Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences

Klaus Zechner
Computational Linguistics Program
Department of Philosophy
135 Baker Hall
Carnegie Mellon University
Pittsburgh, PA 15213-3890, USA
zechner@andrew.cmu.edu

Abstract

This paper describes a system for generating text abstracts which relies on a general, purely statistical principle, i.e., on the notion of "relevance", as it is defined in terms of the combination of tf*idf weights of words in a sentence. The system generates abstracts from newspaper articles by selecting the "most relevant" sentences and combining them in text order. Since neither domain knowledge nor text-sort-specific heuristics are involved, this system provides maximal generality and flexibility. Also, it is fast and can be efficiently implemented for both on-line and off-line purposes. An experiment shows that recall and precision for the extracted sentences (taking the sentences extracted by human subjects as a baseline) is within the same range as recall/precision when the human subjects are compared amongst each other: this means in fact that the performance of the system is indistinguishable from the performance of a human abstractor. Finally, the system yields significantly better results than a default "lead" algorithm does which chooses just some initial sentences from the text.

1 Introduction

With increasing amounts of machine readable information being available, one of the major problems for users is to find those texts that are most relevant to their interests and needs in as short an amount of time as possible.

The traditional IR approach is that the user inputs a boolean query (possibly in a natural language-like formulation) and the system responds by presenting to the user the texts that are a "best match" to his query. In corpora where abstracts are not already provided it might facilitate the retrieval process a lot if text abstracts could be generated automatically either off-line to be stored together with the texts (e.g., as ranked sentence numbers), or on-line, in accordance with the user's query.

So far, there have been two main approaches in this field (for overviews on abstracting and summarizing see, e.g., (?) or (?)). One is oriented more towards information extraction, working with a knowledge base in a limited domain ("top down", see e.g., (?; ?; ?)), the other type relies mainly on various heuristics ("bottom up", see e.g., (?; ?)) which are less dependent on the domain but are still at least tuned to the text sort and thus have to be adapted whenever the system would have to be applied outside its original environment. Combinations of these methods have also been attempted recently (see e.g. (?)).

The focus of this paper will be the description and evaluation of an abstracting system which avoids the disadvantages coming along with most of these traditional approaches, while still being able to achieve a performance which matches closely the results of an identical abstracting task performed by human subjects in a comparative study.

The results indicate that it is indeed possible to build a system relying on a simple and efficient algorithm, using standard tf*idf weights only, while still achieving a satisfying output.[1]

2 A System for Generating Text Abstracts

Kupiec et al. (?) present the results of a study where 80% of the sentences in man-made abstracts were "close sentence matches", i.e., they were "either extracted verbatim from the original or with minor modifications" (p.70). Therefore, we argue that it is not only an easy way but indeed an appropriate one for an automatic system to choose a number of the most relevant sentences and present

[1] By "satisfying" we mean at least indicative for the content of the respective text, if not also informative about it.

these as a "text abstract" to the user.[2] We further argue that coherence, although certainly desirable, is impossible without a large scale knowledge based text understanding system which would not only slow down the performance significantly but necessarily could not be domain independent.

Our design goal was to use as simple and efficient an algorithm as possible, avoiding "heuristics" and "features" employed by other systems (e.g., (?)) which may be helpful in a specific text domain but would have to be redesigned whenever it were ported to a new domain.[3] In this respect, our system can be compared with the approach of (?) who also present an abstracting system for general domain texts. However, whereas their focus is on the evaluation of abstract readability (as stand-alone texts), ours is rather on abstract relevance. A further difference is the (non-standard) method of tf*idf-weight calculation they are using for their system.

Our system was developed in C++, using libraries for dealing with texts marked up in SGML format. The algorithm performs the following steps:[4]

1. Take an article from the corpus[5] and build a word weight matrix for all content words across all sentences (tf*idf-computation, where the idf-values are retrieved from a precomputed file).[6] High frequency closed class words (like A, THE, ON etc.) are excluded via a stop list file.

2. Determine the sentence weights for all sentences in the article: Compute the sum over all tf*idf-values of the content words[7] for each sentence.[8]

3. Sort the sentences according to their weights and extract the N highest weighted sentences in text order to yield the abstract of the document.

To reduce the size of the vocabulary, our system converts every word to upper case and truncates words after the sixth character. This is also much faster than a word stemming algorithm which has to perform a morphological analysis. For our experiments, the amount of new ambiguities thereby introduced did not cause specific problems for the system.

For the test set, we chose 6 articles from the corpus which are close to the global corpus average of 17 sentences per article; these articles contain approx. 550 words and 22 sentences on the average (range: 19–23). All these articles are about a single topic, probably because of our choice about a representative text length. We do not address the issue of multi-topicality here; however, it is well-known that texts with more than one topic are hard to deal with for all kinds of IR systems. E.g., the ANES system, described by (?), tries to identify these texts beforehand to be excluded from abstracting.

