The automatic identification of discourse units in Dutch text

Nynke van der Vliet, Gosse Bouma, Gisela Redeker
University of Groningen, The Netherlands
`n.h.van.der.vliet@rug.nl`, `g.bouma@rug.nl`, `g.redeker@rug.nl`

Abstract
The identification of discourse units is an essential step in discourse parsing, the automatic construction of a discourse structure from a text. We present a rule-based algorithm to identify elementary discourse units (EDUs) in Dutch written text. Contrary to approaches that focus on the determination of segment boundaries, we identify complete discourse units, which is especially helpful for the recognition of interrupted EDUs that contain embedded discourse units. We use syntactic and lexical information to decompose sentences into EDUs. Experimental results show that our algorithm for EDU identification performs well on texts of various genres.

Keywords: discourse analysis, elementary discourse units, segmentation.

1 Introduction

Discourse structures can be useful as input for several other tasks, such as automatic summarization (Marcu, 2000; Thione et al., 2004; Bosma, 2008; Louis et al., 2010), question answering (Verberne et al., 2007) and information extraction (Maslennikov and Chua, 2007). However, the manual analysis of discourse relations in a text is a time-consuming task. The use of discourse relations for other applications thus presupposes that these relations can be added to a text automatically with relatively high accuracy.

This paper addresses the first step in automatic discourse analysis, namely the identification of suitable Elementary Discourse Units (EDUs) in a text. This process involves segmenting the text into sentences, and decomposing complex sentences into smaller units (typically clauses) that express states of affairs and form the basis for discourse analysis. The task thus concerns fine-grained discourse segmentation, not to be confused with segmenting a text into paragraph-size chunks reflecting the topic structure (e.g. Hearst (1997) and Eisenstein (2009)), which is also sometimes called discourse segmentation.

We describe a rule-based approach to identify EDUs in Dutch sentences using syntactic and lexical information, and present the results of applying the algorithm to an annotated corpus that contains texts from four different genres.

2 Related work

Several successful segmentation systems have been developed for English (e.g. Tofiloski et al. (2009), Subba and Di Eugenio (2007) and Bach et al. (2012)), German (Lüngen et al., 2006) and French (Afantenos et al., 2010) written text. Tofiloski et al. (2009) show for their English data that good segmentation results can be obtained by a rule-based approach. For English, the RST Discourse Treebank (Carlson et al., 2002), a substantial corpus segmented and manually annotated for discourse structure, has been widely used for the development of machine learning approaches to discourse segmentation for English text (Soricut and Marcu, 2003; Sporleder and Lapata, 2005; Fisher and Roark, 2007; Subba and Di Eugenio, 2007; Sagae, 2009; Hernault et al., 2010; Bach et al., 2012).

For Dutch, there is insufficient annotated data to apply machine learning techniques. In van der Vliet (2010) we present a basic rule-based segmenter that makes use of syntactic information and punctuation for the automatic identification of segment boundaries in Dutch text. This automatic discourse segmentation algorithm uses the common approach of identifying EDUs by determining the segment boundaries in sentences. A well-known complication for this type of approach is the occurrence of embedded discourse units that interrupt an ongoing EDU, as in sentence (1).

(1) [Echter gedurende de nacht, [die op Mercurius maanden lang kan duren,] daalt de temperatuur tot zo'n -185 graden Celsius,][wat weer tot de laagste in ons zonnestelsel mag worden gerekend.]
[However during the night, [which can last for months on Mercury,] the temperature drops to about -185 degrees Celsius,][which in turn can be counted among the lowest in our solar system.]

In this case the non-restrictive relative clause 'which can last for months on Mercury' is an embedded discourse unit, and the sentence parts before and after this clause together form another discourse unit. As Afantenos et al. (2010) point out, a segmenter that is only able to identify whether there is a boundary or not in the text cannot distinguish between an embedded discourse unit within another discourse unit and three separate subsequent EDUs. Afantenos et al. solve this problem by identifying three types of segment boundaries: boundaries that only start a segment, boundaries that only end a segment and boundaries that do both.

