Examining Topic Shifts in Content-Oriented XML Retrieval

Elham Ashoori, Mounia Lalmas, Theodora Tsikrika

Queen Mary, University of London, London, E1 4NS, UK, e-mail: {elham,mounia,theodora}@dcs.qmul.ac.uk

Abstract. Content-oriented XML retrieval systems support access to XML repositories by retrieving, in response to user queries, XML document components (XML elements) instead of whole documents. The retrieved XML elements should not only contain information relevant to the query, but also provide the right level of granularity. In INEX, the INitiative for the Evaluation of XML Retrieval, a relevant element is defined to be at the right level of granularity if it is *exhaustive and specific* to the query. Specificity was specifically introduced to capture how focused an element is on the query (i.e., discusses no other irrelevant topics).

To score XML elements according to how exhaustive and specific they are given a query, the content and logical structure of XML documents have been widely used. One source of evidence that has led to promising results with respect to retrieval effectiveness is element length. This work aims at examining a new source of evidence deriving from the semantic decomposition of XML documents. We consider that XML documents can be semantically decomposed through the application of a topic segmentation algorithm. Using the semantic decomposition and the logical structure of XML documents, we propose a new source of evidence, the number of *topic shifts* in an element, to reflect its relevance and more particularly its specificity.

This paper has three research objectives. Firstly, we investigate the characteristics of XML elements reflected by their number of topic shifts. Secondly, we compare topic shifts to element length, by incorporating each of them as a feature in a retrieval setting and examining their effects in estimating the relevance of XML elements given a query. Finally, we use the number of topic shifts as evidence for capturing specificity to provide a focused access to XML repositories.

Key words: content-oriented XML retrieval – topic segmentation – INEX relevance – right level of granularity – topic shifts vs. length – focused access

1 Introduction

The enormous amount of information accessible from the World Wide Web (the Web) has transformed it 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 markup language that cannot accurately describe its content and structure. Therefore, modern Web applications like digital libraries have been increasingly publishing their information using the eXtensible Markup Language (XML)[1] in order to bring some order on the Web [42].

The continuous growth of XML information repositories has been matched by increasing efforts in the development of XML retrieval systems (e.g. [9,4–6,13,15, 17,16]), that, in large part, aim at supporting **content-oriented XML retrieval**. Given a user query, content-oriented XML retrieval systems are concerned with returning, to the user, document components marked up in XML – the so-called XML *elements* – instead of complete documents. Their aim is to reduce users' effort to locate relevant content by directing them not just to the documents containing the relevant information, but to their most relevant parts. Such retrieval paradigm is of particular benefit for information repositories containing

---

[1] http://www.w3.org/XML/

long documents or documents covering a wide variety of topics (e.g. documents such as books and user manuals, or legal documents).

Such XML retrieval systems consider XML elements of any granularity (e.g. a paragraph or the section enclosing it) as potential answers to a query, as long as they are relevant. This constitutes a major departure from traditional Information Retrieval (IR). This is not only because XML retrieval systems deal with document components (the XML elements), rather than only considering complete documents. Most importantly, these XML retrieval systems need to not only score elements with respect to their relevance to a query, but to also determine the appropriate level of element granularity to return to users.

What constitutes **a relevant element at the appropriate level of granularity** is a research question actively being investigated in the INitiative for the Evaluation of XML Retrieval (INEX)[2]. In INEX, a *relevant* element is defined to be at the *right level of granularity*, if it discusses fully the topic requested in the user's query (it is *exhaustive* to the query) **and** does not discuss other topics (it is *specific* to that query). With specificity, it is possible to differentiate, for example, between the only relevant section in an encyclopaedia from the whole encyclopaedia. In the case the section is relevant (exhaustive) to a given user query, the encyclopaedia will also be relevant; however, the former is likely to trigger higher user satisfaction as it will be more specific to the query than the latter.

XML retrieval systems need to score XML elements according to how exhaustive and specific they are given a query. To this end, various sources of evidence have been exploited. These include the content, the logical structure represented by the XML markup and the length of XML elements [13,15,17,16]. In this work, we consider a different source of evidence, **the number of topic shifts in an XML element**. Our motivation stems from the definition of a relevant element at the appropriate level of granularity in INEX, which is expressed in terms of the "quantity" of topics discussed within each element. Consequently, we hypothesize that a measure of the shifts of the topics within an element could reflect its relevance and whether it lies at the appropriate level of granularity for that query.

Topic shifts in XML elements constitute a novel source of evidence, which, to the best of our knowledge, has not been previously employed in the context of XML retrieval. Therefore, our first objective in this paper is to study the characteristics of XML elements as reflected by their number of topic shifts.

A source of evidence that has already led to promising results with respect to retrieval effectiveness is ele-

ment length. Generally when the length of an element increases, it is highly likely that it will discuss more topics. Therefore, it might be argued that the number of topic shifts reflects evidence already captured by their length and as such it does not constitute a distinct feature. This motivates our second objective which is to compare the number of topic shifts and length features of XML elements. We perform a comparative analysis by incorporating each of these sources of evidence as a feature in a language modeling retrieval setting to investigate the effect of each in estimating the relevance of XML elements.

Finally, we use the number of topic shifts as evidence for capturing specificity to provide a *focused* access to XML repositories, i.e. identifying relevant elements at the right level of granularity for a given topic of request. For this purpose, we also use a language modeling framework.

We investigate our three research objectives by carrying out extensive experiments using the testbed built by INEX.

The paper is organised as follows. Section 2 discusses related work. In Section 3, we define the notion of topic shifts and how we formalise it. Section 4 describes the methodology and the experimental setting used in our investigation, including the INEX testbed. The experiments and results, for our three research objectives, are discussed in Sections 5, 6, and 7, respectively. Section 8 concludes the paper and outlines future work.

2 Related work

Content-oriented XML retrieval aims at identifying the most relevant part(s) of documents, a research objective that has been previously investigated by passage retrieval research. Section 2.1 presents the related work in the area of passage retrieval, highlighting the similarities and differences with XML retrieval and also discussing how the definition of passages in passage retrieval influenced our definition of topic shifts for XML retrieval. Section 2.2, on the other hand, reviews retrieval approaches employed in the context of content-oriented XML retrieval.

2.1 Passage Retrieval

Passage retrieval approaches decompose documents into components (referred to as *passages*), and, given a query, either retrieve the most relevant passage(s) from each document or use these passages as evidence in retrieving the most relevant documents [7,27,55,23].

Content-oriented XML retrieval is also concerned with the identification of the most relevant part(s) of the documents. However, whereas in passage retrieval, there is a need to define the parts of the documents that can

---

[2] INEX (http://inex.is.informatik.uni-duisburg.de/) aims to establish an infrastructure and means, in the form of a large XML test collection and appropriate effectiveness metrics, for the evaluation of content-oriented retrieval of XML documents.

act as passages, in XML retrieval, these parts simply correspond to the XML elements determined by the logical structure of the document. In fact, XML retrieval has been and is mostly concerned with so-called *element* retrieval[3]. Also, in XML retrieval, nested relations typically exist between XML elements, whereas in passage retrieval, passages usually have a linear, and possibly overlapping, relation to each other (although the idea of hierarchical passage retrieval, which is comparable to the retrieval of elements in XML documents, has also been suggested [47]). In both passage and XML retrieval, the system needs to determine not only the most relevant part, but also the one at the right level of granularity. In passage retrieval, where the best scoring passage from each document is considered the most relevant part to be returned to users in response to their queries, the issue of right level of granularity is only indirectly addressed by experimenting with different passage sizes and overlapping degrees. In XML retrieval, on the other hand, the most relevant part at the right level of granularity is identified by explicitly exploiting the hierarchical relations between XML elements.

Research in passage retrieval has influenced this work with respect to the methodology applied in determining the number of topic shifts within XML elements. In particular, to quantify the topic shifts, we need to decompose documents into suitable passages, each considered to discuss a topic. For this decomposition, we considered the approaches previously adopted in the identification of passages in documents in passage retrieval. Specifically, three types of passages have been investigated: *discourse*, *window-based*, and *semantic* passages [27].

Discourse passages correspond to the discourse components of documents, such as sentences, paragraphs, sections (e.g. [55]). However, such passages only correspond to a single type of component at a time, i.e. they are either all paragraphs, or are all sections, etc. Since these types of discourse components are fixed a priori, it would be a simplification to assume that a part of a document discusses a topic simply because it corresponds to a discourse component. Fixed- or variable-length (overlapping or non–overlapping) *window-based passages* [7,27] correspond to windows of text consisting of a given number of words. With this type of passages, a document is divided into parts without taking into consideration the document content or the topics discussed in it. Therefore, they cannot be used as a basis to calculate the number of topic shifts in XML elements. *Semantic passages*, on the other hand, divide a document into segments, each corresponding to a topic or subtopic. Such decomposition approaches, e.g. [22,44, 48], have been widely used in many IR applications (e.g. [8,46,38]). They are also appropriate for our research purposes, since this semantic decomposition allows us to calculate for each XML element its number of topic shifts. The application of one such algorithm in this work is discussed in detail in Section 3.

2.2 XML Retrieval

The main challenge for content-oriented XML retrieval systems is to identify highly relevant XML elements that would satisfy the users in response to their information needs. Content-oriented XML retrieval is a relatively new research field and many research questions are open to debate, including the question of what elements the users themselves would prefer the system to retrieve in an XML retrieval setting [54]. This is because in XML retrieval, not only must an element be relevant, but it must be at the right level of granularity to satisfy a user's information need.

To address this issue, various approaches have been proposed during the INEX campaigns, (2002-2005) [13, 15,17,16]. The remainder of this section discusses some of these approaches. The discussion is mainly concerned with the sources of evidence and strategies employed by these approaches, and not the actual retrieval models (but see [32] for a report on some of the XML models).

To determine the appropriate level of granularity, one approach is to select as indexable and, therefore, as retrieval units (i.e. as potential answers to user queries) only a subset of, instead of all, available element types. These are referred to as "index nodes" [10] and are usually selected as follows. The collection administrator can manually determine the element types considered as index nodes by analysing the logical structure (i.e. the DTD[4]) of the document collection [19]. Types denoting, for instance, styles could be excluded [30]. Another strategy is to index and/or retrieve only those element types with the highest distribution of relevant elements in past relevance assessment sets. For example, in INEX, selected types included article, section, abstract, subsection and paragraph element types [35].

The main drawback of these approaches is that they use pre-defined element types as indexable and retrieval units, thus they are DTD-dependent and therefore not portable to different XML collections. This motivates us to focus our research on collection-independent approaches.

One such collection-independent approach is to index leaf elements only (elements with no children), and, given a query, first score only the leaf elements. The next step is to score all non-leaf elements based on some (weighted) combination of the scores of their children elements. The propagation of scores starts from the leaf elements. This propagation can take into account the distance between the element being considered and its

---

[3] Although at INEX 2007 (http://inex.is.inf.uni-due.de/2007/index.html), passage retrieval (as opposed to element retrieval) is also being investigated.

[4] The Document Type Definition (DTD) of a document collection is a document that contains definitions of all XML element types in the collection.

descendant leaf elements [18,49]. It can also exploit element relationships, such as the (particular) relationship between an element and its root element. For instance, in INEX, the root element of any element is the article element and considering this relationship has often shown to improve performance [36,1,52].

Other approaches exploit various DTD-independent features beyond the content of elements. The most notable such feature that has shown to be an important factor in XML retrieval, is the length of XML elements [26]. This is due to the fact that, whereas the length distribution of XML elements in the INEX collection is heavily skewed towards shorter elements, the distribution of the prior probability of relevance of XML elements is heavily skewed towards longer elements. To counterbalance this, several techniques have been proposed [26,41,21].

A simple technique is to use an index cut-off based on the length (both as lower and upper bound) of XML elements to be indexed and retrieved [21]. This strategy removes elements which are either too small or too big to be considered as meaningful retrieval units. However, when the technique of simply introducing a lower cut-off length value for XML components was applied in a language modeling framework, it was shown that it was not sufficient [26]. This might be due to the fact that indexing small elements might still be useful, as they might influence the scoring of their enclosing elements [49].

Another technique is to incorporate length as a source of evidence in the scoring function. For instance, the usage of length priors to bias retrieval towards longer XML elements has shown promising results when employed within a language modeling framework [26,41]. This technique is discussed in Section 6.1, as our approach based on the number of topic shifts will be compared to it.