The system's run-time on a SUN Sparc workstation (UNIX, SUN OS 4.1.3) is approx. 3 seconds for an article of the test set.

3 Experiment: Abstracts as Extracts Generated by Human Subjects

In order to be able to evaluate the quality of the abstracts produced by our system, we conducted an experiment where we asked 13 human subjects to choose the "most relevant 5-7 sentences" from the six articles from the test set.[9] To facilitate their task, the subjects should first give each of the sentences in an article a "relevance score" from 1 ("barely relevant") to 5 ("highly relevant") and finally choose the best scored sentences for their abstracts. The subjects were all native speakers of English (since we used an English corpus) and were paid for their task. Compared to about 3 seconds for the machine system, the humans needed

[2] Clearly, there will be less coherence than in a man-made abstract, but the extracted passages can be presented in a way which indicates their relative position in the text, thus avoiding a possibly wrong impression of adjacency.

[3] In fact, it turned out that factors which could be thought of as "specific for newspaper articles", such as increased weights for title words or sentences in the beginning, did not have a significant effect on the system's performance.

[4] Due to space limitations, we cannot give all the details here. The reader is referred to (?) for more information on this algorithm, various other methods that were tested and their respective results. (This paper can be obtained from the author's home page whose URL is:
http://www.lcl.cmu.edu/~zechner/klaus.html.)

[5] We used the Daily Telegraph Corpus which comprises approx. 44.000 articles (15 million words).

[6] tf*idf=term frequency in a document (tf_k) times the logarithm of the number of documents in a collection (N), divided by the number of documents where this term occurs (n_k): $tf_k * \log \frac{N}{n_k}$. This formula yields a high number for words which are frequent in one document but appear in very few documents only; hence, they can be considered as "indicative" for the respective document.

[7] This provides a bias towards longer sentences. Experiments with methods that normalized for sentence length yielded worse results, so this bias appears to be appropriate.

[8] Words in the title and/or appearing in the first/last few sentences can be given more weight by means of an editable parameter file. It turns out, however, that these weights do not lead to an improvement of the system's performance.

[9] This number corresponds in fact well to the observation of (?) that the optimal summary length is between 20% and 30% of the original document length.

about 8 minutes (two orders of magnitude more time) for determining the most relevant sentences for an article.

4 Results and Discussion

4.1 Automatically created abstracts

Table 1 shows the precision/recall values for the tf*idf-method described in section ?? and for a default method that selects just the first N sentences from the beginning of each article ("lead"-method). Whereas the lead method most likely provides a higher readability (see Brandow et al. (?)), the data clearly indicates that the tf*idf method is superior to this default approach in terms of relevance.[10] The computation of these precision/recall values is based on the sentences which were chosen by the human subjects from the experiment, i.e., an average was built over the precision/recall between the machine system and each individual subject.

4.2 Abstracts produced by human subjects

The global analysis shows a surprisingly good correlation across the human subjects for the sentence scores of all six articles (see table ??): in the Pearson-r correlation matrix, 71 coefficients are significant at the 0.01 level (***), 5 at the 0.05 level (*), and only 2 are non significant (n.s.). This result indicates that there is a good inter-subject agreement about the relative "relevance" of sentences in these texts.

4.3 Comparison of machine-made and human-made abstracts

We computed precision/recall for every human subject, compared to all the other 12 subjects (taking the average precision/recall). From these individual recall/precision values, the average was computed to yield a global measure for inter-human precision/recall. Depending on the article, these values range from 0.43/0.43 to 0.58/0.58, the mean being 0.49/0.49. As we can see, these results are in the same range as the results for the machine system discussed previously (0.46/0.55, for abstracts with 6 sentences). This means that if we compare the output of the automatic system to the output of an average human subject in the experiment, there is no noticeable difference in terms of precision/recall — the machine performs as well as human subjects do, given the task of selecting the most relevant sentences from a text.

5 Suggestions for further work

5.1 Dealing with multi-topical texts

It can be argued that so far we have only dealt with short texts about a single topic. It is not clear how well the system would be able to handle texts where multiple threads of contents occur; possibly, one could employ the method of text-tiling here (see e.g., (?)), which helps determining coherent sections within a text and thus could "guide" the abstracting system in that it would be able to track a sequence of multiple topics in a text.

5.2 On-line abstracting

While our system currently produces abstracts off-line, it is feasible to extend it in a way where it uses the user's query in an IR environment to determine the relevant sentences of the retrieved documents. Here, instead of producing a "general abstract", the resulting on-line abstract would reflect more of the "user's perspective" on the respective text. However, it would have to be investigated, how much weight-increase the words from the user's query should get in order not to bias the resulting output in too strong a way.

