eXtended Cumulated Gain Metrics for the Evaluation of Content-oriented XML Retrieval

Gabriella Kazai and Mounia Lalmas
Department of Computer Science
Queen Mary, University of London
London E1 4NS

We propose and evaluate a family of metrics, named the eXtended Cumulated Gain (XCG) metrics, for the evaluation of content-oriented XML retrieval approaches. Our aim is to provide an evaluation framework that allows to consider the dependency that exists among XML document components and, in particular, incorporate mechanisms to reward the retrieval of so-called near-misses and to address issues of overlap. Both system and user-oriented evaluation aspects are considered and both recall and precision-like qualities are measured. Another consideration is that the metrics should be flexible enough so that different models of user behaviour may be instantiated within. We evaluate the proposed XCG metrics, based on the INEX test collection, both with respect to their fidelity and reliability. For example, the effects of assessment variation and topic set size on evaluation stability are investigated, and upper and lower bounds of expected error rates are established. The evaluation demonstrates that the proposed XCG metrics are reliable, stable measures, and in particular that the novel metric, effort-precision and gain-recall (ep/gr), shows comparable behaviour to established measures like precision and recall.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms: Performance, Measurement

Additional Key Words and Phrases: XML retrieval, INEX, evaluation, metrics, overlap, cumulated gain

1. INTRODUCTION

Content-oriented XML retrieval is a domain of information retrieval (IR) that is receiving increasing interest fueled by the widespread use of the eXtensible Markup Language (XML) as a standard document format on the Web and in Digital Libraries. The continuous growth in XML data sources is matched by increasing efforts in the development of XML IR systems (e.g. [Baeza-Yates et al. 2002; Blanken et al. 2003; Fuhr et al. 2005]). XML IR systems aim to harness the enriched source of syntactic and semantic information that XML markup provides. Current work in XML IR focuses on the syntactic layer, aiming to exploit the available structural information in documents in order to implement a more focused retrieval strategy and return document components - instead of complete documents - in response

to a user query. This focused retrieval approach is of particular benefit for collections containing long documents or documents covering a wide variety of topics (e.g. books, user manuals, legal documents, etc.), where the users' effort to locate relevant content can be reduced by directing them to the most relevant parts of the documents. For example, in response to a user query on a collection of scientific articles marked-up in XML, an XML IR system may return a mixture of paragraph, section, article, etc. elements, that have been estimated to best answer the user's query.

The new opportunities that arise in XML IR challenge traditional methods of evaluation. In particular, the lack of a predefined uniform unit of retrieval and the dependency that exists among retrievable components in XML IR has numerous consequences on the evaluation methodology, invalidating the assumptions that standard IR experiments build upon [Gövert and Kazai 2003]. For example, a typical assumption in IR evaluation is that documents are considered atomic and independent units (whose relevance is independent of any other document). This is clearly not the case in XML IR, where multiple and variable granularity components from the same document may be retrieved. Furthermore, similarly to Web IR, the structural links between document components in XML IR allow for richer forms of user interaction, combining searching and browsing [Chiaramella et al. 1996]. This increased richness of the users' interaction with the system requires additional criteria to be considered within the evaluation. For example, users may locate additional relevant information just by browsing from a returned result element to another or simply by reading further down in a document [de Vries et al. 2004]. This motivates the need to allow partial scores to be rewarded to systems for the retrieval of so-called near-miss results, i.e. document components that are structurally related to the relevant content sought-after by the user. For example, a section containing the desired relevant paragraph, or a neighbouring paragraph or a neighbouring section may all be considered as near-misses, as they may lead the user to the desired information. On the other hand, users may get disoriented or frustrated when multiple, structurally related components are returned to them, especially if these are dotted about at different positions in the ranking. For example, an XML IR system may retrieve a section and one or several of its paragraphs in response to a user query. The retrieval of such overlapping components can, however, inundate users with redundant text fragments that represent no further value [Tombros et al. 2005].

Traditional IR measures (i.e. based on precision and recall) rely on assumptions, such as the independence of retrieval units, and assume a simple model of user interaction, ignoring aspects of browsing. Such measures are, hence, not suitable for the evaluation of content-oriented XML retrieval. The invalidity of the traditional assumptions in XML retrieval, and the requirement to consider additional aspects of the retrieval model and user interaction make it necessary to build new test collections and develop new measures and procedures for the evaluation of XML retrieval systems.

The Initiative for the Evaluation of XML retrieval (INEX)[1] has, for the past four years, been building an XML test collection with the aim to establish an

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

infrastructure for the evaluation of content-oriented retrieval of XML documents. However, since its launch, INEX has been challenged by the issue of how to measure an XML IR system's effectiveness. The official metric of INEX 2002-2004, inex-eval [Gövert and Kazai 2003], is based on traditional notions of precision and recall and has been shown to have several weaknesses [Kazai et al. 2004]. One such issue is that the metric does not take into account the overlap of result elements and hence produces better effectiveness scores for systems that return multiple nested components, contradicting the aim of XML IR. Another issue is that inex-eval calculates recall based on a so-called overpopulated recall-base[2], which contains large amounts of overlapping components (see Section 3). This means that 100% recall can only be reached by systems that return all elements of the full recall-base, including all overlapping components. An effect of the latter issue is that precision scores of systems that actually make an effort not to inundate users with overlapping, and hence redundant, elements, are plotted against lower recall values than merited. Finally, inex-eval only counts elements included in the recall-base as hits and does not allow for the partial scoring of near-misses.

In this paper, we propose an alternative evaluation framework, complete with a family of evaluation metrics that extend the cumulated gain based measures of [Järvelin and Kekäläinen 2002] and aim to address the issues that the dependency of retrieval units introduces into the evaluation. We also evaluate the proposed metrics, testing both their fidelity and reliability based on established methodology. The conclusions of our study are encouraging regarding the reliability of the XCG metrics, which have in the meantime been adopted as the official metrics of INEX 2005.

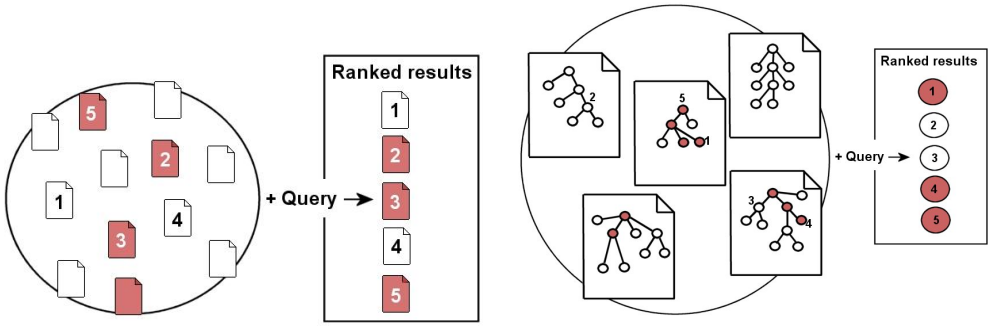
The paper is structured as follows. First, in the next section, we examine in more detail the retrieval and user model of XML IR and list the requirements that suitable evaluation metrics should satisfy. In Section 3 we describe the INEX test collection, which is used in the experiments later on. Section 4 defines the notion of an ideal recall-base and details a possible methodology for deriving it from the original, over-populated set of assessments. We describe the proposed XCG metrics in Section 5 and evaluate them in Section 6. We close with conclusions and future work.

2. CONSIDERATIONS FOR THE EVALUATION OF XML IR

2.1 IR evaluation

Evaluation in IR has a long and rich history with work dating back to the development of the first IR systems in the 1950's, resulting in a wealth of evaluation studies and initiatives [Rijsbergen 1979; Baeza-Yates and Ribeiro-Neto 1999]. Over the years, it has become common practice to evaluate retrieval systems' effectiveness using test collections, consisting of a fixed set of documents, user requests and relevance assessments. Fixing the control variables of the evaluation this way allows to best focus on the retrieval approaches to compare their relative effectiveness [Tague-Sutcliffe 1992]. This so-called *Cranfield tradition* of experimental evaluation

[2]The term recall-base refers to the collection of relevant elements within the test collection - obtained through relevance assessments -, which forms the ground-truth for evaluation experiments.



a) Sketch of traditional IR retrieval model b) Sketch of XML IR retrieval model

Fig. 1. Traditional IR versus XML IR. Filled shapes represent relevant documents or components.

has since converged into what is now known as 'standard IR evaluation practice'. It has become universal through the retrieval evaluations organised at the Text REtrieval Conference (TREC) [Harman 1992].

These traditional evaluation experiments rely on implicit assumptions regarding the underlying retrieval task and user model. A typical retrieval task of a traditional IR system, often referred to as "flat document retrieval", is to return a ranked list of relevant documents in response to the user's query. The user of such a system is typically associated with a simple model for interacting with the system. He/she poses a query that represents an information need and usually takes the form of a bag of keywords. In response, the system returns a ranked list of documents that have been estimated to satisfy the information need. The user is then assumed to examine the ranked list in a linear fashion, moving from the top of the list down, reading each returned document, either until the end of the list is reached, or until the information need has been satisfied or until he/she gives up. Each result in the ranking is assumed to be independent and users typically do not have access to other documents, e.g. through browsing. This is illustrated in Figure 1a.

Based on the above assumptions, established measures, such as precision and recall, provide suitable and intuitive mechanisms for evaluating the effectiveness of IR systems. They typically measure the quality of a system's output as a function of the retrieved (not-retrieved) and relevant (non-relevant) documents. Both set-based and probabilistic interpretations of precision and recall build on the assumption of a document representing an atomic unit of retrieval, which is considered independent from other documents. When computing precision at certain ranks, it is implicitly assumed that a user spends a constant time per document. Based on the implicit definition of effectiveness as the ratio of output quality vs. user effort, quality is measured for a fixed amount of effort. The ranking is typically considered by taking counts at various recall levels, e.g. precision and recall graphs.

2.2 XML IR evaluation

In XML IR, shown in Figure 1b, the collection that is searched by the user consists of XML documents composed of different granularity nested XML elements, each of which represents a possible unit of retrieval. The relevance of a component in the

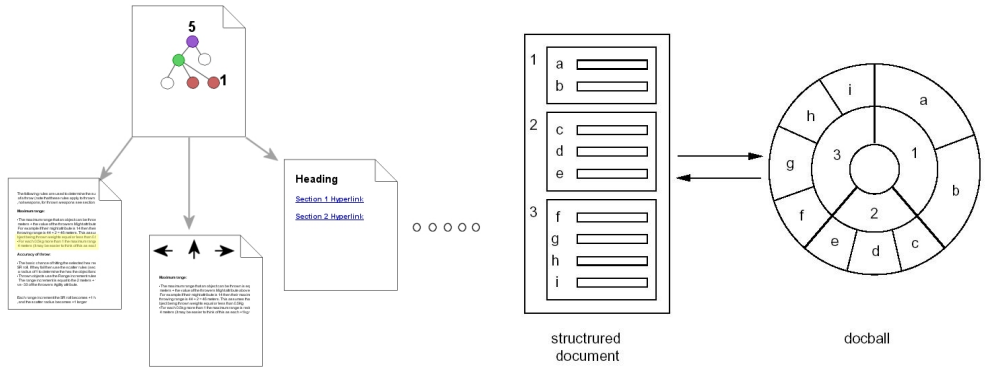
collection may therefore be directly dependent on the relevance of other structurally related components. This is clearly the case, for example, for nested components where the contents of the contained components is subsumed by the container component. The user's query may also differ from the usual bag of keywords as it may contain structural constraints or hints in addition to the content conditions.

In this setting, the task of an XML IR system - as defined in INEX - is to identify the most appropriate granularity XML elements to return to the user (with or without the help of possible structural hints within the query). Following the findings of the FERMI project [Chiaramella et al. 1996], these most appropriate elements have been defined as document components that are most specific while being exhaustive with respect to the topic of request [Fuhr et al. 2004; Malik et al. 2005]. This reflects user preference for document components that contain as much relevant information and as little non-relevant information as possible. Compared to flat document retrieval, the task of an XML IR system is then not only to identify relevant content, but to also identify the best unit of retrieval that should be returned to the user.

Assuming a linear ranking of the result components, users of an XML IR system follow the usual method of examining the ranked list, moving from the top of the list down, either until the end of the list is reached, or until the information need has been satisfied or until they give up. Following standard IR practise, users are assumed to view returned elements in their entirety[3]. However, unlike in the traditional IR model, users in XML IR have access to other, structurally related components from a returned result element. They may hence locate additional relevant information by ways of browsing or scrolling - depending on the actual user interface. For instance, a result element may be displayed to the user as a highlighted text fragment within its context; as an expanded hypertext node (containing contents of its sub-nodes) with links to related nodes; as a hypertext node with links to its sub-nodes; or as a unit within a clearly defined document structure (see Figure 2). Different result presentation approaches will support different forms of user interaction, where the ease of access to structurally related components will also vary.

The aim of XML IR evaluation is to provide a measure of retrieval effectiveness that reflects a system's ability to return these most appropriate XML elements that best answer the user's query. A major difference compared to traditional IR evaluation is that due to the lack of an atomic predefined unit of retrieval as well as the increased richness of the user's interaction with the system (i.e. searching and browsing), it becomes necessary to consider near-misses within the evaluation. For example, a system may return a container section or a neighbouring paragraph instead of the actual desired relevant paragraph. Given that users of an XML IR system may have access to a returned result element's structurally related nodes and/or context, such near-misses may be useful for the user as it leads to otherwise lost relevant information. The idea is then to allow systems to pick up partial scores for near-miss results. The alternative, to simply consider near-misses as

[3]An alternative model is investigated within the T2I (Tolerance to Irrelevance) framework, where users read the contents of a document from a returned entry point until their tolerance to irrelevance is exhausted.



a) Hypothetical interfaces b) Docball interface [Vegas et al. 2002]

Fig. 2. Possible result presentations in XML IR.

non-hits would result in a rather strict evaluation scenario, especially when dealing with fine-grained XML documents. This problem is similar to that encountered in the evaluation of video retrieval, where marginal shifts in the shot boundaries of retrieval results can directly affect the obtained effectiveness scores [de Vries et al. 2004].

