# Content and structure summarisation of XML documents for effective information access

Zoltán Szlávik, Anastasios Tombros, Mounia Lalmas
Department of Computer Science
Queen Mary, University of London
London, U.K.
{zolley, tassos, mounia}@dcs.qmul.ac.uk

**Abstract**

Digital libraries and other information providers make extensive use of the XML standard when publishing information. One of the benefits that XML presents is that it makes the logical structure of documents available. Overviews of the logical structure, as well as of the content, of XML documents can be used for providing effective access to the information stored within DL systems. In this paper, we describe three steps of an exploratory research into the use of automatic summarisation of XML documents for providing effective information access: we investigate the usefulness of the summarisation of the content of XML document elements, we examine the summarisation of the structure of XML documents by means of query-dependent table of contents, and we describe our current work into estimating query independent element features that can be used for generating generic summaries of document structure.

## Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries;

## General Terms

Algorithms, Design, Experimentation

## Keywords

Automatic summarisation, XML, document structure, information access

## 1 Introduction

The amount of information accessible has transformed the Web into a universal public information repository. A major outcome of this transformation has been, and still remains, the promotion of knowledge sharing. This has forced traditional information providers, like libraries, to also publish their information on the Web. However, the fact that the Web is growing at a phenomenal rate makes it difficult to effectively access all the published information. One reason is that this information is mostly published using HTML, a mark-up language that cannot accurately describe a page's content and structure. Therefore, modern Web applications, like digital libraries, have been increasingly publishing their information using the eXtensible Markup Language (XML) in order to bring some order to the Web.

The continuous growth in XML information repositories has been matched by increasing efforts in the development of XML retrieval systems (Lalmas and Tombros, 2007), in large part aiming at supporting content-oriented XML retrieval. These systems exploit the available structural information, as marked up in XML, in documents, in order to implement a more focussed retrieval strategy and return document components - the so-called XML elements - instead of complete documents in response to a user query. This focussed retrieval approach is of particular benefit for repositories containing long documents, or documents covering a wide variety of topics (e.g. books, user manuals, legal documents), where users' effort to locate relevant content can be reduced by directing them to the most relevant parts (called *elements*) of these documents (e.g. a

section or subsection, or even a paragraph).

Research into the design of approaches and systems for effective content-oriented retrieval of XML document elements has received interest over the past years, mainly through the INEX initiative (Lalmas and Tombros, 2007). However, even when retrieval systems return high quality results to users, providing effective access to the retrieved information still remains of high importance. For example, the provision of support for browsing within the elements of documents, especially for long documents with rich structural breakdown, and the provision of support for interacting with the retrieved document elements, are two key areas in the information access and retrieval process in DL.

In this paper, we focus on one particular approach to effective information access, namely summarisation, and we explore its possible applications to the context of structured XML documents. Text summarisation has been used for more than four decades to automatically generate abstracts, "snippets" of textual documents by selecting, or constructing, sentences based on the text of whole documents that provide a short, but concise, overview of the document's textual content (Maizell et al., 1971). For XML documents, however, given the additional structural information, it is not straightforward how summaries of document elements should be generated, nor is it straightforward how they should be presented to users.

Our work is using the infrastructure and methodology (i.e. document collection, search topics) developed as part of INEX, and is closely related to the research carried out in the INEX interactive track (Tombros et al., 2005).

The first part of our exploratory research into XML element summarisation addresses the fundamental question of whether text summarisation can be useful in the context of XML documents. We apply text summarisation to the textual content of XML elements and display the resulting summaries to the users of a retrieval system. Our aim is to investigate whether text summarisation can be effectively used to facilitate access to relevant content within XML documents. This research step is described in detail in Section 2.

The availability of the logical structure of XML documents also allows the creation of overviews of document structure. In other words, it is possible to summarise the structure of the document by means of an automatically created table of contents (ToC). This kind of structure summarisation would require the selection of the most important elements of documents automatically. This would also facilitate the hiding of irrelevant content from the user, e.g. if a user is only interested in Einstein's political influence from Einstein's biography, she would probably not be interested in looking at sections about his childhood in the table of contents of the biography. We address the issue of structure summarisation in Section 3, where we describe a method to automatically generate query dependent summaries of document structures based on a combination of element features.

