DOCUMENT RESUME

ED 289 476 IR 012 943

AUTHOR	Mann, William C.
TITLE	Text Generation: The Problem of Text Structure.
INSTITUTION	University of Southern California, Marina del Rey. Information Sciences Inst.
SPONS AGENCY	Advanced Research Projects Agency (DOD), Washington, D.C.
REPORT NO	ISI/RS-87-181
PUB DATE	Mar 87
NOTE	28p.; ISI Reprint Series. Reprinted from Natural Language Generation Systems, 1987.
PUB TYPE	Reports - Research/Technical (143)
EDRS PRICE	MF01/PC02 Plus Postage.
DESCRIPTORS	*Artificial Intelligence; Comparative Analysis; *Computational Linguistics; Discourse Analysis
IDENTIFIERS	*Natural Language Processing; Rhetorical Structure Theory; *Text Generation; Text Structure

ABSTRACT

One of the major problems in artificial intelligence (AI) text generation is text organization; a poorly organized text can be unreadable or even misleading. A comparison of two AI approaches to text organization--McKeown's TEXT system and Rhetorical Structure Theory (RST)--shows that, although they share many assumptions about the nature of text, they are also in strong contrast. TEXT identifies text organization with whole-text nonrecursive structures, while RST uses small recursive ones. RST has an elaborate apparatus of relations between parts of texts, and of the "nuclearity" of particular parts; TEXT has no correlates for these. RST works with a wide range of relation types, TEXT with just one. TEXT is an implemented system, whereas RST is developmental. More important, TEXT develops text organizations so that they resemble patterns extracted from previous text, while RST strives for an organization which is justifiable as meeting the goals of the text being generated. This contrast raises many of the key issues in current research on the nature of text organization and how it can be created by programs. A 13-item reference list is appended. (Author/RP)

ISI Reprint Series
ISI/RS-87-181
March 1987

University of Southern California

William C. Mann

Text Generation: The Problem of Text Structure

Reprinted from
Natural Language Generation Systems, 1987.

INFORMATION SCIENCES INSTITUTE
213/822-1511
4676 Admiralty Way/Marina del Rey/California 90292-6695

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT This document is approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) ISI/RS-87-181	5. MONITORING ORGANIZATION REPORT NUMBER(S) ---------------
6a. NAME OF PERFORMING ORGANIZATION USC/Information Sciences Institute 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION ---------------
6c. ADDRESS *(City, State, and ZIP Code)* 4676 Admiralty Way Marina del Rey CA 90292	7b. ADDRESS *(City, State, and ZIP Code)* ---------------
8a. NAME OF FUNDING/SPONSORING ORGANIZATION ARPA, NSF, AFOSR 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER MDA903 81 C 0335 IST-8408726 FQ8671-84-01007
8c. ADDRESS *(City, State, and ZIP Code)* Advanced Research Projects Agency (ARPA) 1400 Wilson Blvd. Arlington, VA 22209 (over)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. --------------- PROJECT NO. --------------- TASK NO. --------------- WORK UNIT ACCESSION NO. ---------------

11. TITLE *(Include Security Classification)*
Text Generation: The Problem of Text Structure (Unclassified)

12. PERSONAL AUTHOR(S) William C. Mann

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Research Report	FROM ______ TO ______	1987, March	22

16. SUPPLEMENTARY NOTATION
Reprinted from D. McDonald and L Bolc, eds., *Natural Language Generation Systems.* New York: Springer-Verlag, 1987 (In the series *Symbolic Computation*) (over)

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	artificial intelligence, computational linguistics, discourse analysis, generation, natural language processing, rhetorical structure theory, text generation, text structure, text planning
09	02		

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

One of the key problems in AI text generation is text organization. A poorly organized text can be unreadable or even misleading. Two AI approaches to text organization are compared: McKeown's TEXT system and Rhetorical Structure Theory (RST).

They share many assumptions about the nature of text yet they are found to be in strong contrast. TEXT identifies text organization with whole-text nonrecursive structures, while RST uses small recursive ones. RST has an elaborate apparatus of relations between parts of texts, and of the "nuclearilty" of particular parts; TEXT has no correlates for these. RST works with a wide range of relation types, TEXT with just one of them. TEXT is an implemented system, whereas RST is developmental. Most important, TEXT develops text organizations so that they resemble patterns extracted from previous text, while RST strives for an organization which is justifiable as meeting the goals of the text being generated.

This contrast raises many of the key issues discussed in this paper about the nature of text organization and how it can be created by programs.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Sheila Coyazo Victor Brown	22b. TELEPHONE *(Include Area Code)* 213-822-1511 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

8c.

National Science Foundation (NSF)
1800 G Street NW
Washington, DC 20550

Air Force Office of Scientific Research (AFOSR)
Bolling Air Force Base, Building 410
Washington, DC 20?32

16.

This paper was first presented as an oral presentation at the 1986 COLING Conference in Bonn, West Germany, held August 25-29, 1986

ISI Reprint Series
ISI/RS-87-181
March 1987

University of Southern California

William C. Mann

Text Generation: The Problem of Text Structure

Reprinted from
Natural Language Generation Systems, 1987.

INFORMATION SCIENCES INSTITUTE

213/822-1511

4676 Admiralty Way/Marina del Rey/California 90292-6695

This research was sponsored in part by National Science Foundation grant IST-8408726, in part by Air Force Office of Scientific Research Contract No. FQ8671-84-01007, and in part by the Defense Advanced Research Projects Agency Contract No. MDA903 81 C 0335; the opinions in this paper are solely those of the authors.

