Computer-aided summarisation: How much does it really help?

Constantin Orăsan and Laura Hasler
Research Group in Computational Linguistics
University of Wolverhampton
Stafford St.
Wolverhampton, WV1 1SB, UK
C.Orasan@wlv.ac.uk and *L.Hasler@wlv.ac.uk*

Abstract

Computer-aided summarisation is a technology developed as a complement to automatic summarisation, which produces high quality summaries with less effort. To achieve this, a user-friendly environment which incorporates several well-known summarisation methods has been developed. This paper presents the main features of the computer-aided summarisation environment and evaluates the usefulness of the developed tool. Experiments showed that it is possible to reduce the time necessary to produce the summary by about 20% without any degradation in the summary's quality.

Keywords

computer-aided summarisation, automatic summarisation, evaluation

1 Introduction

Automatic summarisation systems help us deal with the current information overload by reducing it. Unfortunately, despite extensive work in this field, current technology is still not capable of creating human-like summaries. Instead, it usually produces *indicative summaries*, which allow a reader to get a quick gist of the document. The drawback of indicative summaries is that they are quite brief and do not allow the user to explore the structure of the source in more detail or to be used instead of the source, as is possible with some human-produced summaries. In the cases where *informative summaries*, which can replace the source, need to be produced, automatic summarisation does not appear to offer a viable solution yet. As a result, such summaries need to be produced by humans which makes their costs high.

In light of this problem, we propose computer-aided summarisation (CAS) as a complementary approach to automatic summarisation. Whereas automatic summarisation does not require any human input to produce summaries, we argue that computer-aided summarisation is a more feasible approach as it allows the user to post-edit the automatic summaries according to their requirements, resulting in better finished products. CAS is a technology developed at the University of Wolverhampton designed to help humans produce high quality summaries with less effort, in this way also lowering the costs.

The structure of the paper is as follows: the paper starts with a description of the computer-aided summarisation concept. Section 3 briefly presents the computer-aided summarisation tool we developed, whilst Section 4 describes the experiments carried out to prove the usefulness of the computer-aided summarisation concept. The paper finishes with a review of related work in the field of computer-aided language processing, followed by conclusions.

2 The computer-aided summarisation concept

The concept of computer-aided summarisation was inspired by the machine-aided translation approach suggested in 1980 by Martin Kay. Kay [8, 9] proposed the development of *cooperative man-machine systems* as a solution to the unrealistic task of fully automatic high quality translation, allowing the computer and the human translator to perform the translation tasks they are best at. CAS aims to help human summarisers by selecting the important information from a document and presenting it to them, and leaving the task of linking sentences to form a coherent abstract to the summariser. The main advantage of such an approach is that the summariser does not need to read the whole text, instead being presented with only the important parts of the document as a starting point for their summary.

The feasibility of the computer-aided approach is confirmed by research into the human summarisation process. The work of Endres-Niggemeyer [3] provides the theoretical grounding for the idea of human post-editing in computer-aided summarisation in terms of her three-stage human summarisation model of *document exploration*, *relevance assessment* and *summary production*. The first stage, document exploration, involves the summariser exploring the layout and organisation of the document to locate important information. During the next stage, relevance assessment, the summariser assesses information in the document to see if it is relevant to the summary. The final stage of summary production is where the actual creation of the summary as a

unit in itself takes place, and mainly involves cutting and pasting material from the original document using sentence patterns typical of the domain.

The first two stages of Endres-Niggemeyer's model correspond to the automatic summarisation in CAS, which uses automatic methods to identify important information in the text and present these, either in the form of a summary or as highlighted units within the full text, to the user. The third stage, which in Endres-Niggemeyer's analysis involves cutting and pasting operations and reorganising of the text, corresponds to the human summariser's post-editing of the summary, by accepting, rejecting and reorganising the information proposed by our computer-aided summarisation tool (presented in Section 3).

One could argue that for some domains, automatic summarisation methods still perform poorly and are likely to miss important information in the source. However, even when this is the case, computer-aided summarisation can still be useful. In domains where important information cannot be selected reliably, summarisers can use the automatic methods to produce a summary much larger than is actually required, and use this as a starting point, reducing it until the target length is reached. This means that the user will still need to read less text than if they produced a summary manually using the full source text, thereby speeding up the process. As an alternative, the user can choose to use automatic methods which remove unimportant sentences from the full text instead of selecting important ones. Again, this option reduces the length of the document to be read before the user can produce the summary. Because summaries are produced in a computer-aided environment rather than a fully automatic one, the user always has the option to return to the full text to get more information or to clarify uncertainties, and to over-ride the system's decisions.