When a document contains a large number of logical units (i.e. there are many elements within a document), it is important to select the most meaningful elements independent of the users' search query. Identifying such elements is even more important when a query has not been supplied, or is not known (e.g. a user of a DL may serendipitously discover an interesting book while browsing the contents of the library). In such cases, we still need to find out which parts of the document represent its contents best. We call a selection of the most important elements of the XML document without knowing the user's intent a query independent structure summary. To effectively create such a summary we need to combine several element properties, i.e. features, and determine whether they are worth including in an automatically created table of contents. Section 4 looks into how such features may be found.

## 2 Text summarisation for XML documents

The use of summaries in interactive information systems has been shown to be useful for various information seeking tasks in a number of environments such as the web or digital libraries (e.g. (Dumais et al, 2001), (White et al., 2003)). However, in the context of interactive XML retrieval, summarisation has not yet been investigated extensively. In this section, we aim to answer the following research questions:

i) Can text summarisation be useful when the structural overview of a document is also shown to users?

ii) If so, where (e.g. at which structural levels, for which element types) should text summaries be applied?

iii) Are the structural display (ToC) and the use of text summaries perceived similarly by users of XML retrieval systems?

iv) What are the requirements for an interactive XML retrieval system that displays summaries as well as the structure of the XML document?

## 2.1 Experimental system and setup

Due to limited space, this paper describes only the relevant part of the system, i.e. the display of summaries in the document view of the interface. For further details on the retrieval system architecture the reader is referred to the papers by (Szlávik et al., 2006a-b).

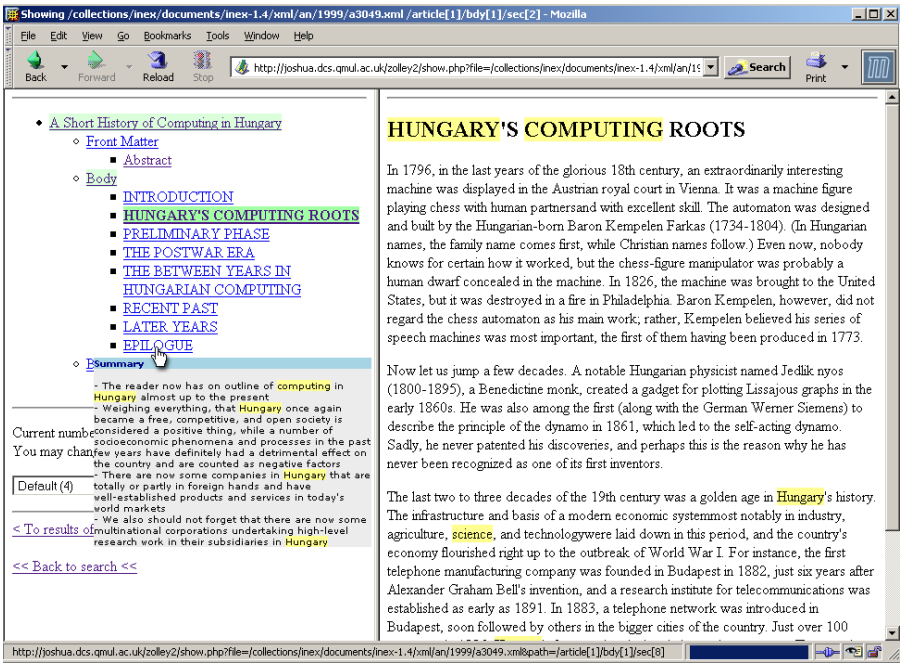


**Figure 2.1. Summary of the text of an XML element in the document view.**

Figure 2.1 displays the system's interface. This window is displayed when users select (click on) a link to a retrieved document element in the retrieved result list. The frame on the right shows the content of the target element with the query words highlighted. On the left, the structural view of the whole document is displayed in the form of a table of contents, where the position of the currently selected element is highlighted.

The structural display is based on the XML structure of the whole document, i.e the root element is shown at the top level, while descendants are displayed at lower levels (indented with bullets). Each structural item (also referred to as table of contents item, ToC item) is also a hyperlink that will show the corresponding XML element in the right window when selected. In this study, the ToC items were preselected after an analysis of their potential relevance.

