UMD/BBN at MSE2005

David Zajic, Bonnie Dorr, Jimmy Lin
Department of Computer Science
University of Maryland
College Park, MD 20742
{dmzajic,bonnie,jimmylin}
@umiacs.umd.edu

Richard Schwartz
BBN Technologies
9861 Broken Land Parkway, Suite 156
Columbia, MD 21046
schwartz@bbn.com

Abstract

We implemented an initial application of a sentence-trimming approach (Trimmer) to the problem of multi-document summarization in the MSE2005 task. Sentence trimming was incorporated into a feature-based summarization system, called Multi-Document Trimmer (MDT), by using sentence trimming as both a pre-processing stage and a feature for sentence ranking. We demonstrate that we were able to port Trimmer easily to this new problem, although the impact of sentence trimming was minimal compared to other features used in the system. The performance of our system in the official MSE2005 task was around the middle of the pack (16 out of 27). After some minor bug fixes and a simple correction (dateline removal) we obtained an improvement on a post-hoc run on the test data.

1 Introduction

This paper presents an initial application of UMD/BBN's single-document summarization approach (Trimmer), to the problem of multi-document summarization. Trimmer uses linguistically-motivated heuristics to trim syntactic constituents from sentences until a length threshold is reached. Given that MSE required a longer summary based on inputs from different sources, we embarked on an investigation of the feasibility of applying sentence trimming approach to multi-document summarization.

We incorporated sentence trimming into a feature-based summarization system, called Multi-Document Trimmer (MDT), by using sentence trimming as both a pre-processing stage and a feature for sentence ranking. Trimmer is used to pre-process the input documents, creating multiple partially trimmed sentences for each original sentence. The number of trimming operations applied to the sentence is used as a feature in the sentence ranker. The UMD/BBN multi-document system is called Multi-Document Trimmer (MDT).

We demonstrate that we were able to port Trimmer easily to this new problem, although the impact of sentence trimming was minimal compared to other features used in the system. The performance of our system in the official MSE2005 task was around the middle of the pack (16 out of 27). After some minor bug fixes and a simple correction (dateline removal) we obtained an improvement on a post-hoc run on the test data.

The next section relates our approach to other existing summarization systems. Following this, we describe the MDT approach and then present the results of running our system in the MSE2005 task. We will present results of our system with and without dateline correction.

2 Background

A successful approach to extractive multi-document summarization is to rank candidate sentences according to a set of factors, iteratively re-

ranking to avoid redundancy within the summary. MEAD (Radev et al., 2004; Erkan and Radev, 2004) ranks documents according to a linear combination of features including centroid, position and first-sentence overlap. Once a set of sentences has been chosen as the summary, all sentences are rescored with a redundancy penalty based on word overlap with the chosen sentences. A new set of summary sentences is chosen based on the re-ranking. This is iterated until there are no changes in the summary. MDT differs in that syntactic trimming is used to provide shorter, but still grammatically correct, variations of the sentences as candidates. Also, MDT treats redundancy as a dynamic feature of unselected candidates.

Syntactic shortening has been used as in multi-document summarization in the SC system (Blair-Goldensohn et al., 2004). The SC system pre-processes the input to remove appositives and relative clauses. MDT differs from SC in that a wider variety of syntactic structures are candidates for trimming, and that multiple trimmed varieties of each sentence are provided.

Minimization of redundancy is an important element of a multi-document summarization system. Carbonell and Goldstein (1998) propose Maximal Marginal Relevance (MMR) as a way of ranking documents found by an Information Retrieval system so that the front of the list will contain diversity as well as high relevance. (2000) demonstrate MMR applied to the problem multi-document summarization. MDT borrows the ranking approach of MMR, but uses a different set of features. MDT, like MEAD, uses feature weights that were optimized to maximize an automatic metric.

3 Multi-Document Trimmer

MDT consists of a three-stage process. First a syntactic trimmer is used to provide multiple trimmed versions of each sentence in each document of the topic set. Each of these trimmed variants is given a relevance score, either to a query or to the topic set. Finally sentences are chosen according to a linear combination of features.

We used five features in ranking the candidate sentences.

- Fixed features
  - Position. The zero-based position of the sentence in the document.
  - Relevance. The relevance score of the sentence to the query or the topic.
  - Trims. The number of trimmer rules applied to the sentence.
- Dynamic features
  - Redundancy. A measure of how similar the sentence is to the currently selected sentences.
  - Sent-from-doc. The number of sentences already selected from the sentence's document.