In our work, we propose and study a different DTD-independent source of evidence: the topic shifts in XML elements. First, we investigate the characteristics of XML elements reflected by their topic shifts. Next, to compare this new feature to element length, we follow the spirit of [26], and incorporate this feature within a language modeling framework, and examine its effects, compared to element length, to estimate relevance in XML retrieval. Finally, we use this new source of evidence for the focused access to XML documents. For this purpose, we also use the language modeling framework, as it allows us to incorporate the number of topic shifts as a parameter in the scoring function. Before doing so, we formally define the notion of topic shifts and how we measure it.

3 Topic shifts

In this section, we describe how we determine the number of topic shifts of the elements forming an XML document. For this purpose, both the logical structure and a semantic decomposition of the XML document are needed. Whereas the logical structure of XML documents is readily available through their XML markup, their semantic decomposition needs to be extracted. To achieve that, we apply a topic segmentation algorithm, previously applied in passage retrieval (Section 2.1). Section 3.1 describes the topic segmentation algorithm we apply in this work and Section 3.2 describes how we measure the number of topic shifts based on the outcome of this algorithm.

3.1 Semantic decomposition of an XML document

A text document can be semantically decomposed through the application of a topic segmentation algorithm. The main goal of such an algorithm is to divide a document into segments, with each segment corresponding to a single topic or subtopic, both referred to, for simplicity, as topics. The granularity of the discourse unit of such segments could range from words or sentences, to paragraphs.

In this work, we consider a topic segmentation algorithm based on lexical cohesion. The linguistic theory of lexical cohesion, first presented in [20], captures a property of text, arising from "the chains of related words that contribute to the continuity of lexical meaning" [40]. In particular we consider the lexical cohesion identified by considering term repetition, and indicated by lexical terms reminding the meaning of earlier terms in the text [53]. Therefore, the underlying assumption of topic segmentation algorithms based on lexical cohesion, is that a change in vocabulary signifies that a topic shift occurs. This results in topic shifts being detected by examining the lexical similarity of adjacent text segments. This motivates us to adopt this type of algorithm in this work as it is particularly well suited to determine the number of topic shifts in text documents.

One such topic segmentation algorithm based on lexical cohesion, which has been successfully used in several IR applications [23,8,46,38], is TextTiling[5] [22]. TextTiling is a linear segmentation algorithm which considers the discourse unit to correspond to a *paragraph* and therefore subdivides the text into *multi-paragraph* segments.

TextTiling is performed in three steps. In the first step, after performing tokenisation, the text is divided into pseudo-sentences of size W, called token-sequences. Next, these token-sequences are grouped together into blocks of size K. A similarity score is computed for all pairs of adjacent blocks based on term repetition. This step is repeated until all possible pairs of adjacent blocks of size K are considered. The gap between two adjacent blocks constitutes a potential boundary for a semantic segment. To identify the actual boundaries, a depth score is computed for each potential boundary, by using the
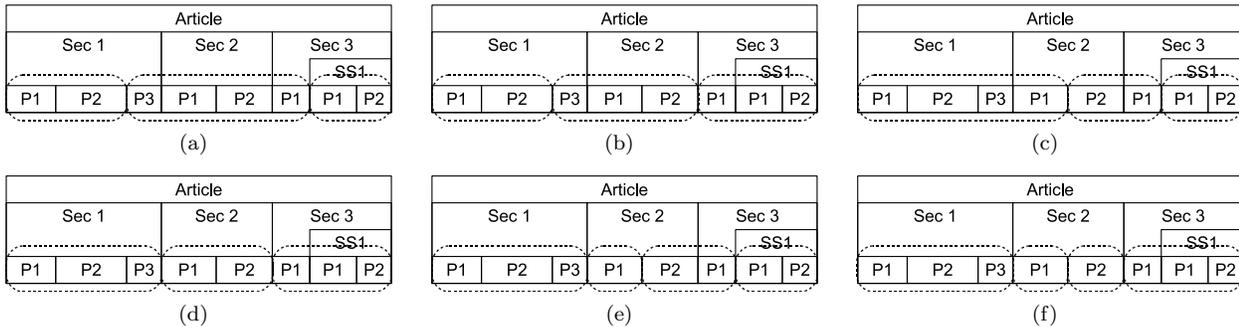
---

[5] http://elib.cs.berkeley.edu/src/texttiles/

Fig. 1. Relations between XML elements and semantic segments.

similarity scores assigned to the neighbouring gaps between blocks, and by applying a smoothing process. The algorithm determines the number of segments, referred to as *tiles*, assigned to each document, by considering segment boundaries to correspond to gaps with depth scores above a certain threshold. The detected boundaries are then adjusted to correspond to the actual discourse unit breaks, i.e. the paragraph breaks.

In this work, XML documents are decomposed into a linear sequence of segments by using the TextTiling algorithm. Such a structure is sufficient for the tasks of interest here, and our choice is further justified by the reasonable results we obtain through the application of the algorithm in our document collection (see Section 5.1). In addition, we chose TextTiling for its computational simplicity.

3.2 Measuring topic shifts in XML elements

The semantic decomposition of an XML document is used as a basis to calculate the number of topic shifts in each XML element forming that document. There are six possible relations between an XML element and the generated semantic segments. These are illustrated in Figure 1, where the XML elements (Article, Sec 1, Sec 2, Sec 3, SS1, P1 and P2) are shown as solid boxes and the outcomes of the topic segmentation, i.e., the segments, are shown in dashed line. For instance, within Sec 2, for case (d), one segment is found, composed of P1 and P2. An XML element might:

– Be part of one segment - e.g. Sec 2 in case (a).
– Be part of one segment and only one element boundary coincides with the segment - e.g. Sec 2 in case (b).
– Overlap with segments that span across other XML elements - e.g. Sec 2 in case (c). This case means that one of the topics discussed in Sec 2 is continuing from the previous element (here Sec 1) and another one is continuing in the next element (here Sec 3).
– Be covered exactly by one segment, i.e. discuss fully one topic - e.g. Sec 2 in case (d).
– Include one segment completely (one boundary coincides with that segment) and overlap with another

segment that spans across other XML elements - e.g. Sec 2 in case (e).
– Include more than one semantic segments and both element boundaries coincide with the segments i.e. discuss fully more than one topic - e.g. Sec 2 in case (f).

As Figure 1 illustrates, we consider the situations where the boundaries of Sec 2 and those of the segments generated by TextTiling completely match, e.g. cases (d) and (f), where they do not match, e.g. cases (a) and (c), and where only one of the boundaries of Sec 2 matches with those of the semantic segments, e.g. cases (b) and (e). The latter case is specifically considered because we want to allow for the tendency exhibited by authors to relate, for instance, the last paragraph of a section to the content of the following section, which will result in TextTiling placing the last paragraph of a section and the first paragraph of the following section in the same segment. This decision is further justified by our experiments, which demonstrate that exact matches do not occur very often (see Section 5.1).

We consider that a topic shift occurs (i) when one segment ends and another segment starts, or (ii) when the starting (ending) point of an XML element coincides with the starting (ending) point of a semantic segment. The *number of topic shifts* in an XML element e is therefore defined as:

$$score(e) := actual\_topic\_shifts(e) + 1 \qquad (1)$$

where $actual\_topic\_shifts(e)$ are the actual occurrences of topic shifts in element e. We are adding 1 to avoid zero values. Indeed, in case (a) of Figure 1, the actual number of topic shifts for Sec 2 is 0. With the above formulation, it is now 1. For simplicity, when we refer to the number of topic shifts, we shall be referring to $score(e)$.

With the above definition, the larger the number of topic shifts – i.e. the larger the $score(e)$ – the more topics are discussed in the element.

Table 1 shows the number of topics (i.e., the number of segments in the element, including segments spanning across previous and next elements), and the number of topic shifts (i.e., as given by Equation 1) for Sec 2 in the

Table 1. Number of topics and number of topic shifts in Sec 2 in Figure 1.

Case	Topics	Topic Shifts
(a)	1	1
(b)	1	2
(c)	2	2
(d)	1	3
(e)	2	3
(f)	2	4

different cases of Figure 1. For instance, in cases (a) and (d), Sec 2 discusses one topic. However, in case (d), Sec 2 fully discusses that topic, whereas in case (a) it discusses only part of that topic. Using the number of topic shifts to measure the "quantity" of topics discussed in an XML element - instead of the number of topics (segments) - we can therefore differentiate between cases (a) and (d); the respective assigned scores are 1 and 3. Similarly, in cases (c), (e) and (f) where Sec 2 discusses two topics, the topic shifts scores differ, and are 2, 3 and 4, respectively.

The number of topic shifts in an element captures how many topics are fully discussed in the element. In fact, any score of 1 or 2 means that the element does not discuss a topic fully. The number of topics in an element cannot detect if these are fully discussed in the element. Although it would be interesting to see whether using the number of topics or the number of topic shifts actually makes any difference in terms of retrieval performance, using the number of topic shifts is more fine-grained, as it allows to differentiate more cases, as illustrated in Table 1. This is why we use in this work the number of topic shifts to quantify the "quantity" of topics discussed in an XML element.

Table 2. Topic shifts scores for Sec 2, P1 and P2 in Figure 1.

Case	Sec 2	Sec 2/P1	Sec2/P2
(a)	1	1	1
(b)	2	1	2
(c)	2	2	2
(d)	3	2	2
(e)	3	2	3
(f)	4	3	3

Another aspect is the relation of the number of topic shifts between parent–children elements. Table 2 shows the number of topic shifts for Sec 2, P1 and P2 in Figure 1. We can see that the number of topic shifts of a parent element can be equal to that of its child element (e.g. Sec 2 and P1), and that it is not necessarily equal to the sum of the number of topic shifts of its children (e.g. Sec 2). Although (by definition) the number of topic shifts of a parent element must be at least equal to (or higher than) the maximum number of topic shifts of its children, we are explicitly measuring the "quantity" of topics fully discussed *within* each element.

Now that we have detailed how we formalise the notion of topic shifts, through the application of a semantic topic segmentation algorithm – in our case TextTiling (see Section 3.1) – we describe next the methodology used to investigate this new source of evidence for XML retrieval.

4 Methodology

In this section, we describe the methodology adopted to examine topic shifts in XML retrieval. Our aims are (i) to examine the characteristics of XML elements reflected by their number of topic shifts (Section 5), (ii) to compare topic shifts to length (Section 6), and (iii) to use the number of topic shifts to provide a focused access to XML documents (Section 7). We conducted extensive experiments on the INEX collection to investigate our three aims. Section 4.1 describes the INEX collection used in our experiments and Section 4.2 discusses our experimental setting.

4.1 The INEX Test Collection

We provide an overview of the INEX test collection which is used for all our experiments. A test collection usually consists of a set of documents, a set of user requests (referred to as topics[6]) and relevance assessments. The latter states which documents – in our case, XML elements – are the "right" answers for a given user request. In our work, we use the INEX 2003-2005 test collections. This section provides the necessary background to understand the experiments carried out in subsequent sections.

4.1.1 Document collection

The INEX document collection contains scientific articles from different IEEE Computer Society journals, marked up in XML. It consists of two versions. *Version 1.4*, used in INEX 2003-2004, contains 12,107 articles (from 21 journals), consisting of over 8 million elements. *Version 1.8*, used in INEX 2005, is an extension of Version 1.4 and contains 16,819 articles (from 24 journals), consisting of over 10 million elements.

The experiments reported in Sections 5 and 6.1 use *Version 1.4*, whereas those reported in Sections 6.2 and 7 use *Version 1.8*.

4.1.2 Topics

The experiments described in this paper make use of the *Content-only* (*CO*) topics, which are requests that ignore the document structure and contain only content

---

[6] The notion of topic in a test collection is different to what we means by the topics discussed in an element. When ambiguity arises, we shall refer to user requests.

related conditions[7]. An INEX CO topic consists of the standard title, description and narrative fields. In this work, the title field of a topic is used for retrieval[8].

The experiments reported in Sections 5 and 6.1 use the relevance assessments (see Section 4.1.4) for the CO topics of *Version* 2.5 of the INEX 2003 data set and *Version* 3.0 of INEX 2004. The experiments reported in Section 6.2 and Section 7 use the CO topics, *Version 2005-003*, of the INEX 2005 data set.

4.1.3 Retrieval tasks

We investigated the incorporation of topic shifts in two retrieval settings, which correspond to two retrieval tasks defined in INEX: the *thorough* and the *focused* tasks.

