Recursion in Text
and its Use in
Language Generation

Kathleen R. McKeown
Department of Computer Science
Columbia University
New York, NY 10027

This paper also appears in Proceedings of the National Conference on Artificial Intelligence. Washington, D.C., 1983.
RECURSION IN TEXT
AND ITS USE IN LANGUAGE GENERATION

Kathleen R. McKeown
Department of Computer Science
Columbia University
New York, NY 10027

ABSTRACT

In this paper, I show how textual structure is recursive in nature; that is, the same rhetorical strategies that are available for constructing the text's macrostructure are available for constructing its sub-sequences as well, resulting in a hierarchically structured text. The recursive formalism presented can be used by a generation system to vary the amount of detail it presents for the same discourse goal in different situations.

1 Introduction

Texts and dialogues often contain embedded units which serve as sub-function of the text or dialogue as a whole. This has been noted both by Grosz [GROSZ 77] in her observations on task dialogues and by Reichman [REICHMAN 81] in analyses of informal conversations. In this paper, I show how textual structure is recursive in nature, that is, the same rhetorical strategies that are available for constructing the text's macrostructure are available for constructing its sub-sequences as well, resulting in a hierarchically structured text. This complements Grosz's view of hierarchical text structure as a mirror of hierarchical task structure. A generation system can use recursion to generate a variety of different length texts from a limited number of discourse plans which specify appropriate textual structures. In the following sections, I present a formulation of recursive text structure, an example of its use in the fully implemented TEXT generation system [MCKEOWN 82b] and finally, a description of some recent work on the application of this mechanism to automatically generating the appropriate level of detail for a user.

2 What is Textual Recursion?

Rhetorical predicates (also termed coherence relations) have been discussed e.g. [GRIMES 75], [HIRST 81], [HOHIS 78] as a means for describing the predating acts available to a speaker. They delineate the structural relations between propositions in a text. Some examples are 'identification' identify an item as member of a generic class, 'analogy' (compare with a familiar object), and 'particular-illustration' (exemplify point). In earlier papers [MCKEOWN 82a, MCKEOWN 82b], I showed how such predicates could be combined to form longer textual sequence serving a single discourse purpose (for example, definition). These combinations were formalized as schemata which embody text structures commonly used in naturally occurring texts, as determined by empirical analysis.

This analysis also indicated that the predicates may be applied recursively to describe the structure of a text at many levels. A predicate may characterize the structural relation of a single sentence or of a longer sequence of text, such as a paragraph, to preceding text. Schemata merely indicate how predicates may be combined to form longer sequences of texts having specific functions. Thus, they describe combinations of predicates which serve the function of a single predicate. Textual recursion is achieved by allowing each predicate in a schema to expand either a single proposition (e.g., a clause or a sentence), or to its associated schema (e.g., a text sequence).

As an example, consider the sequences of text shown in Examples 1 and 2 below. The structure of both these texts is captured by the identification schema a schema which describes the combination of predicates that are commonly used to provide definitions. In the first text, sentence 1 identifies the home cat. 2 describes characteristic attributes, and 3 provides an example. The second text contains the same basic structure except that the identification of the home cat is achieved by a textual sequence instead of a single sentence. This textual sequence (sentences 1-10) is also described by an instantiation of the identification schema. Note that any of the other predicates of either the higher level identification schema or the embedded definition could have been expanded by their associated schemata if the

---

1This work was partially supported by NSF grant #NSF81-07320, awarded by the Dept. of Computer and Information Science of the University of Pennsylvania.

2The schema itself is not shown here. That schemata allow for optional predicates accounts for the variation in the instantiations of the identification schema shown here. See [MCKEOWN 82] for a full description of the schemata themselves.
author/speaker preferred to provide more detail.  

Example 1  

Identification Schema  
1. Identification  
2. Attributive  
3. Particular illustration  

1. A hobie cat is a brand of catamaran, manufactured by the Hobie Company. 2. Its main attraction is that it's cheap. 3. A new one goes for about $5000.

Example 2  

Identification Schema  
1. Identification  
2. Identification  
3. Analogy  
4. Particular illustration  
5. Attributive  
6. Particular illustration  

1. A hobie cat is a brand of catamaran, manufactured by the Hobie Company. 2. Catamarans are sailboats with two hulls instead of the usual one. 3. A catamaran is typically much faster than a sailboat. 4. Hobie cats, tiger cats, and pacific cats are all catamarans. 5. As for the hobie cat, its main attraction is that it's cheap. 6. A new one goes for about $5000.
4 Limits on Recursion

It has been suggested (e.g., CONKLIN 83) that the phenomenon described here may not actually be recursion per se since 1) there may be bounds on how many recursive pushes can be taken and 2) a speaker may not return in reverse order to every higher level dialogue from which a push was taken.