For each ToC item shown in the hierarchical structure on the left, an automatic summary of the corresponding element's content is generated. Summaries are displayed as 'tool tips' when the mouse pointer is over a ToC item. Query terms in the summaries are also highlighted. Sentence extraction is used for summary generation (Edmundson, 1969). Sentences are scored by a combination of features, including the presence of query terms (Tombros and Sanderson, 1998). A maximum of four sentences with the highest ranks are presented as extracts of the source XML elements, in order of appearance in the source element.

Twelve users were asked to complete simulated work tasks (Borlund, 2003) on two versions of the system. One version (the control system $S_c$) showed summaries for every ToC item, the other (experimental system version, $S_e$) for items not deeper than the third structural level. The aim was to examine whether searchers realise the difference in the two systems and display different search patterns. From any observed difference, the usefulness of displaying the document structure and element summaries could be examined.

The INEX IEEE document collection (full-texts, marked up in XML, of 12,107 articles from the IEEE Computer Society's publications) was used in this study (Fuhr et al., 2006). Log data and questionnaires at various stages were recorded and interviews with users were conducted. The next section describes the results obtained from these data.

## 2.2 Discussion of results

The analysis of logs, questionnaires and interviews (discussed in detail in Szlávik et al., 2006a-b) suggested that summarisation can be helpful in interactive XML retrieval.

Searchers in this study did indeed use the provided structure actively and did not only use the whole article in order to identify relevant content. In addition, searchers made good use of the XML element summaries, by spending a significant amount of time reading these. This indicates that results obtained from this study are valid as they come from the extensive usage of the provided ToCs and summaries. We can also say that the experimental system, by the use of text summarisation, facilitated browsing in the ToC level more than that at the INEX 2004 interactive track (Tombros et al., 2005).

Regarding the use of element summaries, searchers in our study tended to read more summaries that were associated with elements at lower levels in the structure (e.g. summaries of paragraphs), and at the same time summaries of lower elements were read for a shorter period of time. The results also suggest that if more summaries are made available, searchers tend to read more summaries in a search session, but for a shorter time.

In order to be able to investigate particular summarisation algorithms in a retrieval system such as the one we used, the display of document structure has to be well controlled. We believe that the structural document display and summarisation for XML elements is strongly connected. If the display is not well designed, development and evaluation of various text summarisation strategies will not be reliable in an interactive environment. To control this effect, the following design guidelines are proposed as a result of this study:

- ToC items should be displayed based on the estimated relevance of the corresponding element. This is because users do not want be pointed at unnecessary irrelevant information. This finding shows the validity of structure summarisation described in the next sections.
- ToC items should be displayed according to their size and not only according to their content type (e.g. section type, chapter type). This is based on that users indicated a relation between the need of display and element length.
- Text summaries should be displayed for each item in the structural display. Alternatively, summaries should be completely avoided as selective summary presence may disturb users.

Based on the close relationship found between the ToC display and summary presentation, it follows that it is important to arrive at an appropriate ToC and summary presentation at the same time. If the ToC is too deep, searchers may lose focus, as the reading of many summaries and short reading times at low levels in this study indicated. Nevertheless, if the ToC is not detailed enough, users may lose potentially useful links to relevant elements. The results suggest that, for the used collection, a one or two-level ToC (containing references to the whole article, body, front and back matter) would be probably too shallow, while displaying the full fourth level (normally to paragraph-level) is sometimes too deep.

The results of this study, by demonstrating the need for a well designed table of contents, provided the motivation to investigate the automatic generation of ToCs for XML documents. We discuss this topic in the following section.

# 3 Structure summarisation of XML documents

In this section, based on the findings of the study in Section 2 and studies carried out by the INEX interactive track (Tombros et al., 2005), we investigate how we can automatically generate ToCs, and what the properties of a 'good' ToC should be.

In addition to the findings of Section 2, we identified two main limitations of interactive XML retrieval systems that have used tables of contents (e.g. the system reported in (Malik et al., 2006)). First, the ToCs used are typically static, i.e. the same ToCs for a given document is displayed for all queries, and second, ToCs are virtually manually defined, i.e. before the documents are displayed, they have to be analysed and several (types of) elements must be selected to be included in ToCs.

