the tree, see *ibid.*, (Mann and Thompson 1987), (Polanyi 1988) or, as we have mentioned, SDRT (Asher 1993). We argue that event summation principles such as those we describe here offer the appropriate constraints for the building of the dialogue tree. Even if possible by means of event sums, building the whole tree may not be relevant. Nevertheless, event summation principles enable us to build relevant local sub-trees, either linguistically justified, as in example (5), or corresponding to recognizable sub-tasks, as in example (6).

Implementation of the model

As we explained, dialogue events do not necessarily match single actions in the task. However, we have to establish links between the utterance and application levels, in order for the system to act. Therefore, we have to group action events together and thus provide each dialogue event with a referent. The implementation of our system **MultiDial2** relies on descriptions of both linguistic and task-based objects. Each object description has a category and contains handles to its parts. For instance, an event sum yields referential access to its constituting sub-events. Event descriptions also contain handles to their participants on the basis of their various roles: agent, patient, etc. The main task for the object/event management module consists in building groups of events. An inferential module is then used to categorize as a single event, if possible, such groups of events. If it fails, the group is considered as non-relevant and forgotten.

Conclusion and future work

In this paper we have advocated an event-based dialogue representation and shown how linguistic and task-based constraints on event summation offer means to build the appropriate dialogue structure for anaphora resolution. As an extension of our model, we now investigate how events could be used to represent utterance productions as well, how to build the dialogue structure on the basis of such events, and how this would be useful to reference resolution, especially for what concerns deixis.

References

Asher N. (1993). *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers.

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

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.

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