ISI Reprint Series

This report is one in a series of reprints of articles and papers written by ISI research staff and published in professional journals and conference proceedings. For a complete list of ISI reports, write to

Document Distribution
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
USA

Text Generation: The Problem of Text Structure

William C. Mann
USC Information Sciences Institute
Marina del Rey, California, USA

1 Overview

What is text generation?[1] The long term view is that it is the process of creating a technology for building computer programs that can function as authors or speakers. I call this the long term view because in the text generation programs in existence today there is very little that deserves the title of "author" or "speaker." Writing and speaking are rightly regarded as complex arts capable of high refinement and great intellectual achievement. In contrast, our programs reflect only fragments of the most basic skills.

Text generation has been studied seriously in computational linguistics only in the last five or ten years, and so it is still sorting out its goals and identifying its problems.

Part of the diversity of approaches in text generation will surely come from a problem that we face now: It is by no means clear how authors do what they do. Even though we are all exposed to text nearly every day, and manipulate it successfully, there is very little explicit knowledge of how text works.

One of the central issues involves text organization. It is evident that natural text is organized, and that its organization is essential to its function. Text has parts, arranged systematically. But what is the nature of text organization or structure? What are the parts, and what are the principles of arrangement?

We must have answers to such questions if we are to create text generators. However, there are no widely accepted answers to these questions. The answers that are available from outside of computational linguistics are partial and complex. There are logicians' answers, grammarians' answers and so forth, often representing mostly the favorite methods and assumptions of their developers -- a priori selectivity rather than comparative results.

---

[1]This paper was created as an invited presentation for the 1986 COLING conference. It is still largely organized as an oral presentation, and so some readers may feel, with me, that does not have all of the attributes one would desire for inclusion in a book. Nevertheless, it seems better to let it appear than not. For additional support for the positions taken here I refer readers to the more formal presentations cited. This research was sponsored in part by National Science Foundation grant IST-8408726, in part by AFOSR contract FQ8671-84-01007, and in part by DARPA contract MDA903 81 C 0335; the opinions in this paper are solely those of the author.

Also, crucially for text generation, there are no accounts of text organization at a level of detail sufficient to support computer programming. Far more detail is needed.

As a result, text generation has been inventing its own answers to these questions. It has had to.[2]

To explore the nature of text structure, we focus in this paper on two of the energetic attempts within Text Generation to describe text structure in a way that is sufficiently detailed and general to serve as a basis for programming. They are:

1. The TEXT system, and

2. Rhetorical Structure Theory

The TEXT system was developed by Kathy McKeown at the University of Pennsylvania, as the centerpiece of her PhD dissertation, and is being followed up at Columbia University and elsewhere [Paris 86]. Rhetorical Structure Theory (RST) was initially defined by Sandra Thompson, Christian Matthiessen and the author; it is under active development at USC Information Sciences Institute.[3]

This paper describes each of these lines of research in its own terms; then they are compared as text structure descriptions. The comparison is extended to the related construction processes, and finally conclusions are drawn about text structures in future text generation work.

2 TEXT Structure Description

The general task of the TEXT system was to explain the structure and terminology, but not the content, of a particular data base.[4]

---

[2]Partly because of these difficulties, some workers in text generation have concentrated on generation tasks that omit the need for extensive text organization [McDonald 80], [Bienkowski 86].

[3]Barbara Fox, Cecilia Ford, and others have made important contributions. It has been influenced significantly by [Grimes 75] and [McKeown 85]; [Mann & Thompson 87] describes its relations to other theories in detail. The support of the National Science Foundation and the Air Force Office of Scientific Research are gratefully acknowledged; the opinions in this paper are solely the author's.

[4]Terminology may be somewhat confusing in this paper, simply because some key technical terms have been used in different ways by different authors. TEXT is McKeown's system, whereas "text" refers to the communication phenomenon. Other words which will have multiple meanings include schema and identification. The problem seems unavoidable.

2.1 The Operating Environment of TEXT

The operating environment of the system included several kinds of resources:

1. Data base of naval information

2. Knowledge base of conceptual categories relevant to naval information

3. Grammar of English.

It also has specific task information, including

1. Task Categories:

    a. Define an entity

    b. Describe an entity

    c. Compare two entities

2. Particular Task Operands, classes or class members.

The knowledge base should be seen as simply a hierarchic categorial extension of the data base, necessary since the data base did not define its own categories. In describing TEXT we will describe only the parts that are expressive of text structure, not the whole system.

2.2 Defined Objects

TEXT had two kinds of defined objects:

1. Predicate Semantics

2. Schemas

The predicate semantics definitions are essentially patterns which can be matched in the data base or knowledge base. They guide the system's search for particular knowledge. For example, Figure 1 shows the pattern for the Identification predicate. In the pattern, "entity" is given, "superord" is the superordinate class category of entity, and "attr" stands for attribute. The "restrictive" and "non-restrictive" lines require finding knowledge which is respectively specific or not specific to the members of the category of <entity>. The example shows the result of matching the pattern given SHIP as the entity.

given-argument: entity

(identification <entity> <superord>
(restrictive <attr-name> <attr-value>)
(non-restrictive <attr-name> <attr-value>))

example:

(identification SHIP WATER-VEHICLE
(restrictive TRAVEL-MODE SURFACE)
(non-restrictive TRAVEL-MEDIUM WATER))

Figure 1: Predicate Semantics for the Identification Predicate

Identification (class & attribute/function)
{Analogy/Constituency/Attributive}*
Particular-illustration/DB-attribute+
{Amplification/Analogy/Attributive}
{Particular-illustration/DB-attribute}

Figure 2: The TEXT Identification Schema in Linear Form

The schemas of TEXT are specifications of how expressions of predicates can be combined to form whole texts. For example, Figure 2 shows the TEXT Identification schema in a linear notation.[5]

In Figure 2, Identification refers to the Identification predicate. (It is reuse of the name, not recursion. Recursion was not implemented in TEXT.) It would be expressed in the generated text by asserting some proposition which matched that predicate. Analogy, Constituency, Attributive, Particular-illustration, DB-attribute,[6] and Amplification are names of other predicates.

