Towards Topic Driven Access to Full Text Documents

Caterina Caracciolo, Willem van Hage, and Maarten de Rijke

Informatics Institute, University of Amsterdam,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
{caterina, wrvhage, mdr}@science.uva.nl

Abstract. We address the issue of providing topic driven access to full text documents. The methodology we propose is a combination of topic segmentation and information retrieval techniques. By segmenting the text into topic driven segments, we obtain small and coherent documents that can be used in two ways: as a basis for automatically generating hypertext links, and as a visualization aid for the reader who is presented with a small set of focused and restricted text snippets. In the presence of a concept hierarchy, or ontology, information retrieval techniques can be used to connect the segments obtained to concepts in the ontology. In this paper we concentrate on the text segmentation phase: we describe our approach to segmentation, discuss issues related to evaluation, and report on preliminary results.

1 Introduction

The full text documents accessible in a digital library can be rather long, potentially with a loose structure or no structure at all. In such a context, a search system that would provide the user with a document relevant to a given information need, and then leave it up to the user to navigate within the document through a combination of "control F" and scrolling, would be very unsatisfactory. We address the issue of providing focused access to full text (scientific) documents in a digital environment, so as to enhance readability and minimize the browsing and scrolling effort. Specifically, we work in the setting of a collection of an electronic handbook consisting of "authoritative" (and usually lengthy) survey chapters. To provide access to the collection, a concept hierarchy (or "ontology") has been developed, consisting of concepts, and lexical relations (e.g., parent-child) and navigational relations (e.g., "see also") between those concepts. The concept hierarchy serves as a map of the handbook's domain, and, after browsing around the map, users jump from a concept to highly relevant text snippets, not complete chapters, in our collection.

We propose to use topic segmentation techniques as a way to subdivide documents into smaller documents (sub-documents), that are homogeneous in topic. Used as link *targets*, these sub-documents should provide readers with a highly relevant document whose coverage of a given topic is as exact as possible: shrinking the subdocument would cause relevant information to be left, and expand-

ing the subdocument would bring in too much non-relevant information. As the *sources* of the links we use the concepts in our concept hierarchy.

In this paper we focus on the task of topic segmentation: in Section 2 we discuss previous work on topic segmentation and present our own approach; in Section 3 we discuss the issue of evaluation for this task and present current results. In Section 4 we draw preliminary conclusions and discuss future work.

2 Topic Segmentation

Previous work on text segmentation has focused on improving retrieval [7], and on topic tracking of broadcast speech data [10]. Text segmentation algorithms are often based on an underlying theory of discourse, or discourse structure. This theory can hypothesize that the text is linear [9] or hierarchical [14]. Skorochod'ko's seminal work [12] has influenced many approaches to topic segmentation; according to Skorochod'ko's topologies, the overlap of words in sentences is an indicator of the semantic structure of the text. One of the methods influenced by Skorochod'ko's works is Hearst's TextTiling algorithm [7], which we take as the basis for our own work. TextTiling performs a linear segmentation by using patterns of lexical connectivity (i.e., repetition of words through the text). The algorithm first compares adjacent *blocks* of text (real paragraphs are not considered because of their variability in length) and assigns them a similarity value. The resulting sequence of similarity values is smoothed. Then the smoothed values are examined and each gap is given a score computed by averaging the difference between the smoothed similarity value at the gap and the peak to the left and to the right. Segment brakes are placed at a gap whose score is lower than a certain threshold. Then, for the sake of the reader, the segment break is rounded to the end of the next paragraph.

The tunable parameters in the algorithm are the size of the blocks, in "sentence", used for comparison, and the number of words forming a sentence. Hearst found that for newspaper corpora a block of 6 sentences, each consisting of 20 words, is optimal. Similarity among two blocks is computed with a cosine similarity. Note that the actual value of similarity is not used for computing a breaks: the algorithm only looks at relative differences.

In this paper we also consider C99, an algorithm for linear topic segmentation described in [5, 6]. It differs from TextTiling in that it takes real sentences as the basic unit and uses a combination of similarity (computed among sentences) and clustering. After a phase of standard preprocessing steps (stop-words removal, stemming), the algorithm computes a matrix of similarity in a sentence by sentence manner, where the adopted similarity measure is the usual cosine similarity. Then, a ranking scheme is applied to the similarity matrix, in order to make more visible the differences in similarity among the sentences. Finally, a hierarchical divisive clustering is applied.

We applied C99[1] to (real) paragraphs, so as to prevent the algorithm from splitting paragraphs, and to be able to compare results with TextTiling. Finally, we did not specify a number of expected segments (as is standard practice when applying divisive clustering), but used the defaults described in [5, 6].