Given a CO topic, the aim in the *thorough* retrieval task is to estimate the relevance of the (potentially retrievable) elements in the collection, and to rank them in decreasing order of their estimated relevance. Within this thorough retrieval setting, we investigate our second research objective, which is to compare topic shifts to length in estimating the relevance of an XML element given an information need (Section 6).

Given a CO topic, the aim in the *focused* retrieval task is to find the most relevant element on a path[9] within a given document, and return only this most appropriate unit of retrieval. Whereas the thorough task implies that the retrieval result might contain several elements from the same document which could be structurally related, the focused task allows no overlapping between the elements, i.e. none of the ascendants or descendants of an element in a path should be returned. This is the formal definition of a focused access to XML documents adopted by INEX [31]. Within this focused retrieval setting, we investigate our third research objective, which is to use the number of topic shifts to provide a focused access to XML documents (Section 7).

4.1.4 Relevance assessments

INEX defines relevance in terms of two dimensions, *exhaustivity* (e) and *specificity* (s), each defined on a scale. These two dimensions are respectively defined as "how exhaustively an element discusses the topic of request" and "how focused an element is on the topic of request (i.e. discusses no other irrelevant topics)" [14]. While

the definition of these two dimensions has remained unchanged during the first four years of INEX, the scale that these dimensions were measured on has changed.

For INEX 2003 and 2004, both dimensions are measured on a four-point scale. For exhaustivity, the scale is defined as highly exhaustive (e=3), fairly exhaustive (e=2), marginally exhaustive (e=1) and not exhaustive (e=0) XML elements. For simplicity, we refer to these four levels as e3, e2, e1, and e0, respectively. Analogously, the four-point scale of the specificity dimension is defined as highly specific (s=3), fairly specific (s=2), marginally specific (s=1) and not specific (s=0) XML elements. As for exhaustivity, we refer to the specificity levels as s3, s2, s1 and s0, respectively. Furthermore, the relevance value of an element is referred to as e-s. For example, e3-s3 refers to a highly exhaustive and highly specific element, whereas e3-s123 refers to highly exhaustive elements with specificity equal either to 1, 2 or 3.

For INEX 2005, exhaustivity is measured on a three-point scale: highly exhaustive (e=2), somewhat exhaustive (e=1), and not exhaustive (e=0). Similarly to INEX 2003-2004, we refer to the three exhaustivity levels as e2, e1, and e0. The specificity dimension is measured on a continuous scale [0,1] as the ratio of the relevant content of an XML element[10], which we refer to as sx where $0 \le x \le 1$. As for INEX 2003-2004, we use e-s to refer to the relevance value of an element. For example, e2-s0.72 denotes a highly exhaustive element, with 72% of its content being relevant.

4.1.5 Evaluation metrics

The metrics used to evaluate the experiments described in Section 6.2 and Section 7 are the eXtended Cumulated Gain (XCG) metrics [29], which include the extended cumulated gain ($MAnxCG[i]$), and the effort-precision/gain-recall measures ($MAep$).

For each returned element, a gain value, $xG[.]$, which is a value between 0 and 1, is calculated through a *quantization function*. The latter provides a relative ordering of the various combinations of exhaustivity and specificity values and a mapping of these to a single relevance scale, reflecting the worth of a retrieved element. The following two quantization functions were used in INEX 2005, the data set in our retrieval experiments:

$$quant_{strict}(e,s) := \begin{cases} 1 \text{ if } e = 2 \text{ and } s = 1, \\ 0 \text{ otherwise.} \end{cases} \quad (2)$$

$$quant_{gen}(e,s) := e * s \quad (3)$$

The strict quantization function $quant_{strict}$ is used to evaluate XML retrieval methods with respect to their

---

[7] INEX has topics that contain explicit references to the XML structure. However, we restrict ourselves to CO topics because our aim is to investigate topic shifts as a new source of evidence in XML retrieval, without the additional complication of interpreting and processing structural constraints.

[8] In INEX 2005, these topics are referred to as CO+S, where the title, description and narrative field correspond to what we refer to as CO topics. For more details, see [34].

[9] A relevant path is a path within the XML tree of a given XML document, whose root node is the root element and whose leaf node is a relevant element that has no or only irrelevant descendants.

[10] Assessors were asked to highlighted text fragments containing only relevant information. The s value of an element was automatically calculated as the ratio (in characters) of the highlighted text (i.e., relevant information) to the element size.

capability of retrieving highly exhaustive and highly specific elements (e=2, s=1). The generalised quantization function $quant_{gen}$ is used to evaluate XML retrieval methods with respect to their capability of retrieving a relevant element, and considering their worth.

Given a ranked list of retrieved elements, the *normalised cumulative gain* at rank i is computed as follows:

$$nxCG[i] := \frac{\sum_{j=1}^{i} xG(j)}{\sum_{j=1}^{i} xI(j)} \qquad (4)$$

where $xG(j)$ is the gain value of jth retrieved element and $xI(j)$ is the gain value of the jth relevant element, where the relevant elements are ranked according to their gain values (this ranking corresponds to the optimal ranking that can be achieved for a given user's request). For a given rank i, $nxCG[i]$ reflects the relative gain accumulated up to that rank, compared to the gain that could have attained by the optimum best ranking; 1 represents best performance. $MAnxCG[i]$ is the mean average $nxCG[i]$ scores up to a given rank i. We use $MAnxCG[i]$ in this paper instead of $nxCG[i]$ because the former reflects on the quality of the ranking, whereas the latter reports a set-based value measured at a single point in the ranking.

MAep is the *non-interpolated mean average effort-precision*, where the effort-precision ep at a given gain-recall value gr is defined as the number of visited ranks required to reach a given level of gain relative to the total gain that can be obtained:

$$ep[r] := \frac{i_{ideal}}{i_{run}} \qquad (5)$$

where i_{ideal} is the rank position at which the cumulated gain of r is reached by the ideal curve and i_{run} is the rank position at which the cumulated gain of r is reached by the system run. A score of 1 reflects ideal performance. The gain-recall gr at rank i is calculated as:

$$gr[i] := \frac{\sum_{j=1}^{i} xG[j]}{\sum_{j=1}^{n} xI[j]} \qquad (6)$$

where n is the number of relevant elements.

For the thorough retrieval task, the full set of relevance assessments is used to derive both the values $xG[.]$ and $xI[.]$. For the focused retrieval task, a subset of the relevance assessments is used to calculate the value $xI[.]$. This subset consists of non-overlapping elements, corresponding to the best elements to retrieve, i.e. elements at the right level of granularity. The construction of the subset is described in [28].

4.2 Experimental Setting

Our experiments are carried out in the following setting. To investigate our first objective of studying the characteristics of XML elements reflected by their topic shifts

(Section 5), we use the INEX 2003-2004 test collection. Following that, to compare length and topic shifts, we use the INEX 2003-2004 test collection to perform some initial analysis (Section 6.1). We then use the INEX 2005 test collection to examine the effects of using topic shifts on retrieval effectiveness compared to length (Section 6.2), and to investigate the use of topic shifts for the focused access to XML documents (Section 7).

In all cases, the first step is to decompose the INEX XML documents into semantic segments through the application of TextTiling (Section 3.1). We consider the discourse units in TextTiling to correspond to *paragraph* XML elements (paragraph elements are any elements of the "para" entity as defined in the INEX document collection DTD[11]). This means that for the purpose of our investigation, we considered paragraph elements to be the lowest possible level of granularity of a retrieval unit. Although this can be viewed as collection-dependent and might indeed change from one collection to the next, it is likely that for many XML content-oriented collections, meaningful content will occur mainly at paragraph level and above.

For the remainder of the paper, when we refer to the XML elements considered in our investigation, we will mean the subset consisting of paragraph elements and of elements containing at least one paragraph element as a descendant element. Accordingly, the generated semantic segments can only correspond to paragraph elements and to their ancestors.

As TextTiling requires a text-only version of a document, each XML document has all its tags removed and is decomposed by applying the algorithm to sequences of paragraphs. Unless otherwise stated, we use the following parameters to $W = 32$ and $K = 6$ for the TextTiling algorithm. These parameters were selected based on preliminary experiments using TextTiling on a small subset of the INEX collection. We examined values between 20 and 40 for the parameter W, while fixing K at 6 (W=20 and K=6 are the recommended values for the TextTiling algorithm [22]). The value $W = 32$ generated the most similar segments to those provided by human judgments (as provided by one of the authors).

In this work, we do not consider the optimization of the parameters of TextTiling when we apply it to the INEX collection. Since, the focus of our work is to examine the characteristics of topic shifts, we consider TextTiling to be sufficient for our purposes even with sub-optimal parameter settings, as long as it produces reasonable results. We however return to the TextTiling parameters in Section 7.2, where we experiment with various settings.

For the experiments based on the INEX 2003 and 2004 data, as described in Section 4.1.1 we use Version 1.4 of the INEX collection. The number of XML elements

---

[11] `<!ENTITY    %    para    "ilrj|ip1|ip2|ip3|ip4|ip5|item-none|p|p1|p2|p3">`.

considered in our experiments is 1,433,539 (18% of the total number of elements in the INEX collection Version 1.4). Although this figure appears to be low, the considered elements form a large part of the actual documents, i.e. they correspond to 80.35% of the actual documents.

Furthermore, to ensure that our reduced element set covers a high proportion of the relevant elements, we looked at the percentage of XML elements in our reduced element set assessed as relevant to at least one of the topics in Version 2.5 of the INEX 2003 and Version 3.0 of the INEX 2004 data sets, compared to those assessed as relevant in the full set. With respect to the INEX 2003 relevance assessments, the elements considered in our experiments correspond to 89% of the relevant elements assessed as e3-s3 in the full set. This number is 81% for the e123-s3 assessments, and 86% for the e3-s123 assessments. With respect to the INEX 2004 relevance assessments, the elements considered in our experiments correspond to 63% of the relevant elements assessed as e3-s3, 74% assessed as e123-s3 and 59% assessed as e3-s123, in the full set.

For the experiments based on the INEX 2005 data, we use Version 1.8 of the INEX collection. The number of XML elements considered in this case is 1,925,673 (17% of the total number of elements in the INEX collection Version 1.8). These considered elements correspond to 79.3% of the actual documents. Using the topics in Version $2005 - 003$ and ignoring elements assessed as too small, the elements considered in our experiments correspond to 82% of the relevant elements assessed as e2-s1 (highly exhaustive, highly specific), 77% assessed as e12-s1 (highly specific with any exhaustivity) and 89% assessed as e2-sx (highly exhaustive with any specificity degree).

After the application of TextTiling in the above data sets, we compute the number of topic shifts in elements. For this computation, we do not remove stopwords. Since the subset of elements considered in our experiments contains both a large part of the actual document collection and a high proportion of the elements assessed as relevant, we can be confident that the results obtained from our investigation are indeed meaningful.

5 Characteristics of topic shifts: Experiments and Results

This section discusses the results of the experiments we conducted to investigate the characteristics of XML elements reflected by their number of topic shifts. First, we examine the relation between the logical structure of the XML documents and their semantic decomposition as obtained using the segmentation algorithm (Section 5.1). Next, we discuss the distribution of the number of topic shifts across element types, as well as the distribution of the difference in the number of topic shifts between

Table 3. Statistics of INEX collection Version 1.4

Logical Structure	
number of paragraphs	938,483
average paragraph length	55
median of paragraph length	60
Semantic Decomposition	
number of segments	$140,949$
number of paragraphs per segment	6.65
minimum number of topic shifts	1
maximum number of topic shifts	156
mean number of topic shifts	1.6523

parent and children elements (Section 5.2). We then examine whether the number of topic shifts of an element reflects its relevance (Section 5.3), and more particularly its exhaustivity and specificity (Section 5.4). We also examine how the patterns of propagation of specificity and exhaustivity from children elements to their parents are affected by the number of topic shifts of the parents (Section 5.5).

5.1 Logical Structure vs Semantic Decomposition

This section discusses the relation between the logical structure of XML documents and their semantic decomposition, through the correspondence between XML elements and the formed semantic segments.

First, we examine the output generated by the Text-Tiling algorithm when applied on Version 1.4 of the INEX collection. Table 3 shows some statistics. The number of paragraph elements considered is 938,483 (65% of all considered XML elements), while the semantic decomposition has detected $140,949$ segments. The fact that the number of detected semantic segments is about 15% of the number of XML element paragraphs clearly shows that often a topic is discussed across several paragraphs. This is also indicated by the average number of paragraphs per segment, which is 6.65.