To address the above described limitations and aims, we developed an experimental system and recruited searchers who were asked to experiment with various features until they reached a ToC that was useful in the context of simulated work tasks (Borlund, 2003) that they had to complete. Searcher actions, comments and generated ToCs were recorded for analysis. Detailed results are presented in (Szlávik et al., 2007).

## 3.1 ToC generation

To automatically generate ToCs, we calculate a score for every XML element in consideration. If the score of an element is higher than a certain threshold value (described below), the element is considered as a ToC element. Ancestors of such elements, i.e elements higher in the XML hierarchy, are also used to place the ToC elements into context. For example, a section reference in a ToC without the chapter it is in would be just 'floating' in the ToC. The titles[1] of the selected elements are displayed as ToC items. The ancestor-descendant relation of elements is reflected, as in a standard ToC, by indentation.

The score of an element is computed using three element features: its depth, length and relevance to a given query. These features have been shown to form important characteristics in various XML retrieval tasks (Fuhr et al., 2006), although other features can also be taken into account.

### 3.1.1 Depth score

Each element receives a depth score between zero and one, based on where it resides in the structure of the document. In the document collection that we use (INEX IEEE), an article element is always at depth level one (i.e it is the root element in the tree structure). Descendants of a depth level one element are at depth level two (e.g. sections in an article), etc. According to the findings of our previous study (Section 2.2), elements at depth level three of a ToC are the most important for accessing relevant content, whereas the adjacent levels (two and four) are deemed less important. Sigurbjörnsson (2006, Chapter 8.) also found, using the INEX IEEE collection, that searchers mostly visited level two and three elements while looking for relevant information. Hammer-Aebi et al. (2006) confirmed that searchers found the highest number of relevant elements at levels two to four. To reflect these findings, the following scoring function was used to calculate an element's depth score (Equation 3.1):

$$S_{depth}(e) = \begin{cases} 1 & \text{if depth(e)} = 3, \\ 0.66 & \text{if depth(e)} \in \{2,4\}, \\ 0.33 & \text{if depth(e)} \in \{1,5\}, \\ 0 & \textit{otherwise} \end{cases} \qquad 3.1$$

where $S_{depth}(e)$ denotes the depth score of element e.

### 3.1.2 Length score

Each element also receives a length score, which is normalised to one. The normalisation is done

[1] If no title is available, the first 25 characters of the text are shown.

on a logarithmic scale (Kamps et al., 2004), where the longest element of the document, i.e the root element, receives the maximum score of one (Equation 3.2):

$$S_{length}(e) = \frac{\log(TextLength(e))}{\log(TextLength(root))} \qquad 3.2$$

where $S_{length}(e)$ is the length score of element e, root is the root element of the document structure and TextLength denotes the number of characters in the element.

### 3.1.3 Relevance score

A score between zero and one is used to reflect how relevant an element is to the current search topic. The score is provided by the search engine used in INEX for document collection exploration (Theobald et al., 2005) (i.e a normalised retrieval status value).

### 3.1.4 Feature weighting

The scores of the above three features are combined by using a weighted linear combination (Equation 3.3). Searchers are allowed to alter the weights of the features themselves while interacting with the system. This allows us to investigate what features searchers find important for ToC generation, and also, to determine what weights should be used to generate ToCs based on such features.

$$S(e) = \sum_{f \in F} W(f) \times S_f(e) \qquad 3.3$$

where S(e) denotes the overall score of element e, F is the set of the three features, W(f) is the weight of feature f and $S_f(e)$ denotes the score that is given to element e based on feature f.

### 3.1.5 Threshold

To determine the lowest score an element must achieve in order to be included in the ToC, we use a threshold value. As well as the feature weights described above, this value is also set by the searchers. This allows us to determine what the desirable size of a ToC should be: if the threshold is set to 100% only elements with the maximum depth, relevance and length scores will be included in the ToC (i.e. the sum of $S_f$-s equals to one), while if the threshold is set to zero, every element with greater than zero score will be in the ToC. We use a default value of 50%.

## 3.2 Experimental methodology and system

We created ten simulated work tasks that were presented to 31 users in random order. We used documents from the INEX IEEE and Wikipedia collections (Denoyer and Gallinari, 2006)(Fuhr et al., 2006). Searchers were asked to view as many documents as they wished for each task (at least three documents were available per task), and adjust their preferences for the three element features (length, relevance, depth) and the threshold by moving sliders on the interface (Figure 3.1). By adjusting the sliders, searchers were able to alter the characteristics of the current ToC. When they felt that the displayed ToC was helpful enough to assist them in finding relevant information for the task at hand, they could move on to the next document or topic.

