Text Summarization of XML documents in Croatian

Nives Mikelić, Tomislava Lauc, Damir Boras
Research Assistant, Docent
University of Zagreb, Faculty of Philosophy, Department of Information Sciences, I. Lučića 3
nmikelic@ffzg.hr, tlauc@ffzg.hr , dboras@ffzg.hr

Abstract. *The paper describes automatic summarization of the XML documents in Croatian language. The goal of the summarizer is to generate extracts with high percent of extract-worthiness and similarity to the author's abstract. Our research shows that extracts generated using our algorithm are well formed, but it also shows that algorithm is very domain dependant.*
The research brought us to conclusion that we should develop the implementation of the Porter's stemming algorithm in order to improve the text summarization for Croatian language.

Keywords. Automatic summarization, XML documents, Croatian language, Perl

1. Introduction

The goal of automatic summarization is to take an information source, extract content from it, and present the most important content to the user in a condensed form and in a manner sensitive to the user's or application's needs.

Automatic text summarization has been under development for many years, and there has recently been much more interests in it due to the increased use of Internet. For example, this technique can be used to summarize news to SMS or WAP-format for mobile phone, or to let a computer synthetically read the summarized text because the written text could be too long and tedious to listen to. Also, summarization can be used in search engines in order to present compressed descriptions of the search results or in keyword directed news subscriptions of news which are summarized and sent to the user. Finally, it can be used for searching in foreign languages in order to obtain an automatically translated summary of the automatically summarized text.

The word summary is associated with a variety of meanings and is used in a variety of contexts. Depending on the output one can have extract- or abstract-like summaries.

Abstracts of scientific articles represent the interpretation of the original text. The process of producing it involves rewriting the original text in a shorter version by replacing wordy concepts with shorter ones. Implementation of abstract methods requires symbolic world knowledge which is too difficult to acquire on a large enough scale to provide a robust summarization. On the other hand, extract is therefore a summary extracted from the original text on a statistical basis or by using heuristic methods or a combination of both. The extracted parts are not syntactically or content wise altered, but are the most representative of relevant information.

Furthermore, depending on the usage, a summary can be indicative or informative.

An indicative summary can provide only an indication of the main topics in the input text and it aims to help the user to decide whether to read the information source, or not. On the other hand, an informative summary covers all the salient information in the source at some level of detail. The summaries generated by our algorithm are all extracts that aim to give the clue about the main topics in the article, but also to extract the most relevant information from the article.

2. The system for XML document summarization

Many different approaches have been proposed for text summarization. Luhn [4] utilized word-frequency-based rules to identify sentences for summaries, based on the intuition that the most frequent words represent the most important concepts of the text. Edmundson [3] incorporated new features such as cue phrases, title/ heading words, and sentence location into the summarization process, in addition to word frequency. The ideas behind these older

approaches are still used in modern text extraction research.

In this paper, we describe the summarization of the scientific papers, since in the scientific literature words are mostly unambiguous. The important words are repeated throughout the text and therefore most of the relevant information should be included in the extract, which is a kind of indicative summary that helps readers to judge the relevance of the associated document and decide whether or not the full text is worth reading.

Although sentences in extracts may lack cohesion if they contain the anaphoric reference or the topic shifts, these problems can be solved through post-processing. Extracts proved to be very useful for people to form an opinion of the context of the original scientific paper.

2.1 Program Overview

Our summarization system is fairly linear and it consist of seven sections, where the first one checks the program arguments, the second one extracts sentences from XML file, while the third one weights each sentence according to the given rules. Each sentence is weighted according to the location criterion:

- beginning of paragraph
- end of paragraph
- title
- heading

Titles and headings are considered to be of infinite weight. As a result they will always be chosen first. Their weight is defined as a large number and that makes impossible for some other sentence to accumulate enough weight to print first. Sentence location feature is based on the hypothesis that sentences occurring after the titles should be relevant and that topic sentences tend to occur at the beginning of a text and a new paragraph.

Furthermore, the fourth section creates a hash frequency table of all words, after which, in the fifth section, all stop words are being removed from the frequency table.
The list of stop words is checked and occurrences removed from the frequency table. Stop words are grammatical words like prepositions (u, na o, po), conjunctions (i, ali), adverbs (kako, tako), pronouns (ja, ti, on) with all of the case forms (meni, njoj), auxiliary verbs (biti, htjeti) and modal verbs (moći, smjeti) with all of their forms in all verb tenses.

In the sixth section the top percentage of most frequent words that left over are extracted and sentences are weighted again according to the given rules. (According to Luhn's method [4], frequency of word occurrence in an article seems to be a useful measurement of word significance.)