3 The computer-aided summarisation tool (CAST)

As mentioned above, computer-aided summarisation is seen here as a complement to existing automatic summarisation techniques, as it allows human intervention in the summarisation process. However, in order to make the approach worthwhile, this intervention should be minimal, so that the effort required for a human to produce the summary using CAS is significantly less than that required to write a summary without the help of an advanced tool. To achieve this, several automatic summarisation techniques which have been extensively used were implemented in CAST. The purpose of these methods within CAST is to present to the user an extract which contains the most important sentences from a text, allowing them to post-edit it in order to improve its quality. As not all the sentences identified automatically will be worth including in a summary, the user has the option to override the program's decisions and delete irrelevant sentences, as well as to extract additional sentences.

After careful consideration of the existing automatic summarisation methods commonly used to produce extracts, we decided to implement the following methods: term-based summarisation methods, methods based on indicating phrases, surface clues, and discourse information. The term-based summarisation methods assume that the importance of a sentence can be determined on the basis of the words it contains. Indicating phrases are phrases such as *in this paper*, *we conclude that* which are specific to a domain and normally indicate the important sentences [16]. Surface clues can also help the summarisation process by assuming that words in titles and headings are more important than the rest, whilst text in brackets can usually be discarded. Finally, the discourse structure of the text can also be utilised. In CAST, this information is exploited in the form of lexical chains which are used to determine links between sentences [6]. A more detailed description of these methods can be found in [15]. Given that each method depends on a host of parameters, we offer users a high level of flexibility without compromising the simplicity of the tool by giving them the option to adjust all these parameters in a user friendly way.

The automatic methods embedded in the tool are used not only to identify important sentences in a text, but also to remove sentences which do not contain important information. For example, as well as extracting sentences containing certain indicating phrases or having their term-based score above a certain threshold, it is also possible to remove sentences which contain certain indicating phrases or have a term-based score lower than a given threshold. As with the case of important sentences, the user can review the system's decisions over-riding it whenever the decision is wrong.

The results of the summarisation methods can be viewed in different ways, depending on the user's preferences. They can be viewed either in isolation, when the results are presented as an automatic extract, or the sentences extracted can be highlighted within the source text using formatting defined by the user. The advantage of highlighting the results in the text is that the user can easily see the sentences in their original context. Given the friendly graphical interface available to the user and the different styles which can be defined for each method, the user can quickly identify sentences selected by different methods. A screenshot of the tool is presented in Figure 1.

Once a user decides that a sentence is important enough to be included in a summary (either indicated by the program or on the basis of their understanding), it can be copied into the summary window at the bottom of the tool and edited. In order to facilitate the editing task further, a common set of errors such as dangling pronouns and phrases which could indicate a problem with the summary (e.g. "on the other hand", "secondly", etc.) are highlighted to draw attention to them.

4 Evaluation

CAST is intended to help human summarisers to produce abstracts. To assess the extent to which

