Tailoring Text Using Topic Words: Selection and Compression[*]

Timm Euler
University of Dortmund, Computer Science VIII
D-44221 Dortmund, Germany
Email: euler@ls8.cs.uni-dortmund.de

Abstract

In the context of unified messaging, a textual message may have to be reduced in length for display on certain mobile devices. This paper presents a new method to extract sentences that deal with a certain topic from a given text. The approach is based on automatically computed lists of words that represent the desired topics. These word lists also give semantic hints on how to shorten sentences, extending previous methods that rely on syntactical clues only. The method has been evaluated for extraction accuracy and by human subjects for informativeness of the resulting extracts.

1. Introduction

One of the tasks in unified messaging is to adjust texts to various display formats of receiving devices. Consider the aim of reducing urgent email texts such that they can be sent as short messages in SMS format (limited to 160 characters). For this task, it is necessary to distinguish important information from less important information within the original text, where different users will have different notions of what is important for them. Thus, a combination of information filtering and text reduction is needed.

The corresponding research area is domain-specific text summarization, a specialization of generic summarization, where mostly extraction from the original text, rather than reformulating content, is the method of choice. For filtering tasks, models for a user's thematic focus are needed, which have been comparatively complex in information extraction and filtering, while domain-specific summarization approaches have mostly applied queries from information retrieval systems, which tend to be simple and short.

In this paper, a combination of a more complex user model with content-based text reduction is examined. A user specifies her interests by marking texts or text passages as being interesting or not. From this collection, a weighed word list ranking the most important words for the topics of interest is computed. Using the word weights as a user model, sentences from the original text are selected if their weight sum is above a certain threshold. This amounts to a filter for interesting text passages. For further reduction, as needed for 160 available characters, the sentences themselves are shortened using the results of a shallow parser for syntactic and the word list for semantic hints on where to cancel words or phrases.

This approach was applied to filter texts and convert a collection of emails to SMS messages. Evaluation was done (a) by comparing the extracts with human labeling of relevant passages, using standard measures like precision and recall, and (b) by questioning human subjects in a systematic way to judge the informativeness of the resulting messages. The latter type of evaluation is notoriously difficult to do for generic summaries, as the quality of a summary depends on the purpose for which it is used, which is not usually known beforehand. In contrast, specific summaries address a known information need and can be evaluated using specific questions.

In the following, section 2 relates previous work to this approach. Section 3 describes how to gather word lists from marked texts, while section 4 shows how they can be applied to select and compress sentences, and to form extracts of limited length. Section 5 describes the data and experiments, including the evaluation by humans. Section 6 gives concluding remarks.

2 Related work

In text summarization, it is a standard approach to select sentences from the original text for the summary based on characteristic words [10]. Such words and ways to find them are of special interest here. One approach is to use lexical knowledge from sources such as WordNet [11]; compare [2]. Another is to use statistical information gathered from a corpus [1]. A well-known measure that relies

[*] To appear in: Proceedings of 3rd International Workshop on Natural Language and Information Systems (NLIS) 2002, IEEE Computer Society Press

only on the given text collection is *Tf-Idf* (used, for example, in [3, 4]), which measures how characteristic a word is for its text.

Considering domain-specific summarization, words that are characteristic for a given topic, not for a text as a whole, must be found. In the context of Information Retrieval, words from the user query indicate the topic for specific summarization [13, 1], but only coarsely as queries tend to be rather short.

Sentence compression has only recently been explored. In most approaches, syntactical clues were used to find phrases within a sentence that can be cancelled [7, 9]. However, as will be argued in section 4.2, important information can show up in many different syntactical contexts, so that semantic hints for cancelling are needed. In [8], lexical links between words are used in addition to syntactical information. The links provide information on how related the word in question is to the main topics of its text, or to the user query. The next section introduces topic-specific word lists which provide extended and more precise semantic information for sentence compression.

The idea to convert (for example) emails to shorter messages motivates the work in [5], where even single words are shortened by deleting certain vowels.

3 Word lists

To be able to do specific summarization, the system must know about a user's interests. An easy way to achieve this is to learn from examples that the user provides. This is done by marking texts or text passages as relevant or not. While only labeling whole texts is less work, a more fine-grained procedure renders more precise calculations of word weights, as many relevant texts will also contain less interesting passages. Marking each paragraph or even each sentence thus requires fewer texts to achieve the same precision, but amounts to more work for the user. With a suitable environment, where marking interesting sentences is reduced to a mouse click, this may be not unreasonable. However, the methods presented here work with each granularity and in the following, *passage* is used as a synonym for *text*, *text passage* and *sentence*.

