Reducing redundancy in multi-document summarization using lexical semantic similarity

Iris Hendrickx, Walter Daelemans
University of Antwerp
Antwerpen, Belgium
iris.hendrickx@ua.ac.be
walter.daelemans@ua.ac.be

Erwin Marsi, Emiel Krahmer
Tilburg University
Tilburg, The Netherlands
e.j.krahmer@uvt.nl
e.c.marsi@uvt.nl

Abstract

We present an automatic multi-document summarization system for Dutch based on the MEAD system. We focus on redundancy detection, an essential ingredient of multi-document summarization. We introduce a semantic overlap detection tool, which goes beyond simple string matching. Our results so far do not confirm our expectation that this tool would outperform the other tested methods.

1 Introduction

One of the main issues in automatic multi-document summarization is avoiding redundancy. As the source documents are all related to the same topic, at least some of their content is likely to overlap. In fact, this is in part what makes multi-document summarization feasible. For example, news articles that report on a particular event, or that are based on the same source, often contain similar information expressed in different ways. A multi-document summarizer should include this overlapping information not more than once. The backbone of most current approaches to automatic summarization is a vector space model in which a sentence is regarded as a bag of words and a weighted cosine similarity measure is used to quantify the amount of shared information between a pair of sentences. Cosine similarity (in this context) essentially amounts to calculating word overlap, albeit with weighting of the terms and normalization for differences in sentence length. It is clear that this approach to detecting redundancy is far from satisfactory, because it only covers redundancy in its most trivial form, i.e., identical words. In contrast, the redundancy that we ultimately want to avoid in summarization is that at the semantic level. As an extreme case in point, two sentences with no words in common can still carry virtually the same meaning.

The remainder of this paper is structured in the following way. In Section 2 we introduce a tool for detecting semantic overlap. In section 3 we present a Dutch multi-document summarization system, based on the MEAD summarization toolkit (Radev et al., 2004). Next, in section 4 we describe the experimental setup and the data set that we used. Section 5 reports on the results, and we conclude in section 6.

2 Detecting semantic overlap

In this section, we detail the semantic overlap detection tool and the resources we build on.

Parallel/comparable text corpus The basis for our semantic overlap detection tool is a monolingual parallel/comparable tree-bank of 1 million words of Dutch text (Marsi and Krahmer, 2007). Half of the text material has so far been manually aligned at the sentence level. Subsequently, the sentences have been parsed and the resulting parse trees have been aligned at the level of syntactic nodes. Moreover, aligned nodes have been labeled according to a set of semantic similarity labels that express the type of similarity relation between the nodes. The following five labels are used: *generalize, specify, intersect, restate*, and *equal*. The corpus serves as the basis for developing tools for automatic alignment and relation labeling.

Word aligner The word alignment tool takes as input a pair of source and target sentences and produces a matching between the words, that is, a (possibly partial) one-to-one mapping of source to target words. This aligner is a part of the full fledged tree aligner currently under development.

The alignment task comprises several subtasks. First, the input sentences are tokenized and parsed with the Alpino syntactic parser for Dutch (Bouma et al., 2001). Apart from the syntactic analysis, which we disregard in the current work, the parser

Proceedings of the 2009 Workshop on Language Generation and Summarisation, ACL-IJCNLP 2009, pages 63–66,
Suntec, Singapore, 6 August 2009. ©2009 ACL and AFNLP

performs lemmatization, part-of-speech tagging and compound analysis, all of which are used here.

In addition, the aligner uses lexical-semantic knowledge from Cornetto, a lexical database for Dutch (40K entries) similar to the well-known English WordNet (Vossen et al., 2008). The relations we use are *synonym*, *hyperonym*, and *xpos-near-synonym* (align near synonyms with different POS labels). In addition we check whether a pair of content words has a least common subsumer (LCS) in the hyperonym hierarchy. As path length has been shown to be a poor predictor in this respect, we calculate the Lin similarity, which combines the Information Content of the words involved (Lin, 1998). A current limitation is that we lack word sense disambiguation, hence we take the maximal score over all the senses of the words.

The components described above can be considered as experts which predict word alignments with a certain probability. Since alignments can support, complement or contradict each other, we are faced with the problem of how to combine the evidence. Our approach is to view the alignment as a weighted bipartite multigraph. That is, a graph where source and target nodes are in disjoint sets, multiple edges are allowed between the same pair of nodes, and edges have an associated weight. Our goal is on the one hand to maximize the sum of the edge weights, and on the other hand to reduce this graph to a model in which every node can have at most one associated edge. This is a combinatorial optimization problem known as *the assignment problem* for which efficient algorithms exist. We use a variant of the *The Hungarian Algorithm*[1] (Kuhn, 1955), for the computation of the matches.

