2004－ICS－138　(36)
2004／12／7

Using Syntactic Structures and Cohesive Devices in Recognizing Discourse Structure of Text

Huong LeThanh
Faculty of Information Technology,
Hanoi University of Technology
1 Dai Co Viet, Hanoi, Vietnam
huonglt@it-hut.edu.vn

Geetha Abeysinghe
School of Computing Science,
Middlesex University
The Burroughs, London, NW4 4BT, UK
G.Abeysinghe@mdx.ac.uk

This paper presents a system for automatically generating the discourse structure of text. The system is divided into two levels: sentence-level and text-level. At the sentence-level, the discourse analyzer uses syntactic structures and cue phrases to derive discourse structures of sentences. This approach prevents combinatorial explosions while still get accurate analyses. At the text-level, constraints about textual adjacency and textual organization are integrated in a beam search to reduce the search space of the discourse analyzer and generate the best discourse structure. Two new factors using to recognize discourse relations are proposed: noun-phrase cues and verb-phrase cues. Our experiments show that this system achieves a good performance compared to existing discourse analyzing systems.

1 Introduction

The current boom in information technology has produced an enormous amount of information. From the abundance of information available, getting the information that we need is not an easy task. For example, a doctor needs information about a specific disease. He looks for such information from medical digital libraries. What he needs is a document that summarizes all information from this search. Obviously, we do not have the time to read every document presented by a search engine (e.g., Google) to find the most relevant documents and then summary them. Therefore, effective methods of text extraction and text summarization are necessary. Most existing text summarization systems use statistical techniques to extract key sentences or paragraphs from an article (Rau et al., 1994; Mitra et al., 1997). However, this approach often provides incoherent results. A new trend in text summarization solves the incoherence problem by using discourse strategies (Rino and Scott, 1994; Marcu, 2000), which analyze the coherence of a text by a discourse structure that describes discourse relations between different parts of a text. This new approach has proved that using discourse strategies achieves better results than using other strategies.

An example of a discourse structure, which is based on the Rhetorical Structure Theory (RST) proposed by Mann and Thompson (1988), is given in Figure 1. The leaves of the discourse tree correspond to *elementary discourse units* (edus), whereas the internal tree nodes correspond to larger text spans. Each internal tree node represents a discourse relation (*Sequence*, *Circumstance*) that holds between two adjacent, non-overlapping spans. Each span in a discourse relation is a *nucleus* (N) or a *satellite* (S). The nucleus plays a more important role than the satellite in respect of the writer's intention. If both spans have equal roles, they are both considered as *nuclei*.

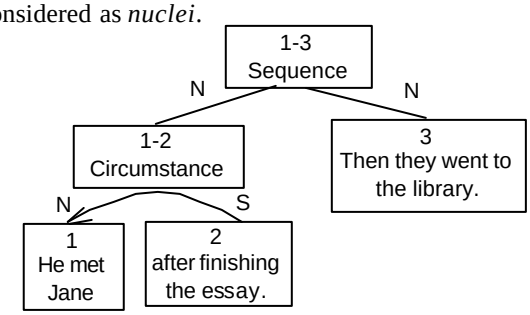

Fig. 1. The Discourse Structure of a Text

Although discourse structure has been found to be useful in many fields of text processing (Rutledge et al., 2000; Torrance and Bouayad-Agha, 2001), only a few algorithms for implementing discourse analyzers have been proposed so far. Most research in this field concentrates on specific discourse phenomena (Schiffrin, 1987; Litman and Hirschberg, 1990). The amount of research available in discourse segmentation is considered small; in constructing discourse structures it is even smaller. The performances of existing discourse systems are still low. Realizing the lack of discourse systems and the great demand for text processing applications, we have carried out research in discourse analysis, aiming to construct a Discourse Analyzing System (DAS) that automatically derives discourse structures of text.

In this paper, different factors were investigated to achieve a better discourse system. Sentential syntactic structures and cue phrases are applied to get an accurate discourse segmenter. Two new recognition factors,

noun-phrase cues and verb-phrase cues, are proposed to recognize discourse relations. With a given text and its syntactic information, the search space in which well-structured discourse trees of a text are produced is minimized.