In the segmentation of the RST Discourse Treebank the two parts of an EDU that is split by an embedded EDU are kept as separate units. In the discourse relation annotation process that follows, the two parts of the EDU are linked by the pseudo-relation SAME-UNIT that was introduced by Carlson and Marcu (2001) for this purpose. Carlson and Marcu's approach thus relegates the problem of embedded units to the relational level, which we consider undesirable.[1] In our corpus, discontinuous EDUs are represented as complete EDUs in the segmentation. As explained above, a binary classification algorithm that identifies for each text position whether there is a boundary or not is not sufficient for our purpose. In this paper, we therefore improve on van der Vliet (2010). We identify the EDUs that form the starting point for discourse relation annotation that uses the established set of Rhetorical Structure Theory (RST) relations (originally introduced by Mann and Thompson (1988); see www.sfu.ca/rst). In addition, we add lexical features and apply the algorithm to two more genres, namely popular scientific news texts and advertisements.

3 Principles of segmentation

Our definition of an elementary discourse unit is guided by the question of whether a discourse relation could hold between the unit and another segment. We identify independent clauses, adverbial clauses and non-restrictive relative clauses (separated by a comma) as EDUs. Sentence fragments (subclausal expressions ending with a period and (sub)headings in the text) are also treated as EDUs. If forward conjunction reduction (see (2)) or gapping occurs in clause-level coordination, we consider both parts of the coordination as EDUs. If VP coordination occurs as in (3) we consider this as two separate propositions about the subject and thus distinguish two EDUs.

(2) [De planeet draait in 58.6 dagen om haar as][en in 88.0 dagen om de zon.]
[The planet turns around its axis in 58.6 days][and around the sun in 88.0 days.] (EE02)

(3) [Dankzij een project van Dark & Light Blind Care is deze aandoening vroegtijdig ontdekt][en behandeld.]
[Thanks to a project of Dark & Light Blind Care this disorder was discovered in time][and treated.] (FL14)

We do not treat restrictive relative clauses, appositives, and complement clauses as separate EDUs. In line with Tofiloski et al. (2009) we do not consider speech parentheticals (e.g. "...", *he said*) as EDUs. For more details about our segmentation principles, see van der Vliet et al. (2011).

4 Implementation

The input for the EDU identification algorithm is a text file containing the text that needs to be segmented. Before applying our segmentation rules we use Alpino (Van Noord et al., 2006), a

[1] Borisova and Redeker (2010) show that the identification of SAME-UNIT pseudo-relations in a boundary-segmented corpus can be problematic.

Dutch dependency parser, to segment the input text into sentences and to create a syntactic tree for each sentence of the text.

We then use the syntactic analysis of the sentences to identify the EDUs for each sentence and produce a text file containing all the EDUs of all sentences. For each sentence we start to identify possible EDUs by applying a set of EDU identification rules. We use two types of rules in our system: EDU rules (see below) that are used to recognize the basis for an EDU, and combination rules that are used to build the complete EDU from the basis by combining it for example with punctuation or prepositional phrases.

EDU rules

1. *Main clauses* are identified by the syntactic category tag *@cat='smain'*. The tag *@cat='sv1'* is used for *imperatives* and *yes/no questions*. Text parts with these two tags are identified as EDUs, unless they start with a verb of reported speech.[2] If they do, they are identified as speech parentheticals (which we do not consider as an EDU on its own in our segmentation) and later combined with the preceding EDU using a combination rule.
2. *Infinitival clauses* are only identified as EDUs if they do not function as a complement. The tag *@cat='oti'* is used for *om te* ('to') - clauses. Only if an *om te*-clause is classified as a modifier with the tag *@rel= 'mod'* is it taken as a separate EDU.
3. *Phrases that start with a subordinating conjunction* are identified by the Alpino tag *@cat= 'cp'*. They are only identified as an EDU if they contain a verb and if they are classified as a modifier with the tag *@rel='mod'*.
4. *Relative clauses* are identified in Alpino with the tags *@cat= 'rel'* and *@cat= 'whrel'*. Text parts with these tags are only identified as an EDU if they are preceded by a comma.[3]
5. *Clauses between parentheses or hyphens* are identified as EDUs.
6. *Complement clauses* are not identified as EDUs. The construction *zo..dat* and *zodanig..dat* (both can be translated as 'so..that') forms an exception to this rule. If the text part that precedes the *dat*-clause contains the word *zo* or *zodanig*, the *dat*-clause is identified as a separate EDU.
7. *Verb-final clauses* are identified with the tag *@cat='ssub'*. These clauses often function as complements and are *not* identified as EDUs. Only when a sentence contains two conjoined verb-final clauses as in (4), the second clause is identified as an EDU.