Before calculating word weights, a reduction of words to stems may be performed, so that different word forms each count as an instance of the same word. In the experiments for this paper, which used German texts, skipping stem reduction did not lead to good results, but this may be different for less inflectional languages like English.

Several ways to compute a word ranking are possible. Of those tried for this work, only the most simple and successful one is described here; refer to [6] for a description of alternatives, which include G^2, information gain and word weights gained from a trained Support Vector Machine.

Given collections of p relevant and n irrelevant passages, let f_p and f_n be the absolute frequencies of a word in them, respectively. Then its weight w is computed as

$$w = \frac{p/f_p}{n/f_n}\,. \tag{1}$$

This weight can be interpreted as an approximation to the probability that a passage containing this word deals with the relevant topic. Note that the word weights only have to be computed once as long as the topic of interest does not change. Also note that words are not separated from their context: if a word has two meanings, of which only one plays a role in the relevant passages, then it will not receive a high weight if it also occurs (with the other meaning) in the other passages. Thus, problems of polysemy and, by a similar argument, synonymy are moderated.

Every word in the corpus receives a weight, but only words with high weights are considered relevant for the topic, so that the list is cut off at a certain length, for example at 10% of the whole list.

4 Sentence selection and compression

This section describes how word lists can be used to reduce a text to a desired length, by selection (4.1), compression (4.2) and ordering of sentences (4.3).

4.1 Selection

Using the truncated word list, every word of an unseen sentence is assigned its weight (zero if it is not on the list). For each sentence, the sum of its word weights is divided by its word count so as not to bias for longer sentences. The result is compared to a threshold to decide whether the sentence should be in the domain-specific extract.

To find the right threshold for a given word list, several candidate thresholds between a minimum and a maximum are tested on the collection of labeled texts. If the labeling was done sentencewise, recall and precision for the sentence extraction can be computed, and the threshold that gains the highest F_1 value (an average of recall and precision) will be chosen. If the labeling was done textwise, the number of sentences from relevant texts above the candidate threshold minus the number from irrelevant texts has to be maximised to choose the best threshold. In this way, the threshold is adjusted to a given word list with its distribution of weights.

The threshold may also be an instrument to tune the method to a user's preferences. A high threshold will lead to the extraction of almost only relevant information, while some relevant sentences will be lost. A lower threshold achieves better recall, meaning that only little relevant information is lost, but more irrelevant sentences are presented. Between these two, a compromise is usually unavoidable.

4.2 Compression

For a 160 character display, even selecting sentences from a longer text does not always achieve enough reduction. Since there are usually more informative and less informative parts within a sentence, it is possible to compress sentences further. However, this must be done in a careful way so as not to destroy the structure of sentences, making the extract unreadable. For example, the verb is central in every sentence both syntactically and semantically and should never be deleted. The same is true for negations.

The results of a shallow, robust syntactical parser give first hints on what parts of a sentence are more important than others. But consider the following example sentences, which all express the same information which might be urgent to some user, namely that an appointment is cancelled.

(1) `On Friday, the meeting at 9 a.m. is cancelled.`

(2) `On no account can I make it Friday morning.`

(3) `Unfortunately, Friday is impossible for me.`

(4) `A cancellation of the Friday meeting is unavoidable.`

The crucial information is in the verb in (1), in the adverbial `On no account` in (2), in the subject attribute `impossible` in (3) and in the subject itself in (4). Basically, relevant information may show up in any syntactical structure. With the list of topic words, it is possible to cancel only parts of a sentence that do not contain topic-related words, making it more likely that important information is retained.

On the other hand, the syntax analysis provides a useful tool to consider different levels of reduction [7]. These levels allow to reduce a sentence flexibly as far as is necessary, but not further. Table 1 shows the different levels that were used here; they are determined by the structures that the shallow parser for German used in the experiments (section 5) provides, namely a linear sequence of phrases without dependencies between them.

Table 1. Reduction levels

Level	Delete if weight is zero
0	(no deletion)
1	stop words
2	articles
3	adjectives
4	adverbial phrases
5	prepositional phrases
6	noun phrases
X	subordinate clauses

Level 1 also includes the replacement of certain words by common abbreviations. The level marked 'X' is used in a special way: it might not be most useful to use it as a last resort (level 7), but earlier if a subordinate clause does not contain any topic words; then the remaining sentence is strongly reduced without a big impact on its readability. On the other hand, it might be unnecessary to remove a whole clause if a reduction to level one or two was enough. Empirically, it was found that the average reduction achieved by levels one through six *without* clause deletion is one half on the extracted sentences (from the corpus described in the next section). Therefore, clause removal is done if the summary length without any compression is more than double the size of the desired summary length, and it is done after level 2, because at this point, the meaning of the sentence is hardly damaged yet. Other solutions could easily be implemented.

