

# An evaluation of structure-preserving and query-biased summaries in web search tasks

F. Canan Pembe and Tunga Güngör

Department of Computer Engineering, Boğaziçi University

Bebek 34342, İstanbul, Turkey

E-mail: cananp@yahoo.com, gungort@boun.edu.tr

***Abstract:** Automatic summarization has started to receive increasing attention in recent years due to the increased amount of information available in electronic form. Especially, summarization techniques can be very useful in improving the effectiveness of information retrieval on the World Wide Web. However, currently available major search engines such as Google show only a limited capability for summarization. We believe that text summarization with more sophisticated techniques can significantly improve the search experience of users. As a novel approach, we propose a query-biased summarization system which incorporates the structure of documents into the summaries. The system also makes use of natural language processing techniques in the summarization process. The effectiveness of the proposed system has been tested on a task-based evaluation.*

***Keywords:** automatic summarization, human-computer interaction, information retrieval, natural language processing*

## 1. Introduction

### 1.1. Automatic Summarization

Automatic summarization is the automated process of distilling the most important information from a source (or sources) to produce a shortened version for particular users and tasks [1,2,3]. Early research in this field began in the late 1950s with surface-level approaches [4,5]. There has been an increasing attention to this field from academia and industry in recent years. The reason is the rapid growth of accessible information sources in digital form, especially the World Wide Web. People now have access to large amounts of information; however, there is no time to read everything individually, widely known as the problem of information overload. The idea of

creating summaries as successful as humans is still a long-term research direction. However, as stated in [6], in parallel, summaries which are not as perfect as human summaries may be utilized in improving the effectiveness of other tasks. Especially, automatic summaries can be very helpful in information retrieval task realized by search engines, where the aim is to find relevant documents in large collections in order to satisfy user requests (queries) for particular items of information.

### 1.2. Motivation

Currently available major search engines, such as Google [7] and AltaVista [8], display their results in response to a user query as a ranked list of Web document links together with a short summary of their content (usually two

lines of text fragments). For example, the query *automatic text summarization* returns a huge number of such results. The user needs to scroll down the results and, based on the short summaries, decide on the relevancy of the actual documents to his/her information need. The problem with those summaries is that they are usually inadequate in directing the user to relevant documents. As a result, the users usually miss some of the relevant documents or lose time with irrelevant ones. Besides, loading each document shown in the results area in order to learn about its actual relevance has also disadvantages. Page loading takes time depending on the Internet connection of the user and the determination of the relevance based on the actual document may also be difficult in the case of long and complex documents. Also, it is not feasible for the user to load each document when such large number of results are returned. To overcome these problems and improve the effectiveness of Web search, better approaches regarding the summarization are needed. In this paper, we present a novel approach to summarization of Web documents using both the content and the structure of documents in a query-biased way.

An attempt in the literature to improve the search experience of Web users is WebDocSum [9]. A novelty of that system is the longer summaries it provides in a separate window whenever the user points the mouse on a particular link in the search engine results. The system is based on simple surface-level extraction techniques; that is, sentences are scored and extracted based on features such as title, location, relation to query, and text formatting. The system was tested on a task-based evaluation where its outputs were shown to be more effective than the summaries provided by Google and AltaVista. However, unlike our work, it does not incorporate the structure of documents into the output summaries. Another related work in the

literature uses also the structure of documents in the output summaries [10]. The system builds a “table of content”-like hierarchy of sections and subsections for each document using heuristics on HTML tags present in the documents and incorporates this structural information in the output summaries. However, that system only creates general-purpose summaries, not tailored for particular user queries or Web search task.

### 1.3. Our Approach

In order to improve the effectiveness of Web search, we decided to provide longer summaries than the ones provided by current search engines. However, if these longer summaries were again displayed under the corresponding titles, then the user would need to scroll too much to see consecutive results. To prevent this problem, similar to the previous work in the literature [9], only the titles of each link are listed, and in a separate frame, the summary for that document is displayed when the user moves the mouse on a particular link.

Then, we considered that Web documents are not just flat text and they usually have structure in them; i.e., they consist of some sections and subsections; e.g. the abstract and introduction sections, and corresponding subsections (as in this document). Therefore, in our approach, some of this structural information is incorporated into the summaries in order to help the users to judge the relevancy of each document more effectively. The structure of the documents can be identified using some heuristics on the HTML tags present in the documents; e.g. headings, fonts, boldness, etc., as described in [10]. A novelty of the proposed system is the use of this structural information for providing the context of the text fragments selected as a part of the summary, in a query-biased way, as will be detailed later in the text.

