

ELABORATION IN OBJECT DESCRIPTIONS THROUGH EXAMPLES

Vibhu O. Mittal

Department of Computer Science
University of Southern California
Los Angeles, CA 90089

USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292

Abstract

Examples are often used along with textual descriptions to help convey particular ideas - especially in instructional or explanatory contexts. These accompanying examples reflect information in the surrounding text, and in turn, also influence the text. Sometimes, examples replace possible (textual) elaborations in the description. It is thus clear that if object descriptions are to be generated, the system must incorporate strategies to handle examples. In this work, we shall investigate some of these issues in the generation of object descriptions.

INTRODUCTION

There is little doubt that people find examples very beneficial in descriptions of new or complex objects, relations, or processes. Various studies have shown that the inclusion of examples in instructional material significantly increases user comprehension (for e.g., (Houtz, Moore & Davis, 1973; MacLachlan, 1986; Pirolli, 1991; Reder, Charney & Morgan, 1986; Tennyson & Park, 1980)). Users like examples because examples tend to put abstract, theoretical information into concrete terms they can understand. Few generation systems have attempted to make significant use of examples, however. In particular, most systems have not integrated examples in the textual descriptions, but have used them mostly on their own, independently of the explanation that may also have been provided at that point. However, examples cannot be generated in isolation, but must form an integral part of the description, supporting the text they help to illustrate.

Most previous work (especially in the context of tutoring systems) focused on the issue of *finding* useful examples (for e.g., Rissland's CEG system (1981) and Ashley's HYPO system (Ashley, 1991; Rissland & Ashley, 1986; Rissland, 1983)). Work by Woolf and her colleagues considered issues in the generation of tutorial discourse, including the use of examples (Woolf & McDonald, 1984; Woolf & Murray, 1987), but their

analysis did not address specific issues of integrated example and language generation.

In this paper, we build upon some of these studies and describe the issues in generating descriptions which include examples in a coordinated, coherent fashion, such that they complement and support each other.

AN EXAMPLE

Consider for instance, the example in Figure 1, from a well known introductory book on the programming language LISP. It describes an object (a data structure) called a "list." There are a number of issues that can be immediately seen to be relevant:

1. Should the system choose to elaborate on the object attributes in text, or through the use of examples? For instance, the information in Figure 1 could also have been expressed textually as: "A list always begins with a left parenthesis. Then come zero or more pieces of data (called the elements of a list), and a right parenthesis. Data elements can be of any LISP type, including numbers, symbols and strings". In the figure, the examples are used to elaborate on two aspects of the data-elements: the variable *number* of the elements, and the different *types* of which these elements may belong to. In some contexts, the examples tend to re-iterate certain aspects (in this case, the number was mentioned in the explanation as well), while in others, the examples tend to elaborate on aspects that are not mentioned explicitly in the description (in our case, the *type* information).
2. Should the system use one example, or multiple examples? Consider for instance, the following example of a LISP list:

```
(FORMAT T "~2% - A - A - A" 12345678  
      'James 'Smith (address person))
```

It is not entirely obvious that single examples of the type above are always the most appropriate ones,

A list always begins with a left parenthesis. Then come zero or more pieces of data (called the elements of a list) and a right parenthesis. Some examples of lists are:

(AARDVARK) ;;; an atom
 (RED YELLOW GREEN BLUE) ;;; many atoms
 (2 3 5 11 19) ;;; numbers
 (3 FRENCH FRIES) ;;; atoms & numbers

A list may contain other lists as elements. Given the three lists:

(BLUE SKY) (GREEN GRASS) (BROWN EARTH)

we can make a list by combining them all with a parentheses.

((BLUE SKY) (GREEN GRASS) (BROWN EARTH))

Figure 1: A description of the object LIST using examples (From (Touretzky, 1984), p.35)

even though such examples are frequently seen in technical reference material. The system must therefore be able to make reasonable decisions regarding the granularity of information to be included in each example and structure its presentation accordingly.