[5] In the figure, elements in parentheses are comments, {} represent optionality, * represents zero or more repetitions, + represents one or more repetitions, and / represents alternation of the choose-one variety.

[6] my name for the Evidence predicate.

For convenience schemas were also expressed in a directed graph transition network notation reminiscent of ATN grammars, as shown in Figure 3. Generation of a text is represented by passing through the graph from the entry node, named ID/, to one of the nodes which bears an arc labeled "POP" such as ID/END. The other graphs are used as subroutines.

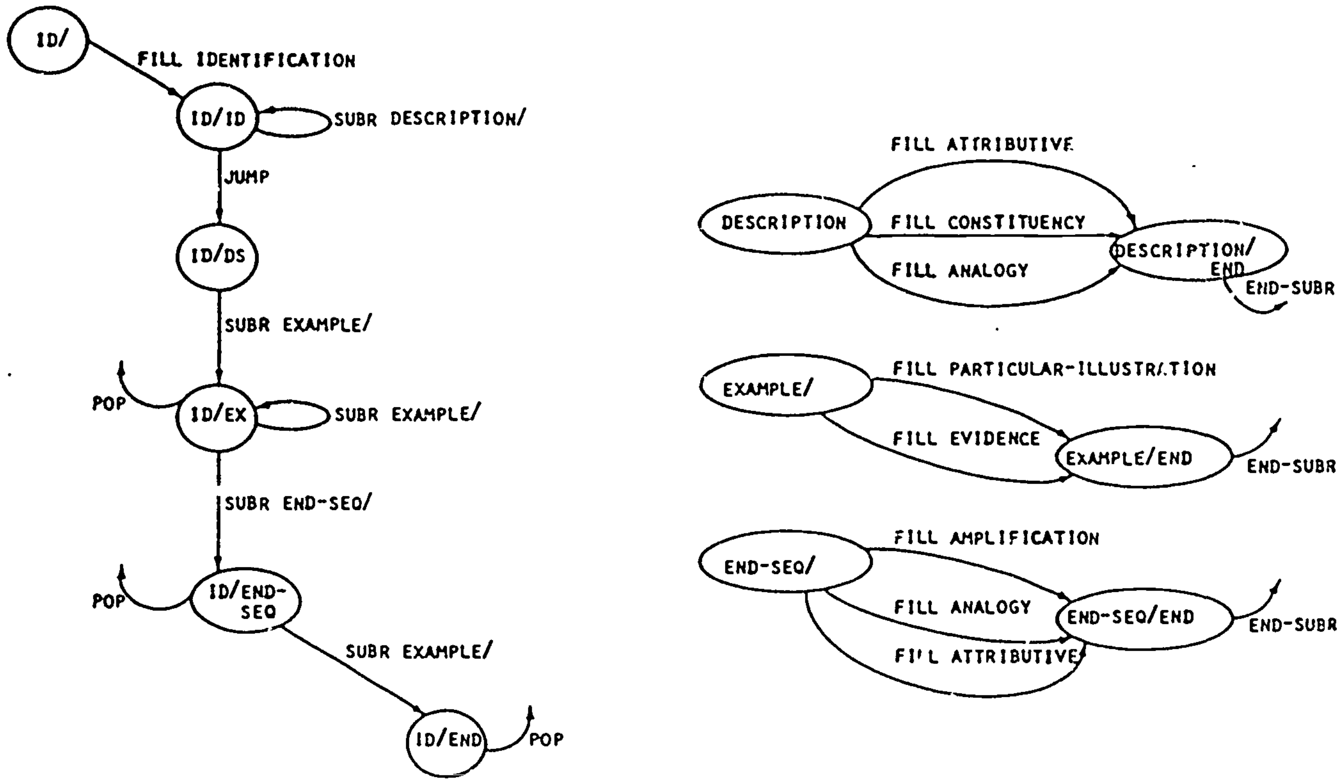


Figure 3: Transition Network Form of the TEXT Identification Schema

>From the point of view of text structures, three features of the TEXT schemas are particularly significant:

1. The scale of schemas is the whole text. Texts are specified as whole units, whose structural possibilities are prespecified rather than created for the individual instance.

2. Optionality of elements is built in at the whole-text level.

3. Order of elements is prespecified.

Where do the schemas come from? They are abstracted from previous texts which are judged to be doing the same task. By studying texts from other knowledge domains, the high frequency patterns are identified and represented in the schemas.

Figure 4 shows a simple identification text generated by TEXT. It is minimal in the sense that all of the optional parts are absent.

The ship is a water-going vehicle that travels on the surface. Its surface-going capabilities are provided by the DB attributes DRAFT and DISPLACEMENT.

Figure 4: Sample Identification Text from TEXT

---

3 RST Structure Description

Now we turn to Rhetorical Structure Theory.[7] Most of this presentation, and most of the work on RST, is on text description rather than text construction. The early focus was entirely on description so that the constructive work would rest on a strong descriptive foundation.

3.1 Defined Objects in RST

There are four classes of defined objects in RST, shown in Figure 5. They are shown in a dependency order. All of the classes are presented below, but not in order.

1. Relations
2. Schemas
3. Schema Applications
4. RST Structures

Figure 5: The Classes of Defined Objects in RST

3.2 Schemas

Figure 6 shows the diagram conventions for the simplest RST schema. It represents graphically a configuration in which there is a span of text (represented by the whole figure) which is composed of two spans, one called the *nucleus* and the other the *satellite*. In such diagrams the nucleus is always below the vertical line. The two spans are related by a named relation.

Figure 7 shows some of the currently defined schemas of RST.[8] The first schema, the Elaboration schema, is typical in that it is defined in terms of only a single relation, in this case the elaboration relation. The second one, the Motivation/Enablement schema, is defined in terms of two relations, motivation and enablement. All of the remaining schemas shown are defined in terms of only a single relation.