The need to reward systems partial scores for the retrieval of near-miss results raises a critical issue with traditional precision/recall type measures. First, the issue of the lack of a predefined atomic retrieval unit needs to be addressed. There are two possible solutions: a suitable recall-base can be constructed (1) by defining the smallest units that make up the documents in the collection (as done in [Kazai et al. 2003] and at TRECVID after 2001) or (2) by collecting all possible units of retrieval. For XML documents, both these cases are viable as they both lead to a finite set of elements (i.e. set of leaf nodes, or set of all (nested) nodes). INEX followed the latter approach. However, given a fixed set of relevant results in the recall-base, how can we reward the retrieval of near-misses and maintain reliable recall and precision? If we include all possible near-misses in the recall-base then we increase recall and end up indirectly penalising systems that return only the desired relevant results (as is the case in INEX when using inex-eval). If we count hits at the level of the smallest units then we indirectly encourage systems to return relevant information broken up into the smallest fragments in order to increase overall precision [Allan 2004]. It is therefore necessary to provide alternative measures whithin which the retrieval of near-misses can be accommodated.

The exact mechanism for calculating the actual scores for near-misses may vary for different models of user interactions and different user interfaces. Different result presentation approaches will require different considerations to reflect the associated cost of accessing related components. For example, in a hyperlinked environment the chances of a user following a particular link may depend on the presentation and informativeness of the anchor text or how many other links are displayed. In the case where relevant information is presented to the user as highlighted text fragment within its context, the user may be more likely to access closely neighboring elements than elements that require extensive scrolling up or

down. The complexity of these factors, together with several other aspects of a user's interaction and preferences that can influence the user's satisfaction with a retrieval system, motivates the need to employ metrics where the user model is separated and can be instantiated as appropriate for a given evaluation experiment.

In addition to near-misses, the evaluation should also consider the retrieval of overlapping components. Due to the structural relationships among the retrievable XML elements (and depending on the retrieval algorithm), an XML IR system may return related elements within the ranked result list to the user. For example, the ranking in Figure 1b contains structurally related results at ranks 1 and 5. Research shows that such redundant elements do not represent any additional value to the user [Tombros et al. 2005] and can lead to user disorientation and frustration [Chiaramella et al. 1996], while at the same time contradict the aim of XML IR to reduce user effort.

As a counter argument, it may be said that overlap - from a system evaluation point of view - should not be seen as an issue since systems could be assumed to be able to deal with it when presenting their results to users. For example, systems may remove overlapping nodes via some filtering strategy or cluster them together, and so on. Therefore, overlap should not be a concern when system-oriented evaluation is applied. This said, a crucial (implicit) assumption of this argument is that overlap should not represent a potential gain factor either [Piwowarski and Gallinari 2004; de Vries et al. 2004; Kazai et al. 2004; Piwowarski et al. 2005]. This means that systems should not be penalised for not retrieving overlapping nodes. However, as demonstrated in [Kazai et al. 2004], the inex-eval metric of INEX actually shows better effectiveness for systems that exploit overlap and inundate users with redundant components, while systems that avoid overlap are penalised on recall. The reason for this is that the recall-base in INEX consists of sets of overlapping components (e.g. a paragraph and its container section and article), which when combined with the use of traditional precision/recall measures, leads to skewed effectiveness scores. As mentioned above, in this scenario systems can only achieve 100% recall if all components of the recall-base, i.e. including the overlapping components, are retrieved. Therefore, unless an evaluation metric that explicitly addresses this issue is employed, unfair advantage can be gained by systems that exploit overlap over systems that actually put effort into not to inundate users with such redundancy. Given a suitable metric, systems would be free to follow a retrieval strategy that outputs overlapping results, but such a strategy would not present an advantage (and may in fact not prove beneficial as it may result in the output list being filled with elements of no value while pushing other non-overlapping relevant nodes further down the list).

To conclude, an accurate and meaningful evaluation for XML IR, where the goal of XML IR system is to direct users to the most specific relevant content, requires an evaluation measure that considers near-misses and overlapping elements, while at the same time being flexible enough to support the adaptation of different user models. We aim to propose such an evaluation framework in this paper. Before we do, however, we next describe the INEX test collection followed by the definition of the ideal recall-base, which provides the first step towards our goal.

3. THE INEX TEST COLLECTION

This section provides a brief overview of the INEX test collection, which is used for the experiments in Section 6.

3.1 Document collection

The document collection of the INEX test collection[4] consists of 12,107 articles of the IEEE Computer Society's publications, from 1995 to 2002, totaling 494 megabytes [Gövert and Kazai 2003]. The collection contains over 18.5 million XML nodes including over 8.2 million element nodes of varying granularity, each representing a valid unit of retrieval. The average depth of a node is 6.9. All documents in the collection are tagged using XML conforming to one common schema, i.e. DTD. The overall structure of a typical article consists of a frontmatter (containing e.g. title, author, publication information and abstract), a body (consisting of e.g. sections, sub-sections, sub-sub-sections, paragraphs, tables, figures, lists, citations) and a backmatter (including bibliography and author information).

3.2 Topics

The topics of the test collection include typical IR queries, where no constraints are formulated with respect to the structure of the retrieval results, and XML queries (in a modified XPath [Clark and DeRose 1999] syntax, named NEXI [Trotman and Sigurbjörnsson 2005]) that contain explicit references to the XML structure. The former query type is referred to as content-only (CO), while the latter is content-and-structure (CAS).

Up to date, the INEX has accumulated a total of 201 topics: INEX 2002 created 30 CO (1-30) and 30 CAS (31-60) topics, INEX 2003 added 36 CO (91-126) and 30 CAS (61-90) topics, and INEX 2004 produced 40 CO (162-201) and 35 CAS (127-161) topics. The experiments in Section 6 make use of the INEX 2004 CO queries.

3.3 Relevance assessments

For the construction of the relevance assessments, INEX employs two relevance dimensions: *exhaustivity* and *specificity* [Fuhr et al. 2004; Malik et al. 2005]. Exhaustivity is defined as a measure of how exhaustively a document component discusses the topic of request, while specificity is defined as a measure of how focused the component is on the topic of request (i.e. discusses no other, irrelevant topics) [Kazai et al. 2004]. Both dimensions are based on the topicality aspect of relevance: exhaustivity measures the amount of relevant information, while specificity measures the relative amount of relevant and irrelevant information contained within a document component. They are both measured on four-point scales with degrees of highly (3), fairly (2), marginally (1), and not (0) exhaustive/specific. The combination of the two dimensions is used to identify those relevant document components that represent the most appropriate unit of information to return to the user.

We denote the relevance degree of an assessed component, given by the combined

[4]The statistics reported here are for INEX 2004.

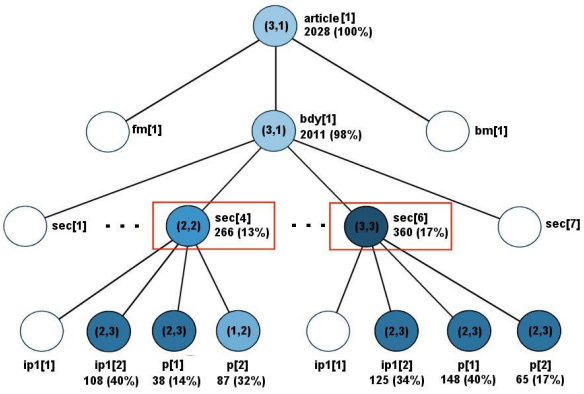


Fig. 3. Sample assessments showing relevant nodes (i.e. $e > 0$ and $s > 0$) - as filled circles - for topic 163 in the article file co/2001/r7022.xml. For each relevant node, its relative XPath, its assessment value pair (e, s), its size in number of words and its size ratio to its parent node is shown. Nodes within the red rectangles are the selected ideal nodes based on the $quant_{sog}$ quantisation function (see Equation 3.

values of exhaustivity and specificity, as $(e, s) \in ES$, where

$$ES = \{(0,0), (1,1), (1,2), (1,3), (2,1), (2,2), (2,3), (3,1), (3,2), (3,3)\}$$

For example, $(2, 3)$ denotes a fairly exhaustive and highly specific component, which means that it discusses many aspects of the topic of request and the topic of request is the only theme of the component (i.e. it contains no irrelevant information).

An important property of the exhaustivity dimension is its propagation effect, reflecting that if a component is relevant to a query then all its ascendant elements will also be relevant. This is because the content of a component is subsumed by its container component, hence any relevant information that is part of, e.g., a paragraph is also part of its container section.

Due to this property, all nodes along a relevant path[5] are always relevant (with varying degrees of relevance), hence resulting in a recall-base comprised of sets of overlapping elements [Kazai et al. 2004]. For example, from the XML article of co/2001/r7022.xml, all relevant elements in Figure 3, shown as filled circles, form part of the recall-base for INEX 2004.

4. DEFINITION OF AN IDEAL RECALL-BASE

As we mentioned earlier, in INEX an XML IR system's task is to return the most appropriate granularity relevant XML elements to the user. These elements have been defined as those being the "most" exhaustive and "most" specific[6] with re-

[5]A relevant path is a path in an article file's XML tree, whose root node is the article element and whose leaf node is a relevant component (i.e. $(e > 0, s > 0)$) that has no or only irrelevant descendants. For example, the XML tree in Figure 3 has six relevant paths.

[6]Note that the term "most" exhaustive/specific does not equate to highly exhaustive/specific, but refers to the highest available exhaustivity and specificity score in a given XML tree. For example,

spect to the user's topic of request [Lalmas and Malik 2004]. We follow the interpretation that the task of identifying the most appropriate granularity relevant elements implies that overlapping, redundant components should not be returned to the user[7]. This is reflective of the findings in [Tombros et al. 2005], which reported that "searchers generally recognise overlapping components, and find them an undesirable 'feature' ".

However, as mentioned in the previous section, the INEX test collection's recall-base consists of sets of overlapping relevant XML elements. This means that systems are actually forced to return overlapping elements in order to achieve better recall.

A solution to this issue is to mark a subset of the relevant elements in the recall-base as ideal answers, i.e. those elements that should be returned to the user. We will refer to this set as the ideal recall-base. The basic idea is that given a set of preference relations among (e, s) value pairs, we can pick from an arbitrary XML tree of related relevant elements those components that represent the best element for the user.

What is needed then is a preference function on the ES space and a methodology for traversing an XML tree and selecting ideal nodes based on their relative preference relations to their structurally related nodes. However, how should we decide about the preference relations? Intuitively, we may decide that highly exhaustive and highly specific components are preferred over all other related relevant nodes, but what if from two related relevant components, one is highly exhaustive but only fairly specific ($(e, s) = (3, 2)$) and the other is only fairly exhaustive but highly specific ($(e, s) = (2, 3)$). Which one should be returned to the user?

Rather than limiting ourselves to a specific answer, we employ so-called quantisation functions to provide a flexible means of modeling various sets of possible user preferences, adjustable to a given user model [Gövert and Kazai 2003]. Any number of different quantisation functions, $quant(\cdot)$, can be defined according to possible user models, each providing a mapping of the two relevance dimensions to a single relevance value: $quant(e, s) : ES \rightarrow [0, 1]$.

The following three functions are typically employed in INEX: $quant_{strict}$ (Equation 1) and $quant_{gen}$ (Equation 2) from [Gövert and Kazai 2003], and $quant_{sog}$ (Equation 3) from [Kazai et al. 2004]. The strict function models a user for whom only highly specific and highly exhaustive components are considered worthy. The generalised (gen) and the specificity-oriented generalised (sog) functions credit document components according to their *degree* of relevance, hence allowing to model varying levels of user satisfaction gained from not perfect, but still relevant components or near-misses. 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 components, while the latter as-

it may be that amongst all relevant components in an article, the most exhaustive node is one with $e = 1$, or the most specific node is one with $s = 2$.

[7]In INEX 2005, this criteria is made explicit within the Focused task definition.

sumes that more specific components are of greater value to the user.

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

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

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

In addition to a quantisation function, a suitable methodology is needed, which, given a tree of relevant elements with associated quantised scores, can identify the set of non-overlapping ideal nodes. In this study, we adopted the following methodology: Given any two components on a relevant path, the component with the higher quantised score is selected. In case two components' scores are equal, the one deeper in the tree is chosen. The procedure is applied recursively to all overlapping pairs of components along a relevant path until one element remains. After all relevant paths have been processed, a final filtering is applied to eliminate any possible overlap among ideal components, keeping from two overlapping ideal paths the shortest one. The reason for selecting the descendant from two overlapping components with equal quantised score is to minimise the propagation effect, while the reason to select from two ideal paths the shortest is to ensure that no relevant information is actually lost in the process.

The resulting ideal recall-base is said to contain the best elements to return to a user based on the assumptions that overlap between result nodes should be avoided and that the user's preferences are reflected within the employed quantisation function. As is the case with the quantisation functions, different methodologies for deriving an ideal recall-base may be applied reflecting different user models. Our goal here is to demonstrate the principle of an ideal recall-base and its application within the evaluation and, hence, in our current study we place less emphasis on investigating which methodology would best reflect which user model. Alternative algorithms to the above have been proposed in [Goevert et al. 2005; Piwowarski et al. 2005] and a comparison of two of these methods is given in [Kazai and Lalmas 2005].

With our chosen methodology and using the $quant_{sog}$ function, the ideal nodes selected from the XML tree in Figure 3 are: `sec[6]` and `sec[4]`[8]. Applying the

[8]Note that if the longest ideal paths were to be selected in the final filtering step, then we would arrive at the ideal nodes of `sec[6]`, `sec[4]/p[1]` and `sec[4]/p[2]`, where the relevant content of

$quant_{strict}$ function leads to a single ideal node of `sec[6]`, while $quant_{gen}$ selects the `bdy[1]` node as ideal.

The constructed ideal recall-base could be used (by itself) for evaluating XML retrieval systems using traditional metrics (i.e. recall and precision). However, as mentioned earlier, in such an evaluation setting systems would be measured against a rather strict ideal scenario, where only exact matches between retrieved and ideal reference elements are considered a hit. Given the possibly fine graded structure of an XML document, the judgement to only credit systems that are able to return exactly the ideal components may seem too harsh, especially since the retrieval of near-misses may still be considered useful for a user when the ideal component is not found.