The system also uses natural language processing techniques for summarization

purposes as well as the traditional term-frequency statistics. The documents and user queries are processed in morphological and syntactical levels in order to obtain content-carrying phrases, such as noun phrases, using finite-state techniques. Then, the text fragments with words and phrases similar to the user query are included in the summary according to sentence scoring metrics. The documents are represented in XML-based form in the system since this is a natural representation and is very suitable to the approaches used. The tags include both structural information (e.g. sections, subsections, etc.) and linguistic labels, such as phrases. Then, given a user query, the system creates a summary using this representation.

The rest of the paper is organized as follows. First, the system implementation is given together with the structural and linguistic processing. Then, the evaluation of the system is presented. This is followed by sections on further work and conclusion.

## 2. System Implementation

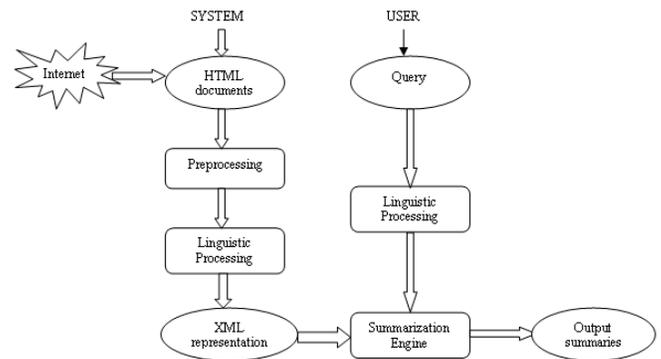
The system architecture is given in Figure 1. Each HTML document in the system is preprocessed in order to obtain its structure such as its sections and subsections. Both the documents and user queries are processed linguistically in order to obtain phrases. For each document, an XML representation is obtained. The user query which is linguistically processed and XML representations of the documents are used by the summarization engine to create the summary outputs.

We used GATE framework for text engineering as the underlying development environment which is an open source project using component-based technology in Java [11,12]. GATE is being used as an infrastructure by many academic and commercial projects. Using such a framework has several advantages because it includes commonly used natural

language functionalities such as part-of-speech tagging and it is a modular environment into which new components can be easily added.

### 2.1. Structural Processing

Currently, most of the documents on the Web are prepared in HTML format. Usually, the documents are not prepared as flat text with all the content in a fixed format. Instead, the documents are usually prepared as consisting of sections and subsections using a limited number of HTML formatting tags including bold (<b>), underlined (<u>), font (<font>), and heading (<h1>, <h2>, <h3>, <h4>, <h5>, <h6>).



*Fig. 1. Overview of the system architecture*

The structure of a document may be considered as a hierarchy where each document may have sections; each section may have subsections, and so on. In the proposed system, the sections and subsections are identified using some heuristics on HTML tags. When a document contains heading tags, it is easier to identify the headings and corresponding subsections. The heading tags <h1> through <h6> usually correspond to headings of different levels in the hierarchy; e.g. <h1> is a first level heading and <h2> is a second level heading. However, in some documents, section headings are displayed using bold and relatively larger fonts. Section headings can also be identified using these relative features. We have implemented a structural analysis module which

is able to identify main sections and subsections of a document using some of the heuristics described.

## 2.2. Linguistic Processing

In the system, both documents and queries are processed linguistically. First, each single word is processed morphologically. Then, part-of-speech tagging is performed for each sentence. For example, consider the text portion “*the analysis of natural language processing systems...*” annotated with part-of-speech tags as in the following:

*the/det analysis/n of/prep natural/adj  
language/n processing/n systems/n ...*

Then, noun phrases, which are important content carriers in almost every natural language, can be identified using finite-state transduction based on predefined patterns. For example, an English noun phrase can be defined as follows where \* is used for multiple occurrences of the components [13]:

NP = det\* pre\* head post\*

In this representation, the components of the noun phrase are:

- det (determiner): e.g. article, number
- pre (premodifier): e.g. adjective, noun
- head: usually a noun
- post (postmodifier): e.g. prepositional phrase

Such patterns based on part-of-speech tags are used to identify the phrases; e.g., the noun phrase *automatic text summarization* is composed of the components *automatic* (premodifier adjective), *text* (premodifier noun) and *summarization* (head noun).

