A Reflection of the Whole Picture Is Not Always What You Want, But That Is What We Give You

Martin Hassel and **Jonas Sjöbergh**
KTH KOD
SE-100 44 Stockholm, Sweden
{xmartin,jsh}@kth.se

Abstract

We evaluate a novel method for automatic text summarization through text extraction. It attempts to find the summary most similar to the original text, thus giving an overview of all the contents. It also evaluates whole summaries, making no judgments on for instance individual sentences. A greedy search strategy is used to search through the space of possible summaries and select the best summary of those found. When evaluated on English abstracts from the Document Understanding Conferences our method performed fairly well. In this paper we evaluate it on Swedish human produced extracts. It performs poorly, which was expected since these extracts were not produced to reflect the whole contents of the texts. They only cover the most important topic.

keywords: computational semantics, text representation, extract corpus

1 Introduction

When a computer automatically creates a summary of one or more texts, we call it automatic summarization. This has been an ongoing research field for quite some time (Luhn 58; Edmundson 69; Salton 88), and is still an active research area. There are mainly two different approaches to automatic summarization, abstraction and extraction. Abstraction is when a text is analyzed in a relatively deep way and then a new, shorter, text with the same information content is generated. Extraction is when a shorter text is created by selecting and presenting passages from the original text, usually using quite shallow analysis of the text, but deeper analysis can also be used (Leskovec *et al.* 04). There are also methods that fall in between extraction and abstraction, by for instance extracting fragments from the text and then transforming them in some way, such as deleting subordinate clauses or joining incomplete fragments (Jing & McKeown 00; Jing 00). Our method is a purely extraction based method.

Usually extraction based summarization is accomplished by ranking individual segments, such as sentences or paragraphs. Then the best ranked segments are selected for inclusion one after the other. Often adjustments are made so the ranking of later segments is sensitive to choices made earlier, so as to avoid redundancy (Carbonell & Goldstein 98; Hovy & Lin 99; McDonald & Chen 02).

Our method tries to find an extractive summary of a given length that is as similar to the original text as possible. One novel idea in our method is to compare whole summaries, the individual segments are not considered. The main problem is how to define similarity between texts, especially when all texts we want to compare are extracts from the same text.

In this paper we evaluate our method on a corpus of human produced extracts. These extracts were intended to cover mostly the main topic of the original text, not as in our method give an overview of all contents. Thus it is possible to see if the generated extracts differ a lot from these single topic extracts. We also include a quick overview of an earlier evaluation of the method, to show that the reason for being different from the reference extracts is not simply that our method works poorly.

2 Random Indexing

Since Random Indexing (RI) (Sahlgren 05) plays a quite important role in our method we here give a short overview. RI is a word space model, i.e. it uses statistics on word distributions. A high-dimensional vector space is created and words that are close in this space are assumed to have related meanings. This is based on the distributional hypothesis, that words that occur in similar contexts tend to have similar meanings.

The most well known and studied word space model is probably Latent Semantic Analysis/Indexing (LSA) (Landauer *et al.* 98). LSA has been used in different ways for summarization, see for instance (Hovy & Lin 99; Yeh *et al.*

05; Miller 04; Murray *et al.* 05). One advantage that RI has over LSA is that it is an incremental method, so new texts are easily added.

In RI, each context is given a label. A context can be a document, paragraph, co-occurring word etc. This label is a very sparse vector. The dimensionality of the vector can be chosen to be quite short if high compression of the word space is desired, or long if compression is not needed. A few percent of the vector elements are set to 1 or -1, the rest to 0. This means that the labels are approximately orthogonal.

Each word is given a context vector. Whenever a word occurs in a context, the label of the context is added to the context vector of the word. Words that often occur in the same contexts will thus have similar context vectors.

We use co-occurring words as contexts, more specifically the three preceding and three following words, weighted so that closer words are given more weight.