A better solution may be to combine the use of the full recall-base and the derived ideal recall-base within the evaluation. In this case, document components in the ideal recall-base represent the desired target components that should be retrieved, while all other elements in the full recall-base (or even in the full collection) may be rewarded partial scores. The main significance of the definition of an ideal recall-base is hence that it supports the evaluation viewpoint whereby components in the ideal recall-base *should* be retrieved, while the retrieval of near-misses *could* be rewarded as partial successes, but other systems *need not* be penalised for not retrieving near-misses.

Once an ideal recall-base has been built, an ideal run can be created by ordering the components of the ideal recall-base in decreasing value of their quantised relevance scores.

5. EXTENDED CUMULATED GAIN (XCG) METRICS

The XCG metrics are a family of metrics that are an extension of the cumulated gain (CG) based metrics of [Järvelin and Kekäläinen 2002]. The motivation for the CG metrics were to develop a measure for multi-graded relevance values, which allow to credit IR systems according to the retrieved documents' degree of relevance. The motivation for our XCG metrics was to extend the CG metrics to the problem of content-oriented XML IR evaluation, where the dependency of XML elements is taken into account. The extension lies partly in the way the gain value for a given document - or in this case document component - is calculated via the definition of so-called relevance value (RV) functions, and partly in the definition of ideal recall-bases. The former allows to consider the dependency of result elements within a system's output, while the latter regards the dependency of elements within the test collection's recall-base (see Section 4).

In [Kazai et al. 2004; 2005], we reported two metrics of the XCG family: xCG and normalised xCG ($nxCG$)[9]. In this section we will first provide a brief recap of these measures followed by their critical analysis. We then go onto define a novel new XCG metric: effort-precision and gain-recall (ep/gr) and its variations which aim to address the limitations of the previous metrics.

`sec[4]/ip1[2]` would have been removed from the recall-base.

[9]We use a slightly different notation in this paper to that in [Kazai et al. 2004] in order to reduce the overload on the term "XCG".

5.1 xCG and normalised xCG ($nxCG$)

As with the CG measures of [Järvelin and Kekäläinen 2002], the XCG metrics compute the cumulated gain the user obtains by examining the retrieval results up to a given rank. The xCG metric accumulates the relevance scores of retrieved documents along the ranked list. Given a ranked list of document components, xG, where the document 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:

$$xCG[i] := \sum_{j=1}^{i} xG[j] \tag{4}$$

For example, the ranking $xG_q = <3, 1, 0, 0, 1, 3, 2, 2, 0, 0>$ produces the cumulated gain vector of $xCG_q = <3, 4, 4, 4, 5, 8, 10, 12, 12, 12>$.

The above definition of cumulated gain can be considered as an extension of the binary function of counting the number of retrieved relevant documents, which forms the basis of the traditional set-based recall and precision measures.

Assuming that users prefer to be returned more relevant documents first, we can derive an ideal gain vector, xI, for each query by filling the rank positions with the relevance scores of all documents in the recall-base (or as in the case of INEX, with the relevance scores of all elements in the ideal recall-base) in decreasing order of their degree of relevance. The corresponding cumulated ideal gain vector is referred to as xCI. For our toy example, the ideal gain vector may be $xI_q = <3, 3, 3, 3, 2, 2, 2, 1, 1, 0, ...>$, for which we obtain $xCI_q = <3, 6, 9, 12, 14, 16, 18, 19, 20, 20, ...>$.

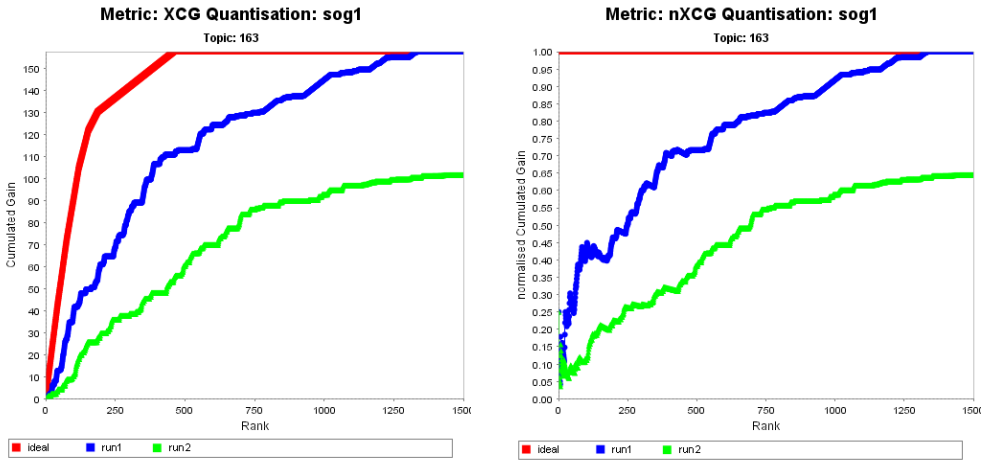
A retrieval run's xCG vector can then be compared to this ideal ranking by plotting both the actual (xCG) and ideal (xCI) cumulated gain functions against the rank position. We obtain two monotonically increasing curves, leveling after no more relevant documents can be found. Figure 4a shows the xCG curves obtained for two arbitrary sample runs (`run1` and `run2`), along with the corresponding ideal curve (`ideal`).

By dividing the xCG vectors of the retrieval runs by their corresponding ideal xCI vectors, we obtain the normalised xCG ($nxCG$) measure:

$$nxCG[i] := \frac{xCG[i]}{xCI[i]} \tag{5}$$

For a given rank i, the value of $nxCG[i]$ 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 best ranking. For our example gain vector xG_q, we obtain $nxCG_q = <1, 0.67, 0.44, 0.33, 0.36, 0.5, 0.56, 0.63, 0.6, 0.6>$. For our two sample runs of Figure 4a, Figure 4b shows the obtained $nxCG$ curves. For any rank the normalised value of 1 represents ideal performance. The area between the normalised actual and ideal curves represents the quality of a retrieval approach.

The normalised cumulated gain measure can be compared to the traditional set-based measure of recall. Unlike recall, however, instead of normalising the number of retrieved relevant documents by the total number of relevant documents in the collection, the gain accumulated by a system up to a given rank is compared to the



a) Example xCG graph

b) Example $nxCG$ graph

Fig. 4. Sample graphs. An xCG graph plots the cumulated gain obtained by the user up to a given rank. A $nxCG$ (normalised xCG) graph plots the cumulated gain relative to the ideal gain achievable up to a given rank.

maximum possible gain that can be achieved up to that rank.

In an evaluation experiment, we may employ several cutoff values that we wish to compare systems at, e.g. $nxCG[5]$ or $nxCG[100]$. Alternatively, we may measure performance over a range of cutoffs and average their results [Hull 1993]. We denote this averaged measure as $MAnxCG$ and define it as the mean average of $nxCG[i]$ values calculated over the range of $[1, i]$ ranks:

$$MAnxCG[i] := \frac{\sum_{j=1}^{i} nxCG[j]}{i} \tag{6}$$

For example, we obtain $MAnxCG_q[6] = 0.55$ for our xG_q vector.

5.2 Calculating a component's relevance value

We based the definition of the xCG and $nxCG$ metrics on the gain value, $xG[i]$, that a user obtains when examining a returned result component. In this section we detail how this gain is calculated.

5.2.1 *Relevance value (RV) function.* We define a relevance value function, $r(c_i)$, as a function that returns a value in $[0, 1]$ for a component c_i in a ranked result list, representing the component's relevance or gain value to the user. The meaning of such a value may be compared to the notion of utility, reflecting the worth that a retrieved component holds for the user. A score of 0 reflects no value to the user, 1 is highest relevance score and values in between represent various gain levels.

The relevance value of a result component will depend on a number of factors, like the component's exhaustivity and specificity degree (i.e. its (e, s) values). When retrieval results are assumed independent, the gain value of a document can be obtained as a direct function of these parameters. When results are dependent,

a component's gain value will likely be influenced by additional aspects, such as what the user may have been returned already at earlier ranks or what additional elements he/she can access from the current returned component (e.g. assuming that retrieval results are presented as entry points into XML documents, users may discover additional relevant information just by reading on [de Vries et al. 2004]). Given the richness of the parameters that may contribute to a model of a user, above and beyond those mentioned here, a wide variety of RV functions may be defined, each capturing a different type of user behaviour. In this study we consider a simple model, focusing on overlap and near-misses.

We define the following result-list dependent relevance value (RV) function:

$$rv(c_i) := f(quant(assess(c_i))) \tag{7}$$

where $assess(c_i)$ is a function that returns the assessment value pair (e, s) for the i-th component in the ranking if it is given within the recall-base and $(0, 0)$ otherwise. The function $quant(\cdot)$ is a quantisation function, providing a mapping of the (e, s) assessment value pair to a real number in $[0, 1]$ (see Section 4). The $f(\cdot) : [0, 1] \rightarrow [0, 1]$ function allows us to take into account the dependency among retrieved elements and to reward near-misses as well as to take into account overlap. The next sections examines this in detail.

5.2.2 *Considering near-misses.* Our aim here is to consider the retrieval of near-misses within the evaluation. The basic idea is to reward a partial score for the retrieval of non-ideal elements that are structurally related to ideal components.

As mentioned in Section 2.2, the value of a near-miss to the user will vary depending on the result presentation. Therefore, the actual score rewarded to a near-miss may, for example, be calculated based on the difference in component length if the components are contained within one another, or based on their distance from each other otherwise. Additional factors may regard reading-order [de Vries et al. 2004] and may, for example, decrease the score of a near-miss result if it comes after an ideal node, i.e. when the user would be required to scroll up in a document (or browse to a previous page) - a perhaps less natural action than reading forward.

In this study, our aim is not to investigate and propose appropriate methods to derive near-miss scores, but to provide a framework within which such scores can be incorporated and hence the retrieval of near-misses can be rewarded. We therefore limit ourselves to the scenario, where only those relevant elements of the full recall-base are considered near-misses which are not included in the ideal recall-base. For example, in Figure 3, all relevant nodes with the exception of the two ideal nodes (`sec[4]` and `sec[6]`) are considered as near-misses. The advantage of this methodology is that it minimises the additional complexity that would be required in order to estimate the relevance values, but its disadvantage is that it does not allow rewarding non-relevant near-misses no matter how close they may be to the relevant content (e.g. a neighbouring non-relevant paragraph). To address this issue, one may consider the whole collection as the set of near-misses and employ a suitable function to estimate the gain value based on, e.g. proximity parameters.

Given a set of near-misses and an ideal recall-base, we can apply the XCG metrics to evaluate XML retrieval approaches in INEX, whereby the ideal gain vector of a query, xI, is derived from the ideal recall-base, while the gain vectors, xG,

corresponding to the system runs under evaluation are based on the full recall-base. Since the relevance degrees of elements in the full recall-base have already been judged by the INEX assessors, we can directly rely on this information when scoring near-misses. The relevance score of a near-miss component can then be calculated simply by Equation 7, where $f(x)$ is simply defined as $f(x) = x$.

However, all we have done so far is that we defined a relevance value for a retrieved near-miss independently of other retrieved related nodes. To consider the dependency of result components on the whole, we define the maximum gain that can be achieved when retrieving any subset of an ideal node's structurally related near-misses[10] (and the ideal node itself) as that of the ideal node's gain. Therefore, we define a dependency normalisation function, which ensures that the total score for any such set cannot exceed the maximum score achievable when the ideal node itself is retrieved. Take as an example the XML tree of Figure 3. Since the two ideal nodes represent the best elements for the user, a system returning these should be ranked above others. However, without dependency normalisation, a system that retrieved all the leaf nodes would achieve a better overall score as the total RV score for these nodes exceeds that of the ideal nodes. The following dependency normalisation function, rv_{norm}, safeguards against this by ensuring that for any $c_j \in S$, $rv(c_i) + \sum^S rv(c_j) \leq rv(c_{ideal})$ holds:

$$rv_{norm}(c_i) = min(rv(c_i), rv(c_{ideal}) - \sum^S rv(c_j)) \tag{8}$$

where c_{ideal} is the ideal node that is on the same relevant path as c_i, S is the set of nodes in the ideal node's sub-tree that have already been retrieved (before c_i).

5.2.3 *Considering overlap.* Due to the possible dependency relations among retrieval results, we need to consider the possible overlap of result elements with already seen components when calculating the relevance value of a given component. Our goal is to reward the retrieval of a relevant component only once. This reflects the viewpoint of a user for whom any already viewed components become irrelevant, and the value of a component seen in part is reduced.

To achieve this, different $f(\cdot)$ functions are applied depending on the kind of dependency that exist among result elements within the ranking. Here we only consider the dependency of a given component to already retrieved components (i.e. those earlier in the ranking). We have three possibilities: 1) a component may be returned for the first time (i.e. no related nodes have been seen by the user before), 2) it may have already been seen by the user in full (e.g. if a container section of the current paragraph result was returned at an earlier rank), 3) some parts of it may have been seen before (e.g. if a paragraph of the current section result has already been returned).

[10]By limiting ourselves to the full recall-base as the set of near-misses, we consider all ascendant and descendant nodes of an ideal node as its structurally related near-misses.

Based on this, we define $rv(c_i)$ for each of these cases:

$$rv(c_i) := \begin{cases} quant(assess(c_i)) & \text{if} \quad c_i \text{ has not yet been seen,} \\ (1-\alpha) \cdot quant(assess(c_i)) & \text{if} \quad c_i \text{ has been fully seen,} \\ \alpha \cdot \frac{\sum_{j=1}^{m} (rv(c_j) \cdot |c_j|)}{|c_i|} & \\ \quad +(1-\alpha) \cdot quant(assess(c_i)) & \text{if} \quad c_i \text{ has been partially seen before.} \end{cases} \tag{9}$$

where m is the number of c_i's child nodes and $|\cdot|$ is the length of an element (in characters or words). The $\alpha \in [0,1]$ weighting factor reflects a user's intolerance to being returned redundant components or component-parts. The higher the α value, the less the relevance value of a redundant relevant component.

According to the above, for a not-yet-seen component, the component's relevance value is only dependent on the component's quantised assessment value: $quant(assess(c_i))$.

