Evaluating XML Retrieval Effectiveness at INEX

Mounia Lalmas and Anastasios Tombros
Queen Mary, University of London,
Mile End Road, London, UK
{mounia,tassos}@dcs.qmul.ac.uk

Abstract

The INitiative for the Evaluation of XML retrieval (INEX) was set up in 2002 to establish an infrastructure and provide means, in the form of large test collections and appropriate scoring methods, for evaluating the effectiveness of content-oriented XML retrieval systems. This report provides an overview of the evaluation methodology developed in INEX from 2002 to 2006.

1 Introduction

XML retrieval systems have been and are being developed to implement content-oriented retrieval approaches to XML documents. A common feature of these systems, which makes them different to traditional document retrieval systems, is that, instead of retrieving whole documents, XML retrieval systems aim at retrieving document components, i.e. XML elements of varying granularity that fulfill the user's query. As the number of XML retrieval systems increases, so is the need to evaluate their benefit to the users.

The predominant approach to evaluate a system retrieval effectiveness is with the use of test collections and effectiveness scoring methods. The INitiative for the Evaluation of XML retrieval (INEX)[1] [3, 4, 6, 5, 7] was set up in 2002 to establish an infrastructure and provide means, in the form of large test collections and appropriate scoring methods, for evaluating content-oriented XML retrieval systems. This report provides an overview of the evaluation methodology developed in INEX from 2002 to 2006. Section 2 describes the INEX test collections and Section 3 describes the scoring methods used to measure effectiveness. We conclude in Section 4. We only report the evaluation of the ad hoc track[2].

2 The INEX test collections

XML documents organize their content into small, nested structural elements. Each of these elements in the document's hierarchy, along with the document itself (the root of the hierarchy), represent a retrievable unit. With the use of XML query languages, users of an XML IR system can express their information need as a combination of content and structural conditions. Consequently, the relevance

[1] http://inex.is.informatik.uni-duisburg.de/

[2] Some of the INEX tracks are described in separate reports of this forum.

assessments for an XML collection must also consider the structural nature of the documents and provide assessments at different levels of the document hierarchy. In addition, a multitude of retrieval tasks can be specified, depending on how the relationships between elements are exploited when deciding what answers to return for given queries.

2.1 Document Collections

Up to 2004, the collection consisted of the full-text of 12,107 articles, marked-up in XML, from 12 magazines and 6 transactions of the IEEE Computer Society's publications, covering the period of 1995-2002, and totalling 494 MB in size, and 8 million in number of elements. The collection contained scientific articles of varying length. On average, an article contains 1,532 XML nodes, where the average depth of the node is 6.9. In 2005, the collection was extended with further publications from the IEEE Computer Society. A total of 4,712 new articles from the period of 2002-2004 were added, giving a total of 16,819 articles, and totalling 764 MB in size and 11 million in number of elements.

INEX 2006 uses a different document collection, made from English documents from Wikipedia[3] [2]. The collection consists of the full-texts, marked-up in XML, of 659,388 articles of the Wikipedia project, covering a hierarchy of 113,483 categories[4], and totaling more than 60 GB (4.6 GB without images) and 30 million in number of elements. The collection has a structure similar to the IEEE collection, but has a richer set of tags (1,241 unique tags compared to 176 in the IEEE collection), and include a large number of links between documents (represented as XLinks). On average, an article contains 161 XML nodes, where the average depth of an element is 4.8.

2.2 Topics

Querying XML documents can be with respect to content and structure. For example, users that are able to exploit the structural nature of the data can restrict their search to specific structural elements within an XML collection. Taking this into account, INEX identified the following two types of topics:

- Content-only (CO) topics are requests that do not include reference to the document structure. They are, in a sense, the traditional topics used in information retrieval test collections. However, the results to such topics are elements of various complexity, e.g. at different levels of the XML documents' structure.

- Content-and-structure (CAS) topics are requests that contain conditions referring both to content and structure of a document. These conditions may refer to the content of specific elements (e.g. the elements to be returned must contain a section about a particular topic), or may specify the type of the requested answer elements (e.g. sections should be retrieved).

As in TREC, an INEX topic consists of the standard title, description and narrative fields. For CO topics, the title is a sequence of terms. For CAS topics, the title is expressed using the NEXI query language, which is a variant of XPATH defined for content-oriented XML retrieval evaluation [30][5].

[3] http://en.wikipedia.org

[4] Articles in Wikipedia are organised into categories.

[5] The topic title format used in INEX 2002 was not based on a variant of XPATH, and as such led to often ambiguous queries, with respect to constraints imposed on the structure. Using the NEXI title topic format, not only removed ambiguity, but was also more in line with current development of XML query languages.

```
<inex_topic topic_id="76" query_type="CAS">
<title>
  //article[(./fm//yr = '2000' OR ./fm//yr = '1999') AND about(.,
  '"intelligent transportation system"')]//sec[about(.,'automation
  +vehicle')]
</title>
<description>
  Automated vehicle applications in articles from 1999 or
  2000 about intelligent transportation systems.
</description>
<narrative>
  To be relevant, the target component must be from an
  article on intelligent transportation systems published in 1999 or
  2000 and must include a section which discusses automated vehicle
  applications, proposed or implemented, in an intelligent
  transportation system.
</narrative>
</inex_topic>
```

Figure 1: A CAS topic from the INEX 2003 test collection

Both CO and CAS titles are made of terms, i.e. words or a phrases, where the latter are encapsulated in double quotes. Furthermore the terms can have either the prefix + or -, where + is used to emphasize an important concept, and - is used to denote an unwanted concept. An example of a CAS topic is given in Figure 1. The `about` clause refers to a content criterion and is different to the `contain` criterion of the XPath query language. An element can be `about` "intelligent transportation system" without actually containing any of the three words "intelligent", "transportation" and "system".