The rest of this paper is organized as follows. The discourse segmentation process is described in Section 2. Different recognition factors used in DAS are introduced in Section 3. Section 4 presents our method to posit discourse relations between spans. A brief description of the process to construct discourse trees is given in Section 5. In Section 6, we describe our experiments and discuss the results we have achieved so far. Section 7 concludes the paper and proposes possible future work.

2 Discourse Segmentation

The purpose of discourse segmentation is to split a sentence into edus, which are clauses or clause-like units with independent functional integrity. DAS solves this task by using two processes: discourse segmentation by syntax (Step 1) and discourse segmentation by cue phrases (Step 2).[1] The input of Step 1 is a sentence and its syntactic structure; documents from the Penn Treebank (1999) were used to get the syntactic information. A syntactic parser is going to be integrated into our system in future work. Based on the syntactic structure of a sentence, Step 1 splits the sentence into clauses, which are considered as discourse segments, and initiates discourse relations between them. This process is illustrated by Example (1):

(1) Mr. Silas Cathcart built a shopping mall on some land *he owns.*

In Example (1), DAS splits the clause "*he owns*" from the noun phrase (NP) "*some land he owns*". Then, DAS detects that "*Mr. Silas Cathcart built a shopping mall on*" is not a complete clause without the NP "*some land*". Therefore, these two spans are combined into one. The sentence is now split into two segments "*Mr. Silas Cathcart built a shopping mall on some land*" and "*he owns.*" Next, the discourse relation between these two discourse segments is initiated. The name of this relation and the span nuclearity (nucleus or satellite) are determined later in a relation recognition-process (see Section 4).

Since several NPs are considered as edus when they are accompanied by a strong cue phrase, DAS needs to carry out another segmentation process to solve these cases. This process is called discourse segmentation by cue phrase (Step 2). This process searches for a strong cue phrase in each discourse segment generated by Step 1. When a strong cue phrase is found, the algorithm splits the discourse segment into two edus: one edu is the NP that contains the strong cue phrase, and another edu is the rest of the discourse segment. An example of this process is shown in Example (2) below.

(2) [*According to* a Kidder World story about Mr. Megargel,] [all the firm has to do is "position ourselves more in the deal flow."]

Similar to Step 1, Step 2 also initiates discourse relations between edus that it derives. The relation name and the span nuclearity are posited later in a relation recognition-process.

3 Factors Used in Recognizing Relations

In this research, we applied several recognition factors that have been used by other researchers such as cue phrases (Marcu, 2000; Forbes et al., 2003), VP-ellipsis (Kehler and Shieber, 1997), and proposed two new recognition factors -- noun-phrase cues and verb-phrase cues.

3.1 Cue Phrases, Noun-Phrase Cues, and Verb-Phrase Cues

Cue phrases (e.g., *however*, *as a result*), also called discourse connectives, are words or phrases that connect spans. They are the most simple and obvious means of signaling discourse relations because of two reasons. First, they explicitly express the cohesiveness among textual units. Second, identifying cue phrases is essentially based on pattern matching. The cue phrase "*when*" in Example (3) determines a *Circumstance* relation between two clauses "*He was staying at home*" and "*the police arrived*".

(3) [He was staying at home][*when* the police arrived.]

In addition to cue phrases, we proposed two new recognition factors -- noun-phrase cues (NP cues) and verb-phrase cues (VP cues) -- as shown in Examples (4) and (5) below:

(4) [New York style pizza meets Californian ingredients,][and the *result* is the pizza from this Church Street pizzeria.]

(5) [By the end of this year, Chairman Silas Cathcart retires to his Lake Forest, Ill., home.][And that *means* Michael Carpenter will for the first time take complete control of Kidder.]

The noun "*result*" indicates a *Result* relation in Example (4); whereas the verb "*means*" signals an *Interpretation* relation between two sentences in Example (5). The phrases in the main NPs (i.e., subject or object) of a sentence that signal discourse relations are called NP cues. The phrases in the main verb phrase

[1] A detailed description of these segmentation processes is described in LeThanh et al. (2004).

(VP) of a sentence that signal discourse relations are called VP cues.

Unlike the cue phrases that are identified based on pattern matching, NPs or VPs have to be stemmed before being compared with the NP or VP cues. The sets of NP cues and VP cues were created by us, basing on our research on different linguistic resources and on the RST Discourse Treebank (RST-DT, 2002). A detailed description of the application of these recognition factors in DAS is discussed in Section 4.