The score for a sentence is a linear combination of these five features.

3.1 Syntactic Sentence Trimming

We use Trimmer (Dorr et al., 2003; Zajic et al., 2004) to provide multiple trimmed versions of the sentences in the documents. Trimmer uses linguistically-motivated heuristics to remove low-content syntactic constituents until a length threshold is reached. In the context of multi-document summarization, each intermediate stage of trimming is presented as a potential summary sentence.

The following example shows the behavior of Trimmer on a sentence from the MSE2005 test set. Ideally, each stage of Trimmer yields a grammatically acceptable sentence.

(1) after 15 years and an investigation involving thousands of interviews, canada's police have arrested the men they say masterminded the deadliest-ever bombing of an airplane.

(2) after 15 years and an investigation involving thousands of interviews, canada's police have arrested the men they say masterminded the deadliest-ever bombing.

(3) after 15 years and an investigation involving thousands, canada's police have arrested the men they say masterminded the deadliest-ever bombing.

(4) canada's police have arrested the men they say masterminded the deadliest-ever bombing.

(5) canada's police have arrested the men.

After the results of the MSE2005 evaluation were published we examined our system output. We discovered that our system had often included datelines in the summaries. To deal with this, Trimmer was modified to skip over initial short sentences. [1] This had the desirable effect of eliminating datelines in written news and low-content introductory sentences from broadcast news.

We have also fixed some bugs in Trimmer regarding time expression removal and punction [2] and corrected a bug in aligning information from the parser and the named entity tagger that was discovered on the MSE2005 test data. We will present results of our primary submitted system and of our system in its current state.

3.2 Relevance Scoring

The relevance score measures the similarity between a trimmed sentence and the entire set of documents that defines the topic at hand. We assume that sentences having higher term overlap with the relevant documents are preferred for inclusion in the final summary. Lucene, a freely-available off-the-shelf information retrieval system,[3] was employed to calculate this measure. In order to get an accurate distribution of term frequencies, we indexed all relevant documents along with a corpus of newswire text (one year of the LA Times)—this additional text essentially serves as a background model for non-relevant documents. Using Lucene's built-in scoring function, we calculated the similarity between a sentence (or trimmed version thereof) and each relevant document. The relevance score for the sentence is simply the average of all document similarity scores.

[1] No sophisticated techniques, e.g., regular expression matching or parsing was used to recognize datelines. We ignored sentences at the start of the document with fewer than six words up until the first sentence with six or more words.

[2] Punctuation was not an issue in Trimmer's original task of headline generation: it was always removed.

[3] http://lucene.apache.org/

3.3 Redundancy Scoring

To measure how redundant a sentence is with respect to the current state of the summary, we imagine that the words in the potentially redundant sentence are drawn from either the current summary or from the general language. The probability of a word in a potentially redundant sentence is a linear combination with parameter λ of the probability that the word occurs in the summary and the probability that it occurs in the general language:

$$P(w|D) = \frac{count\ of\ w\ in\ D}{size\ of\ D}$$

$$P(w|C) = \frac{count\ of\ w\ in\ C}{size\ of\ C}$$

$$P(w) = \lambda P(w|D) + (1-\lambda)P(w|C)$$

where D is the current state of the summary and C is the corpus, in this case the concatenation of all the documents in the topic set. We have set $\lambda = 0.3$ as a conventional starting value, but have not yet tuned this parameter. We take the probability of a sentence to be the product of the probabilities of its words. The redundancy of a sentence with respect to the current summary is the degree to which the sentence is more typical of the summary than it is typical of the general language:

$$Redundancy(S) = \prod_{s \in S} \frac{\lambda P(s|D) + (1-\lambda)P(s|C)}{P(s|C)}$$

For ease of calculation, we actually use log probabilities:

$$\sum_{s \in S} \log(\lambda P(s|D) + (1-\lambda)P(s|C)) - \log P(s|C)$$

Prior to calculating the redundancy score, we remove stopwords and apply the Porter Stemmer (Porter, 1980) to the sentence, the current summary and the corpus.