- If there are multiple examples that are to be presented, their order of presentation is important too. Studies has shown that users tend to take into account the sequence of the examples as a source of implicit information about the examples (Carnine, 1980; Litchfield, Driscoll & Dempsey, 1990; Tenyson, Steve & Boutwell, 1975). For instance, in Figure 1, the first and second examples taken together illustrate the point that the number of data elements is not important.
- When are 'prompts' necessary? Examples often have attention focusing devices such as arrows, marks, or as in the Figure 1, extra text, associated with them. These help the user disambiguate the salient from the irrelevant. What information should be included in the prompts, and in the case of text, how should be be phrased?
- How should the example be positioned with respect to the text? Studies of instructional texts reveal that examples can occur before the text (and the text elaborates upon the example), within the text, or (as in our figure), after the text (Feldman, 1972).

There are other issues that need to be considered in an integrated framework – some of these that affect most of the issues raised above are the audience-type, the knowledge-type (whether the concept being described is a noun or a relation for instance) and the text-type (tutorial vs. reference vs. report, etc). The

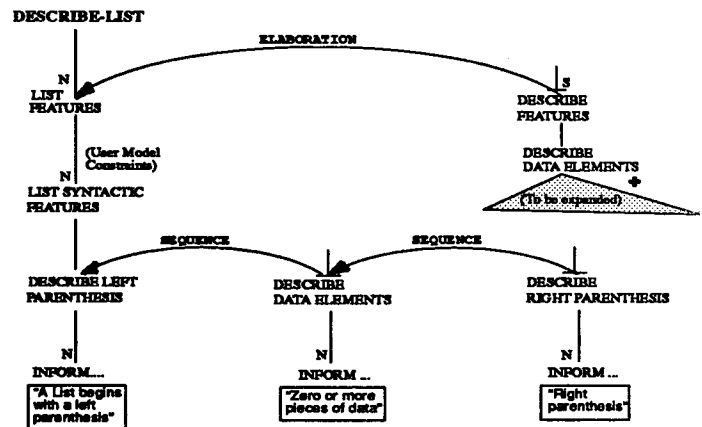


Figure 2: Plan skeleton for listing the main features of a LIST.

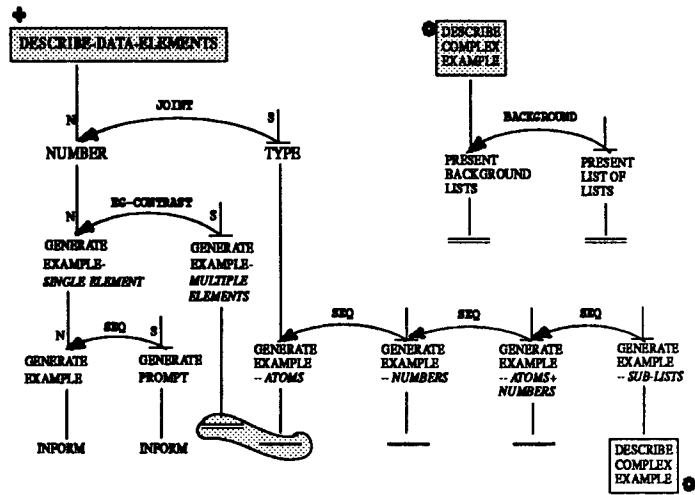


Figure 3: Partial text plan for generating the LIST examples.

issue of how the examples are selected (generated vs. retrieved is also an important issue, but we shall not discuss that here.

STATUS OF WORK

We are investigating these issues by implementing a system that can generate examples within explanatory contexts (within the EES framework (Neches, Swartout & Moore, 1985; Swartout & Smoliar, 1987)) using the Moore-Paris planner (1992, 1991) for discourse generation. Our initial system is for the automatic generation of documentation for small sub-sets of programming languages. One reason for this choice is that it allows us to study a variety of example-rich texts in a

relatively unambiguous domain. A partial text-plan generated by our planner for the description given in Figure 1 is given in Figures 2 and 3. It shows some of the communicative goals that the planner needs to be able to satisfy in order to generate some of the simple object descriptions in our application. These descriptions can make use of examples (instead of text) to list and describe feature elaborations, or use them in conjunction with a textual description to clarify and illustrate various points.

Among issues that we plan to study are the differences between opportunistic generation of examples and top-down planning of text with examples, and the effects arising from differences in the knowledge type, the text-type and other sources of information.