3.2 Syntactic Information

According to Matthiessen and Thompson (1988), clausal relations reflect discourse relations within a sentence. For example, the discourse relation between a main clause and its subordinate clause is an asymmetric relation, in which the main clause is the nucleus and the subordinate clause is the satellite. This proposal is applied in DAS to posit the span nuclearity and to eliminate unsuitable relations. Syntactic information can also be used to find relation names. For example, the reporting and reported clauses of a sentence are considered as the satellite and the nucleus in an *Elaboration* relation, as in Example (6):

(6) [*Mr. Carpenter says*][that Kidder will finally tap the resources of GE.]

In Example (6), the reporting clause "*Mr. Carpenter says*" is considered as the satellite, whereas the reported clause "*that Kidder will finally tap the resources of GE*" is considered as the nucleus.

3.3 Other Recognition Factors

Besides cue phrases, NP cues, VP cues, and syntactic information, which are the most significant factors to recognize discourse relations, other cohesive devices are also used in DAS. They are time references, reiterative devices, substitution words, and ellipses. Among reiterative devices, word repetition and synonyms are used to detect discourse connections and relation names. For example, a *Contrast* relation often occurs when most words in two spans are similar and one span contains the word "*not*". A multinuclear relation (*Contrast*, *List*) often exists between spans whose main NPs are co-hyponyms or antonyms.

Substitution word is a place-holding device, where the missing expression is replaced by a special word (*one*, *do*, *so*, etc.) in order to avoid the repetition. Ellipsis is a special form of substitution word where a part of a sentence is omitted. By replacing or omitting words that have already been used, a strong link is created between one part of the elided text and another. While reiterative devices can be distant from their antecedents, the substitution words only refer to the entities or the actions that have just been mentioned. Therefore, substitution words are used for local focus.

The reiteration devices are recognized by using a thesaurus called WordNet (2004). Meanwhile, the substitution words and ellipses are detected by analyzing the syntactic information of sentences.

4 Relation Recognition

4.1 Recognizing Discourse Relations between Edus

Heuristic rules are used in DAS to recognize discourse relations between edus. These rules are the applications of recognition factors to a specific relation. For example, the heuristic rule that is used to recognize a *List* relation "Unit_2 contains *List* cue phrases" (Section 4.1.2) is the application of the recognition factor *cue phrases*. A description of the process to recognize the *List* relation presenting in Section 4.1.2 will further illustrate this idea. Before that, let us introduce our method of scoring heuristic rules and computing the score of a relation based on all evidences that contribute to the recognition of that relation.

4.1.1 Scoring Heuristic Rules

Cue phrases, NP cues, VP cues, and cohesive devices have different strengths in recognizing discourse relations. The cue phrases explicitly signal discourse relations most of the time. Meanwhile, ellipses can only create a link between spans and cannot determine a relation name. Therefore, the heuristic rules using cue phrases are stronger than the heuristic rules using ellipses. To control the influence of these factors to the relation recognition, each heuristic rule is assigned a heuristic score. The rules involving cue phrase have the highest score of 100 because cue phrase is the strongest factor to signal relations. NP cues and VP cues are also strong factors but weaker than cue phrases since they do not express relations in a straightforward way like cue phrases. As a result, the heuristic rules involving NP cues and VP cues are assigned a score of 90. The heuristic rules corresponding to the remaining recognition factors receive scores ranging from 20 to 80 since these factors are weaker than NP cues and VP cues. At present, heuristic scores are assigned by human linguistic intuitions and optimized by a manually training process.

In this research, we separate two types of scores: the score of a heuristic rule and the score of a specific cue phrase, NP cue, and VP cue. The heuristic rule involving cue phrases has the score of 100, which means DAS is 100% certain that the relation signaled by the cue phrase holds. However, it is only correct when that cue phrase

explicitly expresses a relation. In fact, each cue phrase has a different level of certainty in signaling relations. The cue phrase "*instead of*" always signals an *Antithesis* relation; whereas the cue phrase "*and*" may signal a *List*, *Sequence*, or *Elaboration* relation. That means the cue phrase rule that applies to the cue phrase "*and*" is not 100% certain that a *List* relation holds. Therefore, the score of a cue phrase rule should be reduced when this rule is applied to a weak cue phrase. Since the score of a cue phrase is between 0 and 1, DAS computes the actual score of a heuristic rule involving cue phrases as follow:

Actual-score(heuristic rule)
= Score(heuristic rule) * Score(cue phrase).