The paragraph-per-segment ratio, of 6.65, indicates that our choice of the TextTiling parameters produces reasonable results. This ratio should not be too low as it will be close to the paragraph decomposition, and it also should not be too high as it will be equivalent to article decomposition. For comparison, the average number of paragraphs in articles on Version 1.4 of the INEX collection is 77.52.

We also examine whether the author-specified boundaries of the considered XML elements (indicated by the XML markup) coincide with the boundaries of the semantic segments. Our experiments show that there is an exact match for both boundaries of XML elements to those of semantic segments for only 4.3% of the elements. For 24.7% of the elements, only one of their boundaries coincides with a semantic segment boundary. For the remaining 70% of the elements, none of their boundaries coincides with those of the semantic segments.

Overall, the semantic decomposition generates an additional structure not captured by the logical structure,
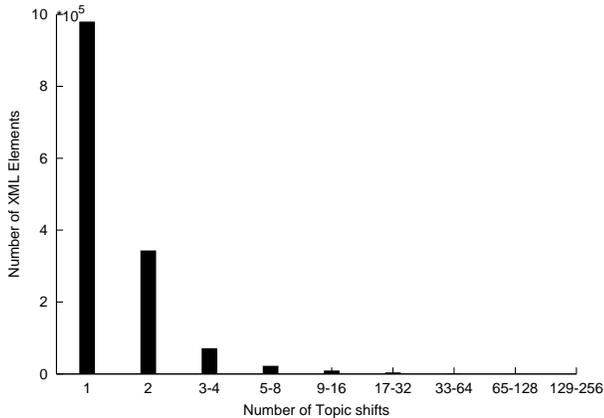
Fig. 2. Distribution of XML elements across topic shift levels

and as such may constitute a new source of evidence for content-oriented XML retrieval.

5.2 Distribution of topic shifts numbers

We examine the distribution of the number of topic shifts of the considered XML elements (i.e. paragraph elements and their ancestors). In our experiments, the number of topic shifts ranges from 1 (no topic shift) to 156 (as shown in Table 3). We rank the XML elements with respect to their number of topic shifts and then group the elements into exponential-sized "bins" to represent the different numbers of topic shifts. We use 9 bins on an exponential scale ranging from $2^0 (= 1)$ to $2^8 (= 256)$. Therefore, we consider 9 levels, which respectively correspond to the number of topic shifts being 1, 2, 3–4, 5–8, 9–16, 17–32, 33–64, 65–128 or 129–256. We refer to these 9 levels as *topic shift levels*. We use exponential-sized bins due to the large number of elements with low number of topic shifts. Base 2 allows us to distinguish between elements with a low number of topic shifts (1, 2, 3–4) and the rest.

Figure 2 depicts the number of XML elements for each topic shift level. The distribution of elements is heavily skewed towards elements with low number of topic shifts. This is however to be expected as 65% of the considered elements are paragraphs, which correspond to the retrieval units with the lowest level of granularity that are allowed in our study (see Section 4.2).

We therefore investigate the distribution of XML elements of different types across topic shift levels. These results are shown in Table 4. Element type p (paragraph) was used as the basic element type for comparison. The other types such as *article*, *dialog*[12], *sec* and *bm* were selected based on having the highest mean in the number of topic shifts and length[13]. Overall, the majority of elements of each particular element type have a low

---

[12] In INEX, *dialog* contains a number of questions and answers.
[13] Here, the length of an XML element refers to the number of terms in the content of the descendant paragraphs of that element.

number of topic shifts, ranging from 1 for *sec* and *p*, to 5–8 for *article*. This shows that even larger elements (i.e. *article*, *dialog* and *sec*), which are among the elements with the highest average length, have themselves a low number of topic shifts. However, as user requests will usually be concerned with low numbers of topics, the difference among these low numbers of topics may be useful in determining the best elements to retrieve.

Finally, we examine the difference in the number of topic shifts between XML children elements and their parents. This allows us to determine whether elements higher in the logical structure discuss more topics than those lower in the structure. Out of the total 1,433,539 XML elements considered in our experiments, 938,483 elements (65%) are paragraphs corresponding to our leaf level, 268,271 elements (19%) have only one child element and 226,785 elements (16%) have two or more children. Since we consider that differences in the number of topic shifts occur only when an element has two or more children, we examine only the 226,785 elements having two or more children.

Table 5 shows the distribution of these elements across the different values in topic shift levels between parent and children elements. These values are calculated as the difference between the topic shift level of a parent element and the maximum topic shift level of its children. The distribution of elements is heavily skewed towards low difference in topic shift levels. Indeed, parent and children elements may have the same number of topic shifts, as observed in 69% of elements having two or more children, which indicates that elements higher in the tree do not necessarily discuss more topics than those lower in the tree.

Overall, our experiments indicate that elements residing higher in the logical structure do not necessarily discuss a large number of topics or more topics than their children elements. As elements higher in the logical structure will be in general larger than those lower in the structure, an increase in the length of an element does not automatically imply that the element discusses more topics. This prompted us to investigate the relationship between element length and topic shifts, in Section 6, where we compare the effects of using length and topic shifts in estimating the relevance of an element.

5.3 Relevance vs Topic shifts

This section examines the number of topic shifts of relevant XML elements, in order to investigate whether it constitutes a feature that could be related to the different degrees of relevance of an element. Our motivation stems from the fact that the definitions of relevance in INEX, and more specifically the definitions of the dimensions of relevance, are expressed in terms of the number of, and extent to which, topics are discussed within each element. Consequently, we hypothesize that the rel-

Table 4. Distribution of different XML elements across topic shift levels

Tag type	1	2	3–4	5–8	9–16	17–32	33–64	65–256	Total
article	0(0%)	0(0%)	1342(11%)	**4182(35%)**	3357(28%)	2087(17%)	896(8%)	107(1%)	11971(100%)
dialog	47(24%)	42(22%)	**50(26%)**	35 (18%)	17(9%)	3(1%)	0(0%)	0(0%)	194(100%)
sec	**22951(33%)**	16660(24%)	16832(25%)	9098 (13%)	2845(4%)	427(1%)	27(0%)	3(0%)	68843(100%)
bm	0(0%)	**6835(75%)**	1398(15%)	614(7%)	165(2%)	36 (1%)	7(0%)	1(0%)	9056(100%)
p	**524920(72%)**	188086(25%)	19928(3%)	0(0%)	0 (0%)	0(0%)	0 (0%)	0(0%)	732934(100%)

Table 5. Distribution of XML elements across difference values in topic shift levels between parent and children elements

Topic shift Difference	0	1	2	3–4	5–8	9–16	17–32	33–64	65–128	Total
Number of elements	156998 (69%)	31042 (14%)	17345 (8%)	12365 (5%)	5659 (2%)	2209 (1%)	946 (0.4%)	208 (0.1%)	14 (0%)	226785 (100%)

Table 6. Distribution of relevant XML elements across topic shift levels

Measure–Year	1	2	3–4	5–8	9–16	17–32	33–64	65–256	Total
e3-s3–2004	731 (45%)	486 (30%)	160 (10%)	119 (7%)	69(4%)	33 (2%)	13 (1%)	4 (0%)	1615 (100%)
e123-s3–2004	3616 (54%)	2028 (30%)	588 (9%)	250 (4%)	147 (2%)	87 (1%)	23 (0%)	4 (0%)	6743 (100%)
e3-s123–2004	1123 (36%)	825 (26%)	384 (12%)	371 (12%)	232 (8%)	125 (4%)	66 (2%)	13 (0%)	3139 (100%)
e3-s3–2003	338 (26%)	315 (24%)	209 (16%)	205 (16%)	131 (10%)	68 (5%)	44 (3%)	3 (0%)	1313 (100%)
e123-s3–2003	2805 (40%)	2021 (29%)	890 (13%)	605 (9%)	333 (5%)	183 (3%)	89 (1%)	12 (0%)	6938 (100%)
e3-s123–2003	560 (22%)	568 (23%)	391 (15%)	417 (17%)	259 (10%)	197 (8%)	110 (4%)	17 (1%)	2519 (100%)

evance of an element could be reflected by the number of topic shifts within that element.

We examine the distribution of relevant XML elements with respect to strict (e3-s3), specificity-oriented (e123-s3) and exhaustivity-oriented (e3-s123) INEX relevance criteria across the topic shift levels. The results of this investigation for the INEX 2003 and 2004 data sets are reported in Table 6. Overall, the number of topic shifts of elements assessed as relevant, with respect to any of the relevance criteria, tends to be low across both data sets.

For INEX 2004, we observe that 84% of the specificity-oriented, 75% of the strict and 62% of the exhaustivity-oriented relevant elements have topic shift scores less than 3. For INEX 2003, 82%, 66% and 60% of the respective relevant elements have topic shift scores less than 5. This indicates that highly specific elements discuss fewer topics compared to highly exhaustive elements, which accords well with the INEX definition of specificity. It also confirms our expectation for the exhaustive elements, since by definition they are the ones covering all themes requested by a query. There is however an upper bound, in the sense that elements with high topic shift level are not necessarily relevant with respect to the exhaustivity-oriented measure[14].

The observed behaviour of the number of topic shifts for the different relevance criteria confirms the intuition that the differences between the relevance criteria and therefore differences between the definitions of exhaustivity and specificity are captured by the number of topic shifts. Thus using the number of topic shifts seems a good source of evidence to estimate the relevance of an element in XML retrieval.

5.4 Specificity / Exhaustivity vs Topic shifts

The observations from the previous section motivate us to further examine the number of topic shifts of XML elements assessed as relevant at various levels of exhaustivity and specificity.

To achieve this, we examine, for each topic shift level, the distribution of relevant XML elements across the various specificity levels (e123–s1, e123–s2, e123–s3) and exhaustivity levels (e1–s123, e2–s123, e3–s123). Table 7 (Table 8) presents, for each topic shift level, the distribution of relevant elements in the INEX 2003 and 2004 data sets across different levels of specificity (exhaustivity).

In Table 7, we observe that for the INEX 2004 data set and low number of topic shifts $(1, 2)$, the number of highly specific relevant elements (e123–s3) is greater than those of elements with lower specificity (e123–s2 and e123–s1). This indicates that relevant elements discussing fewer topics tend to be more highly specific, which again accords well with the INEX definition of specificity. For higher numbers of topic shifts (≥ 3), there is more preference for elements with the lowest specificity e123–s1 (marginally specific). Furthermore, for any number of topic shifts ≥ 5, the numbers of elements assessed as e123-s2 (fairly specific) and e123-s3 (highly specific) are more or less equal.

With respect to the INEX 2003 data set, the relevant elements are distributed in a different manner. For all topic shift levels, elements with the lowest specificity are preferred, followed by the highly specific elements and then by the fairly specific ones.

The differences between the two data sets, especially those observed in the low topic shift levels and for the

---

[14] As a side remark, the tendency for relatively more elements with lower topic shifts to be assessed as relevant in INEX 2004 compared to those in INEX 2003, could be interpreted as an improvement, over time, in the understanding among assessors, of the (relatively unfamiliar) concept of the specificity dimension of relevance [43].