Further issues concerning the human-machine interface are:

- highlighting passages containing the query words
- listing of top ranked keywords in the retrieved text(s)
- indicating the relative position of the extracted sentences in the text
- allowing for scrolling in the main text, starting at an arbitrary position within the abstract

6 Conclusions

In this paper, we have shown that it is possible to implement a system for generating text abstracts which purely operates with word frequency statistics, without using either domain specific knowledge or text sort specific heuristics.

It was demonstrated that the resulting abstracts have the same quality in terms of precision/recall as the abstracts created by human subjects in an experiment.

While a simple lead-method is more likely to produce higher readability judgments, the advantage of the tf*idf-method for abstracting is its superiority in terms of capturing content relevance.

Acknowledgments

The major part of this work has been drawn from the author's dissertation at the Centre for Cognitive Science, University of Edinburgh, UK. I wish to thank my supervisors Steve Finch and Richard

[10] The tf*idf method proved itself better than all the other methods of weight computation which we tested (see (?)); in particular, those using a combination of various other heuristics, as proposed, e.g., in (?).

Table 1: Precision/recall values for default (lead) and tf*idf methods.

number of extr. sent.	(a) lead method	(b) tf*idf method
6	0.38/0.39	0.46/0.55
8	0.38/0.51	0.45/0.68
10	0.37/0.62	0.41/0.74
12	0.34/0.69	0.39/0.83
14	0.33/0.79	0.37/0.91

Table 2: Significance of sentence score correlation between human subjects: All 6 articles

	HS4	HS3	HS8	HS9	HS1	HS5	HS12	HS11	HS13	HS10	HS14	HS15
HS2	***	***	***	***	***	***	***	***	***	***	***	***
HS4		***	***	***	***	***	***	***	***	***	***	***
HS3			***	***	***	***	***	*	***	***	***	***
HS8				*	***	***	***	***	***	***	***	***
HS9					***	***	*	*	***	***	***	***
HS1						***	***	***	***	***	***	***
HS5							***	***	***	***	***	***
HS12								***	***	***	***	n.s.
HS11									***	***	***	***
HS13										***	***	*
HS10											n.s.	***
HS14												***
height												

Shillcock for valuable discussions, suggestions and advice. Also, I am grateful to Chris Manning for his comments on an earlier draft, as well as to the two anonymous reviewers whose remarks greatly helped in improving this paper.

The author has been supported in part by grants from the Austrian Chamber of Commerce and Trade (Bundeswirtschaftskammer) and the Austrian Ministry of Science and Research (BMWF).

References

Brandow, R., Mitze, K., Rau, L.F. 1995. Automatic Condensation of Electronic Publications by Sentence Selection. In: *Information Processing & Management, 31(5).* pp.675–685

Edmundson, H.P. 1969. New Methods in Automatic Extracting. In: *Journal of the ACM, 16(2).* pp.264–285

Hearst, M.A., Plaunt, C. 1993. Subtopic Structuring for Full-Length Document Access. In: *Proceedings of the 16th ACM-SIGIR Conference.* pp.59–68

Hobbs, J.R., Appelt, D.E., Bear, J.S., Israel, D.J., Tyson, W.M. 1992. FASTUS: A System for Extracting Information from Natural Language Text. SRI International, Technical Note 519, Menlo Park, CA

Jacobs, P.S., Rau, L.F. 1990. SCISOR: Extracting Information from On-line News. In: *Communications of the ACM, 33 (11).* pp.88–97

Kupiec, J., Pedersen, J., Chen, F. 1995. A Trainable Document Summarizer. In: *Proceedings of the 18th ACM-SIGIR Conference.* pp.68–73

Mauldin, M.L. 1989. Information Retrieval by Text Skimming. CMU-CS-89-193, Carnegie Mellon University, Pittsburgh, PA

Miike, S., Itoh, E., Ono, K., Sumita, K. 1994. A Full-Text Retrieval System with a Dynamic Abstract Generation Function. In: *Proceedings of the 17th ACM-SIGIR Conference.* pp.152–161

Morris, A.H., Kasper, G., Adams, D. 1992. The Effects and Limitations of Automated Text Condensing on Reading Comprehension Performance. In: *Information Systems Research, 3(1).* pp.17–35

Paice, C.D. 1990. Constructing Literature Abstracts by Computer: Techniques and Prospects. In: *Information Processing & Management, 26(1).* pp.171–186

Salton, G., Allan, J., Buckley, C. 1993. Approaches to Passage Retrieval in Full Text Information Systems. TR 93-1334 (1993), Cornell University, Ithaca, NY

Sparck Jones, K., Endres-Niggemeyer, B. 1995. Automatic Summarizing. In: *Information Processing & Management, 31(5).* pp.625–630

Zechner, K. 1995. Automatic Text Abstracting by Selecting Relevant Passages. M.Sc. Dissertation, *Centre for Cognitive Science*, University of Edinburgh, UK