3.4 Sentence Selection

The score for a sentence is a linear combination of the five features described above. The highest ranking sentence from the pool of eligible sentences is chosen for inclusion in the summary. When a sentence is chosen, all other trimmed versions of that sentence are eliminated. After a sentence is chosen, the dynamic features, redundancy

Feature	Submitted Weight	Revised Weight
Position	-1	-10
Relevance	20	28
Trims	-2	$-\infty$
Redundancy	-20	-20
Sent-from-doc	-0.5	-3

Table 1: Tuned Feature Weights

and sent-from-doc, are re-calculated, and the sentences are reranked.

The weights for the factors were determined by manually optimizing on a set of training data to maximize the ROUGE (Lin and Hovy, 2003) measure for 2-grams on recall on the DUC2004 Task 3 data, using ROUGE version 1.5.5. The weights for our submitted system and our revised system are shown in Table 1. The weight of $-\infty$ for Trims indicates that the highest ROUGE scores were achieved for the training data when no trimmed sentences were selected. It is possible to prohibit selection of trimmed sentences by setting the Trim weight to a very large negative value.

4 Evaluation

Table 2 shows the performance of the version of MDT that was submitted to the MSE2005 evaluation, and an improved version that removed datelines from consideration and made some improvements to the trimming component of the system. We ran the revised system with three settings of the trim weight. A weight of 0 effectively removes the number of trims as a ranking factor. A weight of -2 was the optimal weight in the original submission. When the weight is $-\infty$, no trimmed versions of sentences are used. The ROUGE-2 average recall for these runs are shown in 2. The results on the MSE2005 data show that trimming increases the ROUGE-2 score by a small, non-significant amount when the revised weights are used. When the earlier weights are used, there is a non-significant decrease, however the scores are generally higher. This suggests that the 24 data points (the number of topics in DUC2004 Task 3) may not be sufficient to optimize 5 independent factors, and that better optimization of weights could improve the system.

System	R2 Recall w/submitted weights	R2 Recall w/revised weights
Submitted System 19	0.11849	–
Revised system Trim weight = 0	0.13496	0.13193
Revised system Trim weight = -2	0.14754	0.13143
Revised system Trim weight = $-\infty$	0.14837	0.13039

Table 2: ROUGE scores

5 Future Work

The current state of MDT represents our first effort to incorporate syntactic trimming into multi-document summarization. We plan to analyze MDT's errors to determine why it did not select trimmed versions of sentences that actually removed unimportant syntactic constituents. We also plan to examine the output of the syntactic trimmer to determine if it is actually providing appropriate alternatives to the original sentences.

Acknowledgements

The University of Maryland authors are supported, in part, by BBNT Contract 020124-7157, DARPA/ITO Contract N66001-97-C-8540, and NSF CISE Research Infrastructure Award EIA0130422.

References

Sasha Blair-Goldensohn, David Evans, Vasileios Hatzivassiloglou, Kathleen McKeown, Ani Nenkova, Rebecca Passonneau, Barry Schiffman, Andrew Schlaikjer, Advaith Siddharthan, and Sergey Siegelman. 2004. Columbia university at duc 2004. In *Proceedings of the HLT-NAACL 2004 Document Understanding Workshop, Boston*, pages 23–30.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Research and Development in Information Retrieval*, pages 335–336.

Bonnie Dorr, David Zajic, and Richard Schwartz. 2003. Hedge trimmer: A parse-and-trim approach to headline generation. In *Proceedings of the HLT-NAACL 2003 Text Summarization Workshop, Edmonton, Alberta, Canada*, pages 1–8.

Güneş Erkan and Dragomir R. Radev. 2004. The university of michigan at duc2004. In *Proceedings of the HLT-NAACL 2004 Document Understanding Workshop, Boston*, pages 120–127.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, , and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of ANLP/NAACL Workshop on Automatic Summarization*, pages 40–48.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic Evaluation of Summaries Using N-gram Co-Occurrences Statistics. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, Alberta.

Martin Porter. 1980. An algorithm for suffix stripping. In *Program*, volume 14(3), pages 130–137.

Dragomir R. Radev, Hongyan Jing, Malgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. In *Information Processing and Management*, volume 40, pages 919–938.

David Zajic, Bonnie Dorr, and Richard Schwartz. 2004. Bbn/umd at duc2004: Topiary. In *Proceedings of the HLT-NAACL 2004 Document Understanding Workshop, Boston*, pages 112–119.