An Algorithm for High-Level Organization of Multi-Paragraph Texts

Robert Granville
rgranvil@BBN.COM

Our experience with MACH-III [Kurland et al 1992] showed us that there is more to multiparagraph text than stringing together isolated well-formed paragraphs. The underlying structure of the entire text, depicting interparagraph relationships and emphases, must also be determined for successful generation. Fortunately, RST [Mann-Thompson 1987] is capable of representing interparagraph structure as well as intraparagraph structure. However, RST does not specify how to build large structures representing multiparagraph text. This paper presents an algorithm to construct such multiparagraph structures representing a critique of a student's performance in troubleshooting the HAWK radar, as determined by MACH-III. This critique is based on the functional hierarchy tree (FH tree), which is the heart of the expert system component of MACH-III [Kurland et al 1989]. Each student action is judged based on the structure of the FH tree, and where the student is currently located in that tree. The generated critique thus needs to describe each action and how that action helps or hinders proper navigation through the FH tree.

Functional hierarchy is a new paradigm for organizing expert system knowledge bases, based on the procedural abstraction principles of Liskov and Gutag [Liskov-Gutag 1986]. Functional hierarchy differs greatly from production rules (the customary basis for an expert system) in that functional hierarchy rules define the actions a system can take, rather than the conditions under which actions may take place. The concept of "action" is expanded to include all actions the system takes, including control decisions, rather than just changes to the database, thereby eliminating the need for a separate control structure. These rules are arranged in a hierarchy, where the action of a rule is defined as a combination of other actions.

Our algorithm does not need an elaborate component to build a plan resulting in an RST for two reasons. The first is that the desired output text is a non-interactive monologue of written text, rather than text that models interactive dialogue, as in the text of Moore's system [Moore 1989]. Therefore, we don't need information for revising text in reaction to a listener's misunderstanding. The second is that the database from which we are generating consists of the MACH-III functional hierarchy trees. These FH trees are structured purposely to explicitly reflect the very organization we need to explain. Because of these FH trees, we don't have to build plans to determine text structure, and the job of organizing the text, that is, building RST structures is greatly simplified.

As stated above, the purpose of our generated text is describe student actions in the course of a radar troubleshooting session, and how these actions relate to the organization in the FH trees. In themselves, these actions are completely independent. The way RST organizes events that have no relationship between them other than the order in which they occurred is with the SEQUENCE relationship.

This leaves the problem of the higher organization of the text, that is, where to break paragraphs in the RST. It was argued in [Granville 1990] that the structural organization of a text is as important to the message to be conveyed as its factual content. Therefore, any artificial metric for paragraphs, such as limiting them to a specific number of sentences, must be unsatisfactory. However, the problem is not so daunting when we consider the purpose of a paragraph, which is to describe one idea or topic. The obvious topic category for our text consists of events in the course of troubleshooting that require explanation. The two events in the MACH-III domain that require explanation are mistakes committed by the student, which are mistakes in navigating through the FH trees, and milestones telling the student when branches in the FH trees are entered or completed. The mistake or milestone being explained is obviously the topic of the explanation, and therefore deserves its own paragraph. Mistakes and milestones that have already been explained don't require full explanations again, and therefore don't merit paragraph treatment.

This does not completely solve the problem, however. Depending on the FH tree and the student actions, we may have a set of actions requiring no detailed explanation but is still too large to describe in a single paragraph. This problem can be solved if we remember that the ultimate goal of our text is to explain how actions should reflect the FH tree organization of the troubleshooting knowledge. By causing paragraph breaks when new FH tree branches are entered, our text explicitly reflects the FH tree organization. Another problem is that we may want to avoid a paragraph break after a detailed explanation, if the next text item is closely related, such as a single student action that is an error needing an explanation which also happens to start a new FH branch. Therefore, if two consecutive items are closely related, such as due to arising from the same action, or a repetition of an action, they will appear in the same paragraph, even though the first item would cause a paragraph break under normal circumstances.