Since the Elaboration and Solutionhood schemas are drawn differently, it appears

---

[7]RST was initially defined by Sandra A. Thompson, Christian M. I. M. Matthiessen and the author. Later contributions have been made by Barbara Fox and Cecilia Ford.

[8]A full presentation of schema definitions is in [Mann & Thompson 87]

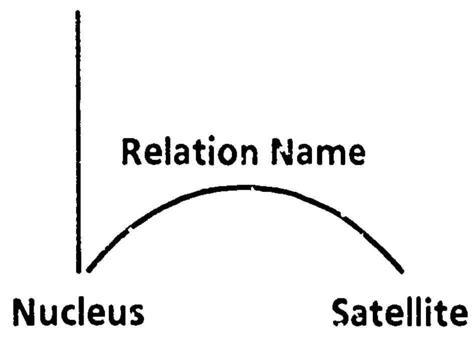


Figure 6: Simplest Generic RST Schema

that the schemas specify the order of the spans of which they are composed, but this is not the case. The spans are unordered in the definition, but the drawings indicate the most frequent ordering. So for example, elaboration satellites tend to follow the nucleus, but satellites in the solutionhood relation (representing presentation of a problem or question) tend to precede the nucleus.

As indicated in Figure 7, there are about 20 relations defined in the current version of RST.

Schema Applications are instances of schemas which correspond to the schema definitions according to a set of application conventions. These conventions allow multiple satellites, but require that a schema application contain only one span as nucleus.

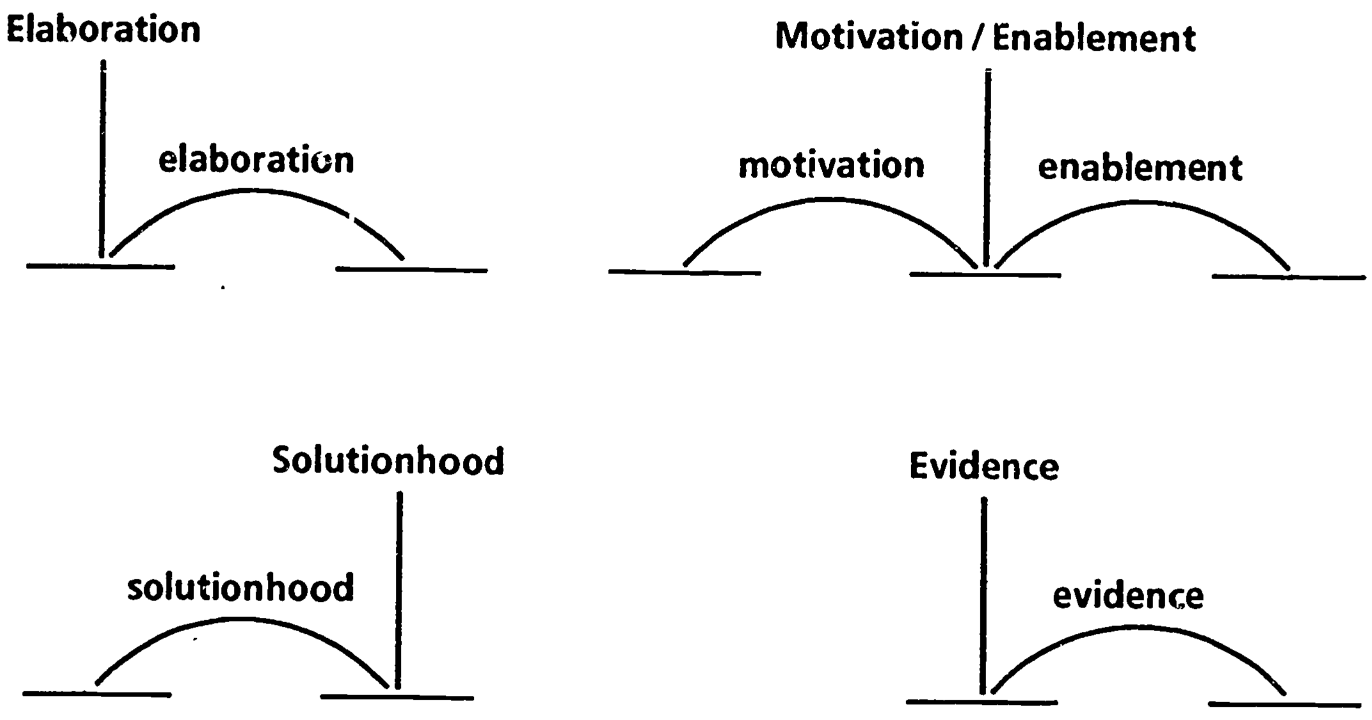
3.3 Text Structures

RST text structures are compositions of schema applications. As an example, Figure 8 shows the RST structure diagram of a 14-unit text that appeared in a political magazine.[9] The units correspond to independent clauses.

At the top of the figure, there is an application of the Motivation/Enablement schema in which Unit 14 is the nucleus and the span of Units 1-13 is a satellite standing in a motivation relation to the nucleus. The nucleus, being a single unit, is not decomposed further.

The satellite is decomposed by an application of the Evidence schema. The

[9]A detailed analysis of this text appears in [Mann 84].



