

NARRATIVE SUPPORT FOR TECHNICAL DOCUMENTS

Formalising Rhetorical Structure Theory

Nishadi De Silva, Peter Henderson

School of Electronics & Computer Science, University of Southampton, Southampton, UK
nhds03r@ecs.soton.ac.uk, ph@ecs.soton.ac.uk

Key words: Document narratives, Technical documents for BPR, Rhetorical Structure Theory, XML

Abstract: Business Process Re-engineering (BPR) is an area that requires a lot of technical documents and an important feature of a well-written document is a coherent narrative. Even though computer software has helped authors in many other aspects of writing, support for document narratives is almost non-existent. Therefore, we introduce CANS (Computer-Aided Narrative Support), a tool that uses Rhetorical Structure Theory to enhance the narrative of a document. From this narrative, the tool generates questions to prompt the author for the content of the document. CANS also allows the author to explore alternative narratives for a document. A catalogue of predefined narrative structures for popular types of documents is provided too. Our tool is still in its rudimentary stages but sufficiently complete to be demonstrated.

1 INTRODUCTION

Written communication is an integral part in many fields of work and study. BPR is an area in particular that requires a lot of technical documents.

A fundamental aspect of a document is the ‘story’ it conveys to the reader. This is referred to as a document’s narrative. A coherent, well-structured narrative will convey the information better and be more convincing. With the advancement of technology, software support for the writing process has been manifold. However, computer support for document narratives is almost non-existent.

There are many theories for the structure of a narrative. We have studied Rhetorical Structure Theory (RST) [Mann & Thompson, 1988] to build a tool which will help authors construct a document with a more coherent, convincing narrative.

CANS (Computer-Aided Narrative Support) allows the author to build, modify and create instances of a narrative for a document. The tool uses this narrative to generate a set of questions that prompts the author for the document’s content. More importantly, CANS also enables the author to explore alternative narrative structures for an important technical document.

This paper describes RST, introduces our tool and discusses further enhancements. We have also looked at other tools that aid the writing process and a brief overview of these are given in section 4.

2 OVERVIEW OF NARRATIVE THEORIES

Studies into narratives have existed for over a century. Many narratologists have identified structures that are optimal for specific genres of writing [e.g. Propp 1928]. For instance, as early as 1863, the German journalist and writer, Gustav Freytag, introduced a five-part pyramidal structure which he believed to be the most successful format for a play [Freytag 1863]. Even formats for technical documents have often been defined [e.g. Paradis and Zimmerman, 2002]. However, defined formats alone do not complete a document. It is important to construct a coherent narrative too.

Many researchers have studied the coherence of narratives in general and with each theory, new notations and understandings of narratives have emerged [Lehnert 1981, Grosz & Sidner 1986, Grosz et al. 1995]. Among them is the Rhetorical Structure Theory (RST) [Mann & Thompson, 1988]. RST is one of the most popular discourse theories of

the last decade [Marcu 2000] and is explained in more detail below.

2.1 Rhetorical Structure Theory

This theory uses relationships between segments of text to explicate the coherence of a narrative.

In RST, a text segment assumes one of two roles in a relationship: the nucleus (N) or satellite (S). Nuclei express what is more essential to the understanding of the narrative than the satellites. The size of a text segment is arbitrary but each should have independent functional integrity.

Relations hold between non-overlapping text segments and are of two kinds: *hypotactic* and *paratactic*. Hypotactic relations connect one nucleus and one satellite. Paratactic relations connect one nucleus and one or more satellites. There are 23 relations defined in Mann & Thompson's original paper. Two of them are illustrated below. In these diagrams, the arrow always points towards the nucleus in a hypotactic relationship.

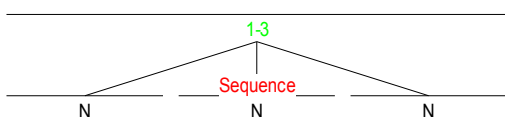


Figure 1: A paratactic relationship

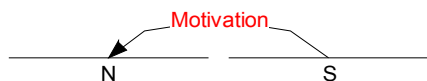


Figure 2: A hypotactic relationship

Text coherence arises due to an overall effect associated with each relation. For instance, in a MOTIVATION relation, the satellite presents some information that increases the reader's desire to perform the action presented in the nucleus.