In 2005, in an effort to investigate the usefulness of using structural constraints in content-oriented XML retrieval, variants of the CO and CAS topics were developed. CO topics were extended into so-called Content-Only + Structure (CO+S) topics. The aim was to enable the performance comparison of an XML system across two retrieval scenarios on the same topic, one when structural constraints are taken into account (+S) and the other when these are ignored (CO) [29]. The CO+S topics included an optional field called CAS title (<castitle>), which was a representation of the same information need contained in the <title> field of a CO topic but including additional knowledge in the form of structural constraints. CAS titles were expressed in the query language of NEXI. An example of a CO+S topic is given in Figure 2.

Regarding CAS topics, the actual definition of CAS topics has not changed over the years. CAS topics are topic statements that contain explicit references to the XML structure, and explicitly specify the contexts of the user's interest (e.g. target elements) and/or the contexts of certain search concepts (e.g. containment conditions). More precisely, a CAS query contains two kinds of structural constraints: where to look (i.e. the support elements), and what to return (i.e. the target elements). What has evolved over the years is how to interpret the structural constraints, since each structural constraint could be considered as a strict (must be matched exactly) or vague (do not need to be matched exactly) criterion. In 2002, the structural constraints of CAS topics were interpreted strictly. In 2003, both interpretations,

```
<inex_topic topic_id="231" query_type="CO+S">
 <title>markov chains in graph related algorithms</title>
 <castitle>//article//sec[about(.,+"markov chains" +algorithm +graphs)]
 </castitle>
 <description>Retrieve information about the use of markov chains in
    graph theory and in graphs-related algorithms.
 </description>
 <narrative>I have just finished my Msc. in mathematics, in the field
    of stochastic processes. My research was in a subject related to
    Markov chains. My aim is to find possible implementations of my
    knowledge in current research. I'm mainly interested in
    applications in graph theory, that is, algorithms related to graphs
    that use the theory of markov chains. I'm interested in at
    least a short specification of the nature of implementation (e.g.
    what is the exact theory used, and to which purpose), hence the
    relevant elements should be sections, pagargaphs or even abstracts
    of documents, but in any case, should be part of the content of the
    document (as opposed to, say, vt, or bib).
 </narrative>
</inex_topic>
```

Figure 2: A CO+S topic from the INEX 2005 test collection

strict and vague, was followed, whereas in 2004, only the latter was followed. In addition in that same year, structural constraints, were to be viewed as hints as to where to look for relevant information. Two lines of research resulted from this view in 2005.

The first one led to the definition of CO+S topics, as described earlier. An investigation, reported in [29], showed that although improvements in some cases were seen when adding structural constraints to the query, these were not significant. A closer look showed that this was because the structural constraints did not seem to correspond to actual hints; instead they appear to be a function of the document collection rather than the query. At this stage, it is not clear if this is true of only the given collection or all collections. Further investigation is required on the INEX 2006 data set.

The second line was to look whether the two interpretations of the structural constraints mattered when evaluating the performance of XML systems. In a CAS topic, depending on how the target elements and/or the containment conditions are treated, four interpretations are possible: when the structural constraints of both target and support elements are vague, when the structural constraints of the target element is vague and that of support elements is strict, etc. An investigation reported in [28] suggests that, in terms of comparing retrieval effectiveness, there are two separate interpretations of CAS that matter, one in which the target element is interpreted strictly and the other in which it is interpreted vaguely. The interpretation of the containment conditions does not appear to be important. These results have implication in how to assess the relevance since it does not seem to matter, for comparing retrieval effectiveness, whether the structural constraint in the containment condition should be assessed strictly or vaguely, which simplifies the assessment process greatly.

In INEX, the topics are created from the participating groups. Each year, each participating group is asked to submit up to a given number of candidate topics, following detailed guidelines regarding

year	number of candidate topics	number of selected topics	number of assessed topics
2002	143	60	54
2003	120	66	62
2004	191	71	60
2005	139	87	63
2006	203	125	114
Total	796	409	350

Table 1: Statistics on the INEX topics

the creation of meaningful topics for evaluating XML retrieval effectiveness. The topic creation process consists of four main steps: initial topic statement creation, collection exploration, topic refinement, and topic selection. The first three steps were performed by the participants themselves while the selection of topics was decided by the INEX organisers. Table 2.2 gives statistics about the topics that were collected in each round of INEX. We also show the number of topics assessed to date. The yearly INEX proceedings contain more statistics on the topics [3, 4, 6, 5, 7], and e.g. [11] provides some analysis of the type of structural constraints found in the CAS topics.

2.3 Relevance

Since retrieved elements can be at any level of granularity, an element and one of its child elements can both be relevant to a given query, but the child element may be more focussed on the topic of the query than its parent element, which may contain additional irrelevant content. In this case, the child element is a better element to retrieve than its parent element, because not only it is relevant to the query, but it is also more specific to the query. To accommodate the specificity aspect, INEX defined in 2002 relevance along two dimensions:

- Topical relevance, which reflects the extent to which the information contained in an element satisfies the information need, i.e. measures the *exhaustivity* of the topic within an element.
- Component coverage, which reflects the extent to which an element is focussed on the information need, and not on other, irrelevant topics, i.e. measures the *specificity* of an element with regards to the topic.

A multiple degree relevance scale was necessary to allow the explicit representation of how exhaustively a topic is discussed within an element with respect to its child elements. For example, a section containing two paragraphs may be regarded more relevant than either of its paragraphs by themselves. Binary values of relevance cannot reflect this difference. INEX therefore adopted a four-point relevance scale based on [18]:

- Irrelevant (0): The element does not contain any information about the topic of request.
- Marginally relevant (1): The element mentions the topic of request, but only in passing.
- Fairly relevant (2): The element discusses the topic of request, but not exhaustively.
- Highly relevant (3): The element discusses the topic of request exhaustively.