(4) [Het probleem is echter wel dat Phobos zich onder het minimum van de geschikte synchronisatiehoogte bevindt][en elke eeuw 1.8 meter dichter bij Mars komt te liggen.] *[The problem however is that Phobos is situated under the minimum of appropriate synchronisation hight][and ends up 1.8 meters closer to Mars each century.* (EE02)

The combination rules are used to create complete EDUs by combining the identified EDUs with tokens that directly precede or follow the EDU and should be part of it. For example, punctuation symbols are assigned to the top of the syntactic trees by Alpino; so we use a punctuation rule to combine a punctuation symbol that immediately precedes an EDU with the EDU, and another rule to combine a punctuation symbol that immediately follows an EDU with

[2] We used the following list of verbs of reported speech: zeggen (to say), uitleggen (to explain), vinden(to think, find), vertellen (to tell), beweren (to claim), denken (to think), vermoeden (to suspect), concluderen (to conclude), melden (to report). Note that Dutch requires subject-verb inversion in postposed tags.

[3] In our segmentation we only identify *non-restrictive relative clauses* as EDUs, which Dutch punctuation rules require to be preceded by a comma.

the EDU. In the same way, complementizers and any remaining prepositional phrases and noun phrases in a sentence are incorporated into an adjacent EDU. A separate combination rule is used to combine speech parentheticals with a preceding EDU.

In the next step the possible EDUs, which are strings of tokens (e.g. words, punctuation), are used to determine the EDUs of the sentence. This is done with a straightforward method:

1. An EDU that spans the whole sentence is added to the list of possible EDUs to make sure that there are no sentence parts left out in the identified EDUs of a sentence.
2. If overlapping EDUs (i.e. two EDUs of which the second EDU starts before the end of the first EDU and ends after the end of the first EDU) or duplicate EDUs occur in the list of possible EDUs, the second EDU is removed from the list.
3. Textual overlap in the EDUs in the list is eliminated by removing tokens from the longer EDUs. For each EDU in the list of EDUs, we check if there are EDUs in the list that fall within the boundaries of the EDU under consideration, i.e. EDUs that start at the same word but end earlier, EDUs that start later but end at the same word, and EDUs that start later and end earlier than the EDU under consideration. We then remove the text tokens of those EDUs from the string of tokens of the longer EDU. For example, if the EDU list consists of the EDUs [1 2 3] and [1 2 3 4 5 6], the list will be transformed in this step to a list containing the EDUs [1 2 3] and [4 5 6].
4. The list of EDUs is sorted by the start positions of the EDUs in the sentence.

The final step is to make the output suitable for discourse relation annotation within the RST framework. The output of the algoritm is a text file containing one EDU per line. The text files can be opened with the O'Donnell's RSTTool3 (O'Donnell, 1997) for annotating the discourse structure. One issue in the production of the text file containing the EDUs is the handling of embedded EDUs. Note that the sorting procedure in step 4 above solves the problem of embedded EDUs by placing the embedded EDU after the EDU inside which it occurs in the text.

5 Data

Our corpus consists of 80 Dutch texts varying in length between a minimum of approximately 190 words and a maximum of approximately 400 words. The corpus consists of 40 expository texts and 40 persuasive texts. The expository subcorpus contains 20 texts from online encyclopedias on astronomy and 20 astronomy news texts from a popular-scientific news website. The persuasive texts are 20 fundraising letters from humanitarian organizations and 20 commercial advertisements from lifestyle and news magazines. Table 1 shows the total number of sentences, EDUs and embedded EDUs in each genre. For more details about the corpus, see van der Vliet et al. (2011) and Redeker et al. (2012)).