A brief example will help make these ideas concrete. Assume a MACH-III student is presented with an LO fault, indicating that the local oscillator (the LO) is either malfunctioning or not being monitored correctly. Our student performs four actions, each with a mistake. He first tests the W2 cable, a component of the HAWK radar that can't cause an LO fault even if the cable itself is faulty. In MACH-III, this is known as an OUT OF TREE mistake, because since the W2 can't cause an LO fault, it doesn't appear in the subtree of the functional hierarchy that deals with LO faults.

Our student next replaces the A11 card, a component that could cause the fault. However, he fails to follow this action with a test of the system (known as a BITE test) to see if the A11 was in fact the cause of the problem. MACH-III calls this an EXPECTED BITE TEST mistake.

The student's third action is to test the W8 cable. Like the W2, the W8 cannot possibly cause an LO fault, so this is again an OUT OF TREE mistake.

Finally, the student tests the P12J3 plug. This component can cause an LO fault, so it isn't an OUT OF TREE problem. However, testing the A11 card is part of determining whether the local oscillator is being monitored correctly, while testing the P12J3 is part of determining whether the local oscillator itself is malfunctioning. As stated above, one of the goals of MACH-III is to teach students the organization exhibited in the functional hierarchy tree. Therefore, jumping around from topic to topic, as our student is doing here, is flagged as a SKIPPING fault.

When the first action is encountered, the OUT OF TREE mistake hasn't been explained yet, so the action of testing the W2 and the OUT OF TREE problem warrant their own paragraph detailing why OUT OF TREE actions are mistakes. Similarly, replacing the A11 card requires a separate paragraph to explain the EXPECTED BITE TEST problem. The third action, testing the W8, does not force a paragraph break on its own, since it exhibits the OUT OF TREE mistake, which has already been explained. Instead, it merely mentions the problem, and the paragraph continues. The final action of testing the P12J3 does exhibit a new problem, though, namely SKIPPING. It is added to the third paragraph in a sequence relationship, and this paragraph is closed off because of the required explanation. The resulting RST looks like the following:

```
                            TEXT
                            ----
                             |
         ----------------------------------------
         |        PARAGRAPH  | SEQUENCE         |
 ---------              ---------               |
  EXPLAIN                 EXPLAIN             ---
OUT-OF-TREE          EXPECTED-BITE-TEST         |
     W2                     A11            ------------
                                           | SEQUENCE |
                                     ----------  ---------
                                      MENTION     EXPLAIN
                                    OUT-OF-TREE   SKIPPING
                                        W8         P12J3
```

References:

[Granville 1990] Granville, Robert A., "The Role of Underlying Structure in Text Generation," in Proceedings of the Fifth International Workshop on Natural Language Generation, 1990

[Kurland et al 1989] Kurland, Laura C., Robert Granville, and Dawn MacLaughlin, "HAWK MACH-III Explanations of the Receiver Troubleshooting Tree," Technical Report, BBN Systems and Technologies, Cambridge, Massachusetts, 1989

[Kurland et al 1992] Kurland, Laura C., Robert Alan Granville, and Dawn M. MacLaughlin, "Design, Development and Implementation of an Intelligent Tutoring System (ITS) for Training Radar Mechanics to Troubleshoot," in "Intelligent Instruction by Computer," edited by Marshall J. Farr and Joseph Psotka, Taylor & Francis, Washington, DC, 1992

[Liskov-Guttag 1986] Liskov, Barbara, and John Guttag, Abstraction and Specification in Program Development, MIT Press, Cambridge, Massachusetts, 1986

[Mann-Thompson 1987] Mann, William C., and Sandra A. Thompson, "Rhetorical Structure Theory: A Theory of Text Organization," ISI/RS-87-190, Information Sciences Institute, Marina del Rey, California, 1987

[Moore 1989] Moore, Johanna Doris, "A Reactive Approach to Explanation in Expert and Advice-Giving Systems," Ph.D. thesis, University of California, Los Angeles, 1989