As for topical relevance, a multiple degree scale was also necessary for the component coverage dimension. This is to allow to reward retrieval systems that are able to retrieve the appropriate ("exact") sized elements. For example, a retrieval system that is able to locate the only relevant section in a book is more effective than one that returns a whole chapter. A four-point relevance scale for component coverage was adopted:

- No coverage (N): The topic, or an aspect of the topic, is not a theme of the element.
- Too large (L): The topic, or an aspect of the topic, is only a minor theme of the element.
- Too small (S): The topic, or an aspect of the topic, is the main or only theme of the element, but the component is too small to act as a meaningful unit of information.
- Exact coverage (E): The topic, or an aspect of the topic, is the main or only theme of the element, and the element acts as a meaningful unit of information.

Based on the combination of the topical relevance and component coverage, it becomes possible to identify those relevant elements, which are both exhaustive and specific to the topic of request and hence represent the most appropriate unit to return to the user. In the evaluation we can then reward systems that are able to retrieve these elements.

In a study of the collected assessments for 2002, the use of "too small" led to some misinterpretations while assessing the coverage of an element [17]. The problem was that, for CAS topics, the "too small" and "too large" coverage categories were incorrectly interpreted as the relation between the actual size of the result element and the size of the target element instead of the relation between the relevant and irrelevant contents of the result element. To address this issue, INEX 2003 renamed the two dimensions to exhaustivity and specificity:

- Exhaustivity, which measures how exhaustively an element discusses the topic of the user's request.
- Specificity, which measures the extent to which an element focuses on the topic of request (and not on other, irrelevant topics).

The scale for the exhaustivity dimension, which replaces the topical relevance dimension, was redefined by simply replacing the word relevant to exhaustive. To avoid direct association with element size, the specificity dimension, which replaces the component coverage, adopted an ordinal scale similar to that defined for the exhaustivity dimension:

- Not specific (0): the topic of request is not a theme discussed in the element.
- Marginally specific (1): the topic of request is a minor theme discussed in the element.
- Fairly specific (2): the topic of request is a major theme discussed in the element.
- Highly specific (3): the topic of request is the only theme discussed in the element.

Although there have been arguments against the separation into two relevance dimensions, this was believed to provide a more stable measure of relevance than if assessors were asked to rate elements on a single scale. One reason for this is that assessors are likely to place varying emphasis on these two dimensions when assigning a single relevance value. For example, one assessor might tend to rate highly specific elements as more relevant, while another might be more tolerant of lower specificity and prefer high exhaustivity.

However, obtaining relevance assessments is a very tedious and costly task [24]. An observation made in [1] was that the assessment process could be simplified if first, relevant passages of text were identified by highlighting, and then the elements within these passages were assessed. As a consequence, at INEX 2005, the assessment method was changed, leading to the redefinition of the scales for specificity. The procedure was a two-phase process. In the first phase, assessors highlighted text fragments containing only relevant information. The specificity dimension was then automatically measured on a continuous scale [0,1], by calculating the ratio of the relevant content of an XML element: a completely highlighted element had a specificity value of 1, whereas a non-highlighted element had a specificity value of 0. For all other elements, the specificity value was defined as the ratio (in characters) of the highlighted text (i.e. relevant information) to the element size. For example, an element with specificity of 0.72 has 72% of its content highlighted.

In the second phase, for all elements within highlighted passages (and parent elements of those), assessors were asked to assess their exhaustivity. Following the outcomes of extensive statistical analysis performed on the INEX 2004 results [21] - which showed that in terms of comparing retrieval effectiveness, the same conclusions could be drawn using a smaller number of grades for the exhaustivity dimension[6] - INEX 2005 adopted the following $3 + 1$ exhaustivity values:

- Highly exhaustive (2): the element discussed most or all aspects of the query.
- Partly exhaustive (1): the element discussed only few aspects of the query.
- Not exhaustive (0): the element did not discuss the query.
- Too Small (?): the element contains relevant material but is too small to be relevant on it own.

The category of "too small" was introduced to allow assessors to label elements, which although contained relevant information, were too small to sensibly reason about their level of exhaustivity. In 2002 the "too small" category was with respect to the specificity aspect of relevance, whereas in 2005, it is a degree of exhaustivity, and was deemed necessary to free assessors from the burden of having to assess very small text fragments whose level of exhaustivity could not be sensibly decided.

As the ultimate aim of an evaluation is to be able to state that a system performs consistently better than another system, a continuous discussion in INEX was whether such a sophisticated definition of relevance, and in particular the exhaustivity dimension, was needed. An extensive statistical analysis was performed on the INEX 2005 results [21], which showed that in terms of comparing retrieval performance, not using the exhaustivity dimension led to similar results in terms of comparing retrieval effectiveness. As a result, INEX 2006 dropped the exhaustivity dimension, and relevance was defined only along the specificity dimension.

Each year, INEX participants provided the relevance assessments. This was done through the use of an on-line relevance assessment tool, called X-RAI [24]. Figure 3 shows a screenshot of the interface used at INEX 2006, where assessors had to highlight relevant text fragments from articles identified as candidates after pooling was performed [24]. These highlighted passages were then automatically converted into element specificity scores.

[6]The same observation was reached for the specificity dimension, but as the assessment procedure was changed in INEX 2005, the new highlighting process allowed for a continuous scale of specificity to be calculated automatically.