Other Defined Single-Relation Schemas

- *Circumstance
- *Background
- *Justify
- *Volitional Cause
- *Non-Volitional Cause
- *Volitional Result
- *Non-Volitional Result
- *Antithesis
- *Concession
- *Condition
- *Otherwise
- *Interpretation
- *Evaluation
- *Restatement
- *Summary
- *Sequence
- *Contrast

Figure 7: Some RST Schemas

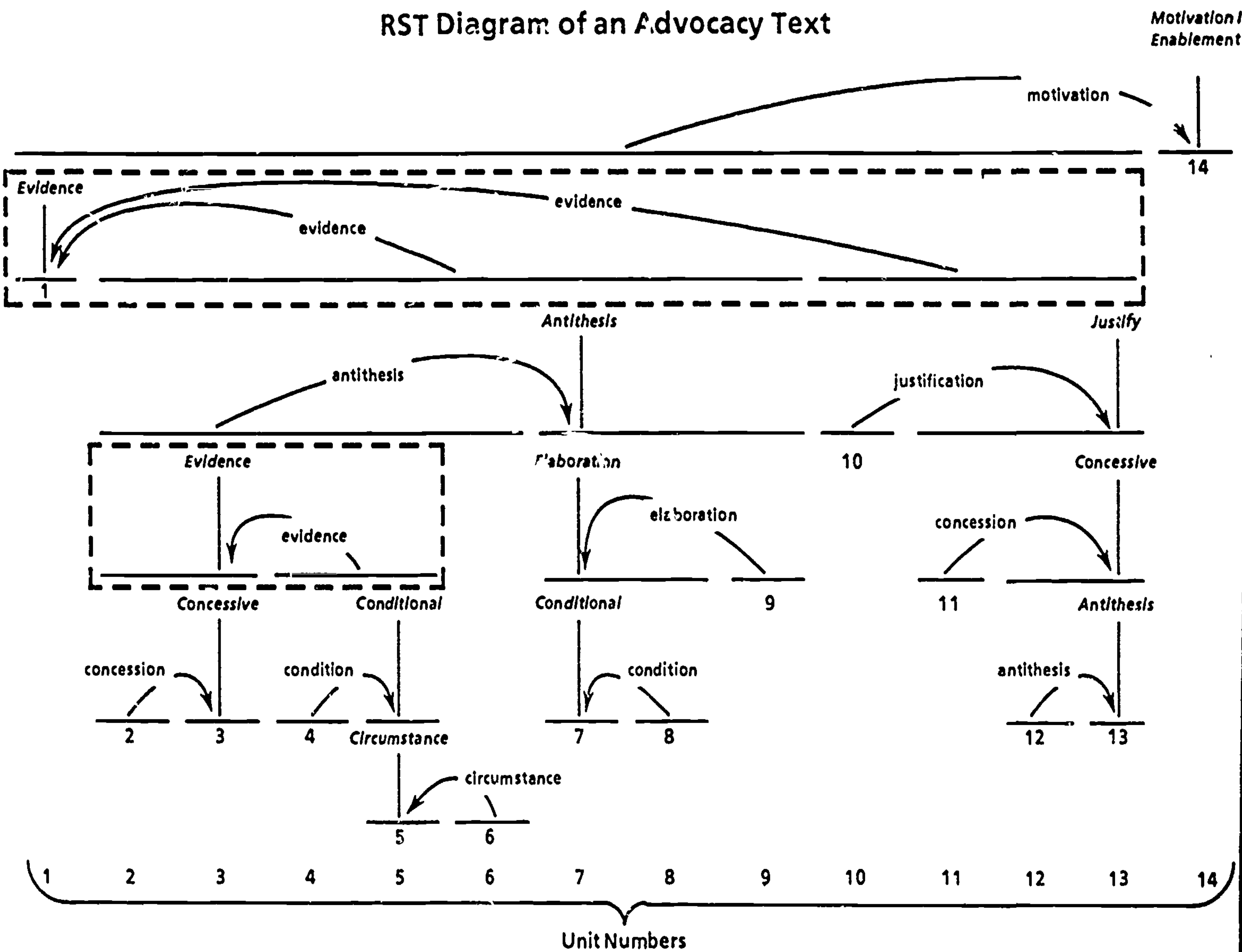


Figure 8: RST Diagram of an Advocacy Text

nucleus, which we will call the *claim*, is Unit 1, and there are two satellites, Units 2-9 and 10-13, standing in an evidence relation to the nucleus. The multiple evidence relations follow a general schema application convention that allows multiple occurrences of any relation in any schema. Within the span of Units 2-9 an Antithesis schema contains a satellite which is decomposed by another use of the Evidence schema. This use of Evidence within Evidence represents the general recursiveness of RST analysis -- that any schema can be used to decompose any span.

Notice that the top-down decomposition using schemas produces a tree structure. This particular structure contains 12 schema applications and 8 different relations.

3.4 Relation Definitions

All of the above rests on the definitions of relations. How are relations defined?

A relation definition has four fields, named in Figure 9.

1. **Constraints on the Nucleus**
2. **Constraints on the Satellite**
3. **Constraints on the combination of Nucleus and Satellite**
4. **The Effect**

Figure 9: Fields of a Relation Definition

In describing a text, constructing an RST structure, it is necessary to decide whether a relation holds between two spans or not. It holds if its definition holds; there are no side conditions.

The decisions on whether a definition holds are plausibility judgments made by the analyst; they are based solely on the text and its context (and not, for example, on the analyst personally knowing the writer or intended reader.) Given two spans, the analyst decides whether it is plausible that each field applies.

The first 3 fields are stated as constraints which must hold for the identified nucleus and satellite spans. These constraint fields are sometimes empty.

The Effect field is a statement by the analyst that it is plausible that <u>the writer desired a particular effect</u> from the spans which the relation relates. It is not necessarily

the only effect, or even the most important effect, that the writer might have intended, but if it is not plausible that the writer had the particular intent for the spans in question, then the relation definition does not hold. The Effect field of a relation definition is never empty.

The significance of the fields becomes clearer with an example. Figure 10 shows the definition of the Evidence relation.

1. **Constraints on the Nucleus (the *claim*):**

   The reader possibly does not already believe the claim.

2. **Constraints on the Satellite (the *evidence*):**

   The reader believes the satellite or will find it credible.

3. **Constraints on the combination of Nucleus and Satellite:**

   Comprehending the evidence will increase the reader's belief in the claim.