Sentence similarity score Given a word alignment between a pair of sentences, a similarity score is required to measure the amount of semantic overlap or redundancy. Evidently the similarity score should be proportional to the relative number of aligned words. However, some alignments are more important than others. For example, the alignment between two determiners (e.g. *the*) is less significant than that between two common nouns. This is modeled in our similarity score by weighting alignments according to the idf (inverse document frequency) (Spärck Jones, 1972) of the words involved.

$$sim(s_1, s_2) = \frac{\sum\limits_{w_i \in A} idf(w_i)}{\sum\limits_{w_j \in S} idf(w_j)} \qquad (1)$$

Here s_1 and s_2 are sentences, S is the longest of the two sentences, w_j are the words in S, A is the subsequence of aligned words in S, and w_i are the words in A.

3 Multi-document summarization

The Dutch Multi-Document Summarizer presented here is based on the MEAD summarization toolkit (Radev et al., 2004), which offers a wide range of summarization algorithms and has a flexible structure. The system creates a summary by extracting a subset of sentences from the original documents. The summarizer reads in a cluster of documents, i.e. a set of documents relevant for the same topic, and for each sentence it extracts a set of features. These features are combined to determine an importance score for each sentence. Next the sentences are sorted according to their importance score. The system starts a summary by adding the sentence with the highest weight. Then it examines the second most important sentence and measures the similarity with the sentence that is already added. If the overlap is limited, the sentence is added to the summary, otherwise it is disregarded. This process is repeated until the intended summary size is reached. The module that performs this last step of determining which sentences end up in the final summary is called the *reranker*.

We use two baseline systems: the random baseline system randomly selects a set of sentences and the lead-based system which selects a subset of initial sentences as summary. We investigated the following features. A simple and effective features is the *position*: each sentence gets a score of $1/position$ where 'position' is the place in the document. The *length* feature is a filter that removes sentences shorter than the given threshold. The *simwf* feature presents the overlap of a sentence with the title of the document computed with cosine similarity. One of MEAD's main features is *centroid*-based summarization. Centroids of clusters are used to determine which words are important for the cluster and sentences containing these words are considered to be central sentences. The words are weighted with tf*idf.

[1] Also known as the *Munkres algorithm*

The aim of query-based summarization is to create summaries that are relevant with respect to a particular query. This can easily be done with features that express the overlap between the query and a source sentence. We examined three different query-based features that measure simple word overlap between the query and the sentence, cosine similarity with tf*idf weighting of words and cosine similarity without tf*idf weighting.

The MEAD toolkit implements multiple reranker modules, we investigated the following three: the *cosine*-reranker, the *mmr*-reranker and *novelty*-reranker. We compare these rerankers against the semantic overlap detection (sod) tool detailed in section 2. The cosine-reranker represents two sentences as tf*idf weighted word vectors and computes a cosine similarity score between them. Sentences with a cosine similarity above the threshold are disregarded. The mmr-reranker module is based on the maximal margin relevance criterion (Carbonell and Goldstein, 1998). MMR models the trade-off between a focused summary and a summary with a wide scope. The novelty-reranker is an extension of the cosine-reranker and boosts sentences occurring after an important sentence by multiplying with 1.2. The reranker tries to mimic human behavior as people tend to pick clusters of sentences when summarizing.

4 Experimental setup

To perform proper evaluation of the summarization system we constructed a new data set for evaluating Dutch multi-document summarization. It consists of 30 query-based document clusters. The document clusters were created manually following the guidelines of DUC 2006 (Dang, 2006). Each cluster contains a query description and 5 to 25 newspaper articles relevant for that particular question. For each cluster five annotators wrote an abstract of approximately 250 words. These summaries serve as a gold standard for comparison with automatically generated extracts.

We split our data set in a test set of 20 clusters and a development set of 10 clusters. We use the development set for parameter tuning and feature selection for the summarizer. We try out each of the characteristics discussed in section 3. The best combination found on the development set is the feature combination *position*, *centroid*, *length* with cut-off 13, and *queryCosine*. We tested the different rerankers and vary the similarity thresholds to determine their optimal threshold value. As the novelty-reranker scored lower than the other rerankers on the development set, we did not include it in our experiments on the test set.

For the experiments on the development set, we compare each of the automatically produced extracts with five manually written summaries and report macro-average Rouge-2 and Rouge-SU4 scores (Lin and Hovy, 2003). For the experiments on the test set, we also perform a manual evaluation. We follow the DUC 2006 guidelines for manual evaluation of responsiveness and the linguistic quality of the produced summaries. The responsiveness scores express the information content of the summary with respect to the query. The linguistic quality is evaluated on five different objectives: *grammaticality, non-redundancy, coherence, referential clarity* and *focus*. The annotators can choose a value on a five point scale where 1 means 'very poor' and 5 means 'very good'. We use two independent annotators to evaluate the summaries and we report the average scores.