All the sentences (apart from title and header) are weighted according to the following rules:

- title word
- heading word
- keyword

Title words are given higher values than header words, as these are normally what the text is about, similarly for heading words and keywords. A certain percentage of most frequent words in the text are used as the set of keywords. This percentage, as well as all the weights, is set manually and testing needs to be performed in order to determine the best weights for a certain domain.

Sentences are then weighted according to cue phrases and stigma/bonus words.

The cue phrases are considered as very useful indicators for locating important sentences in the paper. Existing methods of identifying cue phrases in English are usually frequency based i.e. high-frequency cue phrases are identified in the original paper.

We analyzed the scientific papers and their abstracts from the Croatian Scientific Bibliography database and looked for the cue phrases such as "in this paper" and "we conclude".

Unfortunately, we discovered that authors tend to use a huge variety of the cue phrases and it is hard to say which of them have common use along the corpus since the frequency count was very low. Frequency based methods are, as we realized, not that effective in Croatian, because they can not identify low-frequency cue phrases, which we found to be of high relevance in text summarization for Croatian.

Therefore, we investigated the possibility of exploiting co-occurrence method rather than frequency. Co-occurrence method is based on the observation that cue phrases and keywords appear in the same sentences.

Also, we will look for the bonus words, such as "Significant" or "Greatest" [5], which should positively affect the relevance of a sentence and also for the stigma words, such as "Impossible" or "Hardly", which should affect the relevance of a sentence negatively.

Then, if the sentence contains cue phrase and bonus phrase as well, we add a score to the sentence, or we can penalize it if it contains a stigma phrase.

This combined method [2] definitely outperforms the frequency-based method.

Finally, in the seventh section, the top percentage of sentences (as specified by user) are sent to the XML parser that outputs sentences from the original source file to preserve capitalization and punctuation lost in the weighting calculations. This simple parser is by no means perfect, it ignores most tags, but for the preliminary research it achieved fine results.

2.2 Problems/Difficulties

The program obviously has limitations. The sentences chosen could be further rectified with more cue phrases or additional stigma/bonus words, which would just require finding them manually. Although this approach works well, it can be very domain dependant (e.g. scientific research papers differ quite a lot from newspaper articles, so a stigma word relevant to one may not reflect in the other).

Also, deciding on the weighting to give each rule is difficult and much more testing would produce better values.

After the preliminary testing, the highest weight is assigned to title and headers; it is followed by weight for cue phrases; on the third place are title, header and bonus words which are assigned similar weights, keyword is allocated half of the header word weight, while beginning of paragraph and end of paragraph have double weight of the header word.

The length of a sentence seems to be important, as some sentences are obviously longer than other and naturally more likely to have more keywords than some smaller sentence. On the other hand, a long sentence hopefully contains more information than a short one, but not always.

3. Document pre-processing

All the documents are in Microsoft Word format. In the first step of pre-processing the Visual Basic macro program removes footnotes and automatically denotes paragraph beginning and paragraph end with the XML tag. Paragraph mark is replaced with paragraph tag: <P> for beginning of the paragraph and <\P> for paragraph end.

Also, Microsoft Word styles (combination of formatting characteristics, such as font, font size, and indentation, that are automatically named and stored as a set) for Title and Headers (Header 1 to Header9) are replaced with corresponding XML tag <TITLE>, <HEADERID=1>, <HEADERID=2>…etc.

Furthermore, all sentences are marked with XML tag for sentence. Sentence ID increases for each new sentence.

Beginning of the sentence in our program is defined with the capital letter that follows white space and stop/question/exclamation mark or with the manual line break mark that follows stop/question/exclamation mark. (Question mark and exclamation mark can appear at the end of sentence, but it is not so likely in the scientific papers.)

One of the problems is that abbreviations or years can sometimes be misinterpreted as a sentence end. Therefore, we have to replace all the abbreviations with the full words if the abbreviation does not represent a real sentence end. Also, if the year ends with stop mark (*e.g.1960.*), the stop mark is removed if it does not represent a real sentence end. Dictionary of abbreviations is taken from the PhD thesis *Theory and rules of automatic text segmentation in Croatian language* and it contains 467 abbreviations. Although most of the abbreviations were replaced correctly, we still encountered some problems that could be sorted only by further processing (e.g. *dr*. represents both *doktor* and *drugo*).

The other problem is errors the authors made in the original paper: double white space before stop mark, inconsistent citations, unsystematic notation of quotations, etc. Unfortunately, this problem can only be solved manually.

After the title, headers, paragraph and sentences are clearly marked with XML tags, another Visual Basic macro program replaces characters č, ć, đ, ž and š with cx, cy, dy, zx and sx. This step is performed because Perl module that extracts sentences in summary at this moment does not support Unicode. After the extracts are acquired, Visual Basic macro program returns the original characters to the text of the extract.

