Beyond Structured Dialogues: Incorporating Clark's Models of Language Usage

Peter A. Heeman and Michael Johnston
Computer Science and Engineering
Oregon Graduate Institute
PO Box 91000 Portland OR 97291
heeman@cse.ogi.edu

Abstract

In this paper, we discuss the use of Clark and Wilkes-Gibbs' model of collaboration (1986) and Clark and Schaefer's model of contributions (1989) in using structured models of dialogue. Rather than completely specifying the behavior of a dialogue system by a structured dialogue model, we advocate that the dialogue structure should only specify the behavior of the system with respect to high-level contributions. The realization of these contributions and the associated grounding that accompanies them would be handled by a general dialogue manager based on Clark's models.

1 FINITE STATE DIALOGUE MODELS

In building spoken dialogue systems, the designer needs to ensure that the system is able to properly understand what the user said. This is difficult because the system might misrecognize what the person said, or might not properly interpret it. One technique for overcoming this obstacle is to use a structured dialogue approach. This approach involves keeping initiative solely with the system, and letting it direct the user through a series of pre-ordained steps to accomplish the task. The steps are described as a finite state dialogue model. Each state has an associated system prompt, and a set of possible responses that the user can make. The determination of the next state is a function of both the current state and the user's response. Figure 1 gives a simple structured subdialogue that allows the user to specify information for finding train information, namely the destination city, the initial city, and the time that the train leaves. The advantage of structured dialogues is that the system prompts encourage the user to say something from a limited set of possible responses. This simplifies speech recognition and almost eliminates the need for natural language processing. It also simplifies the dialogue management component, which is completely specified by the specification of the transitions. This technology has proved very popular for building working systems, and the Center for Spoken Language Understanding at Oregon Graduate Institute even provides a toolkit for building such systems with this approach (Novick and Sutton, 1996; Sutton et al., 1996). Although such systems seem overly restrictive for the user, Walker *et al.* (1997) found that users preferred a system-initiative system over a mixed-initiative system, probably due to the former giving more reliable performance.

2 LIMITATIONS OF STRUCTURED DIALOGUE

When a designer specifies a dialogue by means of a finite state dialogue model, the designer must account for all dialogue behavior between the system and the user. As such, he must hand-code the grounding behavior in the dialogue: the process by which participants ensure what they have said is mutually understood (Clark and Schaefer, 1989). In a structured dialogue, this is often achieved by asking the user to verify the recognition and understanding results of the system after each piece of information

COOP'98 Workshop on Use of Clark's Models of Language for the design of Cooperative Systems, May 1998.

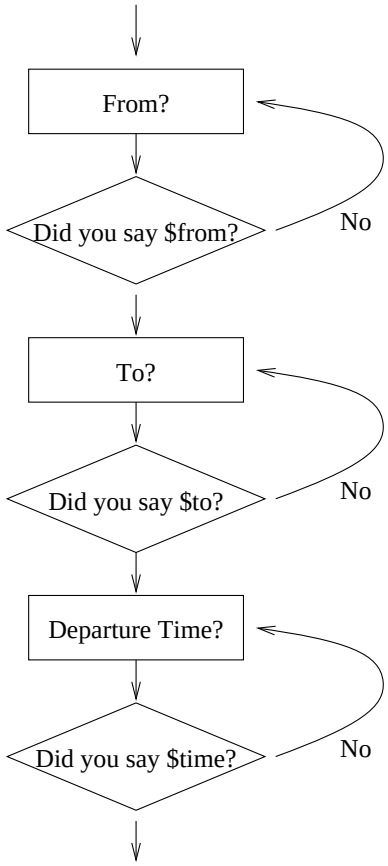

Figure 1: Structured dialogue model for train information

that the user gives. This is shown in Figure 1 by the three verification questions and the loops back to the original queries if user answers that there was a misunderstanding. Another approach to dealing with misunderstandings is to add special commands, such as "scratch that", which the user can utter if he detects that a misrecognition or misunderstanding occurred, which will undo the last operation, and "clear history", which will completely reset the system (Zue et al., 1994).

A second aspect that complicates the specification of a structured dialogue model is that there are many different ways that a speaker might want to break a contribution into installments. For the task of specifying criteria about a train, the user might want to specify the destination, source and time in a single utterance, or break it down into some combination of installments, such as first presenting the destination and source and then presenting the time. The way speakers break this down depends on a number of factors, such as "minimization of collaborative effort" (Clark and Wilkes-Gibbs, 1986). Allowing for all possible installment patterns for a contribution will add to the complexity of the dialogue structure, and hence this variability is typically not supported. Rather, the structured dialogue model prompts the user for a specific ordering of the information as shown in Figure 1.