5 Results

The evaluation of the results on the test set are shown in table 1. The Rouge scores of the different rerankers are all above both baselines, and they are very close to each other. The scores for the content measure and responsiveness show that the values for the automatic summaries are between 2 (poor) and 3 (barely acceptable). The optimized summarizers score higher than the two baselines on this point.

We are most interested in the aspect of 'non-redundancy'. The random baseline system achieves a good result here, and the optimized summarizers all score lower. The chance of overlap between randomly selected sentences seems to be lower than when an automatic summarizer tries to select only the most important sentences. When we compare the three optimized systems with different rerankers on this aspect we see that the scores are very close. Our semantic overlap detection (sod) reranker does not do any better than the other two. The optimized summarizers do perform better than the baseline systems with respect to focus and structure.

setting	Rouge-2	Rouge-SU4	gram	redun	ref	focus	struct	respons
rand baseline	0.101	0.153	4.08	3.9	2.58	2.6	2	2.25
lead baseline	0.139	0.179	3.05	3.6	3.25	2.88	2.38	2.4
optim-cosine	0.152	0.193	3.9	3.18	2.65	3.15	2.43	2.75
optim-mmr	0.149	0.191	3.98	3.13	2.55	3.13	2.38	2.7
optim-sod	0.150	0.193	4.05	3.13	2.85	3.23	2.5	2.7

Table 1: Macro-average Rouge scores and manual evaluation on the test set on these aspects: *gram*maticality, non-*redun*dancy, *ref*erential clarity, *focus*, *struct*ure and *respons*iveness.

6 Discussion and conclusion

We presented an automatic multi-document summarization system for Dutch based on the MEAD system, supporting the claim that MEAD is largely language-independent. We experimented with different features and parameter settings of the summarizer, and optimized it for summarization of Dutch newspaper text. We presented a semantic overlap detection tool, developed on the basis of a monolingual corpus of parallel/comparable Dutch text, which goes beyond simple string matching. We expected this tool to improve the sentence reranking step, thereby reducing redundancy in the summaries. However, we were unable to show a significant effect. We have several possible explanations for this. First, many of the sentence pairs that share the same semantic content, also share a number of identical words. To detect these cases, therefore, computing cosine similarity may be just as effective. Second, the accuracy of the alignment tool may not be good enough, partly because of errors in the linguistic analysis or lack of coverage, and partly because certain types of knowledge (word sense, syntactic structure) are not yet exploited. Third, reranking of sentences is unlikely to improve the summary in cases where the preceding step of sentence ranking within documents performs poorly. We are currently still investigating this matter and hope to obtain significant results with an improved version of our tool for detecting semantic overlap.

We plan to work on a more refined version that not only uses word alignment but also considers alignments at the parse tree level. This idea is in line with the work of Barzilay and McKeown (2005) who use this type of technique to fuse similar sentences for multi-document summarization.

Acknowledgements This work was conducted within the DAESO `http://daeso.uvt.nl` project funded by the Stevin program (De Nederlandse Taalunie). The construction of the evaluation corpus described in this paper was financed by KP BOF 2008, University of Antwerp. We would like to thank NIST for kindly sharing their DUC 2006 guidelines.

References

Regina Barzilay and Kathleen R. McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2001. Alpino: Wide-coverage computational analysis of Dutch. In *Computational Linguistics in the Netherlands 2000.*, pages 45–59. Rodopi, Amsterdam, New York.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of SIGIR 1998*, pages 335–336, New York, NY, USA. ACM.

H.T. Dang. 2006. Overview of DUC 2006. In *Proceedings of the Document Understanding Workshop*, pages 1–10, Brooklyn, USA.

Harold W. Kuhn. 1955. The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97.

C.-Y. Lin and E.H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of HLT-NAACL*, pages 71 – 78, Edmonton, Canada.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the ICML*, pages 296–304.

Erwin Marsi and Emiel Krahmer. 2007. Annotating a parallel monolingual treebank with semantic similarity relations. In *Proceedings of the 6th International Workshop on Treebanks and Linguistic Theories*, pages 85–96, Bergen, Norway.

Dragomir Radev et al. 2004. Mead - a platform for multidocument multilingual text summarization. In *Proceedings of LREC 2004*, Lisabon, Portugal.

Karen Spärck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

P. Vossen, I. Maks, R. Segers, and H. van der Vliet. 2008. Integrating lexical units, synsets and ontology in the Cornetto Database. In *Proceedings of LREC 2008*, Marrakech, Morocco.