In the implementation, ANNIE English Tokeniser, Sentence Splitter, and Part-of-Speech Tagger provided as a part of GATE [11] are

used to split the text into individual tokens, to divide the text into sentences and to produce a part-of-speech tag on each token, respectively. GATE also has built-in finite-state transduction capabilities. The transducer runs based on grammars written in a language called JAPE (Java Annotations Pattern Engine) language. Then, to identify phrases, only patterns need to be described. In order to identify noun phrases, we used Noun Phrase Chunker provided as a GATE plugin.

## 2.3. Summarization Engine

In the system, retrieved documents are summarized according to the user query. The sentences in a given document are scored based on the frequency counts of terms (words and phrases). The sentences containing the query phrases are given higher scores than the ones containing single query words. Then, the sentences with highest scores are incorporated into the output summary together with their structural context. Portions of text may be extracted from different sections or subsections. The resulting summary is the union of such extracts. The number of extracted sentences and the extent to which their context is displayed depends on the summary frame size which is fixed to the size of the screen that can be seen without scrolling.

We have implemented a summarization engine module which is able to create summaries using the linguistic and structural information as described. An example summary output by the system for the query “*natural language processing and information retrieval*” is given in Figure 2. As can be seen, both the document structure and the highest scored fragments with important phrases according to the user query are incorporated into the summary. Consecutive fragments and headings are separated with dots (...) indicating that more material follows in between. The structure of the actual document as well as its coverage, main

theme, size, etc. is much more explicit compared with two-line summaries. In this way, it is expected that the user can judge the relevancy of each document better than the traditional approaches without the necessity to load the actual page.

```

1. Natural Language Processing
...
1.1. Good Places to Start
...
Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition .
...
1.2. Readings Online
...
Natural Language Processing and Knowledge Representation: Language for Knowledge and Knowledge for Language .
...
In this view, inferencing and other interesting information and knowledge processing tasks are not part of natural language processing.
...
1.3. Related Web Sites
...
- Text mining attempts to discover new, previously unknown information by applying techniques from information retrieval, natural language processing and data mining.
...
2. Resources relevant to the various levels of language processing 3. Other useful links for NLP students, relating to any aspect of Natural Language Processing that might be encountered in an academic course, from the lowest levels of language processing to the highest levels."
...
1.4. Related Pages
...

```

**Fig. 2.** A summary output of the system

### 3. Evaluation

The system was evaluated on an extrinsic evaluation method of summarization where the quality of summaries is determined based on how the summaries affect the completion of another task. Here, the task is to correctly identify the relevancy of a document with respect to a given query (also called a topic) by using the summary. Then, accuracy is used as the performance measure.

For the evaluation, test sets of documents were created for three different natural language queries (Table 1). The documents were taken from the results of Google for each query (30 documents for each query). Then, the subjects were presented first with summaries of Google, then summaries of the proposed system and finally original documents. The subjects were asked to determine the relevance of the original document with respect to the given query (either relevant or not) using the summaries. In the last part, the user performed the same assessment for the original documents whose results were taken as the actual relevance of each document. The assessments of users for the summaries were

compared with the actual relevancies. Here, four different types of results can be identified depending on the accuracy provided by the summary and the actual relevancy: TP (true positive), FP (false positive), FN (false negative), and TN (true negative) as given in Table 2 similar to [14]. Each query was evaluated by more than one subject and the results were averaged.

**Table 1.** Queries used in the evaluation

Query	Content
1	Natural language information retrieval systems
2	Spam mail
3	Environmental pollution prevention

Based on these outputs, precision (P), recall (R) and F-measure (F) which are aggregate accuracy measures can be computed using the following formulas [14]:

$$\begin{aligned}
 (1) \quad & P = TP / ( TP + FP ) \\
 (2) \quad & R = TP / ( TP + FN ) \\
 (3) \quad & F = 2 * P * R / ( P + R )
 \end{aligned}$$

**Table 2.** Accuracy measures used for evaluation

Actual relevancy	Judgment based on summary	
	relevant	irrelevant
relevant	TP	FN
irrelevant	FP	TN

The average results of the initial evaluation are given in Table 3. Considering the individual results of the experiment and the average results, we noticed that the proposed system usually provides a slight increase in the precision whereas higher increase in recall is obtained, compared to Google [7]. A more comprehensive evaluation is left as a future work.