Furthermore, the list of the stop words: grammatical words like prepositions (u, na o, po), conjunctions (i, ali), adverbs (kako, tako), pronouns (ja, ti, on) with all of the case forms (meni, njoj), auxiliary verbs (biti, htjeti) and modal verbs (moći, smjeti) with all of their forms in all verb tenses was extracted from the Lexical

Database of the Croatian Language [1] automatically.

The preliminary text corpus used for producing extract summaries comprises articles from the Proceedings *Models of Knowledge and Natural Language Processing*, published on the Department of Information Sciences at the Faculty of Philospohy in 2003.

Finally, extract summaries were compared to the authors' hand written summaries (summary length in words is narrowly distributed around 150-200 words per summary). The results obtained are given in the next section.

```
TITLE:  Leksička infleksijska baza podataka svih hrvatskih imena i prezimena
H:  Infleksijska baza osobnih imena za hrvatski jezik
H:  Pravila slaganja osobnih imena i prezimena
H:  Poteškoće pri pripremi baze podataka
H:  Obilježja hrvatskog jezika
H:  Morfološka obrada
H:  Zaključak
H:  Uvod
  U ovom su radu opisana struktura i izrada infleksijske baze podataka hrvatskih
imena i prezimena (za pisani jezik), njezin paradigmatski model te njezina
moguća primjena u sustavima za pretraživanje podataka, sustavima za segmentaciju
teksta, korektorima pogrešaka, te sustavima za gramatičku analizu teksta.
  Ukoliko se nekom imenu nije mogla pridružiti nijedna postojeća paradigma,
dodavala se nova, tako da se na kraju pojavilo sveukupno 38 paradigmi, od čega 6
za ženska imena, 27 za muška te 5 isključivo za prezimena, iako su se, naravno,
prezimenima pridruživale i paradigme osobnih imena.
  Primjena ove baze moguća je, osim kao tvorbene baze za izvođenje svih mogućih
padežnih oblika hrvatskih osobnih imena, odnosno kao sustava za pronalaženje
svih oblika za određeno osobno ime i kao modul za prepoznavanje osobnih imena u
sustavima za pretraživanje teksta [12], kao dodatni izvor za pripremu hrvatskog
korektora pogrešaka (spelling checker), kao sustav za pripremu normativnih
datoteka imena i prezimena u bibliografskim i leksikografskim primjenama, a
također i kao izvor različitih statističkih podataka i sredstvo za određivanje
morfoloških i gramatičkih svojstava hrvatskoga jezika koja se odnose na imena i
prezimena u hrvatskome jeziku.<\P>
  Cilj automatske morfološke obrade je automatska morfološka analiza i/ili
generiranje nekih oblika riječi.
  Model je i jezično i strukturalno (kao baza podataka) jednostavan, ali je baš
zato djelotovoran i lako primjenjiv na bogatu morfologiju hrvatskog jezika te
baza može poslužiti i kao generator i kao analizator svih postojećih standardnih
oblika muških i ženskih imena i prezimena koja se pojavljuju u Republici
Hrvatskoj.<\P>
```

Picture1. Example of the 9% extract summary

4. Evaluation

Evaluation was done on the corpus of the scientific abstracts and comparison was performed with authors' abstracts. Five experiments were performed.

The weight of a sentence in the first experiment is a linear combination of the title, header, location and keyword weights, where stopwords are eliminated.

The second experiment combines title, location and header weights again, but it also includes bonus and stigma words, while the stop words are not eliminated.

The third experiment combines all four weights from the second experiment, but stop words are excluded as well.

The fourth experiment combines cue words instead of bonus or stigma words with all other weights, while the fifth experiment combines both bonus/stigma words and cue phrases with the other weights.

The preliminary results with extracts that differ in length and combination of features were obtained.

Furthermore, when authors' summaries were collected, we noticed that the human selection of sentences in abstracts is very variable. Abstracting the same document two times by the same person with only a few weeks in between, we gained only 60% overlap.

Comparing the extracts of different size and different combination of weights in these five experiments, we found out that 2 out of 9 extracts had the best retention ration and were the most similar to the author abstract in the third experiment. The other seven extracts achieved the best retention ration in the fourth and fifth experiment (actually, results were equal in the 4th and 5th experiment), although 2 out of these 7 extracts had very bad results in third experiment. Furthermore, two out of nine extracts had a good retention ratio, but not a single sentence from the extract was included in the author's abstract. The reason for this is that sentences in the author's abstract were not contained in the body of the article. Also, authors were writing abstracts using the phrases such as: *author claims, author describes, author explains*, that are obviously impossible to be found in the body of the article written by that author.

In order to avoid this situation, it would be wise to give authors clear instructions before they start writing their abstracts.

Hence, we can conclude that all those extracts and abstracts contain the same significant terms and present the most important content, in spite the fact that they consist of different sentences.