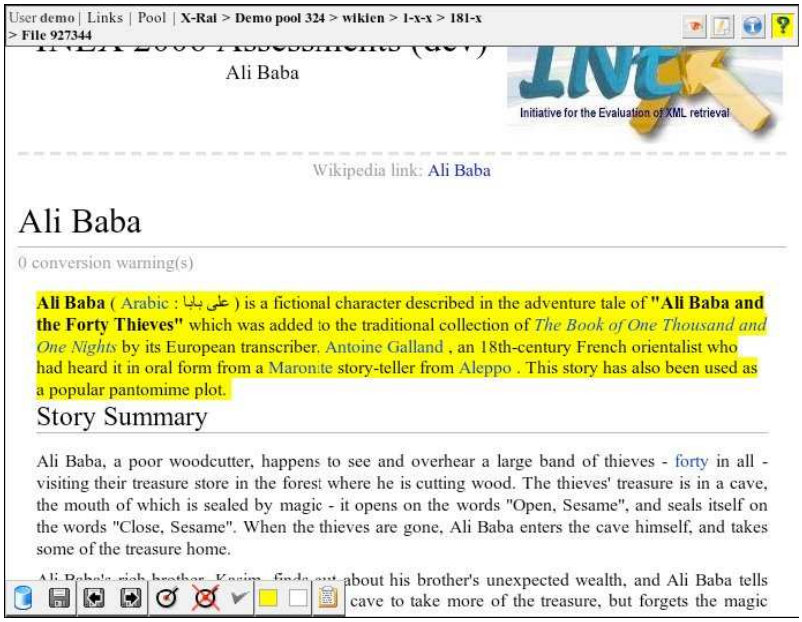

Figure 3: INEX 2006 assessment interface

2.4 Retrieval tasks

The main INEX activity is the ad-hoc retrieval task, where the collection consists of XML documents, composed of different granularity of nested XML elements, each of which represents a possible unit of retrieval. A major departure from traditional IR is 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. In INEX, the specificity dimension makes it possible to differentiate, for example, between the only relevant section in an encyclopaedia from the whole encyclopaedia. Although both may be relevant to a given user query, the former is likely to trigger higher user satisfaction as it will be more specific to the query than the encyclopaedia.

Up to 2004, ad-hoc retrieval was defined as the *general* task of returning, instead of whole documents, those XML elements that are most specific and exhaustive to the user's query. In other words, systems should return components that contain as much relevant information and as little irrelevant information as possible. Within this general task, several sub-tasks were defined, where the main difference was the treatment of the structural constraints.

The *CO sub-task* makes use of the CO topics, where an effective system is one that retrieves the most specific and exhaustive elements and only those, which are relevant to the topic of request. The *CAS sub-task* makes use of CAS topics, where an effective system is one that retrieves the most specific document components, which are relevant to the topic of request and match, either strictly or vaguely, the structural constraints specified in the query. In 2002, a strict interpretation of the CAS structural constraints was adopted, whereas in 2003, both, a strict and a vague interpretations was followed, leading to the *SCAS sub-task* (strict content-and-structure), defined as for the INEX 2002 CAS sub-task, and the *VCAS sub-task* (vague content-and-structure). In that last sub-task, the goal of an XML retrieval system was to return relevant elements that may not exactly conform to the structural conditions expressed within the user's query, but where the path specifications should be considered hints as to where to look. In 2004,

the two sub-tasks investigated were the CO sub-task, and the VCAS sub-task. The SCAS sub-task was felt to be an unrealistic task because specifying an information need is not an easy task, in particular for semi-structured data with a wide variety of tag names.

However, within this general task, the actual relationship between retrieved elements was not considered, and many systems returned overlapping elements (e.g. nested elements). Indeed, the top 10 ranked systems for the CO sub-task in INEX 2004 contained between 70% to 80% overlapping elements. What most systems did was to estimate the relevance of XML elements, which is different to identifying the most relevant elements. This had very strong implications with respect to measuring effectiveness, where approaches that attempted to implement a more selective approach (e.g., between two nested relevant elements, returning the one most specific to the query) performed poorly. As a result, the *focussed sub-task* was defined in 2005, intended for approaches concerned with the so-called focussed retrieval of XML elements, i.e. aiming at indeed targeting the appropriate level of granularity of relevant content that should be returned to the user for a given topic. The aim was for systems to find the most exhaustive and specific element on a path within a given document containing relevant information and return to the user only this most appropriate unit of retrieval. In the 2006, this was translated to the most specific element on a path within a given document. Returning overlapping elements was not permitted. The INEX ad-hoc general task, as carried out by most systems up to 2004, was renamed in 2005 as the *thorough sub-task*.

Within all the above sub-tasks, the output of XML retrieval systems was assumed to be a ranked list of XML elements, ordered by their presumed relevance to the query, whether overlapping elements were allowed or not. However, user studies [26] suggested that users were expecting to be returned elements grouped per document, and to have access to the overall context of an element. The *fetch & browse task* was introduced in 2005 for this reason. The aim was to first identify relevant documents (the fetching phase), and then to identify the most exhaustive and specific elements within the fetched documents (the browsing phase). In the fetching phase, documents had to be ranked according to how exhaustive and specific they were. In the browsing phase, ranking had to be done according to how exhaustive and specific the relevant elements in the document were, compared to other elements in the same document. In 2005, no explicit constraints were given regarding whether returning overlapping elements within a document was allowed. The rationale was that there should be a combination of how many documents to return, and within each document, how many relevant elements to return.

In 2006, the same task, renamed the *relevant in context sub-task*, required systems to return for each article an unranked set of non-overlapping elements, covering the relevant material in the document. In addition, a new task was introduced in 2006, the *best in context sub-task*, where the aim was to find the best-entry-point, here a single element, for starting to read articles with relevant information. This sub-task can be viewed as the extreme case of the fetch & browse approach, where only one element is returned per article.

3 Effectiveness measures

We only report the measures used to measure the effectiveness of the focused and thorough retrieval tasks[7]. Unlike traditional IR, users in XML retrieval have access to other, structurally related elements from returned result elements. They may hence locate additional relevant information by browsing or

[7] An report on the measures used to evaluate the relevant in context and best in context tasks can be found in [19].

scrolling (depending of the interface). This motivates the need to consider so-called near-misses, which are elements from where users can access relevant content, within the evaluation – similar investigations were also conducted at the web track of TREC-8 [9]. The alternative, to ignore near-misses, would lead to a strict evaluation scenario, especially when dealing with fine-grained XML documents.