Table 7. Distribution of relevant XML elements with respect to their specificity for each topic shift level

	2004				2003			
$score(e)$	e123-s1	e123-s2	e123-s3	Total	e123-s1	e123-s2	e123-s3	Total
1	2103(28%)	1740(23%)	3616(48%)	7459(100%)	3256(40%)	2161(26%)	2805(34%)	8222(100%)
2	1529(33%)	1064(23%)	2028(44%)	4621(100%)	2228(39%)	1534(27%)	2021(35%)	5783(100%)
3–4	845(45%)	447(24%)	588(31%)	1880(100%)	1147(45%)	540(21%)	890(35%)	2577(100%)
5–8	1008(66%)	262(17%)	250(16%)	1520(100%)	1072(54%)	313(16%)	605(30%)	1990(100%)
9–16	566(66%)	150(17%)	147(17%)	863(100%)	545(52%)	161(15%)	333(32%)	1039(100%)
17–32	309(66%)	75(16%)	87(18%)	471(100%)	352(52%)	138(21%)	183(27%)	673(100%)
33-64	141(70%)	38(19%)	23(11%)	202(100%)	187(59%)	40(13%)	89(28%)	316(100%)
64–256	27(84%)	1(3%)	4(13%)	32(100%)	26(54%)	10(21%)	12(25%)	48(100%)

Table 8. Distribution of relevant XML elements with respect to their exhaustivity for each topic shift level

	2004				2003			
$score(e)$	e3-s123	e2-s123	e1-s123	Total	e3-s123	e2-s123	e1-s123	Total
1	1123(15%)	2090(28%)	4246(57%)	7459(100%)	560(7%)	1665(20%)	5997(73%)	8222(100%)
2	825(18%)	1215(26%)	2581(56%)	4621(100%)	568(10%)	1269(22%)	3946(68%)	5783(100%)
3–4	384(20%)	500(27%)	996(53%)	1880(100%)	391(15%)	674(26%)	1512(59%)	2577(100%)
5–8	371(24%)	362(24%)	787(52%)	1520(100%)	417(21%)	573(29%)	1000(50%)	1990(100%)
9–16	232(27%)	220(25%)	411(48%)	863(100%)	259(25%)	276(27%)	504(49%)	1039(100%)
17–32	125(27%)	118(25%)	228(48%)	471(100%)	197(29%)	197(29%)	279(41%)	673(100%)
33-64	66(33%)	47(23%)	89(44%)	202(100%)	110(35%)	48(15%)	158(50%)	316(100%)
64–256	13(41%)	7(22%)	12(38%)	32(100%)	17(35%)	14(29%)	17(35%)	48(100%)

highly specific elements, can again be attributed to the better understanding among assessors in 2004 of the specificity dimension of relevance [43] (see Footnote 14).

In Table 8 (related to exhaustivity), we observe that the results are consistent across both data sets and indicate that as the number of topic shifts of elements increases, relatively more elements are assessed on the higher level of exhaustivity than the lower one. This observation also accords well with the INEX definition of exhaustivity.

Our observations on the preference among various specificity and exhaustivity levels within each topic shift level confirm our expectations arising from the INEX definitions for the specificity and exhaustivity relevance dimensions. This suggests that topic shifts could be useful in modeling specificity and exhaustivity and their scale.

5.5 Specificity / Exhaustivity Propagation vs Topic shifts

In this section we examine the patterns of specificity and exhaustivity propagation for XML elements. This enables us to investigate whether the propagation of specificity and exhaustivity from children elements to their parents is affected by the number of topic shifts of the parent elements. In this experiment, we only use the INEX 2004 data set because of the better understanding of the difference between specificity and exhaustivity among assessors.

We consider only XML elements with two or more children, since differences in topic shifts can occur only in elements having two or more children. The specificity dimension of relevance has a propagation property such that, the specificity degree of an element is always less than or equal to the maximum specificity of its children. Regarding the exhaustivity dimension, if an element is exhaustive to a query then all its ascendant elements will also be relevant and will have an exhaustivity degree at least equal to its exhaustivity. More details can be found in [43].

For each relevant element, we denote the propagation of specificity from its children as $s_{children}$tos_{parent}, where $s_{children}$ is the maximum specificity of the children and s_{parent} is the specificity of the parent relevant element. Since the specificity of the parent is less than or equal to the maximum of that of its children, there are 5 possible informative cases: 2to1, 2to2, 3to1, 3to2 and 3to3, referred to as *propagation categories*. We do not consider the 1to1 case, since it is mandatory according to the rules of relevance assessments in INEX (i.e. an element with $s = 1$ will have all its ascendants with the same $s = 1$ value).

Table 9 presents, for each topic shift level, the distribution of the parent relevant XML elements across the propagation categories. For low numbers of topic shifts $(1, 2)$, there is a high preference to propagate the same specificity values (2to2 and 3to3) from children to their parent. For instance, when the number of topic shifts of a parent element is equal to 1, 57% of the relevant elements corresponds to the 3to3 propagation category and 27% to 2to2, compared to the 5%, 4% and 7%, of elements corresponding, respectively, to the 2to1, 3to1 and 3to2 categories. This suggests that when the number of topic shifts in the parent element is low, it is highly likely that the parent element discusses topics at the same level of specificity as its children. It also shows that the specificity of a parent with low number of topic shifts is less affected by the amount of non-relevant text in its children.

The propagation of exhaustivity from children to their parent is denoted as $e_{children}$toe_{parent}, where $e_{children}$ is the maximum exhaustivity of the children of a relevant element and e_{parent} is the exhaustivity of that relevant

Table 9. Distribution of relevant XML elements across specificity propagation categories in INEX 2004 relevance assessments for each topic shift level.

$score(e)$	2to1	2to2	3to1	3to2	3to3	Total
1	84 (5%)	421 (27%)	56 (4%)	108 (7%)	903 (57%)	1572 (100%)
2	112 (10%)	238 (22%)	59 (6%)	136 (13%)	522 (49%)	1067 (100%)
3–4	147 (16%)	193 (21%)	98 (11%)	132 (14%)	355 (38%)	925 (100%)
5–8	137 (20%)	138 (20%)	98 (15%)	92 (14%)	210 (31%)	675 (100%)
9–16	75 (20%)	87 (24%)	28 (8%)	49 (13%)	129 (35%)	368 (100%)
17–32	34 (18%)	47 (25%)	13 (7%)	21 (11%)	74 (39%)	189 (100%)
33–64	17 (22%)	26 (33%)	5 (6%)	10 (13%)	21 (27%)	79 (100%)
65–256	0 (0%)	0 (0%)	0 (0%)	1 (50%)	1 (50%)	2 (100%)

Table 10. Distribution of relevant XML elements across exhaustivity propagation categories in INEX 2004 relevance assessments for each topic shift level.

$score(e)$	1to1	1to2	1to3	2to2	2to3	Total
1	1143 (61%)	81 (4%)	9 (0%)	598 (32%)	47 (3%)	1878 (100%)
2	748 (60%)	54 (4%)	5 (0%)	403 (32%)	35 (3%)	1245 (100%)
3–4	697 (61%)	60 (5%)	13 (1%)	340 (30%)	30 (3%)	1140 (100%)
5–8	742 (65%)	33 (3%)	12 (1%)	324 (29%)	25 (2%)	1136 (100%)
9–16	399 (63%)	7 (1%)	4 (1%)	212 (34%)	10 (2%)	632 (100%)
17–32	227 (64%)	9 (3%)	2 (1%)	109 (31%)	6 (2%)	353 (100%)
33–64	88 (64%)	2 (1%)	0 (0%)	45 (33%)	3 (2%)	138 (100%)
65–256	12 (63%)	0 (0%)	0 (0%)	7 (37%)	0 (0%)	19 (100%)

element. Since the exhaustivity of the parent is equal to or greater than the maximum of that of its children, there are 5 possible informative cases: 1to1, 1to2, 1to3, 2to2 and 2to3. We do not consider the 3to3 case, since it is mandatory.

Table 10 presents the distribution of relevant XML elements for topic shift levels across propagation categories. The relative distribution of propagated exhaustivity is rather similar in all topic shift levels, with higher preference for the 2to2 and 1to1 propagation categories. Therefore, there is no evidence that the number of topic shifts of the parent element influences the propagation of exhaustivity.

Overall, our results suggest that when the number of topic shifts in the parent element is low, it is highly likely that the parent element discusses topics at the same level of specificity as its children. We also observed that the specificity of a parent with a low number of topic shifts is less affected by the amount of non-relevant text in its children. Our results did not show any evidence indicating that the number of topic shifts of the parent element influences the propagation of exhaustivity.

In the last three sections (Sections 5.3, 5.4, and 5.5), our investigations demonstrate that the number of topic shifts in an XML element constitutes a good source of evidence to estimate the relevance of an XML element. This section (Section 5.5) further indicates that the number of topic shifts seems particularly suited to capture the specificity dimension of relevance in XML retrieval. Indeed, we could use the topic shifts score to select which element to return, a parent element or its children elements, when all have been estimated as relevant by an XML retrieval system to a given user request, i.e., the element at the right level of granularity. In Section 7, we describe how we use the number of topic shifts for this purpose.

6 Topic shifts vs. length: Experiments and Results

In this section, we report on the experiments, and their results, that were carried out to investigate our second research objective, which is to compare length and topic shifts.

Length has been shown to constitute a useful source of evidence for XML retrieval [26]. Generally when the length of an element increases, it is highly likely that it will discuss more topics. Therefore, it might be argued that the number of topic shifts reflects evidence already captured by their length and as such it does not constitute a distinct feature. However, this is not always the case, as it was shown in Section 5.2, where the number of topic shifts of parent elements was compared to that of their children. Even though the length from children to their parents increases, the number of topic shifts in the majority of cases stays the same, i.e. it does not vary when the length increases. This motivates us to perform a comparative analysis between the number of topic shifts and length features of XML elements. The particular question we are addressing is whether the number of topic shifts in XML elements provide us with different evidence than that captured by their length.

To perform this comparison, we incorporate this new source of evidence, the number of topic shifts, as a feature in a retrieval setting, and examine its effect on XML retrieval effectiveness, compared to length. There are different ways we could adopt, but for a meaningful comparison between length and topic shifts in XML retrieval, we follow the same approach as that of analysing the length of XML element as [26]. First, we examine the correlation between the prior probability of relevance and the number of topic shifts of XML elements (Section 6.1). Next, we incorporate the length and topic

shifts as priors in a language modeling approach in the context of the thorough retrieval task (Section 6.2).

6.1 Topic shifts as prior

In this section, we examine the correlation between the prior probability of relevance and the number of topic shifts in XML elements. We use the relevance assessments of INEX 2003 and 2004. In this experiment we consider as relevant only those elements assessed as highly exhaustive (e = 3) and highly specific (s = 3). We have restricted ourselves to this subset as it contains the elements at the right level of granularity, which are those we want to identify with the use of topic shifts.

Figure 3(a) shows the topic shift score distribution of relevant XML elements. The distributions for both INEX 2003 and 2004 are heavily skewed towards elements with low numbers of topic shifts and are similar to the topic shift score distribution of all the elements in the collection (see Figure 2).

Next, we investigate the probability of relevance of XML elements by dividing the number of relevant elements for each topic shift level by the number of elements in the collection at the corresponding topic shift level. Results in Figure 3(b) show that the distribution is heavily skewed towards higher topic shift levels, which is in the opposite direction of what is observable in the topic shift score distribution of the collection in Figure 2.

Our analysis on the topic shift scores of XML elements in Figure 3(b) shows that, when estimating the relevance of an XML element, a bias is needed towards elements with a high number of topic shifts. Following a similar analysis, Kamps et al.[26] have previously shown that a bias is also needed towards retrieving relatively long elements. Their investigation was based on the INEX 2002 and INEX 2003 data sets. However, for the INEX 2005 data set, a decrease in the average length of relevant elements (highly exhaustive and highly specific) has been reported [51]. Thus estimating the relevance of an element with bias towards large elements may not always be appropriate. Our view is that topic shifts could provide a more natural bias for estimating relevance in XML retrieval.

To directly compare length and topic shifts, we incorporate each of these features into a retrieval setting, where the aim is to estimate the relevance of XML elements for given an information need.

6.2 Comparing length and topic shifts in XML retrieval

This section presents and discusses the results of comparing length and topic shifts by incorporating each of them in a retrieval setting. The retrieval setting we consider is the thorough retrieval task (described in Section 4.1.3) applied on the INEX 2005 data set, which is also the task followed in [26]. This task consists of estimating the relevance of potentially retrievable elements in the collection, and rank these elements in decreasing order of their estimated relevance.

First, we present the retrieval approaches incorporating length and topic shifts (Section 6.2.1). Next, we describe the setting of our experiments (Section 6.2.2). Finally, we present and analyse our experimental results (Section 6.2.3).

6.2.1 Topic shifts, length and prior probability of relevance

For our experiments, the relevance of an XML element is estimated with a statistical language modeling approach [25], similar to that followed in [26]. Language modeling approaches have shown satisfactory results in content-oriented XML retrieval (e.g. [26, 45, 41, 24, 33]). The language modeling approach allows us to combine "non-content" features of elements (or documents) (e.g. length, topic shifts) with the scoring mechanism. It is a sound, flexible and promising framework, not only for XML retrieval, but also for IR research in general [11].