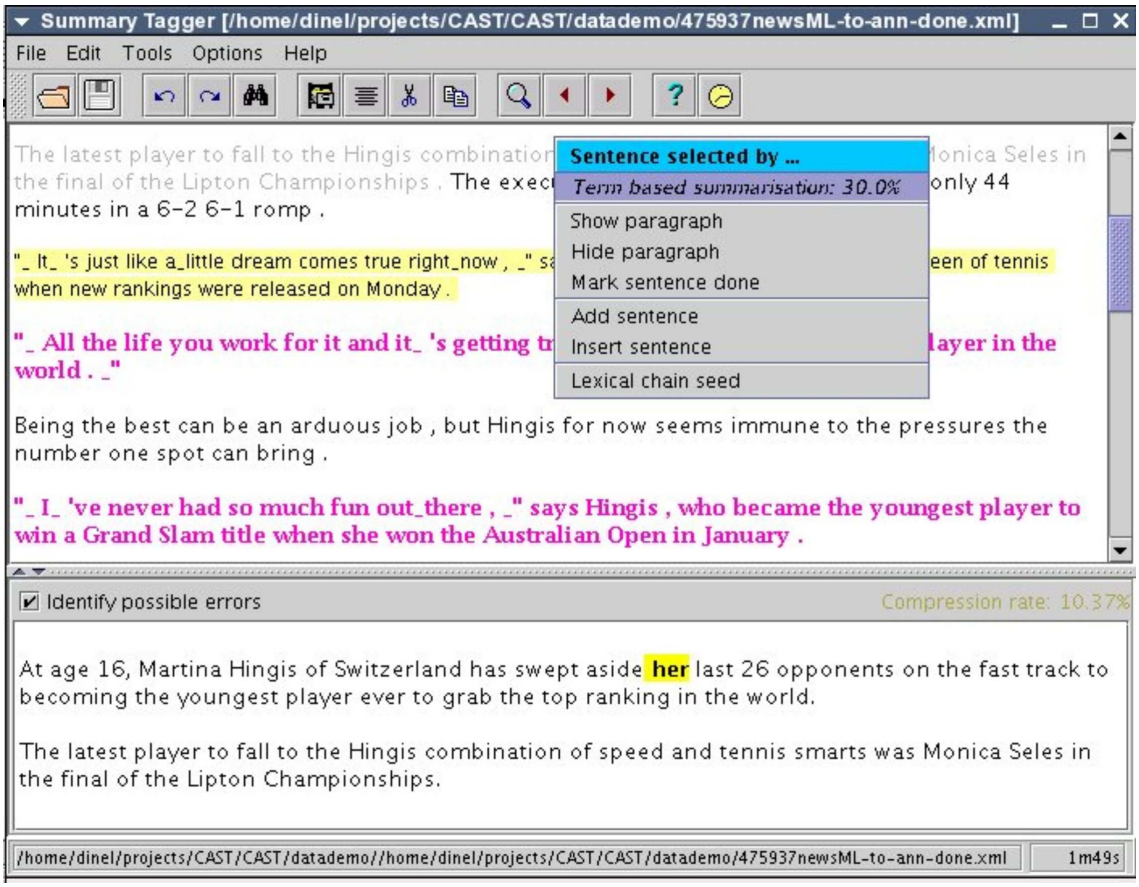

Fig. 1: *Screenshot of the program*

this is achieved, a professional human summariser was asked to use the tool and provide feedback about its usability. On the basis of this feedback, the tool was improved and we learnt how it is used by professional summarisers. These findings are presented in [14]. In this paper, we conjecture that summaries produced with the help of the tool will be as good as those manually produced, but that it will take less time to write them. To prove this hypothesis about the usefulness of the tool, two experiments were conducted. Their results are presented in Sections 4.1 and 4.2.

4.1 Evaluation of the time

As already mentioned, our assumption in the first experiment is that CAS can reduce the time necessary to produce summaries. To this end, a two stage experiment was conducted using a professional summariser. In the first stage, he was asked to produce summaries using a simple interface which records the time necessary to produce a summary, but does not provide any help with the summarisation process. In the second stage of the experiment, the same summariser was asked to produce abstracts of the same texts using the CAST in order to see whether the time necessary to produce them was reduced. The second stage of the experiment occurred one year after the first stage so that any effect of text familiarity was extinguished.

For this experiment, a total of 69 texts extracted from the CAST corpus [4] were summarised. Fifty four of these texts were newswire texts extracted from the Reuters corpus and fifteen were articles from New Scientist. The newswire texts contain on average 800 words whilst for the New Scientist texts the average number of words is 1750. These texts were selected for two reasons. First, they were previously annotated with information regarding the importance of sentences which allowed us to assess the accuracy of the automatic summarisation methods used by the human summariser. Secondly, the texts summarised are old enough (published around 1997) to ensure that the summariser is not very familiar with the topic discussed in the texts so that the first stage of the experiment is not unfairly helped by the summariser's background knowledge.

For both stages the professional summariser was asked to produce 20% summaries of each text. Table 1 shows the average number of seconds necessary to summarise a text with and without CAST. As expected, it takes longer to summarise the texts from New Scientist than the newswire ones due to the fact that they are almost twice in length. The table also indicates that by using CAST the time necessary to produce a summary reduces by almost 2 minutes for newswire texts and almost 2 minutes and a half for those from New Scientist. In both case the reduction

	Without CAST	With CAST	Percentage of reduction
Newswire texts	498sec	382sec	23.29%
New Scientist texts	771sec	623sec	19.19%

Table 1: *The time necessary to produce summaries with and without CAST*

is statistically significant.