The effectiveness of most ad-hoc retrieval tasks is measured by the established and widely used precision and recall measures, or their variants. From 2002 to 2004, INEX used the inex_eval measure, which applies the measure of precall [25] to XML elements. As for precision and recall, inex_eval is based on a counting mechanism, i.e. based on number of retrieved and relevant elements. As a consequence, if we consider near-misses when evaluating retrieval effectiveness, then systems that return overlapping elements (e.g. both a paragraph and its enclosing section) will be evaluated as more effective than those that do not return overlapping elements (e.g. either the paragraph or its enclosing section). If both the paragraph and its enclosing section are relevant, then this family of effectiveness measures will count both these nested elements as separate relevant components that increase the count of relevant and retrieved elements. Therefore, despite not retrieving entirely new relevant information, systems that favour the retrieval of overlapping components would receive higher effectiveness scores [16].

The first step to address this problem - as discussed in Section 2.4 - was to define the two retrieval tasks, thorough and focussed, to distinguish between systems that were interested in estimating the relevance of elements given a topic of request, and those that aimed at providing so-called focused access to XML content. Using the inex_eval measure to evaluate the thorough sub-task is then appropriate.

With respect to the focussed task, as we still want to appropriately reward the retrieval of near-misses, we need to differentiate between those elements that should be retrieved, i.e. the desired elements, and those elements that are structurally related to the desired elements, i.e. the near-misses. It was therefore necessary to separate between these two sets by marking a subset of the relevant elements in the recall-base as ideal answers, i.e. the so-called desired elements. We refer to this set as the ideal recall-base. However, using inex_eval on the ideal-recall base to evaluate the focussed task would mean that near-misses cannot be considered when evaluating retrieval performance.

As a result, INEX adopted in 2005 a new set of measures, called XCG, for both sub-tasks [14]. The XCG measures are an eXtension of the Cumulated Gain based measures [10]. These measures are not based on a counting mechanisms, but on cumulated gains associated with return results, which are appropriate to evaluate the focused task where near-misses are considered. For sake of consistency, the same family of measures were also adopted to evaluate the thorough task. Before describing how the thorough and focused tasks are evaluated using the XCG measures, we first describe quantisations, which deal with the multi-dimension definition of relevance in INEX.

3.1 Quantizations

Given that INEX employs two relevance dimensions (up to 2005), with multiple degree scales, it is necessary to combine these to reflect the worth of a retrieved element. The so-called quantization functions aim to do just that, by providing a relative ordering of the various combinations of exhaustivity (e) and specificity (s) values and a mapping of these to a single relevance scale in $[0, 1]$. Various quantization functions have been used over the years, as a means to model assumptions regarding the worth of retrieved elements. INEX 2003 used the quantisations defined as follows:

$$quant_{strict}(e,s) := \begin{cases} 1 & \text{if } e = 3 \text{ and } s = 3, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The strict function is used to evaluate XML retrieval methods with respect to their capability of retrieving highly exhaustive and highly specific components. This function models the scenario where only highly specific and highly exhaustive components are considered worthy.

$$quant_{gen}(e,s) := \begin{cases} 1 & \text{if } (e,s) = (3,3), \\ 0.75 & \text{if } (e,s) \in \{(2,3),(3,\{2,1\})\}, \\ 0.5 & \text{if } (e,s) \in \{(1,3),(2,\{2,1\})\}, \\ 0.25 & \text{if } (e,s) \in \{(1,2),(1,1)\}, \\ 0 & \text{if } (e,s) = (0,0). \end{cases} \quad (2)$$

$$quant_{sog}(e,s) := \begin{cases} 1 & \text{if } (e,s) = (3,3), \\ 0.9 & \text{if } (e,s) = (2,3), \\ 0.75 & \text{if } (e,s) \in \{(1,3),(3,2)\}, \\ 0.5 & \text{if } (e,s) = (2,2), \\ 0.25 & \text{if } (e,s) \in \{(1,2),(3,1)\}, \\ 0.1 & \text{if } (e,s) \in \{(2,1),(1,1)\}, \\ 0 & \text{if } (e,s) = (0,0). \end{cases} \quad (3)$$

The generalised ($quant_{gen}$) and the specificity-oriented generalised ($quant_{sog}$) functions reward retrieved elements according to their *degree* of relevance, thus also allowing to reward fairly and marginally relevant elements. The difference between $quant_{gen}$ and $quant_{sog}$ is that the former shows slight preference towards the exhaustivity dimension, assigning high scores to exhaustive, but not necessarily specific elements, whereas the latter assumes that more specific elementts are of greater value. $quant_{sog}$ was proposed in [16] with the argument that it better reflects that specificity plays a more dominant role than exhaustivity in XML retrieval. A statistical analysis on the INEX 2004 results reported in [21], however, shows that a very high agreement about which systems perform statistically significantly differently from each other between $quant_{gen}$ and $quant_{sog}$. Thus, although these two quantisation functions express different preferences, they behave very similarly when ranking systems.