3 The Summarization Method

The main idea is that we want to find a summary of a given length that is as similar to the original text as possible. One way to accomplish this would be to generate all possible extracts and see which one is most similar to the original text. The difficulty here lies in judging how similar two texts are. This is even more difficult than usual in this context. Most methods that calculate the similarity between two documents use measures like word or n-gram overlap. Since all candidate summaries generated by our method are extracts from the original text, all words in all summaries overlap with the original text. This is thus not a good way to differentiate between candidates.

3.1 Evaluating Summaries

When measuring similarity we make use of Random Indexing (RI), though if other methods were available it could easily be replaced. As mentioned in Section 2, RI assigns each word a context vector that in some sense represents the semantic content of the word. In our method, each text is assigned its own vector for semantic content. This vector is simply the (weighted) sum of all the context vectors of the words in the text. It should be noted that there is not, at least as far as we know, anything inherent in the Random Indexing model that guarantees that adding vectors in this way should in any way make sense. However, it turns out to work quite well in our experiments. This idea has also been applied for instance for text categorization (Sahlgren & Cöster 04).

Similarity between two texts is then simply measured as the similarity between the directions of the semantic vectors of the texts, in our case between the vector for the full text and the vectors for each of the candidate summaries.

When constructing the semantic vector for a text, the context vector for each word is weighted with the term frequency and the inverse document frequency, by simply making the length of the vector be $tf * log(idf)$. It is of course easy to add other weighting criteria if desired, for instance for slanted summaries where some words are deemed more important, or by giving words occurring early in the document, in document or paragraph headings etc. higher weight. In our experiments we only used $tf * log(idf)$.

Words in a text that have never been encountered during the calculation of a word space representation generally degrade performance, since we have no information on their distributional properties. This is not a problem with RI though, since it allows for continuous updates. By simply adding new texts to the index immediately before summarizing, all words in the texts to be summarized will have been encountered at least once.

For good performance very large amounts of text should be used when generating the Random Index. Since RI is relatively fast and only requires access to raw (unannotated) text this is generally not a problem. It is also an incremental method, so if more text is made available later this can be used to boost future performance without redoing work on earlier texts.

Since our method does not give any consideration to the position in the text a sentence is taken from (though this is possible to do if one so wishes), it should be straightforward to use for multi-document summarization as well. Since it only requires unannotated text as reference data it should also be relatively language independent.

Though we currently only use text extraction when generating summary candidates, the method we use to compare the candidates can be used for any type of summary.

3.2 Finding a Good Summary

To find a good summary we start with one summary and then try to see if there is another summary that is "close" in some sense, that is also a better summary. Better in this context means more similar to the original text. The reason we do not compare all possible summaries to the original text is that there are exponentially many possible summaries. Evaluating all of them would thus not be feasible.

In our experiments we use the "lead" summary, i.e. the first sentences from the document up to a specified length, as a starting point for our search. Using a standard hill-climbing algorithm we then investigate all neighbors, looking for a better summary. The summaries that are defined as neighbors to a given summary are basically those that can be created by removing one sentence and adding another. Since sentences vary in length we also allow removing two sentences and adding one new, or just adding one new sentence. Summaries that differ too much in size from the specified compression rate are discarded.

When all such summaries have been investigated, the one most similar to the original document is updated to be the currently best candidate and the process is repeated. If no other summary is better than the current candidate, the search is terminated. It is also possible to stop the search at any time if so desired, and return the best candidate so far.

In our experiments the generated summaries were quite short. For the English texts they were about three sentences. This means that there were usually quite few, typically around four, search iterations, though some documents required very many iterations before a local maximum was found.

4 Evaluation

In earlier experiments, our summarization method has been evaluated using manually written abstracts from the DUC summarization tasks (Hassel & Sjöbergh 06). In this paper we have evaluated our method on a corpus of human produced extractive summaries of Swedish newspaper articles (Hassel & Dalianis 05).