For a component that has been already fully seen by the user, the component's quantised assessment value, $quant(assess(c_i))$, is weighted by $(1-\alpha)$, where α is a weighting factor introduced to reflect a user's intolerance to being returned redundant components or component-parts. For example, setting $\alpha = 1$, which represents a user who does not tolerate already viewed components, results in an RV score of 0 (due to $1-\alpha = 0$) for a fully seen component, reflecting that it represents no value to the user any more.

Finally, if a component has been seen only in part before, then its relevance value is calculated recursively based on the relevance value of its descendant nodes combined with its own quantised assessment value. The intolerance weighting factor of α is again used to modify the value attributed to already seen components. For example, using $\alpha = 1$ means that only not-yet-seen sub-components will be scored, while using $\alpha = 0$ will return the unmodified quantised score of the component regardless how much of it the user has seen already. An advantage of the proposed recursive method is that it does not assume that relevant information is uniformly distributed within a component, as done in [Goevert et al. 2005], which may still lead to the rewarding of redundant elements. Take for example a relevant section s_1, assessed as $(3,1)$, consisting of a single relevant paragraph, p_1 assessed as $(3,3)$, and nine irrelevant paragraphs, $p_2 \ldots p_{10}$, where all paragraphs are of equal length. Assuming uniform distribution of relevant content in s_1, the ranking $<p_1, s_1>$ would reward the retrieval of s_1 with a score of $9/10 \cdot quant(assess(s_1)) > 0$. The method proposed here, on the other hand, results in a score of 0 for s_1 (using $\alpha = 1$).

Alternative and more elaborate relevance value functions may be defined and employed within an evaluation framework, reflecting more complex user interaction models. For example, when calculating overlap the above heuristics provide a limited view where only explicit overlap observations are considered. It is possible, however, that users browsing from a given result may encounter other components, which may then overlap with other already seen results or with results further down the ranking. Such a model is being investigated in [Piwowarski et al. 2005]. For

the purpose of this study, our aim is not to arrive at such detailed user models and tuned metrics, but to define an encompassing evaluation framework which caters for the dependency considerations, but which is flexible enough so that particular models of user behaviour may be instantiated as seen fit by the evaluator.

We note that it is also possible to ignore overlap within the evaluation by simply setting $\alpha = 0$. We will refer to this mode of evaluation as "overlap=off", while the result-list dependent functions defined above as "overlap=on".

5.2.4 *The final gain value.* The final gain value of a result element in a ranked output list of an XML IR system, taking into account near-misses and overlaps, is given by the normalised relevance score of:

$$xG[i] := rv_{norm}(c_i) \tag{10}$$

where $rv_{norm}(c_i)$ is defined in Equation 8, $rv(c_i)$ is given in Equation 9.

For example, considering overlap and near-misses, the ranking $< sec[6]/p[1], sec[6], sec[6]/p[2] >$ accumulates a total score of $nxCG[3] = 1$ (after normalisation and using $quant_{sog}$).

5.3 A critique of the xCG and $nxCG$ metrics

A main criticism of cumulated gain based metrics is that they do not average well across topics [Kando et al. 2001] (in [Sakai 2004]). The reason for this is that as the total number of relevant documents differs across topics, so does the upperbound performance at fixed ranks. As a result, a reported performance value at a given rank could represent very different performance levels for different topics. For example, a score of $xCG[10] = 10$ may correspond to perfect performance for a topic with 10 relevant documents, each having a gain score of 1, but it may represent only a small portion of the total gain for another topic.

The above problem also applies to other, standard, rank-based IR measures, such as precision at fixed document cutoff (p@DCV) [Hull 1993]. An argument for rank-based measures, however, is that they are based on the evaluation viewpoint that a user is only willing to examine a fixed number of documents for any given query and hence compares queries on the basis of equivalent effort. Precision/recall-like measures rely on a different user model and assume that a certain level of recall must be obtained by every query to meet user satisfaction.

Another criticism of cumulated gain measures is that they only provide a recall component. A score of $xCG[10] = 6$ tells us nothing about how many non-relevant documents the user needs to read to accumulate a gain value of 6. It may be that all 10 ranks contain relevant documents (where the relevance score of each document may be 6/10), or that only 6 documents are relevant. Indeed, as [Järvelin and Kekäläinen 2002] point out, cumulated gain at a given rank is of little use by itself in most evaluation experiments.

On the other hand, the measure of normalised cumulated gain, and hence $nxCG$, have some attractive properties that also make it better suited for averaging across topics. Its advantage is that it is a measure of relative performance compared to an ideal case and hence incorporates a measure of upperbound performance within. This gives the measure a "precision-like" quality. For example, $nxCG[10] = 0.8$ means that up to rank 10 the user attained 80% of the achievable gain. Nevertheless, $nxCG$ also fails to inform about the number of non-relevant results within the

ranking.

Another issue with the normalised cumulated gain measure is that the ideal cumulated gain vector remains a constant value for all ranks $i \geq n$, where n is the number of relevant documents [Sakai 2004]. A consequence of this is that $nxCG$ at sufficiently large cutoffs cannot differentiate between different system performances. For example, a system that returns all relevant documents within the first 10 ranks and another system that requires the user to scan 100 ranks, will receive the same score of $nxCG[100] = 1$.

The $MAnxCG$ measure (Equation 6) provides a partial solution, but it also tends to cover up performance differences as the range grows sufficiently larger than the number of relevant documents.

An alternative solution has been suggested in [Sakai 2004] in the form of the following measures, where the explicit incorporation of the rank position in the denominator ensures that performance is calculated against an always increasing ideal value:

$$Q - measure = \frac{1}{R} \sum_{j=1}^{i} isrel(d_j) \frac{cbg(j)}{cig(j) + j} \tag{11}$$

where R is the total number of relevant documents, d_j is the document retrieved at rank j, $isrel(\cdot)$ is a binary function that returns 1 if the document is relevant (to any degree) and 0 otherwise. The function $cbg(\cdot)$ is a so-called cumulated bonus gain function, which is defined as $cbg(i) := bg(i) + cbg(i-1)$, where $bg(i) := g(i) + 1$ if $g(i) > 0$ and $bg(i) := 0$ otherwise, and $g(i)$ is the gain value at rank i. The function $cig(\cdot)$ is the cumulated bonus gain derived for the ideal vector (analogue to $cbg(\cdot)$).

$$R - measure = \frac{cbg(R)}{cbg(R) + R} \tag{12}$$

We implemented the extended versions of the above measures, adapted to XML through the definition of $g(i) := xG[i] = rv_{norm}(c_i)$. We refer to these extended versions as Q and R, and use them for comparison in the experiments of Section 6.

In summary, given the user-oriented nature of cumulated gain based measures [Kekäläinen 2005], we argue that $nxCG$ provides a useful measure of performance at low cutoffs, while it also suffers less than xCG from a statistical viewpoint.

5.4 Effort-precision: *ep*

The cumulated gain based measures described so far provide a recall-oriented view of effectiveness at fixed rank positions. This is illustrated in Figure 5: $nxCG$ is calculated by taking measurements on both the system's cumulated gain curve and the ideal ranking's cumulated gain curve along the vertical line drawn at rank i. Here, rank position is used as the control variable and cumulated gain (or relative cumulated gain) as the dependent variable.

Switching viewpoints, we may ask what is the amount of effort required of the user to reach a given level of cumulated gain when scanning a given ranking. Furthermore, we may wish to measure the amount of required effort compared to an ideal ranking. The horizontal line drawn at the cumulated gain value of r, shown in Figure 5, illustrates this view. Based on this, and analogue to the definition of

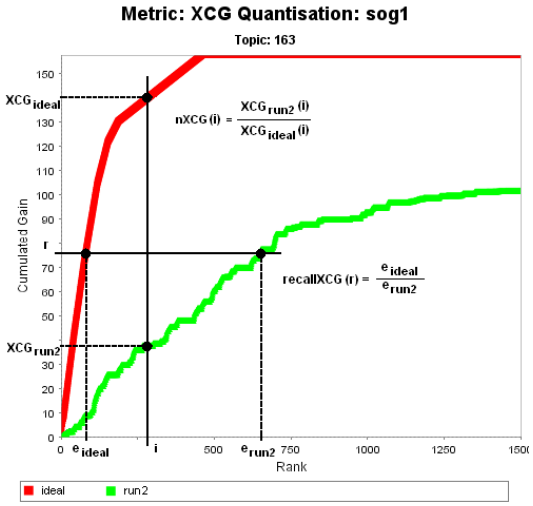


Fig. 5. Calculation of $nxCG$ and effort-precision (ep)

$nxCG$, we can define a precision-oriented XCG measure, effort-precision ep as:

$$ep[r] := \frac{i_{ideal}}{i_{run}} \tag{13}$$

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, where the user need to spend the minimum necessary effort to reach a given level of gain.

We chose to name the measure effort-precision to better describe it and also to differentiate it from an alternative definition of precision for cumulated gain, which was developed in [Kekäläinen and Järvelin 2002].

Instead of taking measurements at absolute cumulated gain values, we can calculate effort-precision, ep, at arbitrary recall points, where recall can be calculated as the cumulated gain value divided by the total achievable cumulated gain [Kekäläinen and Järvelin 2002]:

$$gr[i] := \frac{xCG[i]}{xCI[n]} = \frac{\sum_{j=1}^{i} G[j]}{\sum_{j=1}^{n} I[j]} \tag{14}$$

where n is the total number of relevant documents. The meaning of effort-precision at a given gain-recall value is 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 a given level of gain relative to the total gain that can be obtained.

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 the dependent variable. As with precision/recall, 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. $ep@0.1$. Given the nature of our xCG curve, we employ a simple linear interpolation method. An alternative technique is given in [Amati 2003].

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. For not retrieved relevant documents a precision score of 0 is assigned. We may also calculate an average over the interpolated effort-precision values, which we will refer to as $iMAep$. Analogue to recall/precision graphs, we may also plot effort-precision against gain-recall and obtain a detailed summary of a system's overall performance.

6. EVALUATION OF EVALUATION MEASURES

The evaluation of a metric requires a number of tests. Voorhees in [Voorhees 2003a] identifies two aspects to qualify an evaluation: fidelity and reliability. Fidelity reflects the extent to which an evaluation metric measures what it is intended to measure, while reliability is the extent to which the evaluation results can be trusted. In Section 6.1 we investigate fidelity by providing a walk-through analysis of the results of a simple experiment to give an insight into the behaviour of the different XCG metrics. In Section 6.2 we examine various aspects of reliability, such as what effect assessment variation has on the stability of the different metrics.

For all our experiments, we used the EvalJ evaluation package[11], which implements all XCG metrics and the inex-eval metric within a single Java project.

We tested the following metrics: $nxCG$ at various cutoffs, e.g. $nxCG[100]$, and $MAnxCG[1500]$ (or $MAnxCG$ for short) (Section 5.1), Q and R (Section 5.3), ep at standard recall points and $MAep$ (see Section 5.4). We also investigate the average of ep values at standard recall points: $iMAep$.

It is clear that different metrics measure different aspects of the system. Variations of $nxCG$ provide a more user-oriented view, while overall performance measures that take recall as the control variable follow a system-oriented view.

6.1 Fidelity

6.1.1 *What do they measure?.* To demonstrate the different behaviour of the various XCG metrics, we start with a simple experiment, where we evaluate simulated runs constructed from a single relevant XML tree, taken from the recall-base of topic 163. The XML tree chosen is that of the article file `co/2001/r7022.xml`, the full recall-base of which consists of the relevant nodes shown in Figure 3. We created four simulated runs from elements of the article's full and ideal recall-bases, see Table I. The result elements of each run have been sorted according to our chosen quantization function: $quant_{sog}$ (Equation 3).

We evaluated the four simulated runs using the various XCG metrics. Table II shows the results.

6.1.1.1 *Measures with document cutoffs.* Let us have a look at the results of $nxCG[DCV]$ first. These measures assume that the user is only prepared to scan a certain number of ranks in the output list and stop once this cutoff value is

[11] https://sourceforge.net/projects/evalj/

Table I. Simulated runs constructed from the recall-base of a single XML tree. For each result, its rank, XPath, assessment value pair (e, s) and its quantised value are shown.

`frb: #All relevant nodes in topic 163's assessments for co/2001/r7022.xml, sorted by SOG quantised value (see Figure 3).`		
1.	`/article[1]/bdy[1]/sec[6]`	$(3,3) \rightarrow 1$
2.	`/article[1]/bdy[1]/sec[4]/ip1[2]`	$(2,3) \rightarrow 0.9$
3.	`/article[1]/bdy[1]/sec[4]/p[1]`	$(2,3) \rightarrow 0.9$
4.	`/article[1]/bdy[1]/sec[6]/ip1[2]`	$(2,3) \rightarrow 0.9$
5.	`/article[1]/bdy[1]/sec[6]/p[1]`	$(2,3) \rightarrow 0.9$
6.	`/article[1]/bdy[1]/sec[6]/p[2]`	$(2,3) \rightarrow 0.9$
7.	`/article[1]/bdy[1]/sec[4]`	$(2,2) \rightarrow 0.5$
8.	`/article[1]`	$(3,1) \rightarrow 0.25$
9.	`/article[1]/bdy[1]`	$(3,1) \rightarrow 0.25$
10.	`/article[1]/bdy[1]/sec[4]/p[2]`	$(1,2) \rightarrow 0.25$
`ideal: #Elements of the ideal recall-base, selected topic 163's assessments for co/2001/r7022.xml, and sorted by SOG quantised value.`		
1.	`/article[1]/bdy[1]/sec[6]`	$(3,3) \rightarrow 1$
2.	`/article[1]/bdy[1]/sec[4]`	$(2,2) \rightarrow 0.5$
`reverse_ideal: #Elements of the ideal run above, but in reverse order.`		
1.	`/article[1]/bdy[1]/sec[4]`	$(2,2) \rightarrow 0.5$
2.	`/article[1]/bdy[1]/sec[6]`	$(3,3) \rightarrow 1$
`rel_leaves: #All relevant leaf nodes from topic 163's assessments for co/2001/r7022.xml, sorted by SOG quantised value.`		
1.	`/article[1]/bdy[1]/sec[6]/ip1[2]`	$(2,3) \rightarrow 0.9$
2.	`/article[1]/bdy[1]/sec[6]/p[1]`	$(2,3) \rightarrow 0.9$
3.	`/article[1]/bdy[1]/sec[6]/p[2]`	$(2,3) \rightarrow 0.9$
4.	`/article[1]/bdy[1]/sec[4]/ip1[2]`	$(2,3) \rightarrow 0.9$
5.	`/article[1]/bdy[1]/sec[4]/p[1]`	$(2,3) \rightarrow 0.9$
6.	`/article[1]/bdy[1]/sec[4]/p[2]`	$(1,2) \rightarrow 0.25$