The feedback from the summariser indicated that he first preferred to run the term-based summariser to highlight a set of sentences which could be useful for the abstract he produced. Although he was asked to produce a 20% abstract, he decided to use a 30% automatic summary as the starting point. This allowed him to see a wider selection of important sentences so that no crucial information was missed [14]. Because of the importance given to term-based summarisation by our user, we assessed its performance to establish whether there is any correlation between the accuracy of the automatic method and the time reduction. For evaluation, we used precision, recall and f-measure because the texts included in this experiment were extracted from the CAST corpus [4], a corpus which is annotated with information about the importance of sentences.

Table 2 presents the results of the evaluation. As can be seen, term-based summarisation performs significantly better on the newswire texts than on those from New Scientist. Moreover, a correlation between the reduction in the time necessary to produce the summary and the accuracy of the automatic methods was noticed. In light of this, it can be concluded that term-based summarisation methods included in CAST really help the summarisation process.

The user also at times used lexical chains to determine sentences related to those he considered important. However, this method was run only on an ad-hoc basis and therefore it is not possible to obtain figures about how useful it was. The same applies to the other summarisation methods incorporated, which were used even less often.

4.2 Quality of the summaries

Our second assumption was that the reduction in the time necessary to produce the summaries does not have a detrimental influence on their quality. To this end, we conducted a Turing-like test where pairs of summaries produced with and without CAST were shown to judges who were asked to select the best one in the pair. For this experiment, our hypothesis was that there are no significant differences between the two types of summaries and that human judges will not be able to make a reliable distinction between them.

For this experiment, 17 judges were shown four pairs of summaries each. The summaries were randomly selected from all the summaries produced in the first experiment. The order in which they were displayed in the pair was also random to avoid situations were one judge always selects the same element of the pair. Our judges included undergraduate and post-graduate students as well as members of staff, who were not given any instructions except that they should indicate which summary is better on the basis of their intuition.

Analysis of the results revealed that in 41 pairs the judges preferred summaries produced using CAST, whereas in 27 those produced without CAST were considered better. In order to see whether this difference is significant we calculated chi-square between the observed judgements and the expected judgements according to our hypothesis (i.e. that the votes are equally distributed between the two classes which means that each class gets 34 votes). The chi-square test revealed that there is no statistical difference at 0.05 level which indicates that there is no difference between the quality of the two types of summaries. Despite this, the results indicate that there is a slight preference towards the summaries produced using CAST.

5 Related work

This section presents related work in the field of computer-aided summarisation, but does not try to review existing work in automatic summarisation because it is considered to be beyond the scope of this paper. Good sources of more information about automatic summarisation are [10, 7].

Work related to CAS is relatively sparse in comparison with computer-aided approaches used in other areas such as machine translation and computer-aided language learning. It was also proved to be useful in other areas. Mitkov and Ha has showed that the time taken to generate multiple-choice questions was reduced by 75% when a computer-aided approach was used instead of a manual one, with no decrease in quality [12]. Semi-automatic annotation methods can also speed up the production of annotated corpora [5]. Whilst the idea of some form of automated help for human summarisers may have been around for some time [11, 1, 13], the more specific notion of CAS which combines automatic extracting and human post-editing, has only recently been explored in more depth [15, 14].

Craven [2] focuses on the automatic extraction of keywords and phrases from documents which could be useful when presented to a human trying to summarise the document. He argues that even this simple automatic assistance can help humans produce summaries of a text more easily than they would have done otherwise. The abstracting tool presented by Narita [13] aims to improve summaries of research papers in the field of information engineering written in English by Japanese software engineers who are intermediate or advanced learners of English. The tool provides an organisational template for the human abstractor to flesh out with their own material, helping them in the process by providing examples from a corpus. As with Craven's work, no automatic summarisation methods are employed; instead the tool

	Precision	Recall	F-measure
Newswire texts	44.19%	48.66%	46.32%
New Scientist texts	32.26%	34.05%	33.14%

Table 2: *The accuracy of the automatic summarisation method*

accesses a corpus of human-produced abstracts which have been analysed for their rhetorical structure.

In a working paper in 1995, Mitkov described plans to develop a "computer-assisted and user-friendly abstracting tool" [11] which identifies and highlights sentences considered to be important in terms of content for the user. Once the computer has performed this task, the human abstractor accepts or rejects the selected sentences as they see fit, and perhaps adds new sentences, before connecting the text together into cohesive paragraphs. Mitkov terms this approach semi-automatic and argues that it will make abstracting faster and cheaper as it does not rely on fully human summarisation which is time-consuming and labour-intensive. It is Mitkov's work which provided the basic idea for the CAST system presented in this paper.