4. **The Effect:** The reader's belief in the claim is increased.

Figure 10: RST Definition of the Evidence Relation

The constraint on the nucleus, that the reader possibly does not already believe the claim, simply insures that evidence is needed. The constraint on the satellite, that the reader believes it or will find it credible, insures that the satellite can actually serve as evidence. A non-credible span cannot function as evidence. The constraint on the combination assures that there is actually a support relation of some sort between the two in the writer's view of the reader's view.

The Effect field is a statement by the analyst that it is plausible that one reason the writer may have had for presenting the combination of nucleus and satellite spans was to increase the reader's belief in the claim. If this general statement is not plausible, then the Evidence relation does not hold.

Since the Effect field is never empty, an RST analysis is inherently a statement, for every part of a text, of a plausible reason for including each part in the text. RST is thus an explicitly functional theory, since its analyses are necessarily also accounts of the functions of the parts of a text.

The Effect field is also potentially useful in text generation, since it can help in choosing particular structures.

The kinds of conventions presented above are all that is needed to analyze a text. The analyst finds spans for which the relations hold. When they combine into schema applications, such that those applications form a tree, then the tree is an RST structure.

3.5 Related RST Studies

The methods described above have been used as the basis for a number of linguistic studies. Several of the studies below are PhD dissertations.

*Relational Propositions -- Assertions from Discourse Structure. RST relations have assertional properties, in which they convey information from the discourse structure, distinct from the clausal assertions of the text. In [Mann & Thompson 86] this phenomenon is identified, and in [Mann & Thompson 85] the link between RST and the phenomenon is made.

*Clause Combining -- Hypotaxis, Embedding and "Subordination". In [Matthiessen & Thompson 86], RST is used to establish that so-called subordination is a composite category consisting of two different phenomena.

*Clause Combining -- Antithesis Relations at Large Scale and Clause Scale. In [Thompson & Mann 87], clausal-level and large-scale antithesis relations are found to rest on the same discourse configurations.

*A Linguistic Characterization of the BBC World News Service. In [Noel 86] RST is used to characterize BBC presentational methods.

*Predicting Anaphora. In [Fox 84] RST is used in predicting when anaphora will be used.

These are ongoing studies:

*Clause Combining -- Switch Reference in Quechua

*Contrastive Rhetoric -- Comparing Structures of Essays in Chinese and English

*Rhetorical Questions in English Argumentative Prose

Overall RST is proving to be a flexible descriptive tool for discourse studies.

4 Comparing the Approaches to Structure Description

With the two sketches above we are ready to compare approaches to text structure description. To make the comparison easier, a text produced by TEXT has been analyzed in RST terms. The text is presented in Figure 11 and its RST structure in Figure 12.

Notice first of all that the only RST relation in the analysis is the Elaboration relation. The text contains 11 uses of it. This result actually holds for the TEXT system as a whole. Of the 20 or so RST relations, TEXT is using only one.

On the other hand, within Elaboration TEXT recognizes more detail and diversity in its predicates than RST does. RST has four subtypes of Elaboration: whole to part, object to attribute, set to member and set to subset. Figure 13 shows the correspondence of the two.[10]

Notice that the 16 predicates of TEXT map onto just 3 of the 4 subtypes of Elaboration. TEXT is specifying structure with a much finer grain than RST. But why? What are the benefits to TEXT of this finer grain? There are several:

1. Selectivity that is useful for particular tasks -- e.g. superordinate class information for the Identification schema.

2. Selectivity on what to say when -- predicate diversity interacts with the ordering which is inherent in schema definitions to sequence relevant information.

3. Accommodation of details of search -- e.g. attributes with and without values.

4. Basis for selecting grammatical forms for expressing the information.

Notice that none of these benefits relate to structure-building functions in RST's approach. We will see in Section 5, on text construction, that all of these functions are located somewhere outside of RST structure-building. This could be interpreted in either of two ways: that RST's present level of detail is sufficient for structure building, but needs to be supplemented to support other processes, or that RST's present level of detail is insufficient. Decisive evidence is unavailable.

The most significant differences between the text structure accounts of TEXT and RST are:

*Size of the largest structural unit -- whole text vs. RST schema

*Dual use of TEXT schemas for

1. what constitutes an adequate answer, and

2. text structure

(RST separates these.)

---

[10]The difference between the data base and the knowledge base in TEXT is an implementation detail which has been neutralized away in the table.

1. The cruiser and the ocean escort are surface ships.

2. The ship's surface-going capabilities are provided by the DB attributes DISPLACEMENT and DRAFT.

3. The ship has DB attributes MAXIMUM_SPEED, PROPULSION, FUEL(FUEL_CAPACITY and FUEL_TYPE), DIMENSIONS, SPEED_DEPENDENT_RANGE and OFFICIAL_NAME.