Table II. Evaluation results for our four simulated runs

	nxCG at DCV:									
	1	2	3	4	5	10	25	50	100	1500
ideal	1	1	1	1	1	1	1	1	1	1
frb	1	1	1	1	1	1	1	1	1	1
reverse_ideal	0.5	1	1	1	1	1	1	1	1	1
rel_leaves	0.9	0.66	0.66	1	1	1	1	1	1	1

	Effort-precision at recall points:									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
ideal	1	1	1	1	1	1	1	1	1	1
frb	1	1	1	1	1	1	1	1	1	1
reverse_ideal	0.5	0.5	0.5	0.43	0.5	0.56	1	1	1	1
rel_leaves	0.9	0.9	0.9	0.9	0.9	0.9	0.46	0.47	0.49	0.5

	iMAep	MAep	Q	R	MAnxCG
ideal	1	1	1	1	1
frb	1	1	1	1	1
reverse_ideal	0.6991	0.75	0.875	1	1
rel_leaves	0.732	0.633	0.8751	0.857	0.9995

reached. The measure then reports a single score representing the relative gain the user has accumulated up to this point compared to the gain he/she could have achieved had he/she been scanning a perfect ranking. The reported score is a result of a single measurement taken along the ideal and actual xCG curves, see Figure 5, and does not take into account the order of the results within the scanned ranking. As a result, the measure reports e.g. the `ideal` and `reverse_ideal` runs as equal after rank 1, and all the runs as equal after rank 3, despite the fact that the rankings of `reverse_ideal` and `rel_leaves` are not perfect. The difference in the quality of the `ideal` and `reverse_ideal` rankings is only reflected at rank 1 and the difference between the `rel_leaves` and others can only be observed up to rank 3. At higher ranks the measure is unable to discriminate between differently performing rankings.

The rewarding of near-misses leads to the scores obtained, e.g., for the `rel_leaves` run at ranks 1, 2 and 4. At rank 1, the child node of the ideal `sec[6]` node is rewarded a score of 0.9, its quantised relevance value. At rank 2, the dependency normalisation leads to only an additional $1 - 0.9 = 0.1$ score for the retrieval of another child node of `sec[6]`, and at rank 3 the near-miss score is reduced to 0. At rank 4, again due to the dependency normalisation, the original quantised value of `sec[4]/ip1[2]` is reduced to the maximum ideal score of 0.5 (the quantised score of the ideal `sec[4]` node). The results obtained at ranks 1 and 4 demonstrate a disadvantage of the employed simple mechanism for scoring near-misses (i.e. based only on quantised score, not taking into account distance or length ratios): At rank 1, we only obtain a partial score for one near-miss, while at rank 4 we reward the full ideal score for another near-miss. This situation arises because $quant_{sog}(sec[4]) < quant_{sog}(sec[4]/ip1[2])$. As mentioned earlier, the problem can be easily overcome by employing alternative estimators for calculating the relevance value of near-misses. However, it remains an open question whether such an estimate would lead to differences in the evaluation results which would then warrant its need.

The effect of overlap on the scores can be observed on the `frb` run at ranks > 3. All results retrieved after rank 3 overlap with results retrieved at earlier ranks and are hence rewarded no additional score (although their quantised value is > 0).

Overall, $nxCG$ at rank 1 reports that `ideal` $\equiv$ `frb` $\succeq$ `rel_leaves` $\succ$ `reverse_ideal`, as the `ideal` and `frb` runs both return the same, most relevant result, `rel_leaves` retrieves a near-miss of the best result, while `reverse_ideal` returns a less relevant element. At ranks 2 and 3, we get `ideal` $\equiv$ `frb` $\equiv$ `reverse_ideal` $\succ$ `rel_leaves`. The preference of the `reverse_ideal` run over the `rel_leaves` reflects that with the former the user has found all relevant content, while the latter requires further scanning of the ranking. At ranks beyond 3, `ideal` $\equiv$ `frb` $\equiv$ `reverse_ideal` $\equiv$ `rel_leaves` is reported, reflecting that the user is indifferent about their performance after the maximum relative gain has been obtained.

6.1.1.2 *Overall performance measures.* Next, we examine the results of those metrics that are based on the assumption that the user stops only after all relevant elements have been found (or when the collection is exhausted). The metrics *MAep*, *iMAep*, Q, and R are based on this stopping rule. Both *MAep* and Q report performances measured at rank positions where a relevant document is found, which

is then averaged over the total number of relevant documents in the ideal recall-base. R is a measurement taken at a single rank position equal to the total number of relevant documents in the ideal recall-base. $iMAep$ is the average of ep scores over the standard recall points. Apart from R, all measures are affected by the quality of the ordering of result elements in a ranking.

All three averaged measures ($MAep$, $iMAep$, Q) report the `ideal` and `frb` runs as equal. This is again a consequence of our method for scoring near-misses, where the element of `sec[4]/ip1[2]` at rank 2 of the `frb` run is rewarded the full score of its ideal parent node. I.e. for our user this near-miss is as good as the ideal node. The metrics also agree that both `reverse_ideal` and `rel_leaves` perform worse than the `ideal` and `frb` runs, but they disagree about their relative order. According to $MAep$ the `reverse_ideal` run is better, while $iMAep$ concludes the opposite and Q reports that the two runs have near-equivalent performance. So which one is correct?

Let us examine the two runs. The `reverse_ideal` run delivers all relevant content for the minimum required effort (user needs to scan only 2 ranks), although the ordering of the results is not perfect. On the other hand, the `rel_leaves`, in this case, leads the user first to the most relevant element and then to the less relevant node (i.e. correct ordering), but it does require the most amount of effort from the user to access all relevant content (the user needs to scan 4 ranks, twice the effort of the `reverse_ideal` run).

The $MAep$ measure ranks the `reverse_ideal` run above the `rel_leaves` run, reflecting that on average the user needs to spend less effort when scanning the output of `reverse_ideal` to achieve the same level of gain. The interpretation of Q is not so obvious as it combines recall with precision-like qualities in a single measure. It is however still a primarily recall-oriented measure, and since the `rel_leaves` run[12] achieves higher recall at earlier ranks, it is judged to perform on the same level and even slightly better than the `reverse_ideal` run[13]. The $iMAep$ measure contradicts the result of $MAep$, due to the error introduced by the interpolation/extrapolation process (whereby the ep value of 0.9 at the natural recall point of 0.6 is extrapolated to all lower recall values).

The R measure agrees with $nxCG[2]$ in the relative ranking of systems, although the actual score for the `rel_leaves` run is different due to the modifying factor of the "bonus" gain. The measure of $MAnxCG[1500]$ reports similar preferences to the R measure, but from all averaged measures it is the least able to detect any differences between the systems. It also has a major limitation in that it requires a predefined rank threshold that acts as the upper value of its range. In our example, we set 1500 as the threshold. The choice of such a high threshold is not motivated by the user-oriented character of the metric, but it represents a possible choice when a common threshold is to be employed over several queries and systems. The high value in the case of our toy example query means that any performance differences at low ranks are simply covered up by the averaging process. Appropriate selection of a threshold value can however lead to a performance results that are reflective

[12] $Q_{rel\_leaves} = \frac{cbg(1)}{cig(1)+1} + \frac{cbg(2)}{cig(2)+2} + \frac{cbg(4)}{cig(4)+4} = \frac{0.9+1}{1+1} + \frac{(0.9+1)+(0.1+1)}{1.5+2} + \frac{(0.9+1)+(0.1+1)+(0.5+1)}{1.5+4}$

[13] $Q_{reverse\_ideal} = \frac{cbg(1)}{cig(1)+1} + \frac{cbg(2)}{cig(2)+2} = \frac{0.5+1}{1+1} + \frac{(0.5+1)+(1+1)}{1.5+2}$

of the user's expectation. A more sensitive $MAnxCG[2]$ for example leads to the system ranking[14] of `ideal` ≡ `frb` ≻ `rel_leaves` ≻ `reverse_ideal`.

Finally, effort-precision (*ep*) at standard recall points reflects the amount of relative effort required compared to an ideal scenario in order to reach a predefined level of gain. As mentioned above, in the case of sparse data, the measure does suffer from problems associated with interpolation/extrapolation. In addition, like *nxCG*, an individual *ep* value is only the result of a single measurement taken along the ideal and actual *xCG* curves, see Figure 5, and does not take into account the order of the results within the scanned output ranking. However, analogue to standard precision/recall graphs, a series of *ep* and *gr* values can provide a complete picture of system performances.

In summary, effort-precision/gain-recall graphs and the *MAep* metric seem to provide the most informative and discriminative measures for system-oriented experiments, while the *nxCG* measures lack of discriminatory power at higher ranks, which makes them less attractive for system-oriented evaluations. At lower cutoff values, *nxCG* does provide a useful user-oriented measure.

6.1.2 *Adding non-relevant results.* In this section we examine how the different measures react when increasing number of non-relevant elements are added to an initial ranking of relevant components. We took our simulated `ideal` run (Table I) as our initial ranking and inserted 1, 2, 3, 4, 5, 10, 50, 100, 500 and 1000 non-relevant results between ranks 1 and 2 (Figure 6a), and before rank 1 (Figure 6b). We refer to the former as the "insert" case and the latter as the "precede" case.

The general intuition is that performance scores should decrease as more non-relevant elements are returned to the user. This tendency is confirmed for most metrics, although both *ep*@0.1 (constant 1) and *R* (constant 0.5714) remains unchanged in Figure 6a. In the case of *ep*@0.1 (effort-precision at 10% gain-recall) this is reasonable as the scope of the metric is a constant small window, limited to the first rank where the first relevant result is returned. Similarly, the scope of *R* only stretches to the first two ranks, since we only have a total of two relevant elements in our toy collection. The behaviour of *iMAep*, *MAep*, and *Q* shows high correlation, with slowing change in the reported performance score as more additional non-relevant results are added. This slope of the curves illustrates the effect of outliers (e.g. the last relevant document found) on the evaluation. The *MAnxCG* measure stands out from all other and is the most resiliant to any change. This is due to the fact that it is an average of averaged cumulated gain values, where the high threshold (1500) already contributes to its reduced responsiveness.

The difference between the performance scores across Figures 6a and 6b shows the effect that finding relevant or non-relevant results at early ranks have on the evaluation. Note that the difference in our case is perhaps unusually high at due to the extreme properties of the toy collection (e.g. total gain of 1.5). The two most stable measures are *ep*@1 (effort-precision at 100% gain-recall) and *MAnxCG*. For *Q*, *MAep* and *iMAep* there is a 25%, 30% and 48% drop in performance, respectively, when the first relevant document is moved from rank 1 to rank 2. This behaviour is consistent with standard IR measures, such as precision/recall,

[14] A system ranking is an ordered list of runs sorted by decreasing performance score.

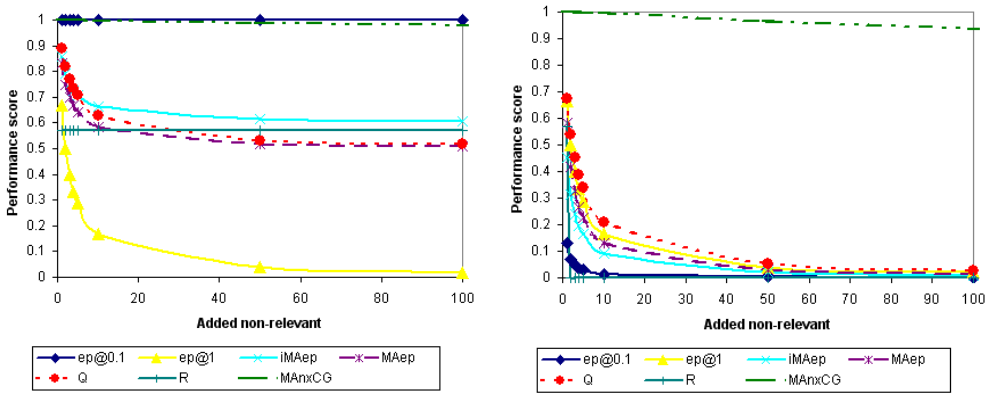


Fig. 6. The effect of inserting non-relevant results into the ranking: a) between ranks 1 and 2; b) before rank 1.

applied on the same toy collection, which results in the same 30% decrease in performance score as *MAep*.

From the figures, it is also clear that some measures are more sensitive or discriminative than others. It is easy to see, and is not surprising, that measures such as p@0.1 are highly sensitive to small changes, while overall performance measures, e.g. *MAep*, are more robust, but can therefore be less informative.

Similarly to the metrics with limited scope, the behaviour of the $nxCG[DCV]$ measures is rather crude (not shown). With the exception of $nxCG[1]$ and $nxCG[2]$, all $nxCG[DCV]$ metrics produce a step function, where the jump shifts further to the right along the X axis as the number of inserted non-relevant results increases. For example, in the "insert" case, the curve for $nxCG[3]$ jumps from the score of 1 to 0.66 after two non-relevant results have been inserted, while $nxCG[5]$ makes the same jump after 4 non-relevant results have been inserted. For the "precede" case, the step function first jumps from 1 to 0.66 and then to 0. Like ep@0.1, $nxCG[1]$ remains a constant value throughout for both cases (a constant 1 in the "insert" case and 0 in the "precede" case), while $nxCG[2]$ has a constant value of 0.66 in the "insert" case, and has a single jump from 0.66 to 0 after one non-relevant result is added in the "precede" case.

The conclusions we can draw confirm that all measures which take a single measurement or have limited scope are highly sensitive to small changes within their scope. Most of our user-oriented measures belong to this category and hence are expected to have larger variations across systems and queries. Averaged measures are much more stable, but are affected by the early ordering of results as well as outliers. These findings correlate with other studies on standard measures of retrieval performance [Buckley and Voorhees 2000; Sanderson and Zobel 2005] and show in particular that *MAep* has very similar characteristics to the standard measure of mean average precision. In addition, the high correlation between *MAep* and Q reinforces that the two metrics are comparable.