**Table 3.** Average results of the initial evaluation

	Google	Proposed System
--	--------	-----------------

<b>TP</b>	46.67	51.11
<b>FP</b>	26.67	27.22
<b>FN</b>	12.22	11.67
<b>TN</b>	14.44	10.00
<b>P</b>	0.79	0.80
<b>R</b>	0.80	0.83
<b>F</b>	0.78	0.80

#### 4. Future Work

The research can be extended in several directions. First, the scoring method of sentences can be improved. Second, the system can be refined considering different types of search task, such as searching for a particular fact, searching for background information about a subject, etc. Also, the heuristics used in the identification of structural information and incorporation in the output summary can be improved.

Natural language processing techniques used within the system are also open to improvement. One further work is about the use of noun phrases for summarization purposes. Such usage promises an increase in the effectiveness; however, the problem of syntactical variation should be considered. For example, all the phrases *text summarization*, *automatic summarization*, *automatic text summarization* are different syntactical formulations with similar meaning. Therefore, some form of regularization is necessary for phrases; i.e. syntactical normalization [13]. A normalization technique that can be used for this purpose is the extraction of word pairs with head-modifier relation from noun phrases [15]. For example, the phrase *automatic text summarization* can be normalized to the head-modifier pairs “*automatic* (modifier) + *summarization* (head)” and “*text* (modifier) + *summarization* (head)”, which can be easily found using heuristics. Then, such subcompounds can be used in scoring sentences in the summarization stage.

[9] R. W. White, J. M. Jose, I. Ruthven, “A task-oriented study on the influencing effects of query-biased summarization in web

This is left as a future work.

#### 5. Conclusion

This paper presented the implementation and evaluation of an automated summarization system as an attempt to improve the search experience of Web users. The system uses both structural and linguistic information in the summarization process. As a novelty, this information is used in a query-biased way suitable to Web search context. The results were evaluated using the widely used information retrieval performance metrics adapted to summarization task and they showed an increase in accuracy compared to Google.

#### Acknowledgement

This work was supported by Boğaziçi University Research Fund, Grant no. 07A106.

#### References:

- [1] I. Mani, M. T. Maybury, *Advances in automatic text summarization*, MIT Press, 1999, Cambridge.
  - [2] I. Mani, *Automatic summarization*, J. Benjamins Pub. Co., 2001, Amsterdam.
  - [3] P. Jackson, I. Moulinier, *Natural language processing for online applications: text retrieval, extraction and categorization*, J. Benjamins Pub. Co., 2002, Amsterdam.
  - [4] H. P. Luhn, “The automatic creation of literature abstracts”, *IBM Journal of Research and Development*, No. 2, 1958.
  - [5] H. P. Edmundsun, “New methods in automatic extracting”, *Journal of the ACM*, Vol. 16, No. 2, pp. 265-285, 1969.
  - [6] K. Sparck Jones, “Automatic summarizing: factors and directions”, in: *Advances in automatic text summarization*, MIT Press, pp. 1-12, 1999, Cambridge.
  - [7] Google. <http://www.google.com>, 2006.
  - [8] Altavista. <http://www.altavista.com>, 2006.
- searching”, *Information Processing and Management*, Vol. 39, No. 5, pp. 707-733, 2003.

- [10]H. Alam, A. Kumar, M. Nakamura, A. F. R. Rahman, Y. Tarnikova, C. Wilcox, "Structured and unstructured document summarization: design of a commercial summarizer using lexical chains", Proceedings of Seventh International Conference on Document Analysis and Recognition, pp. 1147-1150, 2003.
- [11]GATE: a general architecture for text engineering. <http://gate.ac.uk/>, 2006.
- [12]D. Maynard, K. Bontcheva, H. Saggion, H. Cunningham, O. Hamza, "Using a text engineering framework to build an extendable and portable IE-based summarisation system", Proceedings of the ACL Workshop on Text Summarisation, 2002.
- [13]A. Arampatzis, T. Weide, C. Koster, P. Bommel, "Linguistically motivated information retrieval", in: Encyclopedia of Library and Information Science, Vol. 69, pp. 201-222, 2000.
- [14]I. Mani, G. Klein, D. House, L. Hirschman, T. Firmin, B. Sundheim, "SUMMAC: a text summarization evaluation", Natural Language Engineering, Vol. 8, No. 1, pp. 43-68, 2002.
- [15]D. A. Evans, C. Zhai, "Noun-phrase analysis in unrestricted text for information retrieval", Proceedings of 34th Annual Meeting of the Association for Computational Linguistics, pp. 17-24, 1996.