4. Ocean escorts have a DISPLACEMENT between 3400 and 4100.

5. All ocean escorts in the ONR database have REMARKS of 0, FUEL_TYPE of BNKR, FLAG of BLBL and MAST_HEIGHT of 85.

6. Ocean escorts carry between 2 and 22 torpedoes, 16 missles and between 1 and 2 guns.

7. A ship is classified as an ocean escort if the characters 1 through 2 of its HULL_NO are DE.

8. Cruisers have a PROPULSION of STMTURGRD and a LENGTH between 510 and 673.

9. All cruisers in the ONR database have REMARKS of 0.

10. Cruisers carry between 8 and 42 torpedoes, between 4 and 98 missles and between 1 and 4 guns.

11. A ship is classified as a cruiser if the characters 1 through 2 of its HULL_NO are CG.

12. The ocean escort, therefore, has a smaller LENGTH and a smaller DISPLACEMENT than the cruiser.

Figure 11: RST Analysis of a TEXT Text -- Part 1: The Text

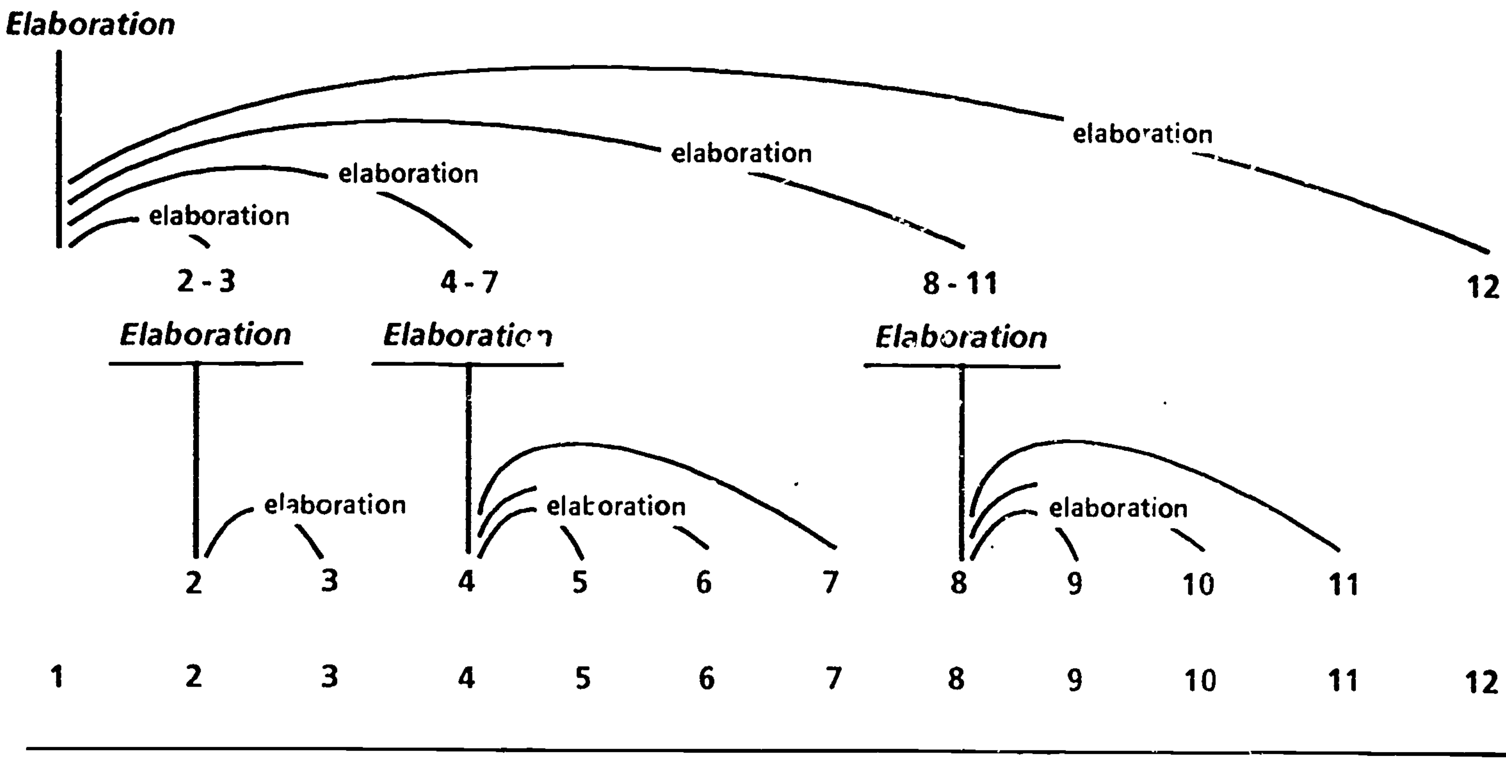


Figure 12: RST Analysis of a TEXT Text -- Part 2: The Structure

TEXT Predicates	Elaboration Subtypes: Whole: Part	Object: Attribute	Set: Member	Set: Subset
Attributive				
attr + value		xxx		
disting.		xxx		
"Evidence"		xxx		
Constituency				xxx
Identification				xxx
Amplification		xxx		
Analogy				
relation		xxx		
range comp.		xxx		
Particular-Illus.				
attr + range			xxx	
attr			xxx	
Explanation		xxx		
Classification				
1			xxx	
> 1			xxx	
Inference (comparability)				
like		xxx		
class		xxx		
different		xxx		

Figure 13: Comparison of TEXT Predicates and Elaboration Subtypes

*Absence of relations in TEXT

*Explicit nuclearity in RST

*Fixeo ordering in TEXT; free in RST.

5 TEXT and RST in Construction

Up to this point little has been said about the text generation methods associated with the text structures of TEXT and RST. Here we compare the two approaches, not explaining either in full but identifying their principal differences.

TEXT has a simple schema-follower that is well documented. It follows the transition network diagrams such as the one in Figure 3, producing one or more sentences that express the propositions of the matched predicate for ea^h arc transition. We need not describe it in detail here.

Constructive RST has a structure-building process as one of several processes involved in developing the pre-grammatical plan for a text. Figure 14 identifies the principal kinds of processes.

In its present state of development, the processor of the procedure is human. Like a very complicated and conditional recipe, it is a procedure that a person can execute but a machine cannot. In tests, it produces identical or acceptable structures for a wide variety of natural texts.

It is now evident that the two processes of generation are very different. Since TEXT relies on predetermined text structures based on study of prior texts, it does not need to reason extensively about the need for particular text elements, nor about how to structure them. Its schemas encode both the task and the method for accomplishing the task. This produces a workable but rigid text structuring method with the major advantages of being implementable and effective in its designed domain.

RST separates text structure building from other processes, (about 8 of which are identified in Figure 14). This has the obvious advantages of flexibility and greater generality, but produces a corresponding need to implement and coordinate a diversity of processes.