We estimate the likelihood for a query $q = (t_1, t_2, ..., t_n)$ to be generated from an element e, $P(q|e) = P(t_1, .., t_n|e)$, using a multinomial language model with Jelinek-Mercer smoothing [25]:

$$P(t_1, .., t_n|e) = \prod_{i=1}^{n} (\lambda P(t_i|e) + (1 - \lambda) P(t_i|C)) \quad (7)$$

where

- t_i is a query term in q,
- $P(t_i|e) = \frac{c(t_i,e)}{|e|}$ is the probability of generating the query term t_i from element e, with $c(t_i, e)$ the number of occurrences of the query term t_i in element e, and $|e|$ the number of terms in element e,
- $P(t_i|C) = \frac{ef(t_i)}{\sum_t ef(t)}$ is the probability of query term t_i in the collection, with $ef(t)$ the total number of XML elements in which term t occurs, and
- λ (weight on the element language model) is a weighting parameter between 0 and 1 which is used in smoothing the element model with the collection model.

The ranking is produced by computing the relevance of an element e to a given query q as:

$$P(e|q) \propto P(e)P(q|e) \quad (8)$$

where $P(e)$ is the prior probability of relevance for element e and $P(q|e)$ is given by Equation 7.

We experiment with two different prior probabilities of relevance $P(e)$. First, we define the prior probability to be proportional to the length of an element (Equation 9), and second to be proportional to the number of topic shifts in an element (Equation 10). We also compare these two approaches with a baseline using a uniform prior. Therefore, we consider three ways to estimate relevance, each leading to a retrieval approach:

(a) Relevant Elements per Topic Shift Level   (b) Probability of Relevance
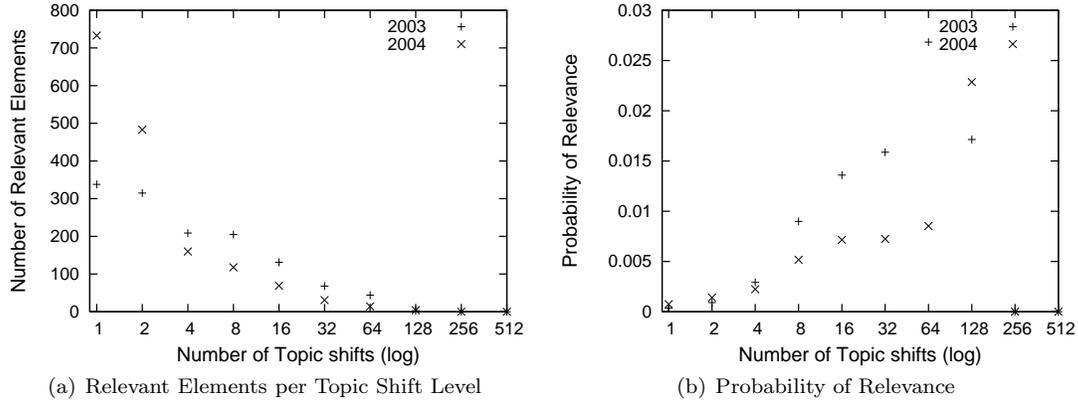
Fig. 3. Topic shifts score distribution of XML elements

1. A retrieval approach based on a language modeling approach with a uniform prior probability of relevance for elements (LM).
2. A retrieval approach based on a language modeling approach where we incorporate length as prior probability of relevance for each element (LM_L) where:

$$P(e) = \frac{\sum_t c(t, e)}{\sum_e \sum_t c(t, e)} \qquad (9)$$

3. A retrieval approach based on a language modeling approach where we incorporate the number of topic shifts as prior probability of relevance for each element (LM_T) where:

$$P(e) = \frac{score(e)}{\sum_e score(e)} \qquad (10)$$

where $score(e)$ is the number of topic shifts in an XML element e as defined in Equation 1.

6.2.2 Experimental Setting

For all of our three retrieval approaches, all elements shorter than 20 terms are removed, as it has been shown effective in XML retrieval by similar strategies (e.g. [26]). The remaining elements are indexed by removing stopwords, but without applying any stemming.

To compare the effects of length and topic shifts on retrieval performance, we set the smoothing parameter, λ, to a fixed value. We examined a range of values between [0,1] for λ. We have made a trade-off between the performance of the above three approaches regarding the two quantization functions, strict and generalised, and set λ to 0.1. This value is close to the traditional setting for document retrieval (λ=0.15), which has shown satisfactory results [25].

In the experiments carried out and reported in this section, our aim is to understand the effect of topic shifts on XML retrieval. For this purpose, no optimization is performed, and no additional sources of evidence are considered.

For each of the retrieval approaches, the top 1,500 ranked elements are returned as answers for each of the CO topics (Version 2005-003). Retrieval effectiveness is evaluated using the XCG metrics: $MAnxCG$ at six different cut-off points (1, 2, 3, 10, 25, 50), and $MAep$. Both strict and generalised quantization functions are used.

To determine whether the differences in performance between two approaches are statistically significant, we use the bootstrapping significance testing method [12]. This method is a non-parametric inference test that has previously been applied in retrieval evaluation in IR [39, 50] and XML retrieval [26]. Improvements at confidence levels 95% and 99% over the baseline are respectively marked with + and ++. Similarly, decreases in performance at confidence level of 95% and 99% are marked with – and – –.

6.2.3 Experimental Results and Analysis

This section presents and discusses our experimental results. Table 11 presents, for each quantization function, the evaluation results for all measures for the three retrieval approaches, with the uniform prior approach LM acting as the baseline. To directly compare the length (LM_L) and topic shifts (LM_T) runs, the last column of Table 11 presents the changes over LM_L.

We first discuss the results with respect to $MAep$. Under the *generalised quantization function*, our results show that using either the length prior or the topic shifts prior leads to significant improvements of +18.34%(++) and +14.25%(++), respectively, over the language model approach with the uniform prior[15]. This confirms that under the generalised case, a bias either towards retrieving longer elements or towards retrieving elements that discuss a high number of topics provides a better estimate of relevance in XML retrieval. Under the *strict quantization function*, the effectiveness drops slightly when

---

[15] The *MAep* scores in Tables 11 and 12 are much lower for XML element retrieval than document retrieval, partly, due to the large number of relevant elements, which comes from their nested nature - if an element is relevant, all its ascendents will also be relevant.

Table 11. Thorough retrieval task using the INEX 2005 data: $MAep$ and $MAnxCG$ at different cut-off points considering LM as baseline

	LM	LM_L	chg over LM	LM_T	chg over LM	chg over LM_L
				General		
$MAnxCG@1$	0.2551	0.2488	(-2.47%)	0.2468	(-3.25%)	(-0.8%)
$MAnxCG@2$	0.2449	0.2434	(-0.61%)	0.2610	(+6.57%)	(+7.23%)
$MAnxCG@3$	0.2420	0.2468	(+1.98%)	0.2663	(+10.04%)	(+7.9%)
$MAnxCG@10$	0.2364	0.2633	(+11.38%)	0.2610	(+10.41%)+	(-0.87%)
$MAnxCG@25$	0.2396	0.2659	(+10.98%)	0.2565	(+7.05%)+	(-3.54%)
$MAnxCG@50$	0.2416	0.2680	(+10.93%)+	0.2571	(+6.42%)+	(-4.07%)
$MAep$	0.0758	0.0897	(+18.34%)++	0.0866	(+14.25%)++	(-3.46%)
				Strict		
$MAnxCG@1$	0.0385	0.0385	(0%)	0.0385	(0%)	(0%)
$MAnxCG@2$	0.0288	0.0385	(+33.68%)++	0.0481	(+67.01%)++	(+24.94%)++
$MAnxCG@3$	0.0321	0.0427	(+33.02%)	0.0534	(+66.36%)++	(+25.06%)++
$MAnxCG@10$	0.0455	0.0393	(-13.63%)	0.0569	(+25.05%)+	(+44.78%)+
$MAnxCG@25$	0.0528	0.0539	(+2.08%)	0.0525	(-0.57%)	(-2.6%)
$MAnxCG@50$	0.0717	0.0739	(+3.07%)	0.0793	(+10.6%)	(+7.31%)
$MAep$	0.0198	0.0191	(-3.54%)	0.0196	(-1.01%)	(+2.62%)

using length (-3.54%) and topic shifts (-1.01%) priors. However, this decrease in performance is not significant. Focussing on the two approaches employing non-uniform priors, LM_L and LM_T, we observe that they perform comparably when evaluated using the $MAep$ measure. Any differences between them for both quantization functions are not significant, as illustrated in the last column of Table 11.

Next, we discuss the results obtained with $MAnxCG$. Under the *generalised quantization function*, using topic shifts leads to significantly improvements in the effectiveness at the early cutoffs, 10, 25 and 50 and substantial improvements albeit not significant at the extremely early cutoffs, 2 and 3. Thus the observed improvements over the range of cutoffs, shows that using topic shift prior leads to a stable improvement over the baseline. Using length prior also leads to substantially improvements over the baseline at rank 10 and 25 with a significant improvement at rank 50. The last column of the Table 11 illustrates that the difference between using the length and topic shifts prior under the generalised quantization function is not significant.

Under the *strict quantization function*, using topic shifts prior leads to significant improvements at ranks 2, 3, 10 and a substantial improvements at rank 50. Using length prior only improves the results at ranks 2 significantly and a substantial improvement at rank 3. In this case, the difference between the approach employing topic shifts prior, LM_T, and the approach using length priors, LM_L is significant in the ranks 2, 3, 10, which shows a stable improvement for using topics shift prior over length prior at the very early ranks. This experimental evidence indicates that for retrieving highly specific and highly exhaustive elements (the strict case) using topic shifts as prior is useful.

The difference in performance could be interpreted by the reported decrease in the average length of relevant elements (highly exhaustive and highly specific) in the INEX 2005 relevance assessments [51]. This confirms that elements with high number of topic shifts are not always the longest elements. This constitutes another indication that these two sources of evidence, length and topic shifts, are different.

To conclude, our results indicate that length and topic shifts are both beneficial. Using topic shifts, however, seems better at identifying the highly specific and highly exhaustive elements. This leads us to our next set of experiments, our third research objective, where we use topic shifts as additional evidence for the focused access to XML documents.

7 Using topic shifts in Focused XML retrieval

In this section, we use the number of topic shifts as evidence for capturing specificity to provide a focused access to XML repositories. For this purpose, we use the language modeling framework proposed in [2], in which an element-based smoothing process formally incorporates the number of topic shifts to rank elements according to how focused they are to a given query. Our approach is described in Section 7.1. Based on this approach, we carried out a number of experiments on the INEX 2005 data set. In this section, we only report those regarding the use of topic shifts combined with element lengths. Additional experiments were carried out to investigate the effect of various settings of the TextTiling segmentation algorithm, which is used as a basis to calculate the number of topic shifts. The experimental settings are described in Section 7.2, whereas Section 7.3 reports our experimental results and their analysis.

7.1 Element-specific Smoothing Using Topic Shifts

A number of techniques could have been used to carry out our investigation, including the language modeling framework used in Section 6.2.1. There, we used topic shifts as a prior to predict the prior probability of relevance for an element. We decided to use a different language modeling framework, where we assume, a uniform

prior, $P(e)$, and instead we exploit the smoothing process to capture the number of topic shifts. This allows us to formally and elegantly incorporate both length and number of topic shifts to rank elements. Furthermore, our aim is to also examine how these two features allow us to capture the two dimensions of relevance. In particular, motivated by the results of our investigations in Sections 5.3, 5.4, and 5.5, we examine the use of topic shifts for capturing specificity.

In language modeling, smoothing refers to adjusting the maximum likelihood estimator for the element language model so as to correct inaccuracy arising from data sparseness. In the smoothing process, the probability of terms seen in an element are discounted mainly by combining the *element language model* with the *collection language model*, thus assigning a non-zero probability to the unseen terms.

In Section 6.2.1 we used the Jelinek-Mercer smoothing, which comes with a fixed smoothing parameter, and as such, cannot incorporate element features (e.g. length and topic shifts). A smoothing approach that allows us to incorporate element features is the "Dirichlet smoothing" approach [56], and is the one adopted here.