After choosing a document, the document view was shown (Figure 3.1). This consisted of four parts: sliders associated with element features (left), the generated ToC for the current slider values (bottom left), the contents of the selected document/element (shown on the right), and links to the topic description, next topic and final page (top left).

We also recorded information about the searchers' perception of the system and ToC generation, e.g. the strategies searchers used when adjusting the sliders on the main screen, through a final questionnaire.

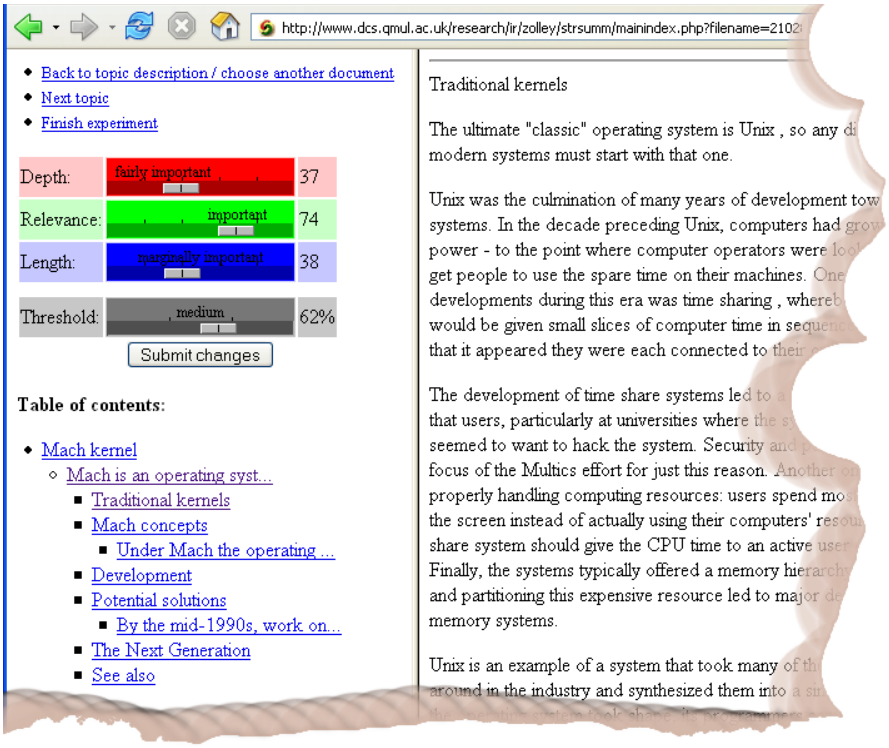


**Figure 3.1. Screenshot of the document view of the structure summarisation interface.**

### 3.3 Results and discussion

The results of this study showed that a ToC generation algorithm that is to be used in an interactive search scenario has to consider the relevance of an element, i.e. it should be query-biased. The other two element features used (length and depth), should also be considered and the relative importance (weight) of these two should be lower than that of the relevance feature. It is also understood that ToCs should not be large in size, i.e. longer documents should still have a relatively small ToC. This also shows that automatic ToC generation has to be more carefully designed when dealing with longer documents.

To ensure better results, ToC generation can be extended to include other element features such as e.g. tag names, titles of elements. Our data also suggest that the size of a ToC does not significantly depend on individual searchers. The selection of the most important elements is much more important, and at this point it may be worth considering searchers' individual preferences. We suggest that if a ToC generation method selects more than a certain number, in our case 20, of ToC-worthy elements, the highest scored 20 elements should be kept regardless of what threshold value the algorithm uses. If the number of ToC elements is lower than this number, these elements should all be used to construct the ToC.

Participants in the study reported various experiences regarding the dynamic nature of the generated tables of contents. The ability to customise ToCs based on various features for a given search task was generally well perceived by the participants. For this specific study, being presented with ToCs that continually changed based on slider selections did prove distracting for some participants, however, one general finding was that if a document's ToC does not change dramatically during a single session then searchers are not distracted from their search task.