4.1 English Abstracts

These experiments are more thoroughly discussed in Hassel & Sjöbergh (05), but a short overview is presented here for reference. We used the evaluation method from the DUC 2004 task 2 (Over & Yen 04), and reference summaries from DUC 2001–2004. We generated summaries of lengths between 75 and 110 words and evaluated them compared to the 100 word reference abstracts provided. A human agreement score was calculated, as well as a baseline, the initial sentences up to the allowed summary length.

	DUC 2004	DUC 2001–2004
Human	42.6	39.7
Baseline-Lead	31.0	28.3
Holistic-250	33.9	32.0
Holistic-500	34.2	32.3
Holistic-1000	34.1	32.4

Table 1: ROUGE-1 scores, i.e. word overlap, in %, for our method (Holistic) with three different dimensionality choices for the Random Indexing context vectors. There are 114 texts in DUC 2004 and 291 in DUC 2001–2004.

The results are presented in Table 1. Different choices of dimensionality for RI differ very little, so there is little need to optimize the parameter choice. Our method is worse than the best systems of DUC 2004, which had about 39% word overlap, but better than about half the systems and well above the baseline.

4.2 Swedish Extracts

Since our method is relatively language independent we also evaluated it on Swedish. We used a corpus of human produced extracts of Swedish newspaper articles (Hassel & Dalianis 05). These extracts were however not produced to give an overview of the whole contents of the texts, which our method attempts to do. The humans were instead more focused on finding the most important topic in the text and then providing mostly information relevant to that.

There are only 15 documents in this corpus. On average there are 20 human generated extracts for each document. These vary quite a lot in compression rate, even for a specific document. There are usually some sentences that are included in almost all extracts, though, so there is agreement on what the main topic is. See Figure 1 for an example of the variation in selected sentences for one of the texts from the extract corpus.

As reference texts for the Random Index-

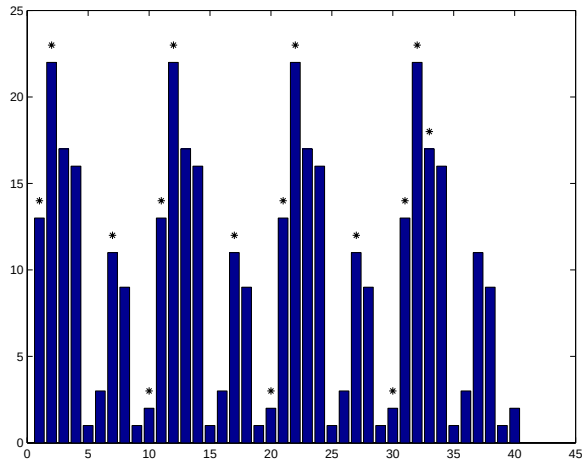

Figure 1: The number of human produced extracts that included each sentence from one of the Swedish corpus texts. There are a total of 27 human produced extracts for this text. Sentences marked with a * are those selected by our system.

ing method we used the Swedish Parole corpus (Gellerstam *et al.* 00), 20 million words, the Stockholm-Umeå Corpus (Ejerhed *et al.* 92), 1 million words, and the KTH News Corpus (Hassel 01), 13 million words. We used stemming and stop word filtering, since this worked well on the English texts.

When evaluating summaries we calculated a weighted precision. The score for a sentence included in the summary is the number of human produced extracts that also included this sentence divided by the total number of human produced extracts. The precision for the summary is then the average for all sentences in the summary.

A recall-like measurement was also calculated, since otherwise it would be best to simply pick a single sentence that the system is sure should be included. Each sentence that was included in at least one human produced extract, but not included in the summary to be evaluated, was also given a score as above, i.e. how often it was included by humans. The recall-like measurement is then the average score for all sentences not included in the summary but included in some human produced extract. Sentences ignored by both the system and the humans have no impact in the evaluation.

Since the extracts vary so much in length we generated two different sets of summaries using our method. The first, called Holistic-long, was the summary most similar to the original text that was longer than the shortest human produced extract and shorter than the longest. This generally produced long summaries, since it is easier to achieve good coverage of the original text with many words than with few. Since long summaries will have lower precision we also generated summaries, called Holistic-short, that, while longer than the shortest human produced extract, were never longer than the average extract.