6.2 Stability testing

In this section we investigate the effects that i) varying relevance assessments and ii) varying topic set size have on the outcome of our performance measures.

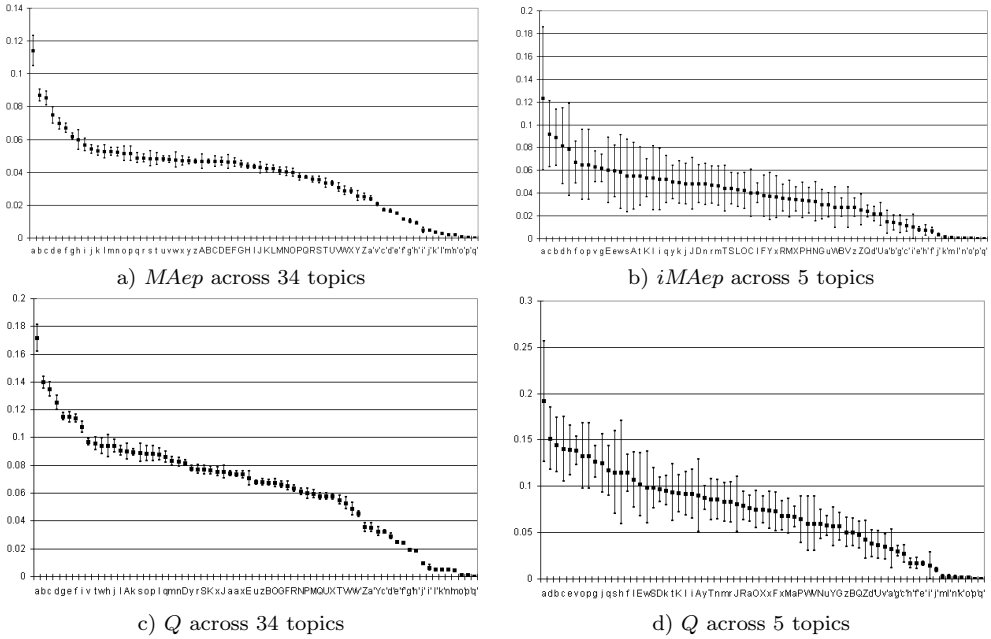
6.2.1 *Effects of varying relevance assessments.* Relevance judgements play a fundamental role in the evaluation of retrieval performance. Due to its dynamic and subjective nature, relevance and its assessment have been the target of much scrutiny in IR research [Borlund 2003]. Several studies concluded that relevance assessors rarely agree [Voorhees 2000; Järvelin and Kekäläinen 2000; Harter 1996; Wallis and Thom 1996; Schamber 1994; Burgin 1992; Lesk and Salton 1969], reporting rates of agreement between assessors' judgements varying between 20-85%, with typical values of around 40% [Voorhees 2000]. From an evaluation viewpoint, an important question is not the actual rate of assessor agreement, but how variations in relevance judgements affect the evaluation results [Burgin 1992; Voorhees 2000]. At INEX[15], the rate of assessor agreement is only 23.7%, when measured at article level and based on binary relevance. At section level the rate drops to 18.5% and at paragraph level it is a mere 10.3%. Exact agreement between all assessed elements, where judges need to agree both on the exact granularity of each relevant fragment and their exhaustivity and specificity degrees, is only around 2%. Given such a low level of agreement, it is even more important to examine how evaluation results change with differences in assessments.

6.2.1.1 *Variation of performance across different qrels test sets*[16]. To test the stability of our metrics, we evaluate system runs using different sets of qrels (selected from multiple judgements on the same topic set) and examine how the reported absolute performance scores change from one qrels test set to the next [Voorhees 2000]. The idea is that stable measures will be least affected by changes in the relevance assessments.

Unfortunately, unlike the study of [Voorhees 2000], we only have a rather limited number of duplicate judgements available: only 5 CO topics, with topic numbers 165, 169, 173, 175, 201 have been judged by two different assessors. This gives us $2^5 = 32$ combinations and two choices: evaluate performance across the set of 5 topics only, or on the full set of 34 topics (i.e. using a static set of 29 topics plus variable 5 topics). The first option leads to an independent result set, but the low number of topics is likely to lead to higher error rates [Voorhees and Buckley 2002] (also see Section 6.2.2). The second option should result in more stable, but somewhat optimistic results due to the use of non-independent topics. We evaluate both options and compare their results with the aim to obtain an upper and lower bound, which should at least provide an indication of the actual effect of assessment variations.

We evaluated all 69 submitted INEX CO runs using $nxCG[DCV]$ with cutoff values of 5, 10, 15, 25, 50, 100, 500, 1000 and 1500, $MAnxCG[1500]$, $iMAep$, $MAep$, Q and R. For each system, we calculated its performance score with each metric for each of the $2^5 = 32$ qrels test set combinations, with 34 topics and again

[15]The agreement rates are based on duplicate assessments derived for only 5 INEX'04 CO topics.
[16]*Qrels* stands for query relevance set and it is the concatenation of one judgement set per topic [Voorhees 2000]. A qrels test set is a set of qrels, containing one judgement set for each topic.



a) *MAep* across 34 topics

b) *iMAep* across 5 topics

c) Q across 34 topics

d) Q across 5 topics

Fig. 7. The effect of assessments variations on performance scores. Mean average performance scores of *MAep* and Q metrics computer for all 69 INEX'04 CO runs over different topic sets is plotted. The run names have been replaced with letters from the alphabet. Error bars indicate the maximum and minimum performance scores obtained.

with 5 topics.

Figure 7 shows the obtained absolute performance scores for the *MAep* and Q measures, averaged over the different qrels test sets containing 34 and 5 topics, respectively. The runs have been arranged in decreasing order of *MAep* score. The error bars indicate the maximum and minimum scores (averaged over all topics in the test set) obtained for a given system over the sample.

We can make a number of observations from the graphs. Both metrics demonstrate that changes in assessments can lead to different system rankings. Clearly, using less topics leads to large variations (taller error bars) where changes in the assessment set have an increased impact. Using more topics leads to more stable performance scores, where the effect of assessment variation is reduced. The graphs also show that changes in relevance assessments influence the better performing systems more, with the size of error bars generally growing from right to left[17]. Both the *MAep* and Q measures have similar levels of tolerance to varying assessments, although they don't exactly agree about the relative ranking of systems. On closer inspection, Q shows to be less sensitive to assessments variations, with average differences between the maximum and minimum scores of 0.0058 and 0.04 (representing 9% and 55% of the average system score) for 34 topics and 5 topics respectively, compared to 0.0046 and 0.03 (12% and 82%) for *MAep*. However,

[17]Similar findings have been reported in a study focusing on the Q-measure in a traditional IR setting [Sakai 2005]

these differences only reflect variation in absolute performance scores based on different qrels test sets and do not indicate changes in the relative ranking of systems. Typically, performance scores increase or descrease in unison across systems with different test sets. I.e. when one system's score increases with a given test set, it is likely that another system's score will also increase (hence the chance of a swap in a system ranking cannot be estimated simply by the difference in the maximum and minimum scores).

For the remaining metrics, we obtained similar graphs to that of Figure 7, but demonstrating slightly higher levels of sensitivity. A general trend for the $nxCG$ measures is that lower document cutoffs lead to taller error bars.

6.2.1.2 *Correlation of system rankings based on different qrels test sets.* To quantify the effect of varying assessments on the resulting system rankings, we measure the association between the different system rankings obtained for the different sets of qrels test sets using Kendall's τ [Conover 1980]. The basic premise is that more stable measures will produce highly correlating system rankings despite variations in the assessments.

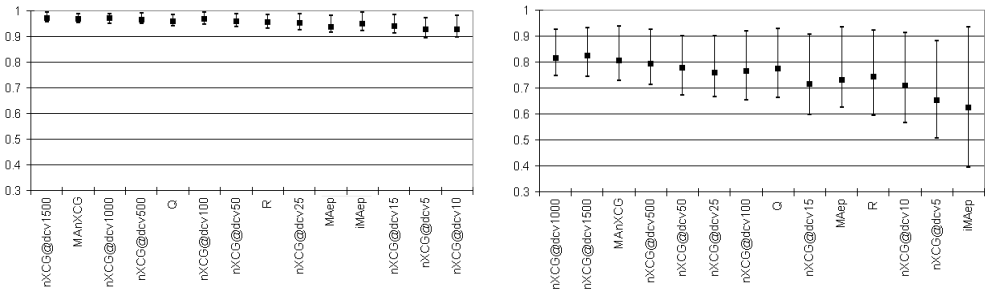
The correlation measure of Kendall's τ is a nonparametric measure of the agreement between two rankings, which is often used in similar experiments, e.g. [Voorhees 2000]. It computes the distance between two rankings as the minimum pair-wise adjacent swaps necessary to turn one ranking into the other. The distance is normalised by the number of items being ranked such that two identical rankings produce a correlation of 1 and two rankings that are a perfect inverse of each other produces a score of -1. The expected correlation of two rankings chosen at random is 0. Previous work has considered all rankings with correlations greater than 0.9 as equivalent and rankings with correlation less than 0.8 as containing noticeable differences [Voorhees 2001].

Table III lists the mean average, maximum and minimum correlations between the system rankings produced by the original qrels test set (containing the official assessments) and the rankings produced by the other 31 test sets, for both topic set sizes of 34 and 5. The correlation figures for *MAep* and Q indicate that *MAep* is in fact more sensitive and does produce larger variations in system rankings than Q. The reported average level of correlation for *MAep* over 34 topics is similar to that reported in [Voorhees 2000], where a mean correlation of 0.9380 (max of 0.9962, min of 0.8712) was observed. This is very encouraging given that we only used 34 topics (compared to their 49, and hence we have higher error rates) and especially since the agreement between judges is much lower in INEX. However, our results do represent an optimistic score, due to the dependency that exists among the topic sets. On the other hand, the correlation values obtained for the topic sets containing only 5 topics provides a pessimistic estimate due to the expected larger variations associated with the small sample size. Without additional data, it is difficult to draw a final conclusion here, but the results thus far do demonstrate that the stability of these metrics is comparable to the stability of the standard measure of mean average precision.

The calculated average, maximum and minimum correlation statistics for all metrics are shown in Figure 8. The metrics have been ordered by the difference between maximum and minimum correlation, with smallest differences on the left.

metric	34 topics			5 topics		
	avg	max	min	avg	max	min
MAep	0.93702	0.98038	0.91642	0.73278	0.933385	0.62385
Q	0.95868	0.98465	0.942	0.77436	0.92872	0.66396

Table III. Mean average, maximum and minimum Kendall correlation of system rankings produced by the performance measures of *MAep* and *Q* for 69 INEX CO runs using 32 different topic sets. The correlation is measured between the original and the other 31 topic sets.



a) 34 topics b) 5 topics

Fig. 8. The effect of variations in assessments. Mean average correlation between system rankings produced by different metrics. Error bars indicate maximum and minimum correlation scores.

Not unexpectedly, *MAnxCG* and *nxCG* at high cutoff values are the most robust measures, while *nxCG* at low cutoffs and *iMAep* are the most sensitive. *Q*, *R*, *MAep*, and *nxCG* at middle range cutoffs provide stable, but still discriminative measures.

6.2.1.3 *Estimating the error rate.* In order to quantify the effect of varying assessments on the evaluation, we next investigate the error rate associated with a measure's ability in deciding if one system is better than another. We use the notation $A \succ B$ to mean the preference relation between systems A and B that system A performs better than system B. Following [Voorhees 2003b; Buckley and Voorhees 2000], we compute the error rate of a metric by counting how often it contradicts its conclusion in judging a system better than another when we vary the topic sets used in the evaluation.

The method is as follows: For all pairs of runs A and B, we count the number of qrels test sets for which $A \succ B$, $B \succ A$, or $A = B$, where two runs are considered to have equal performance if the difference between their scores was less than 5% of the larger score. The error rate of a performance metric is defined as the total number of incorrect decisions made, calculated as $min(\sum(|A \succ B|), \sum(|B \succ A|))$, divided by the total number of comparisons:

$$ErrorRate := \frac{min(\sum |A \succ B|, \sum |B \succ A|)}{\sum |A \succ B| + \sum |B \succ A| + \sum |B = A|} \tag{15}$$

The proportion of ties, indicating a measure's discrimination power, is calculated as the sum over all run pairs of $|A = B|$, divided by the total number of comparisons:

$$ErrorRate := \frac{\sum |B = A|}{\sum |A \succ B| + \sum |B \succ A| + \sum |B = A|} \tag{16}$$

We use the evaluation results we obtained in the previous sections for all 69 INEX runs for our 64 qrels test sets (32 sets of size 34 and 32 sets of size 5). This gives us a total of $(32 \cdot ((69 \cdot 68)/2) = 75072$ comparisons for both the 34 and 5 topic sized tests.

Table IV shows the calculated error rates and proportion of ties for the different measures. The error rates for test sets with topic set size 34 are much lower, e.g. 0.01 for $nxCG[1000]$ or 0.4% for $MAep$, than those reported in [Voorhees 2003b], e.g. 1.4% for mean average precision (for 1000 test sets each containing 50 topics). This would suggest that the XCG metrics are very stable, but again we must remember that the figures shown here represent a rather optimistic estimate due to the dependecy among topics in the test sets. On the other hand, the error rates for the test sets containing the 5 independent topics represents a pessimistic estimate, with values varying between 3.32% and 13.45%, due to the additional error associated with the small size of the topic sets. The actual error that we may associate with a given metric would be somewhere between the two error rates calculated here. For example, based on 30 independent topics the chance that the $MAep$ metric leads to an incorrect preference relation between two systems (e.g. concludes that $A \succ B$ when $B \succ A$ is true) is likely to be more than 0.4% and less than 7.64%.

Comparing the metrics to each other, we can see that $MAep$ (0.4% and 7.64% for 34 and 5 topics) has a comparable, but slightly higher error rate than Q (0.12% and 5,69%, respectively), confirming that it is a similarly stable measure, but slightly more sensitive to assessment changes. $MAnxCG$, $nxCG[100]$ and $nxCG[1000]$ are the most robust measures with th elowest error rates (e.g. 0.01% and 3.32% for $nxCG[1000]$ with 34 and 5 topics). The $iMAep$ measure is the most instable, with the highest error rate of 13.45% when measured on independent test sets.