The Dirichlet smoothing is one of the popular document-dependent smoothing methods which was shown to be more effective than Jelinek-Mercer smoothing for example for title ad hoc queries at TREC [56]. With the Dirichlet smoothing, the likelihood for a query is:

$$P(t_1, .., t_n | e) = \prod_{i=1}^{n} \left(\frac{c(t_i, e) + \mu P(t_i | C)}{\mu + |e|} \right) \qquad (11)$$

$$= \prod_{i=1}^{n} \left((1 - \frac{\mu}{\mu + |e|}) \frac{c(t_i, e)}{|e|} + \frac{\mu}{\mu + |e|} P(t_i | C) \right) \quad (12)$$

$$= \prod_{i=1}^{n} \left((1 - \alpha_e) P_{ml}(t_i | e) + \alpha_e P(t_i | C) \right) \qquad (13)$$

where

- t_i is a query term in q,
- μ is a constant,
- $P_{ml}(t_i | e) = \frac{c(t_i, e)}{|e|}$ is the probability of observing term t_i in element e, estimated using the maximum likelihood estimation, with $c(t_i, e)$ the number of occurrences of the query term t_i in element e, and $|e|$ the number of terms in element e,
- $P(t_i | C) = \frac{ef(t_i)}{\sum_t ef(t)}$ is the probability of observing query term t_i in the collection where $ef(t)$ is the number of XML elements in which the term t occurs, and
- $\alpha_e = \frac{\mu}{\mu + |e|}$ is an element-dependent constant which is related to how much probability mass will be allocated to unseen query terms, i.e., the amount of smoothing.

Since the maximum likelihood estimator will generally underestimate the probability of any term unseen in the element, the main purpose of smoothing is to improve the accuracy of the term probability estimation. If we are concerned with the exhaustivity dimension of relevance, then we may expect most of the query terms to appear in an element for that element to be retrieved. In this case, one would expect that the term probability estimates are more reliable for long elements as they contain more terms compared to the short elements. Therefore, a shorter element needs to be more smoothed with the collection model compared to a longer element. This shows that a higher value of α_e is needed to capture exhaustivity in small elements. The Dirichlet smoothing (Equation 13) satisfies this requirement as the value of α_e depends on the length of the elements.

The above smoothing process is reasonable if we are not concerned with the specificity dimension. With respect to specificity, unseen terms are less of an issue for small elements compared to the above case. Therefore a smaller amount of smoothing (a lower value of α_e) is needed to capture specificity in small elements than the amount of smoothing required to capture exhaustivity. Due to this contradictory behaviour in the required amount of smoothing, Equation 13 in its current form cannot be used to capture both relevance dimensions if only length is taken into account. To accommodate for the specificity dimension, we set α_e, the amount of smoothing, to be proportional to the number of topic shifts in the element.

We, therefore, employ two parameter settings for Equation 13:

1. $\alpha_e = \frac{\mu}{\mu + |e|}$ implies that longer elements need less smoothing. This approach is the original Dirichlet smoothing. We refer to this approach as L.
2. $\alpha_e = \frac{\mu}{\mu + \frac{|e|}{|T|}}$, where $|T|$ is the number of topic shifts in e. In this case we differentiate between two elements with equal length and different numbers of topic shifts so that the presence of a query term in element with a lower number of topic shifts is rewarded. We refer to this approach as L/T.

The first version captures exhaustivity, whereas the second, where we replace length by the combination of length and topic shifts, captures both exhaustivity and specificity. Our hypothesis is that the latter is better for the focused access to XML documents.

7.2 Experimental Setting

For these experiments, we use the INEX collection, *Version 1.8*. The retrieval setting is the focused XML retrieval task (see Section 4.1.3), where the aim is to return a non-overlapping ranked list of the most exhaustive and specific elements on a relevant path.

The two approaches L and L/T will only rank elements, without, though, producing an overlap-free ranking. There are sophisticated ways to remove overlapping elements (e.g., [37]). In this work we restrict ourselves to a post-filtering on the retrieved ranked list by selecting the highest scored element from each of the paths, as our main interest here is to investigate how using topic shifts can help retrieval effectiveness. It is our hypothesis that the L/T approach will rank the more focused elements higher, and as such these will selected to be returned as answers to a given topic of request.

We experimented with a wide range of value for μ between [0, 20000]. To compare the two smoothing approaches, we select a best run (in terms of $MAep$) for each approach and then compare the behaviour of these best runs based on $MAnxCG$.

In the context of these experiments, we also investigate various settings for TextTiling's two parameters: K and W. We used different values for $K = \{3, 4, 5, 6\}$, where K is meant to approximate the average paragraph length in terms of the number of sentences [22], while at the same time maintaining the total window size ($W \times K$) as an approximate constant (in our case equal to 60, the median of the paragraph length in the INEX collection).

7.3 Experimental Results and Analysis

We first report on the results of the experiments for the different parameter settings of TextTiling. Figure 4 shows the impact of varying K with $W * K = 60$ on the $MAep$ values under the two quantization functions. The setting $K = 6$ produces the best $MAep$ under both the generalized and strict cases, which accords well with the TextTiling original setting for K. We then fix K at 6, and set W to its original value of 20 (the default setting of TextTiling), 32 (what we arrived at manually as described in Section 4.2) and 10 (the best setting from Figure 4). These are shown in Figure 5. In the generalized case, the setting of $W = 10$ works well, whereas $W = 32$ leads to best performance in the strict case. Therefore, we only present the results generated when we use the two settings of $W = 10, K = 6$ referred to as W10K6, and $W = 32, K = 6$, referred to as W32K6.

Table 12 shows a summary of these results. We first discuss the evaluation results obtained with $MAep$. Under the *generalised quantization function*, $MAep$ ranks the L/T approach with $W10K6$ above L. However the difference is not significant. To obtain a better understanding, we look at the performance for different values of the parameter μ. Figure 5 shows the $MAep$ values for μ ranging between [0, 20000]. We observe that the L/T approach with $W10K6$ leads to better performance than L regardless of the values of μ. This indicates that elements with equal length and smaller number of topic shifts require less smoothing. This is due to the fact that in the L/T approach, the presence of a query term in

an element with a lower number of topic shifts (a more specific element) is rewarded. This means that we are capturing specificity with the number of topic shifts.

Under the *strict quantization function*, results show L/T with both TextTiling settings is the most effective. These results support the argument that the Dirichlet smoothing in its standard formulation is not sufficient to satisfy the specificity dimension of relevance. These results also show that to retrieve highly specific and highly exhaustive elements, in the strict case, less smoothing is required for elements that contain fewer number of topic shifts than those that contain a higher number of topic shifts. Similar to the observed behaviour for the generalised quantization function, L/T shows better performance than L in most of the values of μ.

Overall, the L/T approach with $W10K6$, where the number of topic shifts combined with length affects the amount of smoothing, performs better than L when evaluated using the $MAep$ measure for both quantization functions.

Next, we discuss the results obtained using $MAnxCG$. Under the *generalised quantization function* and in the early ranks, there is not considerable difference between L/T with $W10K6$ and the baseline approach, L, whereas the setting $W32K6$ leads to a significant decrease in performance. In the *strict* case, however, where the aim is to retrieve highly specific and exhaustive elements, L/T leads to substantially improved performance at all early cut-off points with both TextTiling settings. In particular, for $W10K6$ there is a significant increase of 50.07% at rank 1.

To conclude, the number of topic shifts is a useful evidence in focused XML retrieval, as it seems to properly capture the specificity dimension of relevance. We used a language modeling framework to provide a flexible means to incorporate the number of topic shifts in the scoring function. With this approach, we showed that combining topic shifts with element length - an important parameter in XML retrieval - provided better retrieval performance for the focused retrieval task, than using element length alone.

Our results showed that for our proposed L/T approach, $W10K6$ works better for all quantization functions for the focused task. This does not mean that better performance cannot be obtained with other settings, as we did not experiment with all combinations for the W and K values. We addressed our heuristic approach for parameter setting of TextTiling in Section 7.2. We are going to examine whether this way of parameter setting works well for other collections. More importantly, using a different segmentation algorithm may lead to better performance. This will be the focus of our future work.

Table 12. Focused retrieval task using the INEX 2005 data: *MAep* and *MAnxCG* at different cut-off points considering *L* as baseline
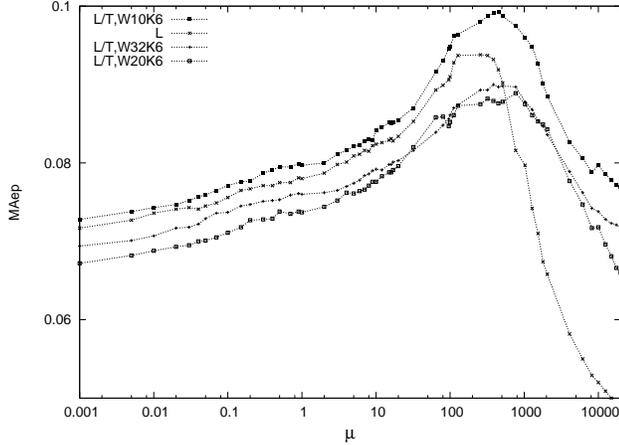
|	General				Strict				
	L	*L/T*	chg	*L/T*	chg	*L*	*L/T*	chg	*L/T*	chg
	$\mu = 256$	$\mu = 448$		$\mu = 384$		$\mu = 256$	$\mu = 1280$		$\mu = 256$	
		W10K6		W32K6			W10K6		W32K6	
MAnxCG@1	0.2954	0.3773	(+27.73%)	0.2119	(-28.27%)	0.0769	0.1154	(+50.07%)++	0.0769	(0%)
MAnxCG@2	0.3039	0.3194	(+5.1%)	0.2096	(-31.03%)-	0.0673	0.1058	(+57.21%)	0.0769	(+14.26%)
MAnxCG@3	0.2961	0.3032	(+2.4%)	0.2206	(-25.50%)-	0.0662	0.1004	(+51.66%)	0.0812	(+22.66%)
MAnxCG@10	0.2772	0.2643	(-4.65%)	0.2423	(-12.59%)-	0.0717	0.0716	(-0.14%)	0.0792	(+10.46%)
MAnxCG@25	0.2572	0.2561	(-0.43%)	0.2396	(-6.84%)	0.0940	0.0976	(+3.83%)	0.0998	(+6.17%)
MAnxCG@50	0.2432	0.2517	(+3.5%)	0.2374	(-2.38%)	0.1135	0.1345	(+18.5%)	0.1249	(+10.04%)
MAep	0.0938	0.0992	(+5.8%)	0.0900	(-4.05%)	0.0290	0.0308	(+6.21%)	0.0331	(+14.14%)
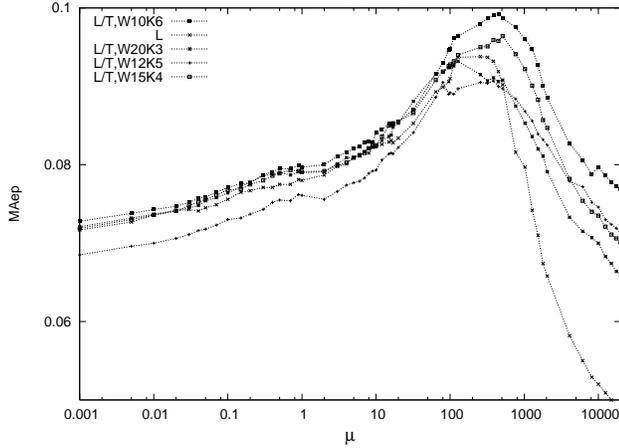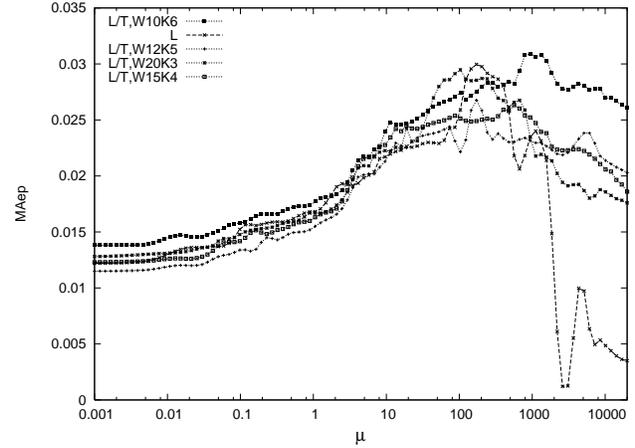
(a) general   (b) strict

Fig. 4. Impact of K on *MAep* with W*K = 60 (INEX 2005 CO topics).
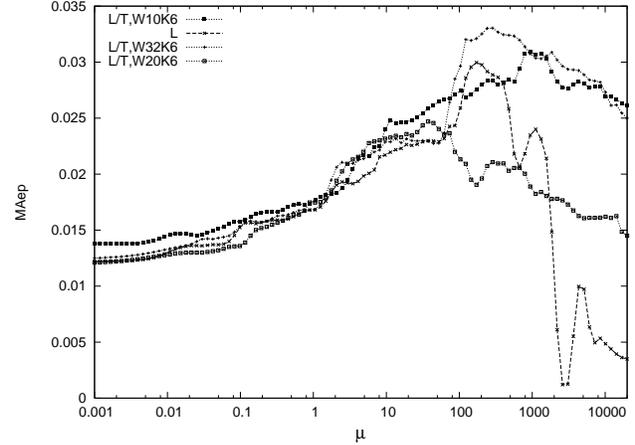
(a) general   (b) strict

Fig. 5. Impact of W on *MAep* with K = 6 (INEX 2005 CO topics).