Text type	*sentences*	*EDUs*	*embedded EDUs*
Encyclopedia texts	395	618	17
Popular Scientific News	435	585	5
Fundraising letters	467	597	2
Advertisements	410	545	3
Total	1707	2345	27

Table 1: Number of sentences, segments and embedded segments per genre

The texts were segmented by two trained annotators following the segmentation principles established in the project. A kappa value of 0.97 shows a high level of inter-annotator agreement.

6 Evaluation

The performance of a discourse segmentation algorithm can be measured by comparing the output of the system with the manual segmentation of the text. The most widely used evaluation measures for automatic discourse segmentation are precision, recall and F-score. Precision is the number of correct units divided by the total number of units given by the system. Recall is the number of correct units divided by the total number of units in the manual segmentation. F-score is the harmonic mean of precision and recall: F = (2 * precision * recall)/(precision + recall).

When comparing the algorithm's performance with other approaches, it is important to note that automatic segmentation approaches differ in the unit that is used for the computation of these evaluation measures. Some approaches (e.g. Subba and Di Eugenio (2007), Tofiloski et al. (2009) and Sagae (2009)) use sentence-internal discourse boundaries, while others (e.g. Le Thanh et al. (2004),Lüngen et al. (2006), Afantenos et al. (2010)) use elementary discourse units. This leads to differences in the scores. For example, sentence (5) consists of three EDUs that are distinguished by two sentence-internal segment boundaries. Consider the segmentation in sentence (6) as the output of the segmentation algorithm. The precision based on sentence-internal boundaries is then $1/1 = 1$ and the precision based on EDUs $1/2 = 0.5$. Recall is $1/2 = 0.5$ based on inside-sentence boundaries and $1/3 = 0.33$ based on EDUs. The F-score is 0.667 based on sentence-internal boundaries and 0.4 based on EDUs.

(5) [Als dat te strak wordt,][breekt het magneetveld los van het gas][en krijgt een nieuwe structuur.] *[If that becomes too tight,][the magnetic field breaks from the gas][and gets a new structure.]*

(6) [Als dat te strak wordt, breekt het magneetveld los van het gas][en krijgt een nieuwe structuur.] *[If that becomes too tight, the magnetic field breaks from the gas][and gets a new structure.]*

The evaluation based on sentence-internal boundaries is problematic for the evaluation of embedded discourse units, because an evaluation based on boundaries cannot distinguish between an EDU that is interrupted by an embedded EDU and three subsequent EDUs. We will thus present our results using precision, recall and F-score based on EDUs.

7 Results

For the development of the segmenter we used a training set of 20 texts from our corpus: 5 texts from each of the four genres. We used the remaining 60 texts (15 texts per genre) for the evaluation of the segmenter. We used precision, recall and F-score based on EDUs as evaluation measures. Table 2 shows the segmentation results on our test set per genre. Although the nature and structure of the text types is quite different, the performance of the segmenter is stable across genres and the results show a reasonable agreement with the manually annotated texts.

Our segmenter makes use of the syntactic parser Alpino to segment the input text into sentences and to produce syntactic trees of the sentences in the text. However, both the tokenization component and the dependency parser can make errors, which could influence the results of the segmentation algorithm. We therefore compared the performance of the segmenter in three different settings. In the first setting (SEG1) we used the Alpino tokenizer and dependency

Genre	*EDUs*	*Precision*	*Recall*	*F-score*
EE	454	0.832	0.784	0.807
PSN	436	0.814	0.823	0.818
FL	453	0.870	0.859	0.864
AD	383	0.831	0.757	0.792
Total	1726	0.837	0.808	0.822

Table 2: Segmentation results in Encyclopedia Entries (EE), Popular Scientific News (PSN), Fundraising Letters (FL) and Advertisements (AD)

parser for EDU identification. In the second setting (SEG2) we used gold standard sentence tokenization and automatically generated syntactic trees. In the third setting (SEG3) we used gold standard tokenization and gold standard syntactic trees as the input for EDU identification. Table 3 shows the results for EDU identification on a subset of the test set of 20 texts (5 texts per genre) for which we manually corrected the Alpino parse trees. For comparison, the table also contains two baselines. BASE 1 takes every comma in a sentence as the end of an EDU and the start of a new EDU. BASE2 uses only EDU rules 1, 3 and 4 and the combination rules (see section 4) for segmentation.