3 BEYOND STRUCTURED DIALOGUES

We are currently extending the CSLU toolkit to go beyond structured dialogs by factoring out the contribution level from the structured dialogue model. Rather than the designer specify the grounding and contribution behavior, the designer will work at a higher level in specifying the dialogue behavior. The designer will specify the dialogue in terms of tasks or subproblems, each having an associated contribution. A general dialogue manager will take the specification of the contribution, or *taskbox*, and engage in the appropriate dialogue behavior to collaborate with the user to realize the contribution in terms of individual installments with the everything properly grounded. This will simplify the specification of

the structured dialogue model as well as make the dialogue system more user-friendly by giving the user some control of the dialogue and by incorporating more sophisticated grounding behavior then can be easily and consistently coded in a structured dialogue model, and more robust in dealing with misunderstandings.

3.1 Clark and Schaefer's Model of Contributions

The structured dialogue model will specify the dialogue behavior in terms of high level contributions while the dialogue manager will collaborate with the user in presenting and accepting these contributions. The model of Clark and Schaefer (1989) tells us that contributions consist of presentation and acceptance processes, and that the presentation can consist of multiple installments, which in term each have an acceptance and presentation process. Their model tells us that after each user presentation, we need to give evidence of our understanding. The amount of evidence should depend on the speech recognition results. If the speech recognition score of the best recognition result is close to the score of the next highest competitor (or a garbage hypothesis), the system should give stronger evidence of understanding, perhaps even paraphrasing the users response, or explicitly asking for a confirmation. In this case, the system should expect that the user might correct a misunderstanding or make a relevant next contribution. If misunderstanding is happening often, the system can adjust its strategy by asking more specific questions that have few possible response, such as yes/no questions. This will ease the speech recognition problem and hence allow more reliable recognition results. Such a strategy should be dictated by reasoning about minimization of collaborative effort.

3.2 Clark and Wilkes-Gibbs' Model of Collaboration

The model of Clark and Schaefer does not give us a complete picture. There model assumes that contributions are built monotonically with each installment. Their model does not address how one of the participants might want to change one of the previous installments nor the collaborative nature of making contributions. Returning to our train information example, assume that the user specified the destination in the first installment, the source in the second and the time in the third, he might then want to change the destination, with an utterance such as "actually, make that to Portland, not Seattle". An even better example comes when we extend what the conversants are doing in the taskbox to include the role of the system in verifying that there is such a train that meets the criteria that the user specified. Here, the user and system must collaborate to find a train that the system knows exists and which the user finds acceptable. Here, either participant should be able to suggest changes to the criteria.

Clark and Wilkes-Gibbs gave a descriptive model of how conversants can collaborate in making a referring expression. They found that after the initial referring expression was presented, the other participant would pass judgment on it, either *accepting* it, *rejecting* it, or *postponing* his decision. If it was rejected or the decision postponed, then one participant or the other would *refashion* the referring expression. This would take the form of either *expanding* it by adding further qualifications, or *replacing* the original expression with a new expression. The referring expression that results from this is then judged, and the process continues until the referring expression is acceptable enough to the participants.

From the above description, we can see that each step in the acceptance process has a referring expression associated with it. The judgment moves serve to judge the expression, while the refashioning moves serve to refashion it, resulting in the referring expression that is associated with the next turn. This view led us (Heeman and Hirst, 1995) to propose that the *current* referring expression is part of the state of the collaborative process, along with beliefs about its validity. Besides having an intention to achieve the goal of the collaborative activity, the participants, in order to co-ordinate their activity, have intentions to keep the state in their common ground. So, the judgment and refashioning moves serve to fulfill these intentions.

In other work (Heeman, 1993), we proposed that this view could be used for collaboration in general, where the referring expression is replaced by a plan that the conversants are collaborating upon. Both agents collaborate in specifying the plan by refashioning the current plan until it is acceptable. Note

that refashionings are contributions which are accepted as longs as it is understood, rather than whether it is an acceptable refashioning. Note that this view of collaboration differs from others proposed in the computational linguistics literature, where agents are assumed to monotonically build up the plan with no revisions allowed of accepted refashionings (Grosz and Sidner, 1990; Litman and Allan, 1990; Chu-Carroll and Carberry, 1994).