8 Conclusions

In content-oriented XML retrieval, elements of any granularity are potential answers to a query. This means that XML retrieval systems need not only score elements with respect to their relevance to a query, but also determine the appropriate level of element granularity to return to users.

INEX defines a relevant element to be at the right level of granularity if it is exhaustive to the user request – i.e. it discusses fully the topic requested in the user's query – *and* it is specific to that user's request – i.e. it does not discuss other topics. The exhaustivity and specificity dimensions are both expressed in terms of the "quantity" of topics discussed within each element. Consequently, we hypothesize that the relevance of an element and its appropriate level of granularity could be

reflected by the number of topic shifts within that element.

We therefore defined a new measure, the number of topic shifts in an XML element, to formally quantify the number of topics discussed in an element. This new measure is collection-independent, as it only requires the specification of the retrieval unit at the lowest level of granularity allowed for a given collection (which will often be paragraph elements in content-oriented XML retrieval).

Using this new measure, we first studied the characteristics of XML elements as reflected by their number of topic shifts. We then compared topic shifts to length by incorporating each of them as a features in a retrieval setting in order to compare their effect on XML retrieval effectiveness. Finally, we proposed a topic shifts-based smoothing process within the language modeling framework and investigate whether using topic shifts is effective for the focused access to XML documents. Our three research objectives were investigated by carrying out extensive experiments on the INEX testbed.

Regarding our examination of the characteristics of XML elements, our main finding was that the number of topic shifts can be used to capture specificity. Therefore, we used the number of topic shifts as evidence for capturing specificity to provide a focused access to XML collections. We showed that using topic shifts combined with length provides a better approach for the focused access to XML documents.

Regarding our comparison between length and the number of topic shifts, our results indicate that although the latter is not unrelated to the former (larger elements can discuss indeed more topics), topic shifts and length are distinct notions and therefore constitute different sources of evidence. Our analysis further indicates that, when estimating the relevance of an element in the context of the thorough retrieval task, biasing retrieval towards retrieving elements with high number of topic shifts is beneficial for enhancing the precision at the early ranks for retrieving highly exhaustive and highly specific elements.

For our future work, our first aim is to incorporate our findings on other test collections. Initial investigation on the effects of using topic shifts on the Wikipedia XML collection, which is the collection used in INEX 2006[16], can be found in [3]. On another direction, we will examine whether other segmentation algorithms or other settings of the TextTiling algorithm are better suited for XML documents, and whether we can eventually obtain other, more effective means to calculate the number of topic shifts of XML elements.

References

1. P. Arvola, M. Junkkari, and J. Kekäläinen. Generalized contextualization method for XML information retrieval. *Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM)*, pages 20–27, 2005.
2. E. Ashoori and M. Lalmas. Using topic shifts for focussed access to XML repositories. *Advances in Information Retrieval: Proceedings 29th European Conference on IR Research (ECIR)*, Volume 4425 of LNCS, Springer, pages 444–455, 2007.
3. E. Ashoori and M. Lalmas. Using topic shifts in XML retrieval at INEX 2006. *Comparative Evaluation of XML Information Retrieval Systems: Fifth Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2006)*, Volume 4518 of LNCS, Springer Verlag, 2007.
4. R. A. Baeza-Yates, N. Fuhr, and Y. S. Maarek. SIGIR XML and Information Retrieval workshop *SIGIR Forum*, 36(2):53–57, 2002.
5. R. A. Baeza-Yates, Y. S. Maarek, T. Rölleke, and A. P. de Vries. SIGIR joint XML and Information Retrieval workshop and Integration of IR and DB workshop *SIGIR Forum*, 38(2):24–30, 2004.
6. H. M. Blanken, T. Grabs, H.-J. Schek, R. Schenkel, and G. Weikum, editors. *Intelligent Search on XML Data, Applications, Languages, Models, Implementations, and Benchmarks*, Volume 2818 of LNCS, Springer, 2003.
7. J. P. Callan. Passage-level evidence in document retrieval. *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 302–310, 1994.
8. C. Caracciolo and M. de Rijke. Generating and retrieving text segments for focused access to scientific documents. *Advances in Information Retrieval: Proceedings 28th European Conference on IR Research (ECIR)*, Volume 3936 of LNCS, Springer, pages 350–361, 2006.
9. D. Carmel, Y. S. Maarek, and A. Soffer. SIGIR XML and Information Retrieval. *SIGIR Forum*, 34(1):31–36, 2000.
10. Y. Chiaramella, P. Mulhem, and F. Fourel. A model for multimedia information retrieval. Technical report, University of Glasgow, 1996. FERMI.
11. W. B. Croft and J. Lafferty. *Language Modeling for Information Retrieval*. Kluwer Academic Publishers, 2003.
12. B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993.
13. N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors. *Proceedings of the First Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Schloss Dagstuhl, Germany, December 9-11, 2002*.
14. N. Fuhr and M. Lalmas. Report on the INEX 2003 workshop, Schloss Dagstuhl, 15-17 December 2003. *SIGIR Forum*, 38(1):42–47, June 2004.

---

[16] http://inex.is.informatik.uni-duisburg.de/2006/index.html

15. N. Fuhr, M. Lalmas, and S. Malik, editors. *INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop. Dagstuhl, Germany, December 15–17, 2003.*

16. N. Fuhr, M. Lalmas, S. Malik, and G. Kazai, editors. *Advances in XML Information Retrieval and Evaluation: Fourth Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2005)*, Volume 3977 of LNCS, Springer, 2006.

17. N. Fuhr, M. Lalmas, S. Malik, and Z. Szlávik, editors. *Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl Castle, Germany, December 6-8, 2004*, Volume 3493 of LNCS, Springer, 2005.

18. S. Geva. GPX - gardens point XML ir at inex 2005. In Fuhr et al. [16], pages 240–253.

19. N. Gövert, N. Fuhr, M. Abolhassani, and K. Großjohann. Content-oriented XML retrieval with HyREX. In Fuhr et al. [13], pages 26–32.

20. M. Halliday and R. Hasan. *Cohesion in English.* Longman, London, 1976.

21. K. Hatano, H. Kinutani, T. Amagasa, Y. Mori, M. Yoshikawa, and S. Uemura. Analyzing the properties of XML fragments decomposed from the INEX document collection. In Fuhr et al. [17], pages 168–182.

22. M. A. Hearst. Multi-paragraph segmentation of expository text. *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 9–16, 1994.

23. M. A. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 59–68, 1993.

24. D. Hiemstra. A database approach to content-based XML retrieval. In Fuhr et al. [13], pages 111–118.

25. D. Hiemstra. *Using Language Models for Information Retrieval.* PhD thesis, University of Twente, 2001.

26. J. Kamps, M. de Rijke, and B. Sigurbjörnsson. The importance of length normalization for XML retrieval. *Information Retrieval*, 8(4):631–654, 2005.

27. M. Kaszkiel and J. Zobel. Effective ranking with arbitrary passages. *J. Am. Soc. Inf. Sci. Technol.*, 52(4):344–364, 2001.

28. G. Kazai and M. Lalmas. Extended cumulated gain measures for the evaluation of content-oriented XML retrieval. *ACM Trans. Inf. Syst.*, 24(4):503–542, 2006.

29. G. Kazai and M. Lalmas. INEX 2005 evaluation metrics. In Fuhr et al. [16], pages 16–29.

30. J. Kekäläinen, M. Junkkari, P. Arvola, and T. Aalto. TRIX 2004: Struggling with the overlap. In Fuhr et al. [17], pages 127–139.

31. M. Lalmas and T. Tombros. INEX 2002 - 2006: Understanding XML Retrieval Evaluation *DELOS Conference on Digital Libraries*, 13-14 February 2007, Tirrenia, Pisa (Italy).

32. M. Lalmas and G. Kazai. Report on the ad-hoc track of the INEX 2005 workshop. *ACM SIGIR Forum*, 40(1):49–57, June 2006.

33. J. List and A. P. Vries. CWI at INEX 2002. In Fuhr et al. [13], pages 133–140.

34. S. Malik, G. Kazai, M. Lalmas, , and N. Fuhr. Overview of inex 2005. In Fuhr et al. [16], pages 1–15.

35. Y. Mass and M. Mandelbrod. Retrieving the most relevant XML components. In Fuhr et al. [15], pages 53–58.

36. Y. Mass and M. Mandelbrod. Using the INEX environment as a test bed for various user models for XML retrieval. In Fuhr et al. [16], pages 187–195.

37. V. Mihajlovic, G. Ramírez, T. Westerveld, D. Hiemstra, H. E. Blok, and A. P. de Vries. TIJAH Scratches INEX 2005: Vague Element Selection, Image Search, Overlap, and Relevance Feedback. In Fuhr et al. [16], pages 72–87.

38. V. Mittal, M. Kantrowitz, J. Goldstein, and J. Carbonell. Selecting text spans for document summaries: heuristics and metrics. *Proceedings of the 16th national conference on Artificial intelligence and the 11th Innovative applications of artificial intelligence conference*, pages 467–473, 1999.

39. C. Monz and B. J. Dorr. Iterative translation disambiguation for cross-language information retrieval. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 520–527, 2005.

40. J. Morris and G. Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48, 1991.

41. P. Ogilvie and J. Callan. Hierarchical language models for XML component retrieval. In Fuhr et al. [17] ,pages 224–237.

42. I. Papadakis and V. Chrissikopoulos. A digital library framework based on XML. In *Proceedings of the 3rd International Conference of Asian Digital Library (ICADL)*, pages 81–88, 2000.

43. B. Piwowarski and M. Lalmas. Providing consistent and exhaustive relevance assessments for XML retrieval evaluation. *Proceedings of the 13th ACM international conference on Information and knowledge management (CIKM)*, pages 361–370, 2004.

44. J. M. Ponte and W. Bruce Croft. Text Segmentation by Topic. *Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries*, pages 113–125, 1997.

45. G. Ramirez, T. Westerveld, and A. P. de Vries. Using structural relationships for focused XML retrieval. In *Proceedings of the Seventh International Conference on Flexible Query Answering Systems (FQAS)*, Volume 4027 of LNCS, Springer, pages 147–158, 2006.

46. J. C. Reynar. *Topic segmentation: Algorithms and applications.* PhD thesis, Computer and Information Science, University of Pennsylvania, 1998.

47. G. Salton and J. Allan and C. Buckley. Approaches to passage retrieval in full text information systems. *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 49–58, 1993.

48. G. Salton and A. Singhal and C. Buckley and M. Mitra. Automatic text decomposition using text segments and text themes. *Proceedings of the the seventh ACM conference on Hypertext*, pages 53–65, 1996.

49. K. Sauvagnat, L. Hlaoua, and M. Boughanem. XFIRM at INEX 2005: ad-hoc and relevance feedback tracks. In Fuhr et al. [16], pages 88–103.

50. J. Savoy. Statistical inference in retrieval effectiveness evaluation. *Inf. Process. Manage.*, 33(4):495–512, 1997.

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

52. B. Sigurbjörnsson, J. Kamps, and M. de Rijke. The effect of structured queries and selective indexing on XML retrieval. In Fuhr et al. [16], pages 104–118.

53. M. Stairmand. *A Computational Analysis of lexical Cohesion with Applications in Information Retrieval*. PhD thesis, University of Manchester, 1996.

54. A. Trotman. Wanted: Element retrieval users. *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology*, Glasgow, July 2005.

55. R. Wilkinson. Effective retrieval of structured documents. *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 311–317, 1994.

56. C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 334–342, 2001.