## 4 Query independent structure summarisation

We saw in Section 3 that if the task that the user of a DL system has at hand is to search (and not, e.g. to browse), then the estimated relevance of an element to the query is highly important when summarising the structure of documents. However, since the other two features examined in Section 3, element length and depth, were also considered useful in ToC generation by searchers, it

is important to estimate how useful an element is for summarisation before we introduce relevance to the generation process. Without relevance in the summarisation algorithm we should still be able to create a list of elements that are more "ToC-worthy" than others. In this way we can create a structural overview for scenarios other than search, where a relevance value may not be applicable, e.g. when a digital library user browses within a book. In such cases it may be useful to gain an overview of what the main parts of the book are even when no conventional ToC is available, or when a ToC spans over several pages.

Generating query independent structure summaries by ignoring relevance estimates, is analogous to methods both in conventional text summarisation and in traditional information retrieval. In text summarisation, such methods produce query independent, or generic, summaries, and they have been widely used, for example, to automatically create abstracts of scientific articles (Kupiec et al. 1995). In information retrieval, we can find query independent prior knowledge about documents directly expressed in statistical language models, where "priors" are used in conjunction with relevance estimations (Hiemstra, 1998). For example, one can find a document prior by counting how often a feature (e.g. document length over a certain value) occurs in relevant and non-relevant elements, and then to estimate whether this feature is helpful in distinguishing relevant from non-relevant documents.

For the problem of structure summarisation, there are not widely available datasets from which to directly investigate whether a feature is discriminative, as we are not aware of training sets consisting of elements marked up according to whether they are "ToC-worthy" or not. However, if we assume that an element should be included in the ToC if it has the characteristics of an element that contains potentially relevant information (since these are the elements that searchers would find useful), then we can acquire training data sets by analysing the properties of elements that have a high likelihood of being relevant. To do so, we use retrieval runs (lists of retrieval results) that are submitted by XML retrieval systems as part of the INEX ad-hoc track. We describe this analysis in the next section.

### 4.1 Retrieval result analysis

For the retrieval result analysis we use retrieval runs that were submitted to the INEX 2006 ad-hoc track, which uses the Wikipedia document collection (Denoyer and Gallinari, 2006). We then categorise runs into good, average and bad quality sets, depending on how they performed in terms of retrieval effectiveness in the ad-hoc track. In order to analyse the characteristics of elements that have a high likelihood of being relevant, we compare the features of the elements of high quality results to those of low quality results. If a feature, which can be anything that can be computed or measured for an element (e.g. element length, number of outgoing links, etc.), displays a different pattern of occurrence in high and low quality results (e.g. it is observed more frequently in high quality results) we can consider it in structure summarisation.

Each retrieval run contains retrieval results for 125 topics, each quality set has 6 runs and each run contains up to 1500 result elements per topic. We therefore believe that the results obtained by the analysis of such a diverse dataset allow us to obtain information about "general relevance" (prior relevance) of document elements. The results over several investigated element features show that there is uniformity among high quality results which, as we can expect, shows that high quality results are more similar to one another than low quality results are to one another. Some preliminary findings from the analysis of retrieval results are as follows:

- The analysis of the depth feature shows that high quality result elements tend to be higher in the structure than those of other results.
- The results also show that the result elements of high quality runs were longer than those of other runs.
- The analysis also suggests that good elements types[2] are those whose frequency in the whole document collection is somewhat average (i.e. they are not the most or least frequent element types).

[2] Examples of element types are sections, paragraphs, etc.

- Elements in high quality runs contain less outgoing links in general than low quality runs. This seems to suggest that hubs are not very promising for element retrieval or structure summarisation.

The incorporation of the results of this analysis into ToC generation, and the evaluation of the effectiveness of the resulting ToCs is ongoing work. Our immediate efforts are focused on the in depth analysis of the retrieval results, and on developing the experimental infrastructure for the evaluation of our approach. We plan to follow a methodology similar to (Kupiec et al., 1995) in order to calculate query independent "ToC-worthiness" scores for document elements.

## 5 Conclusions

We presented an investigation into one particular approach for providing effective information access to users of digital libraries, namely the automatic summarisation of documents formatted in XML. In this paper, we described three steps of an exploratory research: an investigation into the usefulness of content summarisation of XML document elements, a study into the summarisation of the structure of XML documents by means of query-dependent table of contents, and our current work into estimating query independent element features that can be used for generating generic summaries of document structure.