Table1. Percentage of similarity between extracts obtained in the experiments and author's original abstract.

Author's Abstract	Extract
Abstract 7	78%
Abstract 4	76%
Abstract 5	74%
Abstract 1	63%
Abstract 2	57%
Abstract 9	48%
Abstract 3	43%
Abstract 6	30%
Abstract 8	25%

Analyzing the obtained extracts, following characteristics were identified:

- Summary length is definitely dependent of document length (Summary length of a document that contains 3366 words is 213 words or approximately four sentences where the summary percentage is 9%, while the summary. On the other hand, summary length of a document that contains 5478 words is 405 words or approximately 17 sentences.)
- Extract summaries generated using the title, location, header and cue phrase weights (stop words are eliminated) are not different from extract summaries that use the same weights, but also the weight for bonus/stigma words
- Extract summaries generated without stop words, using the title, location, header and bonus/stigma word weights are different from extract summaries generated on the same base, but also with cue phrases weight
- Extract summaries where stop words are eliminated differ very much from extract summaries that still contain stop words, actually, they have much higher retention ratio than the latter summaries
- extracts which summarize the article down to 1-5% appear to be too short to be compared to the authors' abstracts, although the extract worthiness is quite high because they consist of title and headers

Results are expressed as the document vector similarity between an abstract and an extract document. Nearness between authors's abstract and extract obtained by our summarizer is calculated in three steps: list of words is created for each document, word lists are merged together, stop words are removed, the lemmatization is performed and words that are common are extracted in the output. Finally, nearness (document similarity) is expressed as the fraction, which has *number of word common to document pairs* as numerator and *number of words in the shorter of the two documents in the pair* as denominator.

The similarity percentage between extracts and abstracts are presented in Table1.

Another evaluation criterion was extract-worthiness or retention ratio. Results obtained for both the high percentage extracts (10-15%) and the low percentage extracts (5-8%) show that over 90% of the sentences selected are extract-worthy; in other words, extract-worthiness does not get lower with the higher compression ratio, as one may suspect. Also, 1% and 2% extracts contain only the title of the document, while extracts in the range of 2-5% include title and headers as well.

5. Conclusions and future work

Our aim was to generate an extract with the high retention ratio and about the same size as the author's abstract. Extracts are obtained by selecting sentences in the original text. Sentence selection is achieved in two steps. Firstly, each sentence in the text is assigned a score using some features to yield a salience function for sentence selection. Secondly, sentences are ordered in a summary according to their ranking and a predefined number of highest weighing sentences are included in the extract.

We believe that the key facts in a scientific paper are expressed with a range of related words, while redundant information is presented with terms that are not related to the main subject given in the title or the header of the article.

In order to improve our system we plan to keep track of some visual presentation details in the original article, such as font size of words. Words in a larger or bolder font could be weighted higher than other words.

Also, in order to improve the current system, we plan to implement the Porter's stemming algorithm for Croatian language. Up to this point, we have tested the Perl implementation of the Porter's algorithm for English language. The Porter stemming algorithm (or 'Porter stemmer') is a process for removing the commoner morphological and inflexional endings from words in English. The algorithm was originally described in Porter, M.F, 1980. *An algorithm for suffix stripping*, Program, 14(3):130-137. We downloaded the Perl version of the algorithm from the official home page for distribution of the Porter Stemming Algorithm (http://www.tartarus.org/~martin/PorterStemmer) and implemented it into our summarizer.

Finally, we tested it on the English version of one article from the Proceedings Models of Knowledge and Natural Language Processing.

The extract obtained for English encouraged us to implement the same algorithm for Croatian aa well.

6. References

[1] Boras D. Teorija i pravila segmentacije teksta na hrvatskom jeziku. PhD thesis. Zagreb, 1998.

[2] Daille B. Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In: Klavans J, Resnik P, editors. The Balancing Act: Combining Symbolic and Statistical Approaches to Language. Cambridge: MIT Press; 1996. p. 49–66.

[3] Edmundson, HP, Wyllys, RE. Automatic Abstracting and Indexing - Survey and Recommendations. In: Communications of the ACM 1961; 4(5): 226-234.

[4] Luhn H.P. The Automatic Creation of Literature Abstracts in IBM Journal of Research & Development 1958; 2 (2): 159-165.

[5] Paice CD, Jones PA. The Identification of Important Concepts in Highly Structured Technical Papers. In: Proceedings of the 16th International Conference on Research and Development in Information Retrieval (SIGIR'93); 1993. p. 69–78.

[6] Teufel S, Moens M. Sentence Extraction and Rhetorical Classification for Flexible Abstracts; 1998. http://www.cl.cam.ac.uk/users/sht25/papers/aaai98.pdf [4/21/2005]