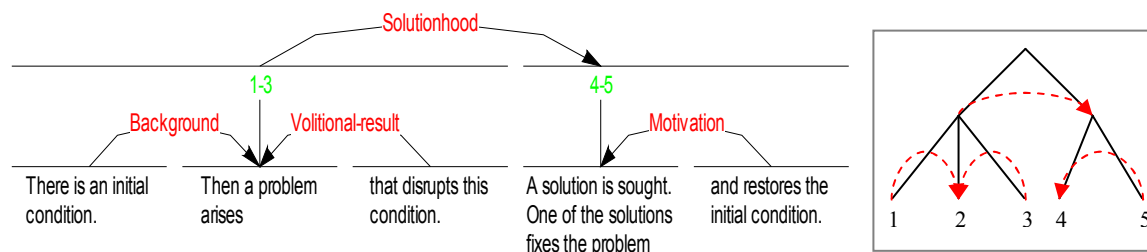


Figure 3: A possible RST tree for the narrative for a simple story (left). A more traditional tree view (right).

Generally, a relation is not expected to dictate the order of the text spans. However, after analysing many texts, Mann & Thompson (1988) identified patterns for the order of the nucleus and satellite for some relations (reproduced below).

Satellite before Nucleus	
Antithesis	Conditional
Background	Justify
Concessive	Solutionhood
Nucleus before Satellite	
Elaboration	Purpose
Enablement	Restatement
Evidence	

Table 1: Order of text spans for some relations [Mann & Thompson, 1988]

In order to illustrate how we apply RST to our work and to explain the theory further, we produce the narrative below for a very simple story.

[There is an initial condition.]¹
 [Then a problem arises]² [that disrupts this condition.]³
 [A solution is sought.
 One of the solutions fixes the problem]⁴
 [and restores the initial condition.]⁵

For a coherent narrative, RST is expected to produce a tree of relations. It is possible to have several valid RST trees for a narrative. One possible RST tree for the narrative above is given below. A more traditional tree diagram also appears on the right with the RST relations superimposed in red.

To make RST-annotated corpus¹ data readable by both humans and machines, Reitter & Stede (2003) introduced URML (Underspecified Rhetorical Markup Language). This is an XML-based data format that allows for the definition of relations between text spans. Below is the URML for part of the RST tree above.

```
<hypRelation id="A" type="Volitional-
  result">
  <satellite id="3" />
  <nucleus id="2" />
</hypRelation>
<hypRelation id="B"
  type="Background">
  <satellite id="1" />
  <nucleus id="2" />
</hypRelation>
```

The simple story below is an example of an instance of the narrative structure above.

Fido is a happy dog.
 Last week Fido got fleas and started scratching. This made Fido unhappy.
 Noticing this, Fido's owner took him to the vet.
 The vet recommended a flea treatment which got rid of the fleas. Fido stopped scratching and was happy again!

It is possible to narrate the same story in several different ways. An alternative narrative is given below (produced by visiting the nucleus first in every relationship of the tree in Figure 3).

Fido's owner took him to the vet.
 The vet recommended a flea treatment which got rid of Fido's fleas.
 Then Fido stopped scratching and was happy again!
 Last week Fido got fleas and started scratching.
 Fido is usually a happy dog but the scratching made Fido unhappy.

3 CANS (Computer-Aided Narrative Support)

We use RST to help the author enhance the document narrative. After the narrative is created, CANS generates a sequence of questions that prompts the user for the document's content. An author can also investigate alternative narratives that better suit the document. These features are elaborated in the following sections. Our tool is still rudimentary and is very much a work in progress. CANS is implemented using JSP [Hall, M. & Brown, L., 2004] and XSLT [Kay, M., 2002]. Central to this tool is an XML database. The user interface is in HTML.