Looking at the proportion of ties, most metrics produce comparable results, showing that they are able to discriminate systems to the same level. While Q is found to be the most discriminating based on the dependent test sets (only 5.96% ties), $iMAep$ shows the highest level of discrimation power for the independent test sets (only 2.79% ties, although this has also led to the higher error rate of 13.45%). $MAnxCG$ and $nxCG[1000]$ have the worst rates for ability to discriminate between systems (above 8% for all test sets).

The dual nature of error rates and proportion of ties has already been noted in [Voorhees 2003b] and can also be observed here. Typically, as the discriminative power of a metric decreases, its error rate increases. This is due to the fact that as more decisions need to be made regarding the direction of a preference relation between two systems, the more likely that some of them will be incorrect, especially since fewer ties imply finer decisions.

6.2.1.4 *Summary of our findings.* In conclusion, the results from our tests are encouraging. We found that our overall performance measures, such as $MAep$, are stable but still suitably discriminating measures. Our calculated lower bound for error is (expectedly, due to the limited set of topics used) magnitudes below the error rates reported in literature for established measures like mean average precision. Our upper bound error rate provides a pessimistic estimate due to the additional error associated with the low sample size. Despite this, the median upper

metric	Test sets with 34 topics		Test sets with 5 topics	
	Error rate (%)	Ties (%)	Error rate (%)	Ties (%)
MAep	0.40	7.61	7.64	4.31
Q	0.12	5.96	5.69	4.81
R	0.12	7.36	6.50	5.82
MAnxCG	0.04	8.71	4.02	8.04
iMAep	0.42	5.49	13.45	2.79
nxCG[10]	0.44	7.99	7.18	4.79
nxCG[100]	0.07	7.29	6.37	6.36
nxCG[1000]	0.01	8.09	3.32	8.21

Table IV. Error rate and proportion of ties for different measures over the 32 topic sets of topic set size 34 and the 32 topic sets of size 5.

bound error rate of 6.4% (over the metrics included in Table IV) is a very positive result.

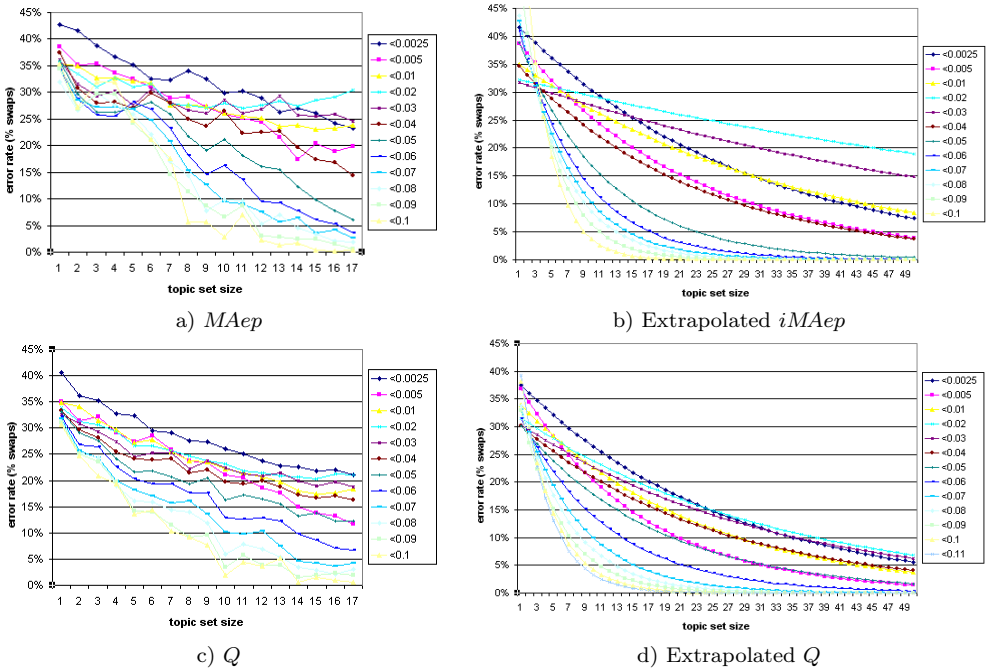
Comparing *MAep* to the *Q* measure, which has been shown to have statistically attractive properties (in a traditional IR setting) [Sakai 2005], showed that the two measures have similar levels of tolerance to varying assessments, although they didn't exactly agree about the relative ranking of systems. On closer inspection, *Q* was shown to be less sensitive to assessments variations, while both measures produced comparable results regarding their discrimination power.

For the other XCG metrics, we found that *MAnxCG* and *nxCG* at high cutoff values proved the most robust measures, while *nxCG* at low cutoffs and *iMAep* are the most sensitive. *Q*, *R*, *MAep*, and *nxCG* at middle range cutoffs provide stable, but still discriminative measures.

6.2.2 *Effects of varying topic set size.* We follow the methodology of [Voorhees and Buckley 2002; Sanderson and Zobel 2005] to examine how the size of the topic set affects the absolute difference in performance scores between two systems, using the various XCG metrics. The core of the procedure is to evaluate our system runs on two disjoint topic sets of equal size and examine the associated error rates as we vary the size of the topic sets in the evaluation. An error occurs when the direction of a preference relation between two systems is reversed from one test set to the other (i.e. $A \succ B$ holds for one of the first set of topics and $B \succ A$ is true for the second set). We count the number of times such a swap occurs separately for different levels of absolute performance differences d. We distinguish 23 performance difference bins: $d < 0.0025$, $d \in [0.0025, 0.005)$, $d \in [0.005, 0.01)$, $d \in [0.01, 0.02)$, ..., $d \in [0.19, 0.2)$ and $d \geq 0.2$.

We use the original CO topic set (official assessment set), containing 34 topics, and the top 75% of the 69 INEX CO runs. For all the metrics, the set of the bottom 25% runs were, with the exception of one or two swaps, always the same. For each metric, we perform 100 different trials using different combinations of topics in the two topic sets in order to increase the sample size. For each trial, we randomly select two disjoint sets of topics from the original topic set, where the size of the topic set is varied from 1 to 17. The total number of pair-wise comparisons for each trial is $51 \cdot 50 \cdot 100/2 = 127500$.

Figures 9a and 9c show the computed error rates against topic set size for the *MAep* and *Q* metrics, respectively. Due to the spareness of data in larger error



a) *MAep* b) Extrapolated *iMAep*

c) *Q* d) Extrapolated *Q*

Fig. 9. The effect of topic set size on evaluation error for the *MAep* and *Q* metrics.

bins, we only show results for the first 12 bins. For all topic set sizes, at least 94% of all comparisons were distributed into these bins, and over 99% for topic set sizes of 7 or over.

The meaning of an error rate of e.g. 25% is that when comparing two systems whose score difference is < 0.01, then out of a 100 different sets of topics of size 17, on average we can expect that 25 of the test sets will favour one system and 75 the other.

Compared to the results obtained in [Voorhees and Buckley 2002; Sanderson and Zobel 2005] for mean average precision, our curves are less smooth (due to fewer systems being compared on less topics), but are similar in shape with the error rate decreasing as topic set size increases. An interesting difference is the faster slope of our curves compared to theirs. For example, the error rate of *MAep* for the bin < 0.01 at 17 topics is just below 25%, while [Voorhees and Buckley 2002] reports a rate of around 40% for the standard IR metric. Similar tendencies apply to all other bins.

An explanation of this relatively high stability of our metrics is that the size of the bins in our case represent different relative performance differences to that in [Voorhees and Buckley 2002]. There the top performance score is 0.4, where an absolute change of 0.01 represents 2.5% of the best score. However, the average *MAep* score for our INEX data is only 0.05 with a standard deviation of 0.02, and the highest score is only 0.107. Therefore, the bin of absolute difference of 0.01 in our case represents 9.3% of the best score (and 20% of the average score). For Q, the average is 0.083, standard deviation is 0.036 and the best score is 0.1639.

The absolute difference of 0.01 represents 6.1% of the best score and 12% of the average score. So the same absolute bin sizes in our case actually cover larger relative differences. The smallest bin of < 0.0025 represents a comparable 2.3% relative score of best performance to the experiments in [Voorhees and Buckley 2002]. A comparison of our first bin's error rate to their < 0.01 bin, however, only confirms that our error rates are indeed lower than those for mean average precision. We hence conjecture that in INEX system performances converge faster with each additional topic. This may be because INEX topics may have a higher discrimination power or because different systems work very differently for different topics. We found evidence to support both claims, but it is not clear how the effect of either could be quantified for this purpose.

Comparing *MAep* and Q, we can see that Q has slightly better statistical properties with smoother curves, leading to lower error rates, especially for the bins of < 0.04, < 0.05 and < 0.06. Aside from these, the error rates are comparable for all bins and topic set sizes. For example, for the topic set size of 17, the error rate of an absolute difference of < 0.0025 is 24% for *MAep* and 21% for Q, while the error rate of a difference between 0.0025 and 0.005 is 18% *MAep* and 13% for Q.

The difference between the two metrics is, however, emphasised by the extrapolated error rates[18], shown in Figures 9b and 9d. At projected 50 topics, the error rate for an absolute score difference of < 0.0025 is 7.4% for *MAep* and 5.6% for Q. A somewhat odd result is the relatively slow linear slope of the error rates for bins < 0.02 and < 0.03 for both metrics but in particular for *MAep*. This could be a result of some anomalies in the random process of topic selection where a particular subset of topics, e.g. those with relatively small recall-base, may have been picked more frequently, leading to coarser and more varied performance results. We intend to further investigate this phenomena in the future by conducting experiments on subsets of the recall-base with varying properties.

A typical error rate of interest in IR experiments is 5%. We want to know what performance score difference is necessary between two systems to have a 95% confidence that one is better than the other. For *MAep* using a set of 17 topics, we would need an absolute difference of > 0.05 (relative difference of 46.7%). The situation is not much better at 34 topics, based on projected error rates, and at 50 topics we need 0.005 absolute difference (or 4.67% relative difference). For Q, we actually need an even higher absolute difference of > 0.06 to reach a 5% error rate at 17 topics. A difference of 0.005 is however enough already at 34 topics.

These results are comparable to the findings of [Voorhees and Buckley 2002]. There, for mean average precision to have an error rate of 5% when using 25 topics required an absolute difference of 0.08 (relative 20%) and when using 50 topics it needed a difference of 0.051 (relative 12.75%) between two systems. While the relative difference required in INEX is higher than in TREC when using fewer topics, it reduces to a lower required difference when using larger topic set sizes.

Very much similar results were obtained for all other metrics, with the exception that all performance differences were more varied and filled all 23 bins and the slope of the curves were somewhat slower. The only trend worth mentioning is the steady

[18]We first used a smoothing function on the original data before extrapolating it using the GROWTH function of Ms Excel.

decrease of the error rates for increasing cutoff values of $nxCG$. For $nxCG[1000]$ (the largest cutoff evaluated) at 17 topics, the average error rate over all bins is 7%, with the highest of 16% for the bin of < 0.03. At the projected 50 topics, all error rates have reduced to below 5%.

6.3 Comparison with other measures

In this section we compare our XCG metrics to the official metric of INEX (inex-eval) and the set-oriented measure of *overlap* that is also reported in INEX, which calculates the ratio of overlapping result elements in a system's output.

The inex-eval metric [Gövert and Kazai 2003] applies the measure of *precall* [Raghavan et al. 1989] to document components and computes the probability $P(rel|retr)$ that a component viewed by the user is relevant:

$$P(rel|retr)(x) : \frac{x \cdot n}{x \cdot n + esl_{x \cdot n}} \quad = \quad \frac{x \cdot n}{x \cdot n + j + \frac{s \cdot i}{r+1}} \tag{17}$$

where $esl_{x \cdot n}$ denotes the *expected search length* [Cooper 1968], i.e. the expected number of non-relevant elements retrieved until an arbitrary recall point x is reached, and n is the total number of relevant components with respect to a given topic. In $esl_{x \cdot n}$, let l denote the rank from which the $x \cdot n$th relevant component is drawn. Then j is the score of non-relevant information within the ranks before rank l, s is the relevant score to be taken from rank l, and r and i are the relevant and non-relevant scores in rank l, respectively.

Table V shows the obtained correlation results between system rankings produced by the different metrics based on all 69 INEX'04 CO system runs and 34 CO topics, with setting "overlap=off" and "overlap=on" for the XCG metrics.

The results show that when overlap is ignored by the XCG metrics (overlap=off), they provide system rankings that highly correlate with the output of the inex-eval metric. In particular the measure of mean average effort-precision (*MAep*) shows an almost perfect correlation. This suggests that the measure of the probability of relevance as adopted in the inex-eval metric measures similar system characteristics to that of effort-precision. Other metrics follow a similar trend, which is not surprising given that they highly correlate between each other. The correlation with the *overlap* measure is similar to the level of correlation between inex-eval and *overlap*. The lower levels of correlation here suggest that the level of overlap in a run is not a sufficient estimator of system performance.

When overlap is considered in the evaluation (overlap=on), the XCG metrics produce very different system ranking from inex-eval. This means that systems ranked better than others by one metric are often actually judged worse by the other metric. This is expected as over half of the system runs contain high ratios of overlapping elements. The average ratio of overlap over the 69 runs is nearly 38% with a maximum of 82%. Since overlapping elements may lead to increased performance scores with inex-eval, but are only rewarded once with XCG, it is eminent that the two types of metrics will result in different preference relations between systems. The even lower level of correlation between the XCG metrics and *overlap* suggests that a low level of overlap in a run does not guarantee a high performance score with the XCG metrics, but systems must also (and more importantly) be able to locate relevant content in the first place.

	MAep	*Q*	*R*	*MAnxCG*	inex-eval	*overlap*
			overlap=off			
inex-eval	0.995	0.985	0.963	0.956	1	0.825
overlap	0.808	0.851	0.879	0.877	0.825	1
			overlap=on			
inex-eval	0.377	0.391	0.412	0.469	1	0.825
overlap	0.238	0.294	0.343	0.388	0.825	1

Table V. Correlation between system rankings produced by different metrics, for both settings of "overlap=off" and "overlap=on" for the XCG metrics. Inex-eval is the official metric of INEX'04 and overlap is a set-based measure - also reported at INEX - that calculates the ratio of overlapping result elements in a system's output.