This treatment is also applied to NP and VP cues. The actual score of a heuristic rule involving a NP or VP cue is:

Actual-score(heuristic rule)
= Score(heuristic rule) * Score(NP cue or VP cue).

The actual score of other heuristic rules that do not involve cue phrase, NP or VP cue is:

Actual-score(heuristic rule) = Score(heuristic rule)

If several heuristic rules of a relation are satisfied, the score of that relation will be the total scores of all factors contributing to that relation.

Total-heuristic-score = $\sum$Actual-score (heuristic rule)

DAS seeks the recognition factors in the following order: cue phrases, NP cues, VP cues, and the remaining recognition factors. A discourse relation will be posited if the *total-heuristic-score* of this relation is greater than or equal to a threshold θ. The threshold is assigned the score of 30 (compare to 100 as the maximum score of a heuristic rule), as by experiments we found that recognition factors can be very weak in many cases. A sample of the recognition process representing by heuristic rules to recognize a *List* relation is introduced next.

4.1.2 *List* Relation

A *List* relation is a multi-nuclear relation whose elements can be listed. The heuristic rules for the *List* relation between two edus, $Unit_1$ and $Unit_2$ ($Unit_1$ precedes $Unit_2$), are shown below:

1. $Unit_2$ contains *List* cue phrases. Score : 100
2. Both units contain enumeration conjunctions (*first, second, third, etc*). Score: 100
3. Both subjects of $Unit_1$ and $Unit_2$ contain NP cues. Score: 90
4. If both units contain attribution verbs, the subjects of their reported clauses are similar, synonyms, co-hyponyms, or hypernyms/hyponyms. Score: 80

We apply the rules to recognize the *List* relation to Example (7).

(7) [Mr. Cathcart is credited with bringing some basic budgeting to traditionally free-wheeling Kidder.$_{7.1}$] [He *also* improved the firm's compliance procedures for trading.$_{7.2}$]

In Example (7), the cue phrase "*also*" signals a *List* relation between the sentences (7.1) and (7.2). Since only the heuristic rule 1 is satisfied, the total-heuristic-score is:

Total-heuristic-score = Actual-score(heuristic rule 1)
= score(heuristic rule 1) * score("*also*").

The cue phrase "*also*" has the score of 1 for the *List* relation, so the total-heuristic-score is 100*1 = 100 > θ. Therefore, a *List* relation is posited between the sentences (7.1) and (7.2).

4.2 Recognizing Discourse Relations between Large Spans

Discourse relations between large spans are converted to discourse relations between their edus. This conversional rule is proposed by us as follow:

> "*Discourse relations between two large spans are recognized either by relations that hold between the nuclei of these spans or by the relations that are signaled by unused cue phrases in the left most edus of these large spans.*"

This rule is illustrated by Example (8) shown below.

(8) [With investment banking as Kidder's "lead business," where do Kidder's 42-branch brokerage network and its 1,400 brokers fit in?$_{8.1}$][*To answer the brokerage question,*$_{8.2}$] [Kidder, in typical fashion, completed a task-force study.$_{8.3}$]

In Example (8), the VP cue "*to*" is used to recognize a *Purpose* relation between (8.2) and (8.3), in which the span (8.2) is the satellite and the span (8.3) is the nucleus. The VP cue "*answer*" of the span (8.2) has not been used for the relation between (8.2) and (8.3). Therefore, it is used to signal a *Solutionhood* relation between the spans (8.1) and (8.2-8.3). In case of no cue phrase remaining in the satellite (8.2) and no cue phrase in the nuclei (8.1) and (8.3), other recognition factors will be checked from (8.1) and (8.3) to posit discourse relations between the two sentences.