6 Concluding remarks

Computer-aided summarisation was proposed as a complementary approach to automatic summarisation and a solution to producing high quality summaries at lower costs. This paper presented two experiments which prove the validity of the computer-aided summarisation concept. In the first experiment, a professional summariser produced summaries with and without the computer-aided summarisation tool. A comparison between the time necessary to produce the summaries revealed that the time is reduced by approximately 20% when CAST is used. A second experiment was carried out to determine whether the reduction in time had any negative influence on the quality of the summaries. The results of this experiment clearly indicate that there is no statistically significant difference between the two types of summaries in terms of quality. However, judges demonstrated a slight preference towards summaries produced using CAST. This is an unexpectedly good result which needs to be investigated further.

In the first experiment presented in this paper only one professional summariser was used. The reason for this is the high cost of employing such users and their limited availability. In the future, we intend to repeat the experiment using more summarisers, including non-professionals to see whether this confirms the results of our experiments.

7 Acknowledgments

This research was partially funded by the Arts and Humanities Research Board through the "Computer Aided Summarisation Tool - CAST" project.[1] We would also like to thank the professional summariser who participated in the first experiment and all the judges involved in the second experiment.

[1] The project's webpage is http://clg.wlv.ac.uk/projects/CAST/

References

[1] T. C. Craven. An experiment in the use of tools for computer-assisted abstracting. In *Proceedings of the ASIS 1996*, Baltimore, MD, United States, 19 - 24 October 1996.

[2] T. C. Craven. Abstracts produced using computer assistance. *Journal of the American Society for Information Science*, 51(8):745 – 756, 2000.

[3] B. Endres-Niggemeyer. *Summarizing information*. Springer, 1998.

[4] L. Hasler, C. Orăsan, and R. Mitkov. Building better corpora for summarisation. In *Proceedings of Corpus Linguistics 2003*, pages 309 – 319, Lancaster, UK, March, 28 – 31 2003.

[5] L. Hasler, C. Orăsan, and K. Naumann. NPs for Events: Experiments in Coreference Annotation. In *Proceedings of the 5th edition of the International Conference on Language Resources and Evaluation (LREC2006)*, pages 1167 – 1172, Genoa, Italy, 24 – 26 May 2006.

[6] E. Hovy. Approaches to the planning of coherent text. In C. L. Paris, W. R. Swartout, and W. C. Mann, editors, *Natural Language in Artificial Intelligence and Computational Linguistics*. Kluwer Academic Publishers, 1990.

[7] E. Hovy. Text summarisation. In R. Mitkov, editor, *The Oxford Handbook of computational linguistics*, pages 583 – 598. Oxford University Press, 2003.

[8] M. Kay. The proper place of men and machines in language translation. Technical Report CSL-80-11, Xerox PARC, Palo Alto, California, 1980.

[9] M. Kay. The proper place of man and machines in language translation. *Machine translation*, 12(1):3 – 23, 1997.

[10] I. Mani. *Automatic Summarization*. Natural Language Processing. John Benjamins Publishing Company, 2001.

[11] R. Mitkov. A breakthrough in automatic abstracting: the corpus-based approach. Technical report, University of Wolverhampton, 1995.

[12] R. Mitkov and L. A. Ha. Computer-aided generation of multiple-choice tests. In *Proceedings of the HLT-NAACL 2003 Workshop on Building Educational Applications Using Natural Language Processing*, pages 17 – 22, Edmonton, Canada, May 2003.

[13] M. Narita. Constructing a tagged E-J parallel corpus for assisting Japanese software engineers in writing English abstracts. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, pages 1187 – 1191, Athens, Greece, 31 May – 2 June 2000.

[14] C. Orăsan and L. Hasler. Computer-aided summarisation - what the user really wants. In *Proceedings of the 5th edition of the International Conference on Language Resources and Evaluation (LREC2006)*, Genoa, Italy, 24 – 26 May 2006.

[15] C. Orăsan, R. Mitkov, and L. Hasler. CAST: a Computer-Aided Summarisation Tool. In *Proceedings of EACL2003*, pages 135 – 138, Budapest, Hungary, April 2003.

[16] C. D. Paice. The automatic generation of literature abstracts: an approach based on the identification of self-indicating phrases. In R. N. Oddy, C. J. Rijsbergen, and P. W. Williams, editors, *Information Retrieval Research*, pages 172 – 191. London: Butterworths, 1981.