For both sets of summaries, four different Random Indexes were used, since there are slight variations in the performance due to the randomness in RI. The results in Table 2 are the mean values of these four sets. All values are within 1.5 percentage units of the mean value.

We also compared our system to two baselines: lead, the first sentences of the original text with a size as close to the system generated summary as possible; and random, randomly chosen sentences up to the same size. We also calculated the agreement between the humans, by taking the average over all human produced extracts when treating them one at a time as a system generated summary instead.

Finally, we include figures for another summarization system, SweSum (Dalianis 00), that has also been evaluated on this data set. SweSum uses both statistical and linguistic methods, as well as some heuristics, and its main domain is newspaper text. SweSum creates extracts, by scoring sentences for various criteria, then extracting high scoring sentences in the original text and joining them together. The sentence scores are calculated based on e.g. sentence position, occurrence of numerical data and highly frequent keywords. Two different sets of summaries were generated by SweSum, one with summaries strictly below the average human produced extract length and one with the shortest summary possible above the average length.

The results are shown in Table 2. It can be seen that our system does not generate the same type of summaries as the others. Since our system tries to include the same proportions regarding different topics in the summary as was found in the original text, it has a quite low score with the precision-like measurement. This is natural, since the reference extracts normally only cover one topic. This also leads to a high (i.e. bad) score on the recall-like measurement, since the reference

	Included	Ignored	Perfect
Human	53	27	8
Baseline, Short Lead	55	29	2
Baseline, Long Lead	48	26	2
Baseline, Short Random	33	36	0.3
Baseline, Long Random	34	37	0
SweSum-above	53	28	3
SweSum-below	54	30	0
Holistic-500, Short	42	34	1
Holistic-500, Long	38	35	0

Table 2: Proportion of human produced extracts that included the sentences chosen by the system, in % (higher is better), and sentences ignored by the system but included by at least one human, also in % (lower is better). "Perfect" indicates for how many of the 15 documents a system generated an extract that was exactly the same as one of the human produced extracts.

extracts include so much information regarding the main topic that our method discards some of it as redundant.

When generating shorter summaries the same sentences are of course still considered redundant by our method, so the recall-like figure is more or less unchanged. Since the extract is shorter, there is room for less information. This gives higher precision, since our method still agrees that the main topic should be covered, but now includes less information regarding other topics. As expected, it seems like using our method when single topic summaries is wanted does not give the best results.

It can also be seen that outperforming the lead baseline on newspaper texts is very hard, since it performs on par with humans when generating shorter extracts. This means that this type of text is not very exciting to do summarization experiments on.

5 Conclusions

We have presented and evaluated an extraction based summarization method based on comparing whole summaries, not ranking individual extraction segments. It produces extracts that include the same proportions of topics as the original text.

The method requires no sophisticated tools, though stop word filtering and simple stemming was used in our experiments. The method is largely language independent and should also work without much modification for multi-document summarization. For good performance, access to large amounts of raw (unannotated) text is needed, but for many languages this is readily available.

Since our method tries to cover all topics covered in the original text, it did not perform very well when evaluated against extracts produced to cover mostly the main topic of a text.

References

(Carbonell & Goldstein 98) Jaime G. Carbonell and Jade Goldstein. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In Alistair Moffat and Justin Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336, Melbourne, Australia, 1998.

(Dalianis 00) Hercules Dalianis. SweSum - A Text Summarizer for Swedish. Technical report, TRITA-NA-P0015, IPLab-174, KTH NADA, Sweden, October 2000.

(Edmundson 69) H. P. Edmundson. New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery*, 16(2):264–285, April 1969.

(Ejerhed *et al.* 92) Eva Ejerhed, Gunnel Källgren, Ola Wennstedt, and Magnus Åström. The linguistic annotation system of the Stockholm-Umeå Corpus project. Technical report, Department of General Linguistics, University of Umeå (DGL-UUM-R-33), Umeå, Sweden, 1992.