This scheme was tested on German texts (see section 5); the following is an artificial English example meant to give an impression of what the reduced sentences look like at different reduction levels. In (5), all levels have been applied, in (6) reduction was stopped after level 4, and (7) is the original sentence. The character ˆ alerts readers to one or more deleted words.

(5) `ˆwhat about havingˆon MO then?`

(6) `ˆwhat about having lunchˆon MO then?`

(7) `Hello, well what about having lunch together on Monday then?`

Note that `on Monday` is a prepositional phrase but is not cancelled as it receives positive weight from the word list; `Monday` is replaced by its common abbreviation `MO`.

Deletion of noun phrases is in most cases too radical (compare example (5)) and produces unreadable sentences. However, in some cases it might be enough to present certain factual fragments like times, dates or money amounts in the extract and this would be achieved on level six. While this could also be done using information extraction technology, it is possible to set a maximum reduction level according to user preferences here.

4.3 Final extract formation

For extracts with a desired maximum length, such as SMS messages, the order of presenting the selected sentences becomes an issue since it may not be possible to accommodate all sentences, even when compression is used.

It may also be necessary to accommodate meta information, such as sender and subject when reducing emails. After selection, there are two possible ways for ordering: (a) by using the original order, (b) by using the rank of the sentences as given by the sum of word weights. The latter order is based on the heuristic that sentences with higher weight contain more, or more urgent, information; the former leaves anaphorical references untouched (unless they refer to unextracted sentences). Both are combined here.

Since sentences should not be compressed more than necessary, they are compressed level by level in an outer loop, starting with level zero (no deletion) and ending with a prespecified maximum level, atmost six. Within the loop, an extract is formed by using those compressed sentences that have highest weight, but in their original order, while the maximum extract length is respected. If the last sentence does not fit completely, it is cut off as necessary. If all sentences fit in the extract, the procedure is stopped; otherwise the next reduction level is tried. If the last level is reached and not all sentences fit, the last extract in which some sentences are missing is returned.

This method allows to trade off readability against informativeness or vice versa: if readability is prioritized, the maximum reduction level should be set low, but this will leave some sentences out of the extract in some cases. By setting a high maximum level, many sentences will fit in the extract but be considerably compressed; however they may still provide enough information to a reader who knows the context of a message, for example.

5 Evaluation

This section describes experimental data (5.1), the experiments on selection accuracy (5.2) and how the informativeness of reduced texts was evaluated (5.3).

5.1 Data

To test domain-specific extraction, two small sets of German texts were collected that each deal with a common topic. The first set consists of 280 emails (47,000 words) that are related to the scheduling or announcement of meetings. The second set contains 93 newspaper articles (97,000 words) reporting results of public elections. Both sets were complemented by equally many emails and newspaper articles from random domains, respectively, to provide a background for the calculation of word weights. The texts were labeled sentencewise, with 13 and 10 percent of all sentences found topic-related, respectively. For stem reduction, tagging and shallow parsing, the German NLP tool Meson, successor of Smes [12], was used.

Both the domains chosen are comparatively suitable for information extraction tasks: certain structural items like times, dates and percentages are rather salient, but these items also show up in other domains like business reports. A look at the resulting word lists reveals that times and dates—but not only these—are weighted very high in the appointment scheduling domain, while the word `percent` receives a rather low weight in the election results domain, where the best indicators are `mandate`, `vote` etc. The evaluation bias due to easily recognizable items is thus limited.

Table 2. Sentence selection results

Texts	Recall	Precision	F_1	Fallout
Emails	83.5 ± 5.6	79.2 ± 6.7	81.2	2.3 ± 0.5
Articles	75.4 ± 6.5	69.6 ± 7.2	71.9	3.5 ± 0.2

5.2 Selection accuracy

The main criteria for successful selection are recall, precision and F_1, measured at sentence granularity. However, the precision value depends on the proportion of irrelevant texts in the collection, as more irrelevant texts increase the likelihood of wrong positive classifications. Therefore, *fallout* is used in addition, which is defined as the ratio of wrong positive classifications to negative (irrelevant) sentences. Low fallout means that little irrelevant information is presented.

Table 2 shows the results on the two collections using 10-fold cross validation, including standard deviations. Extraction was easier in the appointment domain, but is satisfactory for the election results as well. Fallout is very low for both.