Segmenter	*GS tok.*	*GS synt.*	*EDUs*	*Precision*	*Recall*	*F-score*
BASE1	X		563	0.654	0.604	0.628
BASE2			563	0.839	0.776	0.806
SEG1			563	0.858	0.826	0.842
SEG2	X		563	0.860	0.831	0.845
SEG3	X	X	563	0.895	0.865	0.880

Table 3: Segmentation results based on EDUs

As can be seen in table 3 our EDU identification algorithm performs better than the two baselines. The EDU identification results are only marginally better when using gold standard tokenization (SEG2) compared to the automatic tokenizer (SEG1). When compared to the results using automatically identified syntactic trees (SEG2), using gold standard syntactic trees (SEG3) leads to an improvement of the results for EDU identification of about 3.5%.

In table 4 we compare the EDU identification algorithm with our earlier work on automatic segmentation (see van der Vliet (2010)). Our previous segmenter inserts segment boundaries but is not able to identify EDUs, so in our comparison we evaluate the segmentation results based on sentence-internal segment boundaries. When applied to our test set of 60 texts (using gold standard tokenization), our new segmenter (SEG2013) performs better than the old one (SEG2010) on each of the evaluation measures.

Segmenter	*GS tok.*	*Boundaries*	*Precision*	*Recall*	*F-score*
SEG2010	X	480	0.68	0.66	0.67
SEG2013	X	480	0.77	0.73	0.75

Table 4: Segmentation results based on EDU boundaries

Our EDU identification algorithm performs reasonably well compared to that of segmenters for English. Bach et al. (2012) report an F-score of 0.86 based on EDUs using the Stanford parse trees. Subba and Di Eugenio (2007), Sagae (2009) and Tofiloski et al. (2009) use an evaluation based on sentence-internal boundaries and report F-scores of respectively 0.85, 0.87 and 0.83.

Note however that when we compare the results of these systems we should be aware of the differences in segmentation guidelines that are implemented in these systems.

7.1 A qualitative evaluation

The evaluation measures presented above show only quantitative evaluation results. In this section we present a qualitative evaluation and study the kinds of mistakes that occur when applying our EDU identification algorithm. We analyzed the EDU identification results (using gold standard tokenization and gold standard syntactic analysis) in the test set of 20 texts (5 per genre) and categorized the mistakes. We distinguish three different types of errors: errors concerning breaks in the manual segmentation that are not recognized by the system and lead to EDUs that are too long (34 errors), errors concerning extra breaks inserted by the segmentation algorithm that lead to EDUs that are too short (14 errors), and errors concerning an EDU break by the algorithm that is inserted at the wrong place in the sentence (3 errors).

The majority of errors of the first type occur in conjunction constructions. For example, in (7) there is forward conjunction reduction at clause-level coordination: the subject is elided in the second clause. In the syntactic analysis Alpino assigns the category 'prepositional phrase'to the second part. Generally, ouur EDU identification algorithm should not treat prepositional phrases as EDUs. A more specific rule is needed for a correct EDU identification in sentences like (7).

(7) [Tijdens de explosie is de kern van de ster in elkaar gestort][en in de snel ronddraaiende pulsar veranderd].
[During the explosion the core of the star was collapsed][and changed into the fast-spinning pulsar]. (PSN08)

Other errors of the first type concern the recognition of non-restrictive relative clauses. Our algorithm only inserts segment boundaries when non-restrictive relative clauses are preceded by a comma, so it does not recognize other non-restrictive relative clauses as in (8).

