Content planning and generation in continuous-speech spoken dialog systems*

Amanda J. Stent

Department of Computer Science
University of Rochester
Rochester, NY 14627
stent@cs.rochester.edu

December 12, 2000

1 Introduction

Researchers interested in constructing conversational agents that can interact naturally in relatively complex domains face a unique set of constraints. Generation must take place in real, or near-real, time. The language coverage must be extensive, and language use must be varied. A grammar-based approach can be both slow and awkward. On the other hand, it is difficult to provide the required language coverage using templates. In this paper we propose an architecture for generation that combines these two approaches, capitalizing on their strengths and minimizing their weaknesses. In the process, we attempt to answer the question, "How far can templates take us?"

2 The Situation

The TRIPS system is a multi-modal dialog system at the University of Rochester that provides a platform for research in different aspects of dialog and planning ([4]). Currently, generation in this system is done by the dialog manager and the generator. The dialog manager interprets user input and selects content for output. It passes this content to the generator as a set of role-based logical forms with associated speech acts. The generator decides which ones to produce and how to order them, and then passes them on to modality-specific generators.

In an "idealized" generation system, there would be separate components for planning intentions, semantic content, and form ([1]. Our language generator combines these last two components, and in some cases does the work of all three. It finds a rule or rules that match the speech act and content specification, and then selects from the set of meaning-equivalent strings associated with each rule. There is also a set of noun phrase rules for producing noun phrases to insert into utterances. For instance, if the logical form is for an acknowledgment, the generator may select from utterances such as "OK" and "Fine". For location question answers, it may have only one utterance, "There are NP" in which "NP" can be replaced by e.g. "five people at Calypso". For most simple utterances (acknowledgments, indications of lack of understanding or reference failures) and a limited set of more complex ones, this is sufficient. As the system begins

*This work was supported by ONR research grant N00014-95-1-1088, U.S. Air Force/Rome Labs research contract no. F30602-95-1-0025, NSF research grant no. IRI-9623665 and Columbia University/NSF research grant no. OPG: 1307

A	41	And so [breath] I think we need to send a we obviously need to send an ambulance to Marketplace.
	42	We should then send that ambulance to Highland.
B	43	Mm-hm.
A	44	Um so the problem is th- the six people at the airport.
	45	Um we can do the helicopter from the airport to Strong.
	46	Those a- are in fact the only two places that you can do that.
B	47	Right.
A	48	Um [breath] so here's the thing.
	49	We can we we can either uh
	50	I guess we have to decide how to break up this.
	51	We can make three trips with a helicopter a-
B	52	So I guess we should send one ambulance straight off to Marketplace right now right?

Figure 1: Dialog extract

to take more initiative and the domain becomes more complex (answering wh-questions, making statements, asking questions), this approach becomes more difficult to maintain.

The output of generation is a string or set of strings that are sent to the Truetalk speech synthesizer and displayed on the screen, and displays including maps, drawings on maps and charts. Currently, push-to-talk regulates the turn-taking, but we hope soon to move to continuous speech.

Our goal is to develop a conversational agent capable of interacting naturally in task-oriented multi-modal dialog situations. To give us an idea of the kinds of interaction required, we have collected a corpus of twenty mixed-initiative, task-oriented human-human dialogs in a complex domain. Our study of these dialogs informs our hypotheses about how to produce natural interactions.

3 Dialog

In the following discussion we will use the extract in figure 1, which comes from a dialog in our corpus, as a reference point.

It is important to distinguish between the plan for the dialog as a whole, and the plan for the current utterance or turn. The dialog plan is constructed as a byproduct of the agents' collaboration. It is impossible to construct the whole dialog plan at the beginning; planning must be incremental and take place in real-time. Re-planning may have to occur at any point, to deal with interruptions or new information from the world. By contrast, the plan for the current utterance can usually be specified in full (although even here some researchers prefer to interleave planning and execution [7]). Because the dialog plan is incomplete, however, the plan for any individual utterance will necessarily be made on the basis of incomplete information.