Creation of a manually segmented corpus. The experiments on which we report below take place in the setting of a digital library project. Specifically, the *Logic and Language Links* (LoLaLi) project [3] explores methods to extend the traditional form of scientific handbooks with electronic tools. These tools should help the reader explore the content of the handbook and make it easier to locate relevant information. As a case study the project focuses on the *Handbook of Logic and Language* [13] (20 chapters, 1200 pages), and uses a WordNet-like concept hierarchy to provide access to (an electronic version of) the handbook [1]. For the work on which we report in this paper, we use the LaTeX sources of the book as our corpus, which amounted to about 4.5MB of text.

To develop a gold standard to be used for assessing our segments, we selected two chapters from the collection of 20, and annotated the topic segmentation manually. The two chapters were chosen on the basis of the coverage in the LoLaLi concept hierarchy [4, 3] and of the differences in style. Two annotators annotated the text independently, then discussed critical cases to agree on a unique annotation. The annotators were given indications about minimal and maximal size of a segment (respectively a paragraph, and the entire section). No other references to the layout structure of the text were made.

One of the two chapters had a rather formal style, with many tables, figures and formulas, either in-line or as separate objects: here the difficulty was in the treatment of those objects. The second chapter was written in a more narrative style, with fewer tables and pictures: here the annotators had difficulties with the rhetorical style of writing, as almost all paragraphs referred to previous ones.

The annotators agreed on a large number of breaks, that we therefore consider more fundamental or evident than others. We found that within these breaks one of the two annotators would mark additional breaks, while the other would mark fewer breaks, displaying typical "splitter" and "lumper" behavior, respectively [8]. While this is hard to quantify, the resulting gold standard is more of a splitter than a lumper.

3 Evaluation Issues

The evaluation of a topic segmentation system can be either task independent or task dependent. If task independent, the evaluation is done by comparing the result of the system against an annotated corpus, a 'gold standard,' while a task dependent evaluation would look at how the segmentation improves other computational tasks. Here, we concentrate on a task independent evaluation, performed on the basis of our manually annotated corpus. The most commonly

[1] We used the implementation of C99 and TextTiling made available by Choi at http://www.cs.man.ac.uk/∼mary/choif/software.html.

used measures are precision and recall, applied to topic breaks or entire segments. Precision (P) gives the proportion of hypothesized topic breaks (segments) that are correct, recall (R) gives the proportion of correct topic breaks (segments) that are hypothesized. The two measure are often combined by using the F-measure, which can be tuned so as to weigth precision and recall equally, or to privilege one of the two over the other. In Table 1 we report F values when P and R are equally treated, and when precision is twice as important as recall.

When applied to paragraph breaks, precision and recall can be interpreted as measuring how good the system is at recognizing topic shifts; when applied to entire segments, as measuring how good the system is at recognizing homogeneity in topic. Although crude measures (they do not give a measure of how distant the hypothesized segment break is from the real break), precision and recall are well understood measures. Reynar [11] suggests judging a boundary correct if it appeared within a fixed-sized window of words of an actual boundary. The disadvantage of this measure is that it does not distinguish between correct and incorrect boundaries within the window. Beeferman et al. [2] introduce the P_k precision measure, giving the probability that a randomly chosen pair of units (i.e., paragraphs or sentences) are classified accordingly in the gold standard and by the system being evaluated. The disadvantage is that P_k depends on the length of the document, in the sense that in case of non-trivial segmentation, it is likely that two distant units are not hypothesized as belonging to the same segments — i.e., provided that the whole document is not a single segment. For these reasons, we decided not to use the P_k measure. Finally, we remark that none of these measures say anything about how "reader-friendly" is the segmentation.

In our experiments, we compared three segmentation methods: a naive baseline that simply takes every paragraph break to be a segment break, TextTiling, and C99. The algorithms were applied to two chapters from the *Handbook of Logic and Language*, simply called A and B below, for which a gold standard was developed in the manner described previously. Chapter A consists of 13 sections (no subsections), organized into 179 paragraphs spanning 35 pages in the printed version.[2] It contains many tables, examples, explicit definitions and theorems, and many in-line formulas. The manual annotation results in 102 segments, on average 1.6 paragraphs long. Chapter B consists of 3 sections organized into respectively 0, 4 and 5 sub sections, spanning 54 pages in the printed version; the text is distributed into 221 paragraphs. Chapter B does not contain examples and theorems distinguished as such, nor tables and only a few figures, but it does contain many in-line formulas and lists of formulas (axioms or properties). Paragraphs in Chapter B can be quite long, up to the entire length of a subsection (ca. 300 words), on average ca. 80 words long. The annotation distinguishes 90 segments, on average ca. 2.5 paragraphs long.

The results of the evaluation are listed in Table 1. Let us briefly discuss them, starting with Chapter A. The recall value for the baseline is obviously the highest, since by placing a segment break at each paragraph breaks all breaks

[2] Bibliographic items are never considered.