(Gellerstam *et al.* 00) Martin Gellerstam, Yvonne Cederholm, and Torgny Rasmark. The bank of Swedish. In *Proceedings of LREC 2000*, pages 329–333, Athens, Greece, 2000.

(Hassel & Dalianis 05) Martin Hassel and Hercules Dalianis. Generation of reference summaries. In *Proceedings of 2nd Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pages 21–23, 2005.

(Hassel & Sjöbergh 06) Martin Hassel and Jonas Sjöbergh. Towards holistic summarization – selecting summaries, not sentences, 2006. submitted.

(Hassel 01) Martin Hassel. Internet as corpus - automatic construction of a Swedish news corpus. In *Proceedings of Nodalida 2001*, Uppsala, Sweden, 2001.

(Hovy & Lin 99) Eduard Hovy and Chin Yew Lin. Automated Text Summarization in SUMMARIST. In Inderjeet Mani and Mark T. Maybury, editors, *Advances in Automatic Text Summarization*, pages 81–94. The MIT Press, 1999.

(Jing & McKeown 00) Hongyan Jing and Kathleen R. McKeown. Cut and Paste-Based Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference and the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 178–185, Seattle, Washington, April 2000.

(Jing 00) Hongyan Jing. Sentence Reduction for Automatic Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference*, pages 310–315, Seattle, Washington, April 29–May 4 2000.

(Landauer *et al.* 98) Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. Introduction to Latent Semantic Analysis. *Discourse Processes*, 25:259–284, 1998.

(Leskovec *et al.* 04) Jure Leskovec, Marko Grobelnik, and Natasa Milic-Frayling. Learning sub-structures of document semantic graphs for document summarization. In *LinkKDD2004*, Seattle, Washington, 2004.

(Luhn 58) Hans Peter Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research Development*, 2(2):159–165, 1958.

(McDonald & Chen 02) Daniel McDonald and Hsinchun Chen. Using Sentence Selection Heuristics to Rank Text Segments in TXTRACTOR. In *Proceedings of the 2nd ACM/IEEE Joint Conference on Digital Libraries*, pages 25–38, Portland, Oregon, 2002.

(Miller 04) Tristan Miller. Latent semantic analysis and the construction of coherent extracts. In Nicolas Nicolov, Kalina Botcheva, Galia Angelova, and Ruslan Mitkov, editors, *Recent Advances in Natural Language Processing III*, volume 260 of *Current Issues in Linguistic Theory (CILT)*, pages 277–286. John Benjamins, Amsterdam/Philadelphia, 2004.

(Murray *et al.* 05) Gabriel Murray, Steve Renals, and Jean Carletta. Extractive summarization of meeting recordings. In *Proceedings of Interspeech'05*, Lisbon, Portugal, 2005.

(Over & Yen 04) Paul Over and James Yen. An introduction to duc 2004 intrinsic evaluation of generic new text summarization systems.
http://www-nlpir.nist.gov/projects/duc/pubs/2004slides/duc2004.intro.pdf, 2004.

(Sahlgren & Cöster 04) Magnus Sahlgren and Rickard Cöster. Using Bag-of-Concepts to Improve the Performance of Support Vector Machines in Text Categorization. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING 2004*, Geneva, Switzerland, August 23-27 2004.

(Sahlgren 05) Magnus Sahlgren. An Introduction to Random Indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE 2005)*, Copenhagen, Denmark, August 16 2005.

(Salton 88) Gerard Salton. *Automatic Text Processing*. Addison-Wesley Publishing Company, 1988.

(Yeh *et al.* 05) Jen-Yuan Yeh, Hao-Ren Ke, Wei-Pang Yang, and I-Heng Meng. Text summarization using a trainable summarizer and latent semantic analysis. *Information Processing and Management*, 41(1):75–95, 2005.