5 Constructing Discourse Trees

Constructing discourse trees of a text can be considered as the problem of searching for the combination of discourse relations that best describes the text, given all possible relations that hold between spans. In order to take advantages of the clausal relations within a sentence, we divide the task of constructing discourse trees of a text into two levels: sentence-level (Section

5.1) and text-level (Section 5.2), each of which is processed in a different way.[2]

5.1 Constructing Discourse Trees at the Sentence-level

This module takes the output of the discourse segmenter as the input and generates a discourse tree for each sentence. As mentioned in Section 2, the discourse segmenter has already generated edus and information about discourse relations between edus. The sentence-level discourse analyzer only has to posit relation names and the span nuclearity, and then connects all sub-trees within one sentence into one tree. Syntactic information and cue phrases are the main recognition factors for the recognition process at the sentence-level. An example of the role of syntactic information in positing discourse relations is shown in Example (6) (Section 3.2). Example (9) illustrates the use of cue phrases in positing discourse relations.

(9) [He came late] [*because of* the traffic.]

The cue phrase "*because of*" signals a relation between the clause containing this cue phrase and its left adjacent clause. The clause containing "*because of*" is the satellite of that relation. When syntactic information and cue phrases are not strong enough to signal discourse relations, other recognition factors discussed in Section 3 are taken into account.

To construct the sentence-level discourse trees, after all relations within a sentence have been posited, DAS connects all sub-trees within one sentence into one tree. All leaves that correspond to another sub-tree are replaced by the corresponding sub-trees. With the presented method of constructing sentential discourse trees based on syntactic structures and cue phrases, combinatorial explosions can be prevented while DAS still gets accurate analyses.

5.2 Constructing Discourse Trees at the Text-level

Given all discourse relations between spans, DAS has to select a set of relations that best describes the text. The chosen relations should connect non-overlapping spans and cover the entire text. This problem can be considered as searching for the best solution of combining discourse relations. An algorithm that minimizes the search space and maximizes the tree's quality needs to be found. We applied a beam search, which is the optimization of the best-first search where only a predetermined number of paths are kept as candidates.

The search space is reduced further by applying constraints of textual organization and textual adjacency. The constraint of textual organization allows only spans within a textual unit (e.g., paragraph, section) being connected. The reason for this process is as follow. Normally, each text has an organizational framework, which consists of sections, paragraphs, etc., to express a communicative goal. Each textual unit completes an argument or a topic that the writer intends to convey. Therefore, a span should have semantic links to spans in the same textual unit before connecting with spans in a different one.

The constraint of textual adjacency is one of the main constraints in constructing a valid discourse structure that are proposed by Mann and Thompson (1988). Since the spans that contribute to a discourse relation must be adjacent, only adjacent spans are considered to be connected in generating new relations.

6 Evaluation

The evaluation of DAS were done by manually training the system on 20 documents from the RST Discourse Treebank (RST-DT, 2002) and then testing on a different set of 40 documents from the same corpus. The syntactic information of these documents was taken from the Penn Treebank (1999), which was used as the input of the discourse segmenter. A set of 22 discourse relations was used in the experiments. The annotated documents from the RST corpus, which were created by humans, were used as the standard discourse trees for our evaluation.

The human performance was considered as the upper bound for our system's performance. This value was obtained by evaluating the agreement between human annotators using 53 double-annotated documents from the RST corpus. The performance of our system and human agreement are represented by precision, recall, and F-score[3]. These values were computed at three levels: segment boundaries (I), sentence-level discourse trees (II), and the discourse trees for the entire text (III). These values are shown in Table 1.

Table 1. DAS Performance Vs. Human Performance

Level		I	II	III
DAS	Precision	90.7	53.4	38.6
	Recall	88.1	51.5	37.6
	F-score	**89.4**	**52.4**	**38.1**
Human	Precision	98.7	69.2	53.0
	Recall	98.8	68.9	52.5
	F-score	**98.7**	**69.0**	**52.7**

In the experiments carried out in this research, the output of one process was used as input to the process following it. The error of one process is, therefore, the

[2] A detailed description of these processes is described in LeThanh et al. (2004).

[3] We use the F-score version in which precision (P) and recall (R) are weighted equally, defined as 2*P*R/(P+R).

accumulation of the error of the process itself and the error from the previous process. As a result, the accuracy of DAS and that of humans decline as the processing level increases. DAS provides a reliable result at the discourse segmentation level (90.7% precision and 88.1% recall). The system's performance at the sentence-level is acceptable when compared with humans. The low accuracies of DAS for the entire text (38.6% precision and 37.6% recall at Level III) indicate that the discourse trees generated by DAS are quite different from those in the corpus. We found that some documents used in these experiments contain incorrect paragraph boundaries. This problem contributes to the error of DAS output at the text-level.