	Baseline	TextTiling	C99
chapter A	$P = .614$, $R = 1$	$P = .602$, $R = .803$	$P = .571$, $R = .078$
	$F_{P=R} = .760$	$F_{P=R} = .683$	$F_{P=R} = .137$
	$F_{P=2R} = .665$	$F_{P=2R} = .630$	$F_{P=2R} = .253$
chapter B	$P = .408$, $R = 1$	$P = .344$, $R = .681$	$P = .565$, $R = .142$
	$F_{P=R} = .579$	$F_{P=R} = .445$	$F_{P=2R} = .228$
	$F_{P=2R} = .462$	$F_{P=2R} = .370$	$F_{P=2R} = .167$

Table 1. Evaluation results.

will be found. Precision is also high (more than 50% of the hypothesized breaks are correct), because there are almost twice as many paragraphs as segments in the manual annotation. For the same reason, TextTiling achieves a high recall (it hypothesizes 134 segments), while precision is not substantially different from the baseline. C99 returns only 15 segments, therefore recall is very low, but about half of the hypothesized breaks agree with the gold standard. For Chapter B, noticeably different scores are obtained, across the board. Again the baseline shows a total recall, but a lower precision than in the case of Chapter A. TextTiling scores worse than in the case of Chapter B, because it returns 168 segments, against the 90 distinguished by the annotators, and over a total of 221 paragraphs. C99 returns 23 segments, and recall score has doubled, while precision is stable.

4 Conclusions and Future Work

We reported on work in progress on the application of text segmentation techniques in a digital library environment. The overall aim of our work is to apply these techniques for the automatic generation of hypertext links to full text documents. In particular, we are interested in the application of these techniques for generating links from ontologies to corpora of full text documents. The work presented here concentrated on a task independent evaluation of the topic segmentation phase: we applied two well-known algorithms to a domain specific corpus, and evaluated the results against a previously manually annotated segmentation. As a baseline we used the system the identifies a segment breaks at each paragraph break. Our finding is that the baseline performs well when evaluated in terms of precision and recall, though more investigation should be done to assess such a segmentation in a more reader-oriented evaluation. In case of highly structured documents (chapter A), TextTiling gives the best balance between precision and recall; however, scores degrade when the text has a more narrative style (chapter B). C99 turns out to be the worse algorithm to use for such a task, mainly because it returns to few segments, too long.

Future work includes an analysis of the results that take into account the agreement of the human assessors, and the evaluation of the text segmentation algorithms within a larger task, viz. link generation. We plan to use the documents resulting from the topic segmentation as sub-documents to be retrieved

by an IR system, where the concepts in the LoLaLi ontology [1] will serve as queries. Within this setting we also plan to compare the structure provided by our topic segmentation system with the layout structure of the underlying LaTeX documents.

Acknowledgements

We thank Joost Kircz and David Ahn for interesting discussions. Caterina Caracciolo was supported by Elsevier Science Publishers. Maarten de Rijke was supported by grants from the Netherlands Organization for Scientific Research (NWO) under project numbers 220- 80-001, 365-20-005, 612.069.006, 612.000.106, 612.000.207 and 612.066.302.

References

1. Logic and language links project. http://lolali.net.
2. Doug Beeferman, Adam Berger, and John D. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210, 1999.
3. Caterina Caracciolo. Towards modular access to electronic handbooks. *JODI - Journal of Digital Information*, 3(4), 2003. http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Caracciolo/.
4. Caterina Caracciolo and Maarten de Rijke. Structured access to scientific information. In *Proceeding of First Global WordNet Conference*, 2002.
5. Freddy Choi. Advances in independent linear text segmentation. In *Proc. of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL-00)*, pages 26–33, 2000.
6. Freddy Choi. Linear text segmentation: approaches, advances and applications. In *Proc. of CLUK3*, 2000.
7. Marti A. Hearst. *Context and Structure in Automated Full-text Information Access*. PhD thesis, 1994.
8. Judith L. Klavans, Kathleen McKeown, Min-Yen Kan, and S. Lee. Resources for the evaluation of summarization techniques. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, Grenada, Spain*, May 1998.
9. Judith L. Klavans Min-Yen Kan and Kathleen R. McKeown. Linear segmentation and segment relevence. In *Proceedings of 6th International Workshop of Very Large Corpora (WVLC-6)*, pages 197–205, 1998.
10. Jay M. Ponte and W. Bruce Croft. Text segmentation by topic. In *European Conference on Digital Libraries*, pages 113–125, 1997.
11. Jeffrey C. Reynar. *Topic Segmentation: Algorithms and Applications*. PhD thesis, University of Pennsylvania, 1998.
12. E. Skorochod'ko. Adaptive method of automatic abstracting and indexing. *Information processing*, 71:1179–1182, 1092.
13. Johan van Benthem and Alice ter Meulen, editors. *Handbook of Logic and Language*. Elsevier, 1997.
14. Y. Yaari. Segmentation of expository text by hierarchical agglomerative clustering. In *Proceeding of the Conference on Recent Advances in Natural Language Processing*, pages 59–65, 1997.