Other quantisations have been used, for example, applying a strict quantisation with respect to the exhaustivity dimension and allow to consider different degrees of specificity, or inversely, applying the strict quantisation with respect to the exhaustivity dimension and allowing to consider different degrees of specificity[8]. A statistical analysis of the INEX 2004 results [21], however, shows that, although quantisation functions express different preferences, many of them behave similarly when ranking systems. As a consequence, one form of strict and one form of general quantizations were used in 2005, and were modified to adapt to the INEX 2005 scale:

$$quant_{strict5}(e,s) := \begin{cases} 1 & \text{if } \quad e = 2 \quad \text{and} \quad s = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

[8]More details can be found at http://homepages.cwi.nl/∼arjen/INEX/.

$$quant_{gen5}(e, s) := e \cdot s \tag{5}$$

The quantization $quant_{gen5}$ ignores elements assessed as "too small". To consider too small elements within the evaluation, $quant_{genLifted}$ was introduced, which adds $+1$ to lift all values of exhaustivity. The effect of this is that it allows the scoring of too small elements as near-misses.

$$quant_{genLifted}(e, s) := (e + 1) \cdot s \tag{6}$$

A statistical analysis of the INEX 2005 results [21] shows reasonably high agreement between which system pairs are identified as significantly different when using $quant_{gen5}$ and $quant_{genLifted}$. Thus, whether or not the "too small" elements are considered relevant did not make a large difference in the rankings of systems.

In INEX 2006, as the exhaustivity dimension was dropped, the quantization function simply maps an element to its specificity value. Only the results using the generalized function were reported. The motivation came from one the outcomes of the extensive analysis reported in [21], which indicates that using a strict quantization did not distinguish systems well when performing statistical significance tests.

Because of how relevance was assessed in INEX 2006, a high number of fully highlighted elements – the figure reported at the INEX workshop was 18% – (which will then obtain a specificity score (quantizised value) of 1) were of link type (i.e. collectionlink, wikipedialink, outsidelink, weblink, etc.). Many of these elements would have been given an exhaustivity value of "too small" with the previous INEX definition of relevance. A filtered set of assessments was thus created, where all link element types were removed. For the focused task, correlation analysis reported in [19], indicated, that, as evaluated by the XCG measures, how to consider the too small elements is important, as they can affect the ranking of systems. More work is needed to address this issue.

3.2 Evaluation of the thorough task

The aim of the task is to return a ranked list of relevant XML elements, in descreasing order of relevance. The goal here is to test a system's ability to produce the correct ranking. A number of measures from the XCG family were adopted to evaluate this task: effort-precision/gain-recall (ep/gr) graph, mean average effort-precision ($MAep$), and interpolated mean average effort-precision ($iMAep$). These measures provide an overall picture of retrieval effectiveness across the complete range of recall. These were chosen because of the recall-oriented nature of the task, e.g. rank all elements of the collection.

All XCG measures are based on the underlying concept of the value of gain, $xG[i]$, obtained when examining the i-th result in the ranked output of an XML retrieval system. Given a ranked list of elements, where the element IDs are replaced with their relevance scores, the cumulated gain at rank i, denoted as $xCG[i]$, is computed as the sum of the relevance scores up to that rank:

$$xCG[i] = \sum_{j=1}^{i} xG[j] \tag{7}$$

If we consider that the best system is one that returns the more relevant elements first, an ideal gain vector, xI, can be derived for each topic by filling the rank positions with the relevance scores of the relevant elements in decreasing order of their relevance scores. The corresponding cumulated ideal gain

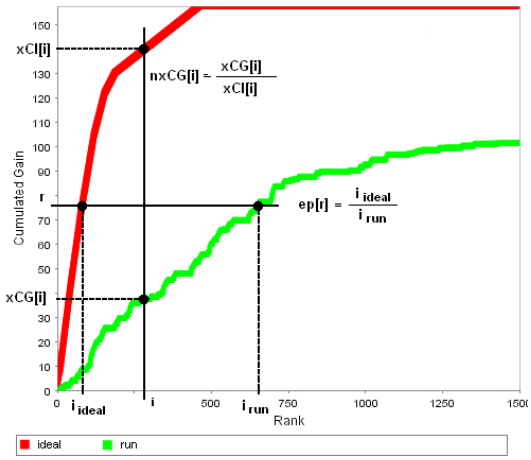

Figure 4: Calculation of $nxCG$ and effort-precision ep

vector is denoted as xCI and is calculated analogue to $xCG[i]$. Both $xG[i]$ and $xI[j]$ are calculated using the element's quantizised value as given by any of the functions listed in Section 3.1:

$$xG[i] = quant(e_i) \tag{8}$$

$$xI[j] = quant(e_j) \tag{9}$$

where e_i is the i-th element in the system ranking, and e_j is the j-th element in the ideal ranking.

Switching viewpoints, we may ask what is the amount of effort required to reach a given level of cumulated gain when scanning a given ranking, compared to an ideal ranking. The horizontal line drawn at the cumulated gain value of r, shown in Figure 4, illustrates this view. This is captured by the effort-precision e/p at a given cumulated gain value r, which measures the amount of relative effort (where effort is measured in terms of number of visited ranks) that the user is required to spend when scanning a system's result ranking compared to the effort an ideal ranking would take in order to reach the given level of gain:

$$ep[r] = \frac{i_{ideal}}{i_{run}} \tag{10}$$

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.

By scaling the recall axis to $[0, 1]$ (i.e. dividing by the total gain), effort-precision can be measured at arbitrary recall points, $gr[i]$ [18]:

$$gr[i] = \frac{xCG[i]}{xCI[n]} = \frac{\sum_{j=1}^{i} xG[j]}{\sum_{j=1}^{n} xI[j]} \tag{11}$$

where n is the total number of relevant elements for the given topic. The range for i is $[0, n]$, where n is the length of a result list. This method follows the same viewpoint as standard precision/recall, where recall is the control variable and precision the dependent variable. In our case, the gain-recall is the control variable and effort-precision is the dependent variable. Analogue to recall/precision graphs,

we can plot effort-precision against gain-recall and obtain a detailed summary of a system's overall performance. Interpolation techniques are necessary to estimate effort-precision values at non-natural gain-recall points, e.g. when calculating effort-precision at standard recall points of $[0.1, 1]$, denoted as e.g. $ep@0.1$. For this purpose, a simple linear interpolation method is used.