Returning to our train example, the plan that the system and user are collaborating upon is a plant that identifies a train that the system knows exists and that the user finds acceptable. Each additional criteria that the user adds to the plan is a contribution in itself and hence needs to be grounded. The contribution also is a refashioning of the current plan since it expands it. While the plan under specifies a unique train, the system should postpone judgment, and perhaps list the trains that do match, as part of the realization of the postponement. If the description over specifies, the system should reject the description, and perhaps refashion the plan so that it will match a train, such as "there is no train in the afternoon, but there is one that leaves at 11 a.m.".

4 DISCUSSION

In this paper, we proposed using Clark and Scheafer's model of contributions and Clark and Wilkes-Gibbs model of collaboration to extend structured dialogue models. The dialogue designer would then just need to specify the high level tasks, *taskboxes*, and use a general dialogue model to collaborate with the user in specifying the plan associated with the taskbox. This will not only simplify the design of the structured dialogues, it will also make the resulting system more robust to misunderstandings and able to collaborate with the user.

To show the feasibility of this approach, we have implemented a simple prototype with the CSLU toolkit. A simple dialogue manager was implemented through the Tcl interface of the toolkit. The dialogue manager implemented the functionality of the subdialogue given in Figure 1. It keeps track of the current plan in its discourse state and uses the discourse state to dynamically specify a relevant grammar for the speech recognizer. The system verifies the user's criteria and allows the user to correct misunderstandings.

References

Chu-Carroll, Jennifer and Sandra Carberry. 1994. A plan-based model for response generation in collaborative task-oriented dialogues. In *Proceedings of the National Conference on Artificial Intelligence (AAAI '94)*, pages 799–805.

Clark, Herbert H., editor. 1992. *Arenas of Language Use*. University of Chicago Press and CSLI.

Clark, Herbert H. and Edward F. Schaefer. 1989. Contributing to discourse. *Cognitive Science*, 13:259–294. Reprinted in (Clark, 1992), pages 144–175.

Clark, Herbert H. and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22:1–39. Reprinted in (Clark, 1992), pages 107–143 and reprinted in (Cohen, Morgan, and Pollack, 1990), pages 463–493.

Cohen, Philip R., Jerry Morgan, and Martha E. Pollack, editors. 1990. *Intentions in Communication*. SDF Benchmark Series, edited byPhilip R. Cohen and Jerry Morgan and Martha E. Pollack. MIT Press.

Grosz, Barbara J. and Candace L. Sidner. 1990. Plans for discourse. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, SDF Benchmark Series. MIT Press, pages 417–444.

Heeman, Peter A. 1993. Speech actions and mental states in task-oriented dialogues. In *Working Notes AAAI Spring Symposium on Reasoning about Mental States: Formal Theories & Applications*, pages 68–73, Stanford, March.

Heeman, Peter A. and Graeme Hirst. 1995. Collaborating on referring expressions. *Computational Linguistics*, 21(3):351–382. Also published as Revised TR 435, Computer Science Dept., U. Rochester, April 1995.

Litman, Diane J. and James F. Allan. 1990. Discourse processing and commonsense plans. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, SDF Benchmark Series. MIT Press, pages 365–388.

Novick, David G. and Stephen Sutton. 1996. Building on experience: Managing spoken interaction through library subdialogues. In *Proceedings of Twente Workshop on Language Technology 11*, The Netherlands, June.

Sutton, Stephen, David G. Novick, Ronald Cole, Pieter Vermeulen, Jacques de Villiers, Johan Schalkwyk, and Mark Fanty. 1996. Building 10,000 spoken-dialogue systems. In *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP-96)*, Philadelphia, October.

Walker, Marilyn A., Donald Hindle, Jeanne Frome, Giuseppe Di Fabbrizio, and Craig Mestel. 1997. Evaluating competing agent strategies for a voice email agent. In *Proceedings of the 5th European Conference on Speech Communication and Technology (Eurospeech)*, Rhodes, Greece.

Zue, V., S. Seneff, J. Polifroni, M. Phillips, C. Pao, D. Goodine, D. Goddeau, and J. Glass. 1994. PEGASUS: A spoken dialogue interface for on-line air travel planning. *Speech Communications*, 15:331–340.