3.1 Creating the narrative structure

The writing process begins by constructing a narrative for the document and producing a RST tree for it. This can be done by typing the narrative, breaking it into segments and defining the relations between these segments. By defining these relations, the existence of each text segment is justified and it is easy to identify segments that are unnecessary or out of place.

This functionality is successfully provided by the free software tool, RSTTool [O'Donnell, 2000]. RSTTool has also been used to produce the diagrams in this paper. We might consider incorporating this tool as part of our work. RSTTool, however, produces .rs3 files which are also in an XML format, but different to URML. We are currently working on an XSL stylesheet that can transform this format to URML.

To demonstrate how our tool can be used by a technical author, we present the narrative below. It was created to represent the typical 'story' of a Research Proposal. The italicized phrases are expected to expand to a section in the Research Proposal and are used in section 3.3 to discuss alternative narratives. We have drawn a RST tree for this narrative and a collapsed version of it is illustrated in Figure 4.

The narrative structures thus created are stored using URML (see section 2.1) in the XML database.

¹ Corpus: A collection of writings or recorded remarks used for linguistic analysis

[We want you to fund us]¹ [because we will achieve these *objectives/results*.]² [We believe these results are important to you]³ [because of *benefits-to-beneficiaries*]⁴ [and to the whole world]⁵ [because there exists an *unsolved-problem*.]⁶ [We know this is unsolved]⁷ [because we have studied the *background*.]⁸ [We will solve this problem]⁹ [by this *method*.]¹⁰ [We know this is the best method]¹¹ [because we have studied *alternative-methods*.]¹² [To achieve this, we will need *total-time*]¹³ [and these *resources*]¹⁴ [because *justification-of-resources*.]¹⁵ [The research will be carried out by these *researchers*]¹⁶ [and they are the most qualified to do this because *justification-of-researchers*.]¹⁷ [The research will be conducted at these *locations*]¹⁸ [because *justification-of-locations*.]¹⁹

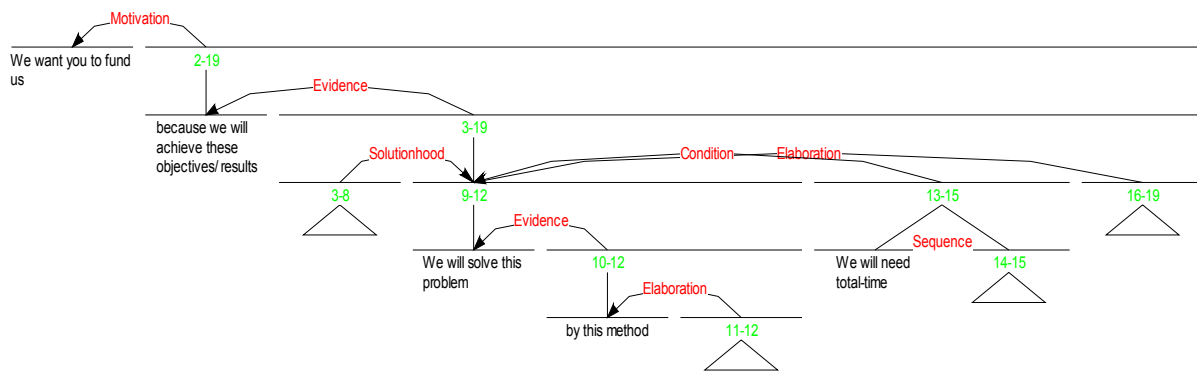


Figure 4: A RST tree for the Research Proposal narrative (collapsed version)

3.2 Generating the questions from the narrative structure

During the second stage of the writing process, the user can select a narrative from a list, along with a mode of traversing the RST tree for this narrative (explained in section 3.3).

At the moment, the questions are relatively simple; there is a question generated for every segment in the narrative. We hope to improve this in the future. Preceding the question is a history of its relations to other segments, so that the author

can realise how the content in the answer integrates with the rest of the document. For instance:

(Motivation:: We want you to fund us)
What are the OBJECTIVES/RESULTS?