Because dialog is collaborative behavior, there are two kinds of intentions behind the production of individual utterances: task/domain-related intentions that contribute to the overall dialog plan; and intentions related to maintaining the collaboration (see figure 2).

Some utterances are related to the topic of the dialog, for instance contributing to the solution of a task. The intentions behind these may come from a task or domain model ([2, 5]), or from a model of rhetorical structure ([6]). The intentions behind utterance 45 probably come from the domain model, while utterance 44 provides **motivation** for utterance 45. These utterances also have semantic content in addition to the speech act, such as references to specific objects. Finally, the form of these utterances matter. For instance, the "then" in utterance 42 is crucial to identification of the **sequence** relation that holds between utterances 41 and 42.

Other utterances (e.g. turn-taking and grounding utterances) maintain the collaboration.

Type of dialog act	Source of intentions	Examples
turn-taking	maintains collaboration	"Um", "Wait a minute"
grounding	maintains collaboration	"Okay", "Well...."
primary acts	furthers task	"Send the A train."
secondary acts	support, coherence	"because it's faster."

Figure 2: The different types of dialog acts

These communicative actions are intentional, but are not part of the overall dialog plan. Often, to produce them one can simply select one of several conventional utterances that satisfy the intention . There is no need to plan semantic content or form for these utterances. For example, utterance 43 is an utterance that performs grounding only.

Looking at the example dialog, we can see that planning utterances does not consist simply of choosing individual dialog acts and then realizing them; multiple dialog acts can be performed by one utterance or a single dialog act may be realized over several utterances ([8]). For example, a *release-turn* act can be performed by performing an *info-request*. We hope that as we annotate our dialog corpus, we will gain insights into the complex interactions between different aspects of content planning and generation. For instance, how frequently do agents reuse surface forms? How do agents decide when they need not plan a grounding act? How do agents combine different dialog acts, and what surface forms do they use to signal these combinations? In what circumstances will agents generate utterances that contribute to rhetorical structure, and when will they limit how much they say?

4 Proposed Architecture

We propose to think of the three stages of generation (planning intentions, planning content and planning form) as three different dimensions along which planning[1] can occur, possibly simultaneously ([7, 3]). The planning of intentions generally consists of selecting intentions from different sources such as interpretation and the agent's internal agent model, and ensuring that none of the selected intentions conflict or are redundant. It usually takes place as part of dialog management. The planning of content is what is more usually referred to as strategic generation, and the planning of form is tactical generation.

If planning need occur along only one dimension, then templates can be effectively used. Grounding and turn-taking acts involve the planning of intentions only. The form can be selected from a set of conventional forms; to try to construct content and generate these surface forms using a grammar adds nothing to the result, and may limit the variability of language use. It can sometimes involve completely unnecessary processing (think of generating the surface form "I heard you" from the discourse act *acknowledge*). Also, turn-taking and grounding acts often begin a turn, so generating these acts quickly can give a conversational agent time to produce other acts that may involve more processing. Therefore, we can use templates to generate these acts.

If planning needs to occur along more than one dimension, it may be better to use a grammar. Otherwise, a template will probably be needed for each combination of, say, intention and content or content and form (see figure 3).

Those utterances that speakers produce to fulfill intentions arising directly from the domain or the task being solved (*primary intentions*) often have content that must be expressed. The form

[1]In this paper, *planning* is any kind of non-trivial processing.

Dimensions that involve planning	Type of dialog act	Grammar/ templates
I	turn-taking, grounding	templates
C		
F		
I, C	some primary intentions	grammar
I, F		
C, F		
I, C, F	some primary intentions, secondary intentions	grammar

Figure 3: Generating different types of dialog acts: dimensions along which planning may occur

may or may not be important. These utterances should be generated using a grammar, unless there is a very limited set of kinds of utterances that can be produced. There is nothing to gain from using templates because there is no way to "skip" stages of processing, and with a grammar greater language coverage can be obtained.