Acknowledgments

Thanks to Cécile Paris for critical discussions, different perspectives and bright ideas. This work was supported in part by the NASA-Ames grant NCC 2-520 and under DARPA contract DABT63-91-C-0025.

References

- Ashley, K. D. (1991). Reasoning with cases and hypotheticals in HYPO. *International Journal of Man-Machine Studies*, 34(6), 753–796.
- Carnine, D. W. (1980). Two Letter Discrimination Sequences: High-Confusion-Alternatives first versus Low-Confusion-Alternatives first. *Journal of Reading Behaviour*, XII(1), 41–47.
- Feldman, K. V. (1972). The effects of the number of positive and negative instances, concept definitions, and emphasis of relevant attributes on the attainment of mathematical concepts. In *Proceedings of the Annual Meeting of the American Educational Research Association*, Chicago, Illinois.
- Houtz, J. C., Moore, J. W., & Davis, J. K. (1973). Effects of Different Types of Positive and Negative Examples in Learning “non-dimensioned” Concepts. *Journal of Educational Psychology*, 64(2), 206–211.
- Litchfield, B. C., Driscoll, M. P., & Dempsey, J. V. (1990). Presentation Sequence and Example Difficulty: Their Effect on Concept and Rule Learning in Computer-Based Instruction. *Journal of Computer-Based Instruction*, 17(1), 35–40.
- MacLachlan, J. (1986). Psychologically Based Techniques for Improving Learning within Computerized Tutorials. *Journal of Computer-Based Instruction*, 13(3), 65–70.
- Moore, J. D. & Paris, C. L. (1991). Discourse Structure for Explanatory Dialogues. Presented at the Fall AAAI Symposium on Discourse Structure in Natural Language Understanding and Generation.
- Moore, J. D. & Paris, C. L. (1992). User models and dialogue: An integrated approach to producing effective explanations. To appear in the ‘User Model and User Adapted Interaction Journal’.
- Neches, R., Swartout, W. R., & Moore, J. D. (1985). Enhanced Maintenance and Explanation of Expert Systems Through Explicit Models of Their Development. *IEEE Transactions on Software Engineering*, SE-11(11), 1337–1351.
- Pirolli, P. (1991). Effects of Examples and Their Explanations in a Lesson on Recursion: A Production System Analysis. *Cognition and Instruction*, 8(3), 207–259.
- Reder, L. M., Charney, D. H., & Morgan, K. I. (1986). The Role of Elaborations in learning a skill from an Instructional Text. *Memory and Cognition*, 14(1), 64–78.
- Rissland, E. L. (1981). Constrained Example Generation. COINS Technical Report 81-24, Department of Computer and Information Science, University of Massachusetts, Amherst, MA.
- Rissland, E. L. (1983). Examples in Legal Reasoning: Legal Hypotheticals. In *Proceedings of the International Joint Conference on Artificial Intelligence*, (pp. 90–93), Karlsruhe, Germany.
- Rissland, E. L. & Ashley, K. D. (1986). Hypotheticals as Heuristic Device. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-86)*, (pp. 289–297).
- Swartout, W. & Smoliar, S. W. (1987). Explaining the link between causal reasoning and expert behavior. In *Proceedings of the Symposium on Computer Applications in Medical Care*, Washington, D. C.
- Tennyson, R. D. & Park, O.-C. (1980). The Teaching of Concepts: A Review of Instructional Design Research Literature. *Review of Educational Research*, 50(1), 55–70.
- Tennyson, R. D., Steve, M., & Boutwell, R. (1975). Instance Sequence and Analysis of Instance Attribute Representation in Concept Acquisition. *Journal of Educational Psychology*, 67, 821–827.
- Touretzky, D. S. (1984). *LISP: A Gentle Introduction to Symbolic Computation*. New York: Harper & Row Publishers.
- Wolf, B. & McDonald, D. D. (1984). Context-Dependent Transitions in Tutoring Discourse. In *Proceedings of the Third National Conference on Artificial Intelligence (AAAI-84)*, (pp. 355–361).
- Wolf, B. & Murray, T. (1987). A Framework for Representing Tutorial Discourse. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, (pp. 189–192).