The user can type the answers in HTML text areas and save the content in the XML database. Later on, other narrative structures can be applied to this same content to transform it to different documents.

Narrative 1	Narrative 2
Objectives/Results	Objectives/Results
Benefits-to-beneficiaries	Methods
Background	Alternative-methods
Unsolved-problem	Total-time
Total-time	Resources
Justification-of-resources	Justification-of-resources
Resources	Researchers
Methods	Justification-of-researchers
Alternative-methods	Locations
Justification-of-researchers	Justification-of-locations
Researchers	Benefits-to-beneficiaries
Justification-of-locations	Unsolved-problem
Locations	Background

Figure 5: Outline of narratives from traversal method 1 (left) and traversal method 2 (right)

3.3 Exploring alternative narratives

The narrative of a technical document often needs to be altered to suit the reader.

For example, the narrative of a proposal pitched to an audience of investors needs an explanation of how the technical plan achieves something that others cannot. The story should convince the investors that the customers will be willing to pay for it. Such a proposal should contain a clear definition of costs and time requirements, along with evidence to show that the research team is capable of using the investors' money wisely. In contrast, a proposal read by other researchers in the field, should enhance the understanding of the unsolved problem and the chosen method of solution [Paradis & Zimmerman, 2002].

Alternative narratives are produced by traversing the RST tree in different ways. For now, there are two traversal methods, each producing a different sequence of questions for the user. The first method visits the nucleus and satellite in an order dictated by the name of the relationship (see Table 1). The second method always processes the nucleus before the satellite for every relationship. To make the traversal easier, the RST tree in Figure 4 was converted to a binary tree. Figure 5 shows the outlines of the narratives produced by each method using just the italicized phrases in the Research Proposal narrative.

More traversal methods will be investigated.

3.4 Viewing the narrative structure

While typing the answers to the questions, the user has the option to view the current narrative structure in either a tree format or as a textual narrative.

3.5 Predefined narrative structures

There is a list of predefined narrative structures for popular types of documents provided by the tool. This list is expected to grow as more research is done into document narratives. For now, we hope to remain within the domain of technical writing.

4 RELATED WORK

In this section we briefly describe a few existing tools that help authors with writing and list some of their features so as to differentiate them from our work.

a) New Novelist software

This software (purchased from www.amazon.co.uk) helps a novice write a novel in 12 steps. The user is asked to choose the genre of the novel, define characters, add attributes to these characters and fill in templates for the content. Each genre has a fixed sequence of sections that fits most novels in that genre, along with the optimum number of pages for each section.

b) ActiveDocs Document creation

Active Docs provides templates for the automatic creation of documents such as Sales Proposals and Lease Agreements by prompting the user for essential information. It has an HTML interface and supports many popular document formats [ActiveDocs, Document Automation Solutions].

c) WiCKEd

This is a prototype tool to assist document authoring in the Semantic Web context [Woukeu, et al. 2004]. As an example, they present the process of writing a research proposal. While the user types in the provided text editor, the tool continuously analyses this text to recognise known words. These words are then used to find relevant information for the proposal on the intranet.

d) Several tools exist that detect RST relations in a given text [Mahmud 2004] and few others make use of RST to enhance the quality of the produced text. For instance, Rizzo et al. (2002) describe a tool that uses RST to produce rhetorically-structured digital puppet presentations.

e) ArtEquAkt

This tool [Kim S., et al. 2002] uses knowledge acquisition and analysis techniques to extract information from web pages on a given subject domain and creates a knowledge base overlaid with an ontology. The ontology can then be used to construct stories by using story templates.

5 CONCLUSIONS AND FUTURE WORK

CANS is still in need of many improvements to its user interface and functionality. Several specific improvements are discussed in this section.

A prominent feature of this tool is the ability to explore different narratives for a document. However, as illustrated by the two simple stories about Fido in section 2.1, a change in the narrative structure requires a change in the words of the sentences. We hope to improve our tool, in a way less pedantic than Natural Language Processing, to mimic this alteration of words so that the alternative narratives remain coherent.