Other utterances are produced primarily to complete an argumentation act, for instance to provide justification for something (we will call these *secondary intentions*). Their production involves the planning of intentions, semantic content and surface form. In particular, the form may determine whether these acts are seen as coherent in the dialog. These utterances should also be generated using a grammar.

These four kinds of dialog acts account for only three of the seven possible combinations of dimensions along which planning may occur (see figure 3). We are unable to think of examples of utterances for the other four possibilities[2], so we have left them blank, but we believe the same reasoning could be used in these cases.

To summarize, we believe that templates are best used when it is possible to eliminate stages of processing (e.g. to go directly from intentions to form), and when speed is necessary. Otherwise, we think a grammar should be used for tactical generation, especially where broad language coverage is needed.

At this point, we may conclude that we have obtained a good architecture for generation for dialog. Intentions, represented as dialog acts ([8]), and associated content come to the generator from the agent's internal agent model (which can reason about the task, the domain, and rhetorical structure), or from the process of interpretation. Each kind of dialog act proceeds through a different path in the architecture. The generation of turn-taking and grounding acts happens quickly, via templates. There is the coverage of a grammar for producing the more "multi-dimensional" acts.

If a grammar that provides incremental output is used, the behavior of the agent will change. One might expect to see fewer utterances that perform only grounding at the start of turns. There will also probably be repairs; the modules producing incremental output would provide the repairs but, if there are pauses, turn-keeping utterances could be interleaved with the incremental output from the other modules.

Unfortunately, this architecture is a little too simple. As we said earlier, many utterances realize multiple dialog acts. For instance, questions can be both *info-request* and *release-turn* acts. We cannot just send the dialog acts through their respective paths without risking over-generation. We have a very preliminary solution for this. The generator maintains a set of sets of intention by content pairs, prioritized mostly by recency (we'll call this the *intention-set*). Each set is sent to all the generation modules. The output from each module is a a surface form and a set of intentions fulfilled by that form. A gate-keeper at the end removes intentions from the intention-set as they are fulfilled. It can also add sets of intentions to the memory, for instance to keep the turn or

[2]If the domain is simple enough that everything to be generated is an *inform*, that might be a case where only content, or maybe only content and form, need be planned. However, few domains are that simple.

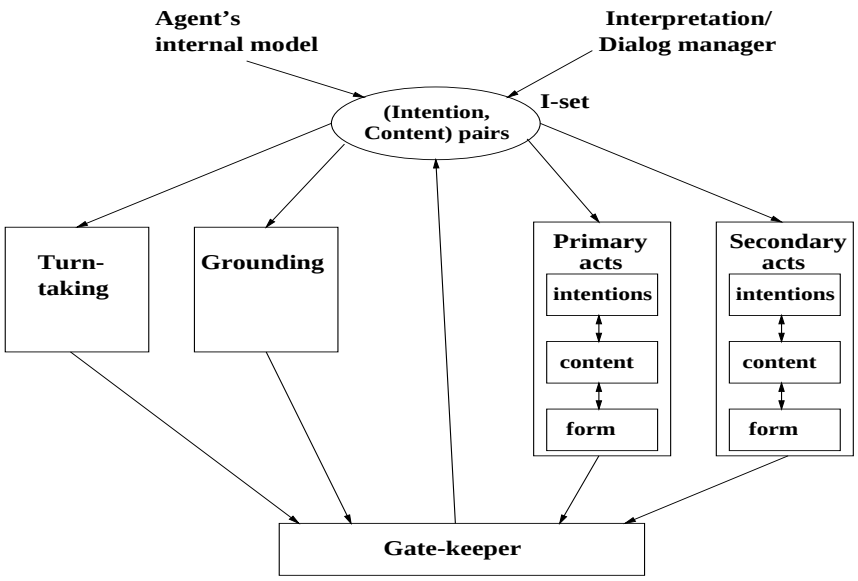

Figure 4: Proposed generator architecture

if the agent is interrupted. Finally, it can minimize over-generation by selecting which results to produce, if it gets simultaneous results that satisfy the same intentions.