Because the RST approach creates structure based on the immediate needs of the text under construction, rather than past texts, it must reason much more about the reader's state and the ways that text affects that state.

Implementation of the RST ap roach depends on finding suitable reductions of it which meet the needs of particular ins. This need for reduction applies both to

1. Before building RST structure:

   a. General decisions about what to accomplish, what knowledge to use. This yields a body of material to convey.

   b. Identification of the audience.

2. RST structure building, including

   a. Organizing the given body of material.

   b. Supplementing it as needed, with evidence, concessives, circumstantials, antithesis, contrast and other supporting material.

   c. Ordering nuclei and satellites.

3. After building RST structure:

   a. Theme control,

   b. Sentence scope,

   c. Conjunction uses,

   d. Lexical choice,

   e. Formulaic text, e.g. "Sincerely yours,",

   f. Grammatical realization.

Figure 14: RST Structure Construction and Related Processes

the oracles of the structure building process and to the processes related to structure building. For better or worse, in a particular text generation application much of the reduction will be forced by the limitations of the current state of the art of knowledge representation.

6 Summary

Considering both text structure and construction processes, the most important comparisons between TEXT and RST are listed in Table 1. Of these differences, the crucial one is the one highlighted -- the difference between structure derived from previous texts and structure derived from goal pursuit. These represent two competing and pervasive influences on language: prior patterns or conventions, and immediate needs.

1. The input to the procedure is a goal.

2. The control structure is topdown and recursive, proceeding by progressive refinement to the single clause level.

3. The procedure contains test conditions determining the need for each RST schema.

4. Use of some schemas, e.g. Conditionals, involves simple goal decompositions, whereas use of others, e.g. Evidence, involves addition of goals.

5. The procedure is based on <u>diverse and detailed appeals to the model of the reader.</u>

Figure 15: Characteristics of the RST Structure Construction Process

Table 1: Major Differences between TEXT and RST

<u>TEXT</u>	<u>RST</u>
Structure Derived from Previous Texts	**Structure Derived from Goal Pursuit**
Diversity of Operations Encoded in Schemas	Diversity of Processes
Narrow Task	Text Diversity
Limited\High Linguistic Utility	Linguistic Utility
Implemented	Developmental

The first type includes fixed text and fill-in-the-blanks text as the simplest cases. The second includes many kinds of responsiveness to present needs beyond the sort represented by RST. Neither of these influences is sufficient by itself in a mature account of text structure or construction. The interactions between them are responsible for a major part of the complexity of language and communication.

It is clear that both classes of methods have important places in the text generation technology of the future. However, it is not clear what those places are. In these very early and formative days of text generation research, as it is sorting out its problems, these will form separate but interacting themes in the development of the art.

References

[Bienkowski 86] Bienkowski, Marie A., *A Computational Model for Extemporaneous Elaborations*, Cognitive Science Laboratory, Princeton University, Technical Report CSL 1, September 1986.

[Fox 84] Fox, Barbara A., *Discourse Structure and Anaphora in Written and Conversational English*, Ph.D. thesis, UCLA, 1984. To appear through Cambridge University Press.

[Grimes 75] Grimes, J. E., *The Thread of Discourse,* Mouton, The Hague, 1975.

[Mann 84] Mann, W. C., *Discourse Structures for Text Generation*, USC/Information Sciences Institute, Technical Report RR-84-127, February 1984. Also appeared in the proceedings of the 1984 Coling/ACL conference, July 1984.

[Mann & Thompson 85] Mann, William C. and Sandra A. Thompson, "Assertions from Discourse Structure," in *Proceedings of the Eleventh Annual Meeting of the Berkeley Linguistics Society*, Berkeley Linguistic Society, Berkeley, 1985. Also available as ISI/RS-85-155.

[Mann & Thompson 86] Mann, William C. and Sandra A. Thompson, "Relational Propositions in Discourse," *Discourse Processes* 9, (1), January-March 1986, 57-90. Also available as ISI/RR-83-115.

[Mann & Thompson 87] Mann, William C. and Thompson, Sandra A , "Rhetorical Structure Theory: A Theory of Text Organization," in Livia Polanyi (ed.), *Discourse Structure*, Ablex, Norwood, N.J., 1987. To Appear.

[Matthiessen & Thompson 86] Matthiessen, Christian and Thompson, Sandra A., "The Structure of Discourse and "Subordination"," in Haiman and Thompson (eds.), *Clause Combining in Grammar and Discourse*, Benjamins, Amsterdam, 1986. To Appear

[McDonald 80] McDonald, D. D., *Natural Language Production as a Process of Decision-Making Under Constraints*, Ph.D. thesis, Massachusetts Institute of Technology, Dept. of Electricial Engineering and Computer Science, 1980. To appear as a technical report from the MIT Artificial Intelligence Laboratory.

[McKeown 85] Kathleen R. McKeown, *Studies in Natural Language Processing. Volume 2: Text generation: Using discourse strategies and focus constraints to generate natural language text,* Cambridge University Press, Cambridge, 1985.

[Noel 86] Noel, Dirk, *Towards a Functional Characterization of the News of the BBC World News Service*. Antwerp, Belgium, 1986. Antwerp Papers in Linguistics, Number 49.

[Paris 86] Paris, Cecile L. and Kathleen R. McKeown, "Discourse Strategies for Descriptions of Complex Physical Objects," in Gerard Kempen (ed.), *Proceedings of the Third International Workshop on Text Generation*, Nijmegen, The Netherlands, August 1986.

[Thompson & Mann 87] Thompson, Sandra A. and William C. Mann, "Antithesis: A Study in Clause Combining and Discourse Structure ," in *Nominal Book Title*, Nominal Press, 1987. Submitted for inclusion in a festschrift book. Publication to be announced.