(8) [De Oortwolk is een grote wolk met miljarden kometen rondom ons zonnestelsel (inclusief de Kuipergordel)][die zich uitstrekt tot ongeveer een kwart van de afstand tussen de zon en de dichtstbijzijnde ster.]
[The Oort cloud is a large cloud with billions of comets surrounding our solar system (including the Kuiper belt)][which extends to about a quarter of the distance between the sun and the nearest star.](EE08)

An example of an error of the second type is shown below. In sentence (9) our algorithm wrongly identifies the text part *'doordat de kometen hier maar weinig ondervinden'* (because the comets are here barely affected) as a separate EDU. The text part functions as a complement clause and is therefore not treated as EDU in our manual segmentation. In the syntactic analysis of Alpino it is labeled with *@cat='cp'* and *@rel='mod'* and it contains a verb, so according to our automatic segmentation rules it is identified as an EDU (see EDU rule 3, section 4). This results in an embedded EDU in the segmentation of sentence (9) produced by our segmenter, as shown in (10).

(9) Dat komt doordat de kometen hier maar weinig ondervinden van de aantrekkingskracht

van de zon.
This is because the comets are here barely affected by the gravitational force of the sun. (EE08)

(10) [Dat komt van de aantrekkingskracht van de zon.][doordat de kometen hier maar weinig ondervinden]
[This is by the gravitational force of the sun.][because the comets are barely affected]

Example (12) shows an error of the third type, wrongly placed segment boundary, in combination of errors of the second type, failure to identify a boundary. Our manual segmentation is shown in (11) and the output of our segmenter is shown in (12). The segmenter does not recognize the segment boundaries after *rompslomp,*, after *rekening*, and after *opnemen* (which separate four elliptical clauses in this sentence), but does identify an embedded discourse discourse unit instead. The text part *'waar en wanneer u maar wilt'* is labeled with the tag *@cat='whrel'* in the syntactic analysis and is preceded by a comma, so the segmenter the EDU rule for non-restrictive relative clauses and identifies it as a separate EDU. Note that the segmenter produces an error of the second type in this sentence as well: it fails to identify the boundary after *opnemen*.

(11) [Dus.. geen papieren rompslomp,][waar en wanneer u maar wilt toegang tot uw rekening][en altijd vrij opnemen][en storten.]
[So.. no paperwork,][access to your account where and when you want][and always freely withdraw][and pay.]

(12) [Dus.. geen papieren rompslomp, toegang tot uw rekening en altijd vrij opnemen en storten.][waar en wanneer u maar wilt]
[So..no paperwork, access to your account and always freely withdraw and pay.][where and when you want]

8 Conclusion

We presented a rule-based algorithm to identify elementary discourse units (EDUs) in Dutch written texts. Our experimental results show that good identification results can be obtained using a relatively simple method. The performance of the EDU identifier is stable across the genres in our data. Together with the work of Tofiloski et al. (2009) and Le Thanh et al. (2004)) our work suggests that for languages for which large corpora annotated with discourse relations are not available, a rule-based approach is a viable alternative to machine learning approaches. In future work we will use the output of the EDU identification algorithm in our experiments on the automatic identification of discourse relations in Dutch text.

Acknowledgments

This work has been supported by grant 360-70-282 of the Netherlands Organization for Scientific Research (NWO) as part of the program Modelling textual organization (http://www.let.rug.nl/mto/).

References

Afantenos, S., Denis, P., Muller, P., and Danlos, L. (2010). Learning recursive segments for discourse parsing. *Arxiv preprint arXiv:1003.5372*.

Bach, N. X., Nguyen, M. L., and Shimazu, A. (2012). A reranking model for discourse segmentation using subtree features. In *SIGDIAL Conference'12*, pages 160–168.

Borisova, I. and Redeker, G. (2010). Same and Elaboration relations in the Discourse Graphbank. In *Proceedings of the 11th annual SIGdial Meeting on Discourse and Dialogue, Tokyo, September 24-25*.