Our work suggests that summarisation can provide effective information access to XML documents and elements, and we identified several conditions that need to be met in order to use summarisation effectively in this context. Additionally, during our investigation into structure summarisation, several recommendations for improving the summarisation of document structure were found, for example, we identified the need for tables of contents that are generated independently from the user's query for usage scenarios other than search-oriented ones (e.g. serendipitous discovery, browsing). Work in this direction is currently ongoing.

## References


Borlund, P. The IIR evaluation model: a framework for evaluation of interactive information retrieval systems. Information Research, 8(3), 2003.

Denoyer, L. and Gallinari, P. The Wikipedia XML Corpus. SIGIR Forum, 40(1):64–69, 2006.

Dumais, S., Cutrell, E. and Chen, H. Optimizing search by showing results in context. In CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems, pages 277–284, New York, NY, USA, 2001.

Edmundson, H. P. New methods in automatic extracting. J. ACM, 16(2):264–285, 1969.

Fuhr, N., Lalmas, M., Malik, S. and Kazai G. editors. Proceedings of INEX 2005, LNCS volume 3977, 2006.

Hammer-Aebi, B., Christensen, K., Lund, H. and Larsen, B. Users, structured documents and overlap: interactive searching of elements and the influence of context on search behaviour. In Proceedings of IIiX, pages 46–55, 2006.

Hiemstra, D.. A linguistically motivated probabilistic model of information retrieval. In Proceedings of the Second European Conference on Research and Advance Technology for Digital Libraries (ECDL), pp. 569-584. 1998.

Kamps, J., de Rijke, M. and Sigurbjörnsson, B. Length normalization in XML retrieval. In Proceedings of ACM SIGIR '04, pages 80–87, 2004.

Kupiec, J., Pedersen, J. and Chen, F. A trainable document summarizer. In Proceedings of ACM SIGIR'95, pages 68–73. ACM Press, 1995.

Lalmas, M. and Tombros, A. INEX 2002 - 2006: Understanding xml retrieval evaluation. In DELOS Conference on Digital Libraries, Tirrenia, Pisa, Italy, February 2007.

Maizell, R, Smith, J. and Singer, T.. Abstracting scientific and technical literature: an introductory guide and text for scientists, abstractors, and management. Wiley-Interscience, 1971.

Malik S., Klas, C. P., Fuhr, N., Larsen, B. and Tombros, A. Designing a User Interface for Interactive Retrieval of Structured Documents - Lessons Learned from the INEX Interactive Track. In Proceedings of ECDL 2006, pages 291–302, 2006.

Sigurbjörnsson, B. Focused Information Access using XML Element Retrieval. PhD thesis, Faculty of Science, University of Amsterdam, 2006.

Szlávik Z., Tombros, A. and Lalmas, M. The use of summaries in XML retrieval. In Proceedings of ECDL 2006, pages 75–86, 2006a.

Szlávik Z., Tombros, A. and Lalmas, M. Investigating the use of summarisation for interactive XML retrieval. In F. Crestani and G. Pasi, editors, Proceedings of ACM SAC-IARS'06, pages 1068–1072, 2006b.

Szlávik Z., Tombros, A. and Lalmas, M. Feature- and query-based table of contents generation for XML documents. In Giambattista Amati, Claudio Carpineto, and Giovanni Romano, editors, Proceedings of ECIR '07, LNCS volume 4425, pages 456–467. Springer, 2007.

Theobald, M., Schenkel, M. and Weikum, G.. An efficient and versatile query engine for TopX search. In Proceedings of VLDB, pages 625–636, 2005.

Tombros, A. and Sanderson, M. The advantages of query-biased summaries in information retrieval. Proceedings of ACM SIGIR '98, pages 2-10, 1998.

Tombros, A., Malik, S. and Larsen, B. Report on the INEX 2004 interactive track. ACM SIGIR Forum, 39(1):43–49, 2005.

White, R., Jose, J. and Ruthven, I. A task-oriented study on the influencing effects of query-biased summarisation in web searching. Inf. Process. Manage., 39(5):707–733, 2003.