As presented in Table 1, the accuracy of the discourse trees given by human agreement is not high, (52.7% F-score). This is because discourse is too complex and ill defined to generate rules that can automatically derive discourse structures. Different people may create different discourse trees for the same text (Mann and Thompson, 1988). Because of the multiplicity of RST analyses, the discourse parser should be used as an assistant rather than a stand-alone system.

7 Conclusions and Future Work

In this paper, we have presented a discourse analyzing system and evaluated it using the RST discourse corpus. The experiments show that syntactic information and cue phrases are efficient in constructing discourse structures at the sentence-level, especially in discourse segmentation (89.4% F-score). At the text-level, the constraints of textual adjacency and textual organization are integrated in a beam search to reduce the search space. The experiments show that the proposed approach can produce reasonable results compared to human annotator agreements.

At present, the scores used in DAS and the threshold θ are assigned manually based on human linguistic intuitions and optimized by a manually training process. The best method to optimize these scores is to train them by machine learning algorithms. These training processes will be considered in future work. Future work also includes investigating a method to identify the correct boundaries of high level textual units (paragraph, section, etc.). We propose to use an approach of topic segmentation (e.g., Choi, 2000) to solve this problem. We hope this research will aid in the future development of text processing such as text summarization, text understanding, and text extraction.

References

Choi, F. 2000. *Advances in domain independent linear text segmentation*. In Proc. of NAACL'00, Seattle, USA.

Forbes, K., Miltsakaki, E., Prasad, R., Sarkar, A., Joshi, A., and Webber, B. 2003. D-LTAG System: Discourse Parsing with a Lexicalized Tree-Adjoining Grammar. *Journal of Logic, Language and Information*, 12(3), 261-279.

Kehler, A. and Shieber, S. (1997). Anaphoric Dependencies in Ellipsis. *Computational Linguistics*, 23(3):457-466.

LeThanh, H., Abeysinghe, G., and Huyck, C. 2004. *Generating Discourse Structures for Written Texts*. In Proc. of COLING'04, pp.329-335.

Litman, D. and Hirschberg, J. 1990. *Disambiguating cue phrases in text and speech.* In Proc. of COLING-90. Vol 2: 251-256.

Mann, W. and Thompson, S. 1988. Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. *Text*, 8(3): 243-281.

Matthiessen, C. and Thompson, S.A. (1988). *The structure of discourse and 'subordination'*. In Haiman and Thompson (eds.), pp.275-329.

Marcu, D. 2000. *The theory and practice of discourse parsing and summarization.* MIT Press, Cambridge, Massachusetts, London, England.

Mitra, M., Singhal, A., and Buckley, C. 1997. *Automatic text summarisation by paragraph extraction*. In Proc. of the ACL/EACL-97 Workshop on Intelligent Scalable Text Summarisation, pp.31-36, Madrid, Spain.

Penn Treebank. 1999. Linguistic Data Consortium. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC99T42

Redeker, G. 1990. Ideational and pragmatic markers of discourse structure. *Journal of Pragmatics*, 367-381.

Rau, L.F., Brandow, R., and Mitze, K. (1994). Domain-Independent Summarisation of News. In *Summarizing Text for Intelligent Communication*, pp.71-75, Dagstuhl, Germany.

Rino, L.H.M. and Scott, D. 1994. *Automatic Generation of Draft Summaries: Heuristics for Content Selection*. In Proc. of the Third International Conference of the Cognitive Science of Natural Language Processing. Dublin City University, Ireland.

RST-DT 2002. *RST Discourse Treebank*. Linguistic Data Consortium. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002T07.

Rutledge, L., Bailey, B., Ossenbruggen, J., Hardman, L., and Geurts, J. 2000. *Generating Presentation Constraints from Rhetorical Structure.* In Proc. of HYPERTEXT 2000.

Schiffrin, D. 1987. *Discourse markers.* Cambridge: Cambridge University Press.

Torrance, M. and Bouayad-Agha, N. 2001. *Rhetorical structure analysis as a method for understanding writing processes*. In Proc. of MAD 2001.

WordNet. 2004. http://www.cogsci.princeton.edu/~wn/index.shtml