Bosma, W. E. (2008). *Discourse Oriented Summarization*. PhD thesis, University of Twente, Enschede, the Netherlands.

Carlson, L. and Marcu, D. (2001). Discourse tagging reference manual. Technical report, ISI Technical Report ISI-TR-545.

Carlson, L., Okurowski, M. E., and Marcu, D. (2002). *RST discourse treebank*. Linguistic Data Consortium, University of Pennsylvania.

Eisenstein, J. (2009). Hierarchical text segmentation from multi-scale lexical cohesion. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 353–361.

Fisher, S. and Roark, B. (2007). The utility of parse-derived features for automatic discourse segmentation. In *Proceedings of ACL '07*, pages 488–495.

Hearst, M. (1997). Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Hernault, H., Bollegala, D., and Ishizuka, M. (2010). A sequential model for discourse segmentation. In *Proceedings of CICLing 2010*, pages 315–326.

Le Thanh, H., Abeysinghe, G., and Huyck, C. (2004). Automated discourse segmentation by syntactic information and cue phrases. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications (AIA 2004), Innsbruck, Austria*.

Louis, A., Joshi, A., and Nenkova, A. (2010). Discourse indicators for content selection in summarization. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 147–156.

Lüngen, H., Puskàs, C., Bärenfänger, M., Hilbert, M., and Lobin, H. (2006). Discourse segmentation of German written text. In *Proceedings of the 5th International Conference on Natural Language Processing (FinTAL 2006)*.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Marcu, D. (2000). *The theory and practice of discourse parsing and summarization*. The MIT Press.

Maslennikov, M. and Chua, T. (2007). A multi-resolution framework for information extraction from free text. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 592–599.

O'Donnell, M. (1997). RST-Tool: An RST analysis tool. In *Proc. of the 6th European Workshop on Natural Language Generation, Duisburg*.

Redeker, G., Berzlánovich, I., van der Vliet, N., Bouma, G., and Egg, M. (2012). Multi-Layer discourse annotation of a Dutch text corpus. In *Proceedings of LREC 2012, Istanbul, May 21-27*, pages 2820–2825.

Sagae, K. (2009). Analysis of discourse structure with syntactic dependencies and data-driven shift-reduce parsing. In *Proceedings of the 11th International Conference on Parsing Technologies*, pages 81–84.

Soricut, R. and Marcu, D. (2003). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of HLT/NAACL 2003*, pages 228–235.

Sporleder, C. and Lapata, M. (2005). Discourse chunking and its application to sentence compression. In *Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP)*, pages 257–264.

Subba, R. and Di Eugenio, B. (2007). Automatic Discourse Segmentation using Neural Networks. In *Proc. of the 11th Workshop on the Semantics and Pragmatics of Dialogue*, pages 189–190.

Thione, G., Van Den Berg, M., Polanyi, L., and Culy, C. (2004). Hybrid text summarization: Combining external relevance measures with structural analysis. In *Proceedings ACL Workshop Text Summarization Branches Out. Barcelona*.

Tofiloski, M., Brooke, J., and Taboada, M. (2009). A syntactic and lexical-based discourse segmenter. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 77–80.

van der Vliet, N. (2010). Syntax-based discourse segmentation of Dutch text. In Slavkovik, M., editor, *Proceedings of the 15th Student Session, ESSLLI*, pages 203–210.

van der Vliet, N., Berzlánovich, I., Bouma, G., Egg, M., and Redeker, G. (2011). Building a Discourse-annotated Dutch Text Corpus. In Dipper, S. and Zinsmeister, H., editors, *Bochumer Linguistische Arbeitsberichte 3*, pages 157–171.

Van Noord, G. et al. (2006). At last parsing is now operational. In *Verbum ex machina: actes de la 13e conférence sur le traitement automatique des langues naturelles (TALN 2006): Leuven, 10-13 avril 2006*, page 20.

Verberne, S., Boves, L., Oostdijk, N., and Coppen, P. (2007). Discourse-based answering of why-questions. *Traitement Automatique des Langues (TAL), special issue on "Discours et document: traitements automatiques"*, 47(2):21–41.