Other traversal methods of a RST tree will be researched along with ways of producing different RST trees for the same narrative. We can get some useful ideas from Marcu (2000) about exploring all valid RST trees for a given text. A further enhancement would be to allow the combination of RST trees so that several narratives could be merged into one document.

Currently the XML database is maintained using the Java API for XML processing. We have studied Xindice as an alternative [Apache Xindice, 2001] and hope to start using it soon. We are also considering other XML formats that can be used to store the narrative structures instead of URML.

We will also implement the ability to define new relations, apart from those specified by RST.

Most deliverables in a technical environment are in the form of various kinds of factual genres. The challenge in our work is to understand narrative forms and then to transform them into professionally acceptable technical documents. We believe this tool is useful because it encourages an organisation of thought and structure which is considered essential for good writing. Our studies show that this feature is absent in most other writing tools. In particular, we hope that the ability to explore alternative, coherent narratives for a document will be helpful for technical authors in BPR.

REFERENCES

- ActiveDocs, Document Automation Solutions.* (n.d.) Retrieved June 8, 2004, from <http://www.activedocs.com/>
- Apache Xindice.* 2001. Retrieved November 11, 2004 from <http://xml.apache.org/xindice/>
- Freytag, G., 1863. *Freytag's technique of the drama.* Benjamin Blom. New York and London, translated from the 6th German edition by Ellias J. MacEwan in 1968.
- Grosz, B. & Sidner, C., 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12, 3, 175-204.
- Grosz, B., Joshi, A. & Weinstein, S., 1995. Centering: A Framework for Modelling the Local Coherence of Discourse. *Computational Linguistics*. 21, 2, 203-225.
- Hall, M. & Brown, L., 2004. *Core Servlets and JavaServer Pages.* Prentice Hall. USA. 2nd edition.
- Kay, M. 2002. *XSLT.* Wrox Press. Canada. 2nd edition.
- Kim, S., et al., 2002. Artequakt: Generating Tailored Biographies from Automatically Annotated Fragments from the Web. In *Proceedings of Workshop on Semantic Authoring, Annotation &*

- Knowledge Markup (SAAKM'02)*, pp: 1-6, Lyon, France.
- Lehnert, W., 1981. Plot Units: A Narrative Summarization Strategy. In *Strategies for Natural Language Processing*, 375-412, edited by Lehnert & M. Ringle in 1982. New Jersey: Lawrence Erlbaum Associates.
- Mahmud, R., 2004. Revealing Discourse Relations Structure: an Approach for a Dynamic Computer Aided Writing. *Computers and Writing conference 2004*. Hawaii.
- Mann, W. & Thompson, S., 1988. Rhetorical Structure Theory: Toward a functional theory of text organisation. *Text*, 8:3:243-281
- Marcu, D. 2000. *The Theory and Practice of Discourse Parsing and Summarization*. The MIT Press.
- O'Donnell, M., 2000. RSTTool 2.4 – A markup tool for Rhetorical Structure Theory. In *Proceedings of International Natural Language Generation Conference (INLG'2000)*, 253-256, Mitzpe Ramon, Israel.
- Paradis, J. and Zimmerman, M., 2002. *The MIT Guide to Science and Engineering Communication*. The MIT Press. 2nd Edition.
- Propp, V., 1928. *Morphology of the Folktale*, (pp:25-65), University of Texas Press. Austin, 2nd edition.
- Reitter, D. & Stede, M. 2003. Step by step: underspecified markup in incremental rhetorical analysis. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03)*, Budapest.
- Rizzo, P., et al. 2002. An Agent That Helps Children to Author Rhetorically-Structured Digital Puppet Presentations. In *Proceedings of the 6th International Conference on Intelligent Tutoring Systems*, pp:903-912.
- Woukeu, A., Carr, L. and Hall, W. 2004. WiCKEd: A Tool for Writing in the Context of Knowledge. In *Proceedings of Hypertext 2004 - Fifteenth ACM Conference on Hypertext and Hypermedia* (in press), University of California, Santa Cruz, USA.