7. CONCLUSIONS AND FUTURE WORK

Evaluating the effectiveness of content-oriented XML IR is a necessary requirement for the further improvement of the state-of-the-art in XML IR research. In this study we drew comparisons between the retrieval and user model of traditional IR and XML IR experiments and highlighted the need for new evaluation methods that go beyond the assumptions made in standard IR evaluation. In XML IR, the dependency (structural relationships) between retrieval units leads to a richer model of user interaction and additional requirements that evaluation metrics should consider, e.g. overlap and near-misses.

We proposed a set of new evaluation metrics that aim to address this need. In addition to the recall-based and user-oriented metrics of xCG and normalised xCG ($nxCG$) metrics, we defined the system-oriented measures of effort-precision and gain-recall (ep/gr), which overcome the shortcomings of the earlier measures. Effort-precision and gain-recall share similar characteristics with the standard IR measures of precision and recall, but measure performance relative to an ideal scenario. Effort-precision for a given gain-recall point measures the relative effort the user is required to spend when scanning an arbitrary ranked list of result elements compared to the effort he/she would need when scanning a perfect ranking in order to reach the same level of gain-recall. All metrics derive from a basic gain value that represents the worth of a given result element to the user and is affected by aspects such as overlap and near-misses. The calculation of the gain value via relevance value functions allows to model different user behaviours and preferences.

During the evaluation of our metrics we confirmed that their behaviour is comparable to standard IR measures, such as mean average precision and p@DCV, showing similar levels of sensitivity to changes in an output ranking. In particular, our mean average effort-precision *MAep* metric has been shown to have similar properties to measures with statistically sound properties, such as mean average precision and the $Q-measure$ of [Sakai 2004], another cumulated gain based measure.

The stability tests examining the effects of varying assessments on the reported evaluation performance scores provided promising results, with correlation results across 32 (non-independent) test sets (Table III) matching the correlation levels reported in [Voorhees 2000] for mean average precision. The calculated error rates were much smaller for our metrics (e.g. 0.4% for *MAep* using 34 topics compared with 1.4% for mean average precision using 50 topics). Due to the repeated topics

in the sample sets, these results, however, only provide an optimistic estimate and lower bound for expected error rates. The pessimistic estimate, using 32 test sets containing only 5 independent topics, resulted in an upper bound of error between 2.79% and 13.45% (Table IV).

The experiments regarding the effects of topic set size on evaluation error confirmed that as topic set size increases, the error rate decreases. Compared to error rates reported for the standard IR measure of mean average precision in [Voorhees and Buckley 2002], our metrics show faster sloping error curves (Figure 9) where smaller error rates are achieved with less topics. For example, using 17 topics our confidence that a system A is better than a system B, given that the difference in their *MAep* score is < 0.0025 is 77% (error rate of 23%). The comparable result of [Voorhees and Buckley 2002] is 40% error rate for mean average precision (for bin < 0.01). To obtain a 95% confidence, after projecting error rates to 50 topics, we found that the required minimum absolute difference between two systems' *MAep* score is 0.005, which represents a relative difference of 4.67% of the best *MAep* score for the INEX'04 runs.

Our current investigations provide evidence that our XCG metrics provide suitable performance measures for the evaluation of the effectiveness of content-oriented XML IR. Although the user-oriented measures of xCG and $nxCG$ are less reliable and rather sensitive, they are useful and informative measures at low cutoff values. The new measure of effort-precision and, in particular, its mean average is shown to equal mean average precision in stability and sensitivity.

The XCG metrics have since been adopted as the official measues for INEX 2005 and are currently being tested in an operational setting with various scenarios and retrieval tasks (e.g. focused, thorough and fetch and browse), where different user models are to be employed.

Our future work will investigate how the XCG metrics compare to alternative metrics proposed for the evaluation of XML IR [Piwowarski and Gallinari 2004; Piwowarski et al. 2005; Goevert et al. 2005; de Vries et al. 2004]. Furthermore, in order to improve on the currently employed user model, we aim to integrate more sophisticated methods for calculating the gain value of result elements e.g. to consider a wider set of near-misses with scores reflecting their distance and size ratio to an ideal node. In particular, we are interested in integrating the probabilistic user model expressed in [Piwowarski et al. 2005] as an RV function in our evaluation framework. In addition, we will explore the use of alternative methods, and specifically the one proposed in [Goevert et al. 2005], for deriving an ideal recall-base.

8. ACKNOWLEDGEMENTS

We are thankful for the help of Paul Ogilvie and Saadia Malik on some of the scripts for the R statistical package.

REFERENCES

Allan, J. 2004. Hard track overview in TREC 2004: High accuracy retrieval from documents. In *Proceedings of the 13th Text Retrieval Conference (TREC 2004)*. Nist Special Publication, SP 500-261.

Amati, G. 2003. Probability models for information retrieval based on divergence from randomness. Ph.D. thesis, University of Glasgow.

Baeza-Yates, R., Fuhr, N., and Maarek, Y., Eds. 2002. *Proceedings of the SIGIR 2002 Workshop on XML and Information Retrieval*.

Baeza-Yates, R. and Ribeiro-Neto, B. 1999. *Modern Information Retrieval*. Addison Wesley.

Blanken, H. M., Grabs, T., Schek, H.-J., Schenkel, R., and Weikum, G., Eds. 2003. *Intelligent Search on XML Data, Applications, Languages, Models, Implementations, and Benchmarks*. LNCS, vol. 2818. Springer.

Borlund, P. 2003. The concept of relevance in ir. *J. Am. Soc. Inf. Sci. Technol. 54,* 10, 913–925.

Buckley, C. and Voorhees, E. M. 2000. Evaluating evaluation measure stability. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 33–40.

Burgin, R. 1992. Variations in relevance judgments and the evaluation of retrieval performance. *Information Processing and Management 28,* 5, 619–627.

Chiaramella, Y., Mulhem, P., and Fourel, F. 1996. A model for multimedia information retrieval. Tech. Rep. Fermi ESPRIT BRA 8134, University of Glasgow.

Clark, J. and DeRose, S. 1999. XML Path Language (XPath) version 1.0. W3C Recommendation. http://www.w3.org/TR/xpath. Tech. Rep. REC-xpath-19991116, WWW Consortium. Nov.

Conover, W. 1980. *Practical Non-Parametric Statistics, 2nd edn*. John Wiley & Sons, Inc., New York, NY, USA.

Cooper, W. 1968. Expected search length: A single measure of retrieval effectiveness based on the weak ordering action of retrieval systems. *American Documentation 19,* 1, 30–41.

de Vries, A., Kazai, G., and Lalmas, M. 2004. Tolerance to Irrelevance: A user-effort oriented evaluation of retrieval systems without predefined retrieval unit. In *Proceedings of the Recherche d'Informations Assistee par Ordinateur (RIAO 2004)*. Avignon, France.

Fuhr, N., Lalmas, M., and Malik, S., Eds. 2004. *Proceedings of the Second Workshop of the INitiative for the Evaluation of XML Retrieval (INEX). Dagstuhl, Germany, December 15–17, 2003*. http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf.

Fuhr, N., Lalmas, M., Malik, S., and Szlavik, Z., Eds. 2005. *Advances in XML Information Retrieval. Third Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2004), Schloss Dagstuhl, 6-8 December 2004*. Lecture Notes in Computer Science, vol. 3493. Springer.

Fuhr, N., Malik, S., and Lalmas, M. 2004. Overview of the initiative for the evaluation of xml retrieval (inex) 2003. See Fuhr et al. [2004], 1–11. http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf.

Goevert, N., Fuhr, N., Lalmas, M., and Kazai, G. 2005. Evaluating the effectiveness of content-oriented xml retrieval. *Submitted to Information Retrieval*.

Gövert, N. and Kazai, G. 2003. Overview of the INitiative for the Evaluation of XML retrieval (INEX) 2002. In *Proceedings of the 1st Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Dagstuhl, Germany, 8-11 December, 2002*, N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, Eds. ERCIM, Sophia Antipolis, France, 1–17.

Harman, D., Ed. 1992. *Proceedings of the First Text Retrieval Conference (TREC-1)*. Number 500–207 in NIST Special publications.

Harter, S. P. 1996. Variations in relevance assessments and the measurement of retrieval effectiveness. *Journal of the American Society for Information Science 47,* 1, 37–49.

Hull, D. 1993. Using statistical testing in the evaluation of retrieval experiments. In *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 329–338.

Järvelin, K. and Kekäläinen, J. 2000. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 41–48.

Järvelin, K. and Kekäläinen, J. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (ACM TOIS) 20,* 4, 422–446.

Kando, N., Kuriyama, K., and Yoshioka, M. 2001. Information retrieval system evaluation using multi-grade relevance judgements - discussion on averageable single-numbered measures (in japanese). Tech. rep.

Kazai, G. and Lalmas, M. 2005. Notes on what to measure in inex. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology, Glasgow, July 2005*.

Kazai, G., Lalmas, M., and de Vries, A. P. 2004. The overlap problem in content-oriented xml retrieval evaluation. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 72–79.

Kazai, G., Lalmas, M., and de Vries, A. P. 2005. Reliability tests for the xcg and inex-2002 metrics. In *Advances in XML Information Retrieval. Third Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2004), Schloss Dagstuhl, 6–8 December 2004*, N. Fuhr, M. Lalmas, S. Malik, and Z. Szlavik, Eds. Lecture Notes in Computer Science, vol. 3493. Springer, 60–72.

Kazai, G., Lalmas, M., and Piwowarski, B. 2004. Inex relevance assessment guide. See Fuhr et al. [2004], 204–209. http://inex.is.informatik.uni-duisburg.de:2003/proceedings.pdf.

Kazai, G., Lalmas, M., and Reid, J. 2003. Construction of a test collection for the focussed retrieval of structured documents. In *Advances in Information Retrieval, Proceedings of the 25th European Conference on IR Research, Pisa, Italy*, F. Sebastiani, Ed. Lecture Notes in Computer Science, vol. 2633. Springer, 88–103.

Kekäläinen, J. 2005. Binary and graded relevance in ir evaluations: comparison of the effects on ranking of ir systems. *Information Processing and Management 41,* 5, 1019–1033.

Kekäläinen, J. and Järvelin, K. 2002. Using graded relevance assessments in ir evaluation. *Journal of the American Society for Information Science and Technology 53,* 13, 1120–1129.

Lalmas, M. and Malik, S. 2004. Inex 2004 retrieval task and result submission specification. See Fuhr et al. [2005].

Lesk, M. and Salton, G. 1969. Relevance assessments and retrieval system evaluation. *Information Storage and Retrieval 4,* 4, 343–359.

Malik, S., Lalmas, M., and Fuhr, N. 2005. Overview of inex 2004. In *Advances in XML Information Retrieval. Third Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2004), Schloss Dagstuhl, 6-8 December 2004*, N. Fuhr, M. Lalmas, S. Malik, and Z. Szlavik, Eds. Lecture Notes in Computer Science, vol. 3493. Springer, 1–15.

Piwowarski, B. and Gallinari, P. 2004. Expected ratio of relevant units: A measure for structured document information retrieval. In *Proceedings of the 2nd Workshop of the INitiative for the Evaluation of XML retrieval (INEX), Dagstuhl, Germany, December 2003*, N. Fuhr, M. Lalmas, and S. Malik, Eds. 158–166.

Piwowarski, B., Gallinari, P., and Dupret, G. 2005. Precision Recall with User Modelling: Application to XML retrieval. *Submitted for publication*.

Raghavan, V. V., Bollmann, P., and Jung, G. S. 1989. Retrieval system evaluation using recall and precision: problems and answers. In *SIGIR '89: Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 59–68.

Rijsbergen, C. J. V. 1979. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA. Out of print, available online from http://www.dcs.glasgow.ac.uk/Keith/Preface.html.

Sakai, T. 2004. New performance metrics based on multigrade relevance: Their application to question answering. In *NTCIR Workshop 4 Meeting Working Notes*.

Sakai, T. 2005. The reliability of metrics based on graded relevance. In *AIRS*. 1–16.

Sanderson, M. and Zobel, J. 2005. Information retrieval system evaluation: effort, sensitivity, and reliability. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 162–169.

Schamber, L. 1994. Relevance and information behavior. *Ammal review of information science and technology (ARIST)*, 3–48.

Tague-Sutcliffe, J. 1992. The pragmatics of information retrieval experimentation, revisited. *Information Processing and Management 28,* 4, 467–490.

Tombros, T., Larsen, B., and Malik, S. 2005. The interactive track at INEX 2004. In *Proceedings of the 3rd Workshop of the INitiative for the Evaluation of XML retrieval (INEX), Dagstuhl, Germany, December 2004*, N. Fuhr, M. Lalmas, S. Malik, and Z. Szlavik, Eds.

Trotman, A. and Sigurbjörnsson, B. 2005. Narrowed extended xpath i (nexi). In *Advances in XML Information Retrieval. Third Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2004), Schloss Dagstuhl, 6-8 December 2004*, N. Fuhr, M. Lalmas, S. Malik, and Z. Szlavik, Eds. Lecture Notes in Computer Science, vol. 3493. Springer, 41–53.

Vegas, J., de la Fuente, P., and Crestani, F. 2002. A graphical user interface for structured document retrieval. In *Proceedings of the 24th BCS-IRSG European Colloquium on IR Research*. Springer-Verlag, London, UK, 268–283.

Voorhees, E. M. 2000. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing and Management 36,* 5, 697–716.

Voorhees, E. M. 2001. Evaluation by highly relevant documents. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 74–82.

Voorhees, E. M. 2003a. Overview of the TREC 2003 question answering track. In *Text REtrieval Conference, Gaithersburg*.

Voorhees, E. M. 2003b. Overview of the trec 2003 robust retrieval track. In *TREC*. 69–77.

Voorhees, E. M. and Buckley, C. 2002. The effect of topic set size on retrieval experiment error. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 316–323.

Wallis, P. and Thom, J. A. 1996. Relevance judgments for assessing recall. *Inf. Process. Manage. 32,* 3, 273–286.