In a second experiment, the word lists were computed using only a textwise labeling (see section 4.1). This lead to 76.5% F_1 for the sentences from emails and a rather low 59.5% F_1 for the election results, probably due to the smaller size of this collection. Thus, labeling texts rather than sentences is sufficient for the word list method if there are enough texts.

5.3 Extract informativeness

As mentioned in the introduction, specific summaries are easier to evaluate than generic ones because it is possible to set up specific questions to readers. Such an intrinsic evaluation was done for SMS messages with 160 characters that resulted from reducing the appointment-related emails. SMS messages were formed from 50 emails using little, medium and strong reduction (up to levels two, five and six, respectively) and 50 informants were presented with different original and reduced texts each (nobody was shown two versions of the same text). For each text, they were asked what kind of a meeting the text dealt with, when and where it was supposed to take place, who takes part, whether the meeting was being announced, canceled, confirmed etc., and how understandable they found the text, on a scale of 1 to 5.

Not the correctness of answers was measured, but differences between answers based on short messages and those based on seeing the whole email. It was found that the time of meeting survived almost all compressions. The "status" (confirmation, cancellation, etc.) was inferrable from 92% of emails and 86%, 79% and 74% of SMS messages based on little, medium and strong reduction, respectively. Information concerning the place and the participants of a meeting is rare in the original texts and hardly ever survives compression, as proper names never receive a high weight. All in all, 70% of the questions could be answered on average for emails, but only 43% for strongly reduced messages. Readers gave an average 4.3, 3.1, 2.4 and 1.8 for intellegibility of emails and little/medium/strongly reduced extracts, respectively.

Taken together, this means that important information (time and status for appointments) is often preserved in the reduced messages, but strong reduction should be avoided as too little information can be inferred from the resulting texts. It seems better to leave some sentences out of the extracts and to compress the others only lightly, to preserve readability.

6 Concluding remarks

In this paper, it was shown how ranked lists of words related to a topic can be used to filter and reduce texts with respect to this topic. No explicit knowledge about semantic relations between words on the list is needed; instead, a semantic relation is assumed implicitly between words with high weight as they are significant for the positively labeled texts. However, methods other than the one presented here can be used to gain such lists. Further, several lists can be applied in parallel, so that users can recycle each other's lists and build filters for their own range of interesting topics. Word lists can also be used to semantically extend syntactic methods to compress sentences, allowing stronger reduction for certain purposes.

Acknowledgements

I would like to thank Prof. Dr. Katharina Morik and Ralf Klinkenberg for their valuable support.

References

[1] B. Baldwin and T. S. Morton. Dynamic coreference-based summarization. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing (EMNLP-3)*, Granada, Spanien, June 1998.

[2] R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In I. Mani and M. T. Maybury, editors, *Advances in Automatic Text Summarization*, pages 111–121. MIT Press, Cambridge, MA, 1999.

[3] R. Brandow, K. Mitze, and L. Rau. Automatic condensation of electronic publications by sentence selection. *Information Processing and Management*, 31(5):675–685, 1995.

[4] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Seeing the whole in parts: Text summarization for web browsing on handheld devices. In *Proceedings of the Tenth International World Wide Web Conference*, 2000.

[5] S. Corston-Oliver. Text compaction for display on very small screens. In *Proceedings of the Workshop on Automatic Summarization at NAACL*, June 2001.

[6] T. Euler. Informationsextraktion durch Zusammenfassung maschinell selektierter Textsegmente. Technical report, LS8 Report 27, Fachbereich Informatik, Universität Dortmund, 2001. In German.

[7] G. Grefenstette. Producing intelligent telegraphic text reduction to provide an audio scanning service for the blind. In *Intelligent Text Summarization, AAAI Spring Symposium Series*, pages 111–117, Stanford, California, 1998.

[8] H. Jing. Sentence simplification in automatic text summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference (ANLP)*, Seattle, Washington, May 2000.

[9] K. Knight and D. Marcu. Statistics-based summarization—step one: Sentence compression. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI)*, pages 703–710, 2000.

[10] I. Mani and M. Maybury, editors. *Advances in Automated Text Summarization.* MIT Press, 1999.

[11] G. A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[12] G. Neumann, R. Backofen, J. Baur, M. Becker, and C. Braun. An information extraction core system for real world german text processing. In *5th International Conference of Applied Natural Language*, pages 208–215, Washington, USA, March 1997.

[13] A. Tombros and M. Sanderson. Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2–10, 1998.