Again as with standard precision/recall, we calculate the non-interpolated mean average effort-precision, denoted as $MAep$, by averaging the effort-precision values obtained for each rank where a relevant document is returned. We can also calculate an average over the interpolated effort-precision values, which we will refer to as $iMAep$. INEX 2006 decided to use $MAep$ because, as reported in [15], $MAep$ and $iMAep$ were found to be highly correlated when evaluating retrieval effectiveness. Finally, [14], using the above XCG measures, leads to the same results in terms of retrieval systems effectiveness than when using the inex_eval measures, which is what we would expect.

3.3 Evaluation of the focused task

In this task, systems are asked to return the ranked list of the top most relevant, relevant XML elements for each given topic, without returning overlapping elements. The task is similar to the thorough task in that it requires a ranking of XML elements, but here systems are required not only to estimate the relevance of elements, but also to decide which element(s) to return from two nested relevant elements.

The normalized cumulated gain $nxCG[i]$ measure is used to evaluate the task. For a given topic, the normalized cumulated gain measure is obtained by dividing a retrieval run's xCG vector by the corresponding ideal xCI vector (see Section 3.2 for the definition of these two vectors):

$$nxCG[i] := \frac{xCG[i]}{xCI[i]} \tag{12}$$

In the above, $xCG[i]$ takes its values from the full set of relevance assessments (so-called full recall-base) of the given topic and $i \in [0, n]$ where n is the length of a result list. $xCI[i]$ takes its values from the ideal recall-base (described below) and i ranges from 0 and the number of relevant elements for the given topic in the ideal recall-base. The gain values $xI[j]$ used in $xCI[i]$ are given by Equation 9. The gain values used in $xCG[i]$ is defined as follows:

$$xG_{norm}[j] = min(xG[j], xG[j_{ideal}] - \sum_{S} xG[k]) \tag{13}$$

for the j-th retrieved element, where j ranges from 1 to i, and where $xG[\cdot]$ is given by Equation 8, j_{ideal} is the rank of the ideal element that is on the same relevant path as the j-th relevant element, and S is the set of elements that overlap with that ideal element and that have been retrieved before rank j. The normalization ensures that a system retrieving all descendant relevant elements of an ideal element cannot achieve a better overall score then if it retrieved the ideal element.

For a given rank i, $nxCG[i]$ reflects the relative gain the user accumulated up to that rank, compared to the gain he/she could have attained if the system would have produced the optimum ranking. As illustrated in Figure 4, $nxCG$ is calculated by taking measurements on both the system and the ideal rankings' cumulated gain curves along the vertical line drawn at rank i. Here, rank position is used as the control variable and cumulated gain as the dependent variable.

The $nxCG[.]$ values can be averaged at various cut-off values. This is denoted as $MAnxCG$, which is defined it as the mean average of $nxCG[i]$ values calculated over the range of $[1, i]$ ranks:

$$MAnxCG[i] := \frac{\sum_{j=1}^{i} nxCG[j]}{i} \tag{14}$$

$MAnxCG[i]$ for various cut-off values i was used in INEX 2005. However correlation analysis, reported in [15], shows that $MAnxCG$ and $nxCG$ at the various cutoffs also report fairly similar results, and hence INEX 2006 only uses $nxCG$ (with cut-off values of 5, 10, 25 and 50).

We return now to the issue of the recall-base, as the evaluation of the focused retrieval task requires two recall-bases. The full recall-base is the list of all elements that contains any relevant information (which therefore includes all parents of any such element), already used in the thorough task. The ideal recall-base is a subset of the full recall-base, where overlap between relevant reference elements is removed so that the identified subset represents the set of ideal answers, i.e. the most focused elements that should be returned to the user.

To construct the ideal recall-base, we need a preference function among exhaustivity and specificty value pairs, or specificity value only in 2006, and a methodology for traversing an XML document (its tree structure) and selecting ideal elements based on their relative preference relations to their structurally related elements. A number of preference relations can be used, based on the quantization functions used in INEX as these reflect the worth of retrieved elements. Given an a chosen quantisation function, it is possible to quantify the value or worth of an element and identify the "best" components within an XML document as those elements with the highest quantized score.

Similarly to the quantization functions, different methodologies for deriving an ideal recall-base may be applied reflecting different strategies. The methodology to traverse an XML tree and select the ideal elements is taken from [23]: Given any two elements on a relevant path,[9] the element with the higher score is selected. In case two elements' scores are equal, the one higher in the tree is chosen (i.e. parent/ascendant). The procedure is applied recursively to all overlapping pairs of elements along a relevant path until one element remains. After all relevant paths in a document's tree have been processed, a final filtering is applied to eliminate any possible overlap among ideal elements, keeping from two overlapping ideal paths the shortest one.

In [13], alternative methodologies that have been proposed to build an ideal recall-base were discussed. A study reported in [12] shows that the chosen methodology can impact on the obtained performance scores. Work is currently on-going looking at this issue more closely.

4 Conclusions

INEX has focused on developing an infrastructure, test collections, and appropriate scoring methods for evaluating the effectiveness of content-oriented XML retrieval. The initiative is now entering its sixth year, with INEX 2007 beginning in April 2007. The major achievements, including outlooks for 2007, in XML retrieval evaluation can be summarised as follows:

- A larger and more realistic test collection has been achieved with the addition of the Wikipedia documents. The content of the Wikipedia collection can also appeal to users with backgrounds

[9] We recall that a relevant path is a path in an article file's XML tree, whose root element is the article element and whose leaf element is a relevant element.

other than computer science, making the carrying out of user studies with this collection more appropriate.

- A better understanding of information needs and retrieval scenarios. The set of retrieval tasks that were used at INEX 2006 is considered as a good representation of actual retrieval tasks that users of an XML retrieval system may wish to perform.