For example, imagine a user has just made a statement to the agent. The agent wants to acknowledge part of the statement (grounding) and ask a question about another part. So the memory looks like:
{{*take-turn, acknowledge(Utt1), info-request(Content)*}}
(items with initial capital letters are variables).
This set gets passed to all modules. The turn-taking module returns "Uh" for *take-turn* and the grounding module returns "Okay" for *take-turn* and *acknowledge(Utt1).* The gate-keeper therefore removes *take-turn* and *acknowledge(Utt1)* and produces "Okay". If a pause of more than, say, half-a-second ensues, the gate-keeper might add the set {*keep-turn*} to the memory which will feed it to the various modules. However, happily the gate-keeper quickly receives a result for *info-request(Content)* which it produces, removing that intention (and therefore the whole set) from the memory.

This architecture is given in figure 4. We have not yet implemented it, but we believe it may be possible to combine the turn-taking and grounding modules into one, and the primary and secondary act modules into one. This would especially help with reasoning about argumentation acts.

Real-time generation is very important in the context of dialog. We have observed that if users have to wait for a response, they may begin to hyper-articulate, resort to saying only one word per utterance, or otherwise begin to use unnatural interactions. In a continuous-speech system in particular, this architecture would be most effective if the interpretation component could produce incremental output, thus allowing the system to, for instance, provide appropriate and timely back-channels.

Of course, agents have many intentions when interacting, among them social intentions such as politeness, and other "global" intentions such as efficiency. We have not discussed how these types of intentions could be used in this architecture. We believe that they could simply be included as constraints on the generation process, as they are in some text-based generators.

We should point out that there is nothing in the architecture that requires that templates or a grammar need be used in any module; either of these, or other forms of generation (finite state models, statistical generators) can be used.

This architecture allows for different levels of processing, incremental generation, and fast generation in some cases. It also allows us to combine different types of generators into one component. We believe it, or something like it, will permit natural interaction in the context of conversational agents.

5 Conclusion

We have highlighted the unique difficulties of performing generation for free-flowing task-oriented dialog and the possibilities inherent in using an approach to generation that combines the use of templates with the use of a grammar and planning. We have also classified some of the ways templates can be used in strategic and tactical generation for dialog.

These are our initial hypotheses based on a preliminary examination of our data. We hope to soon be able to confirm or deny them, and point out any complicating factors of which we become aware during further data analysis and preliminary system development.

References

[1] M. Bordegoni, G. Faconti, S. Feiner, M. Maybury, T Rist, S. Ruggieri, P. Trahanias, and M. Wilson. A standard reference model for intelligent multimedia presentation systems. *Computer Standards and Interfaces*, 18(6, 7):477–496, December 1997.

[2] J. Chu-Carroll and S. Carberry. Collaborative response generation in planning dialogues. *Computational Linguistics*, 24(3):355–400, 1998.

[3] K. De Smedt, H. Horacek, and M. Zock. Architectures for natural language generation: Problems and perspectives. In *Trends in Natural Language Generation: An Artificial Intelligence Perspective*, pages 17–46. Springer-Verlag, Berlin, Germany, 1996.

[4] G. Ferguson and J. Allen. TRIPS: an intelligent integrated problem-solving assistant. In *Proceedings of the fifteenth national conference on artificial intelligence (AAAI-98)*, Madison, WI, 1998.

[5] K. Lochbaum. A collaborative planning model of intentional structure. *Computational Linguistics*, 24(4):525–572, December 1998.

[6] W. Mann and S. Thompson. Rhetorical structure theory: a theory of text organisation. In L. Polanyi, editor, *The structure of discourse*. Ablex, Norwood, NJ, 1987.

[7] N. Reithinger. POPEL – a parallel and incremental natural language generation system. In C. L. Paris, W. R. Swartout, and W. C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 179–199. Kluwer Academic Publishers, Boston, MA, 1991.

[8] D. Traum and E. Hinkelman. Conversation acts in task-oriented spoken dialogue. *Computational Linguistics*, 18(3):575–599, 1992.