The existence of bounds on the depth of recursion is not motivated by the occurrence of recursion in naturally occurring texts. My analysis suggests instead that many levels of nesting are possible, but that when such nesting occurs the text grows in length and may cover several pages. MCKEOWN 82A]. Gross' analysis of hierarchically nested dialogues also indicates that nesting can occur to many levels. Placing arbitrary bounds on the depth of recursion could conceivably limit a generation system in its ability to provide the kind of detail needed by a user in a given situation. The absence of limits on recursive depth, on the other hand, does not have detrimental side-effects as long as the system is capable of determining in what situations recursion is not necessary.

Bounds on recursion are even more severely limiting on the generality of a generation system than this suggests. Note that if no recursion is allowed, the system will only be capable of producing texts of a uniform length unless further changes are made in the system. A single schema will consistently produce paragraph length text if its predicates are always expanded as single propositions. To generate longer texts, the system must either be capable of combining schemata appropriately (requiring further work on a new combination of schemata) or new schemata must be developed which will generate longer sequences of text.

If, on the other hand, recursion is allowed, then a limited number of schemata can be used to generate an infinite number of different length texts. A single schema produces infinitely many texts if its different predicates are expanded to their associated schemata instead of single propositions and this expansion occurs at all levels of the text. The use of unlimited recursion, therefore, allows for less work to be done in determining possible text ordering in theory, for the generation of arbitrarily long texts from a small number of schemata. Currently, schemata for 1 predicate have been developed for the TEXT system which uses a total of 10 predicates.

In the written texts that were analyzed, writers did return in reverse order to higher level texts from which a push was taken, with the exception of cases where a push was taken on the last predicate in a schema. I would speculate that whether a speaker does return to every dialogue from which a push was taken may be affected by his/her memory for the past discourse. That memory is not perfect may cause higher level unfinished discourse to be skipped when finishing a sub-dialogue. If memory is

the cause, then well-planned writing should exhibit the phenomenon of imperfect recursion less often since planning, re-reading, and rewriting is possible. This hypothesis could be empirically tested.

5 Current Directions

The recursive mechanism can be used to allow a generation system to provide either a detailed or succinct response to the same question under different circumstances. Clearly, an analysis of the factors that trigger or inhibit recursion is critical for use of this capability and this is an endeavor that is currently underway. A preliminary analysis indicates that these factors would at least include the following:

1. The user's level of expertise. A user comes to a system with prior knowledge on the subject in question. The system's knowledge of that level (whether deduced from interaction or explicitly stated) will influence how much it should say. Note that this is not a simple influence. An expert may in certain situations be able to handle more detail than a novice.

2. The past discourse. What the user has learned through the past discourse influences level of detail since previous discussion of a subject may mean that less can be said about it in a current response. What the system has learned through the past discourse affects level of detail as well: the user's acceptance of detail or request for detail may indicate to a system that it can provide a particular type of detail without being asked.

3. The user's overall goal in interacting with the system. Whether the user is using the system, for instance, to quickly retrieve a specific fact or to learn about or from the system will require different levels of detail.

4. The user's specific goal in asking a particular question. If the user's is only one step towards acquiring the information necessary for a higher level goal, that goal may dictate how much information is required.

5. Feedback from the user. While the goal of this research is to anticipate the user's needs for detail before she states them explicitly, in actual conversations people often do explicitly state that they have absorbed information and are ready for more (e.g., backchannel noises such as 'um-hum') or that they have not understood. Such feedback can also be used in a system.

While some of these factors are very difficult to implement (e.g., determining the user's goal), others are, in fact, tractable. Tracking of past discourse, for example, has been used previously to avoid repetition [MCDONALD 80, DAVLY 79]. The recursive mechanism
is also viewed as an important element in providing re-explanations. That is, a user's dissatisfaction with a given response may provide the trigger to recurse on a predicate that was previously unexpanded.

This effort is being conducted with the goal of implementing an information/expert system that can provide explanations in the domain of advising students about course schedules. This domain requires the capacity for communicating at different levels of detail and for providing re-explanations since students as users may frequently be dissatisfied with an explanation (for example, why they cannot take a course), may simply want to talk at length about a course of action, and may want to explore alternate solutions to a problem.

6 Conclusions

In this paper, a formalism which represents the hierarchical nature of texts in terms of recursive textual structure has been presented. This augments previous work on the structure of sub-dialogues by capturing another dimension along which sub-sequences of text are related to the text as a whole. Furthermore, this formulation of text structure allows a generation system to use the same schema to generate both short and more detailed descriptions. While this has already been used in a limited way in the TEXT generation system, the eventual goal is to develop a full analysis of decision mechanisms for recursion and embody this in a generation system which can provide explanations at varying levels of detail as well as re-explanations in response to a user's dissatisfaction.

ACKNOWLEDGEMENTS

I would like to thank Michael Lebowitz, Bonnie Webber, and Kathy McCoy for their comments on drafts of this paper.

7 References