  INEX 2007 will add new retrieval tasks; two under consideration include passage retrieval [27] and range retrieval [1].

- A better understanding of how to measure the effectiveness of different retrieval systems by using appropriate metrics. In particular, we now have an understanding of how to deal with near-misses and overlapping elements, and which metrics to use under which retrieval assumptions. There are however still problems to solve, e.g. the consideration of "too small" elements and the construction of the ideal-recall base when using the XCG measures to evaluate effectiveness.

  Since its launch in 2002, INEX has been challenged by the issue of how to measure an XML information access system's effectiveness. Several metrics have been used at various years, e.g. XCG [14] described in this report, HiXEval [22] and PRUM [23]. Investigation into scoring methods is ongoing, where in order to compare the results of different tasks, including new tasks, it is planned to use a single set of measures.

Acknowledgments

INEX is an activity of the DELOS Network of Excellence in Digital Libraries. This report is an extended version of [20] presented at the DELOS Conference on Digital Libraries, Tirrenia, Pisa, Italy, in February 2007, which itself is based on a number of INEX publications (e.g. [8, 21, 14]). Sections 3.2 and 3.3 are based on [19]. We would like to thank Birger Larsen and Jaap Kamps on their feedback to this report and the authors would like to acknowledge the INEX organisers for the definition of the various retrieval tasks, and all participants for their valuable contributions throughout the various INEX campaigns.

References

[1] C. L. Clarke. Range results in XML retrieval. In *INEX 2005 Workshop on Element Retrieval Methodology*, pages 4–5, 2005.

[2] L. Denoyer and P. Gallinari. The Wikipedia XML Corpus. *SIGIR Forum*, 40(1), 2006.

[3] N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors. *INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the First INEX Workshop (INEX 2002)*, ERCIM Workshop Proceedings, 2003.

[4] N. Fuhr, M. Lalmas, and S. Malik, editors. *INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop (INEX 2003)*, 2004.

[5] 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)*. Springer, 2006.

[6] 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)*. Springer, 2005.

[7] N. Fuhr, M. Lalmas, and A. Trotman, editors. *Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2006)*, Springer, 2007.

[8] N. Goevert, N. Fuhr, M. Lalmas, and G. Kazai. Evaluating the effectiveness of content-oriented XML retrieval methods. *Journal of Information Retrieval*, 9(6):699–722, 2006.

[9] D. Hawking, E.M. Voorhees, N. Craswell, and P. Bailey. Overview of the TREC-8 Web Track. In *TREC*, 1999.

[10] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM TOIS*, 20(4):422–446, 2002.

[11] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. Structured queries in XML retrieval. In *CIKM*, pages 2–11, 2005.

[12] G. Kazai. Choosing an Ideal Recall-Base for the Evaluation of the Focused Task: Sensitivity Analysis of the XCG Evaluation Measures. In Fuhr et al. [7].

[13] G. Kazai and M. Lalmas. Notes on What to Measure in INEX. In *INEX 2005 Workshop on Element Retrieval Methodology*, Glasgow, July 2005.

[14] G. Kazai and M. Lalmas. eXtended Cumulated Gain Measures for the evaluation of content-oriented XML retrieval. *ACM TOIS*, 24(4), pages 503–542, 2006.

[15] G. Kazai and M. Lalmas. INEX 2005 Evaluation Metrics. In Fuhr et al. [5].

[16] G. Kazai, M. Lalmas, and A.P. de Vries. The overlap problem in content-orientedXML retrieval evaluation. In *SIGIR*, pages 72–79, 2004.

[17] G. Kazai, S. Masood, and M. Lalmas. A study of the assessment of relevance for the INEX'02 test collection. In *ECIR*, pages 296–310, 2004.

[18] J. Kekäläinen and K. Järvelin. Using graded relevance assessments in IR evaluation. *JASIST*, 53(13), 2002.

[19] M. Lalmas, G. Kazai, J. Kamps, J. Pehcevski, B. Piwowarski, and S. Robertson. INEX 2006 Evaluation Measures. In Fuhr et al. [7].

[20] M. Lalmas and A. Tombros. INEX 2002 - 2006: Understanding XML Retrieval Evaluation. In *DELOS Conference on Digital Libraries, Tirrenia, Pisa, Italy*, 2007.

[21] P. Ogilvie and M. Lalmas. Investigating the exhaustivity dimension in content-oriented XML element retrieval evaluation. In *CIKM*, pages 84–93, 2006.

[22] J. Pehcevski and J. A. Thom. Hixeval: Highlighting xml retrieval evaluation. In *Advances in XML Information Retrieval and Evaluation: Fourth Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2005)*, 2006.

[23] B. Piwowarski and G. Dupret. Evaluation in (XML) information retrieval: expected precision-recall with user modelling (eprum). In *SIGIR*, pages 260–267, 2006.

[24] B. Piwowarski and M. Lalmas. Providing consistent and exhaustive relevance assessments for XML retrieval evaluation. In *CIKM*, pages 361–370.

[25] V. V. Raghavan, P. Bollmann, and G. S. Jung. A critical investigation of recall and precision as measures of retrieval system performance. *ACM TOIS*, 7(3):205–229, 1989.

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

[27] A. Trotman and S. Geva. Passage Retrieval and Other XML-Retrieval Tasks. In *SIGIR 2006 Workshop on XML Element Retrieval Methodology*, pages 43–50, 2006.

[28] A. Trotman and M. Lalmas. Strict and vague interpretation of XML-retrieval queries. In *SIGIR*, pages 709–710, 2006.

[29] A. Trotman and M. Lalmas. Why structural hints in queries do not help XML retrieval. In *SIGIR*, pages 711–712, 2006.

[30] A. Trotman and B. Sigurbjornsson. Narrowed extended XPATH I (NEXI). In *INEX 2004 Workshop Proceedings*, pages 16–40, 2004.