The Interpretation of CAS

Andrew Trotman[1] and Mounia Lalmas[2]

[1] Department of Computer Science, University of Otago, Dunedin, New Zealand
andrew@cs.otago.ac.nz,
[2] Department of Computer Science Queen Mary University of London, London, UK
mounia@dcs.qmul.ac.uk,

Abstract. There has been much debate over how to interpret the structure in queries that contain structural hints. At INEX 2003 and 2004, there were two interpretations: SCAS in which the user specified target element was interpreted strictly, and VCAS in which it was interpreted vaguely. But how many ways are there that the query could be interpreted? In the investigation at INEX 2005 (discussed herein) four different interpretations were proposed, and compared on the same queries. Those interpretations (SSCAS, SVCAS, VSCAS, and VVCAS) are the four interpretations possible by interpreting the target elements, and the support elements, either strictly or vaguely. An analysis of the submitted runs shows that those that share an interpretation of the target element correlate - that is, the previous decision to divide CAS into the SCAS and VCAS (as done at INEX 2003 and 2004) was sound. The analysis is supported by the fact that the best performing VSCAS run was submitted to the VVCAS task and the best performing SVCAS run was submitted to the SSCAS task.

1 Introduction

Does including a structural hint in a query make a precision difference and if so how should we interpret it? At INEX 2005 the *ad hoc* track has been investigating this question. Two experiments were conducted, the CO+S experiment, and the CAS experiment.

In the CO+S experiment the participants were asked to submit topics with content only (CO) queries containing just search terms, and optionally an additional structured (+S) query specified in the NEXI [10] query language. Given these two different interpretations of the same information need it is possible to compare the precision of queries containing structural hints to those that do not *for the same information need*. The details of the CO+S experiment are beyond the scope of this paper.

In a separate experiment participants were asked to submit topics containing queries that contain content and structure (CAS) constraints specified in NEXI [10]. These topics were used to determine how the structural hints, necessarily present in a CAS topic, should be interpreted by a search engine. The two extreme views are the database view that all structural constraints must be

upheld, and the information retrieval view that satisfying the information need is more important than following the structural constraints of the query.

This contribution discusses the mechanics of the CAS experiment from the topic submission process, the document collection, through to the evaluation methods. The different tasks are compared using Pearson's product moment correlation coefficient showing that there were essentially only two tasks, those that in previous years have gone by the name VCAS and SCAS. Further analysis shows that of the tasks SSCAS is the easiest and VVCAS the hardest.

2 CAS Queries

Laboratory experiments in information retrieval following the Cranfield methodology (described by Voorhees [12]) require a document collection, a series of queries (known as topics), and a series of judgments (decisions as to which documents are relevant to which topics). In element retrieval this same process is followed - except with respect to a document element rather than a whole document.

Content and structure queries differ from content only queries in so far as they contain structural hints. Two types of structural hints are present, those that specify where to look (support elements) and those that specify what to return to the user (target elements). In INEX topic 258

```
//article[about(.,intellectual property)]//sec[about(., copyright law)]
```

the search engine is being asked to identify documents about intellectual property and from those extract sections about copyright law. The target element is //article//sec (extract //article//sec elements). The support elements are //article (with support from //article about intellectual property) and //article//sec (and support from //article//sec about copyright law). Full details of the syntax of CAS queries is given by Trotman and Sigurbjörnsson [10]. The applicability of this language to XML evaluation in the context of INEX is also discussed by Trotman and Sigurbjörnsson [11].

2.1 Query Complexity

The simplest CAS queries contain only a single structural constraint. Topic 270,

```
//article//sec[about( ., introduction information retrieval)]
```

asks for //article//sec elements about "introduction information retrieval". A more complex (multiple clause) query can be decomposed into a series of single constraint (single clause) queries (or child queries). Topic 258,

```
//article[about(.,intellectual property)]//sec[about(., copyright law)]
```

could be written as a series of single constraint queries, each of which must be satisfied. In this case it is decomposed into topic 259,

```
//article[about(.,intellectual property)]
```

and topic 281,

```
//article//sec[about(., copyright law)]
```

if both hold true of a document then the (parent) query is true of that document - and the target element constraints can be considered. The same decomposition property holds true for all multiple constraint CAS topics (so long as the target element is preserved) - it is inherent in the distributive nature of the query language.

Having separate parent and children topics makes it possible to look at different interpretations of the same topic. As a topic is judged according to the narrative the judgments are by definition vague. Strict conformance of these judgments to the target element can be generated using a simple filter. This is the approach taken at INEX 2003 and 2004 for the so-called SCAS and VCAS tasks. But what about the sub-clauses of these topics? Should they be interpreted strictly or vaguely? With the judgments for the child topics, vague and strict conformance to these can also be determined. With the combination of child and parent judgments it is possible to look at many different interpretations of the same topic.

2.2 Topic format

INEX captures not only the query, but also the information need of the user. These are stored together in XML. Methods not dissimilar to this have been used at TREC [2] and INEX [1] for many years. As an example, INEX topic 258

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE inex_topic SYSTEM "topic.dtd">
<inex_topic topic_id="258"  query_type="CAS" ct_no="72">
<InitialTopicStatement>
I have to give a computer science lesson on intellectual property
and I'm looking for information or examples on copyright law to
illustrate it. As I'm looking for something which is specific, I
don't think I can find a whole article about it. I'm consequently
looking for section elements.
</InitialTopicStatement>
<castitle>
//article[about(.,intellectual property)]//sec[about(., copyright law)]
</castitle>
<description>
Return sections about copyright law (information or examples) in an
article about intellectual property.
</description>
<narrative>
I have to give a computer science lesson on intellectual property,
and I'm looking for information or examples on copyright law to
```

```
illustrate it. More precisely, I'd like to have information about
authors rights and how to protect your creation. As I'm looking for
something which is specific, I don't think I can find a whole
article about it. I'm consequently looking for section elements.
Information or examples can concern copyright on software,
multimedia or operating systems. Copyright on literary work can help
but only for examples. Information concerning domain names and
trademarks is not relevant.
</narrative>
</inex_topic>
```

contains several parts all discussing the same information need:

- <**InitialTopicStatement**> a description of why the user has chosen to use a search engine, and what it is that the user hopes to achieve.
- <**castitle**> the CAS query specified in the NEXI language [10].
- <**description**> a natural language expression of the information need using the same terms as are found in the <**castitle**>. This element is used by the natural language track at INEX [13].
- <**narrative**> a description of the information need and what makes a result relevant. When judgments are made they are made against this description so it is important that it precisely describes the difference between relevant and irrelevant results. For experiments that additionally take into account the context of a query (such as the interactive track [8]), the purpose for which the information is needed (the work-task) is also given in the narrative.

Both the parent query and the child queries are stored in this way - but an additional element, the <**parent**> element, is present in child topics. This element stores the castitle of the child's parent. This method of linking children to parents was chosen over using identifiers as it was considered less likely to be prone to human input error.

2.3 Query Interpretation

A contentious point about CAS queries is the interpretation. The strict view is that the structural hints are constraints and the search engine should follow them to ensure returning elements that satisfy the user. The vague view is that the structural hints are hints and can be down-played so long as a returned element is relevant in the mind of the user (it satisfies the information need).

A single clause query might be interpreted strictly, or vaguely - that is the constraint might be followed or can be ignored. If, for example, a user asks for an article abstract about "information retrieval", then perhaps an article introduction might just as well satisfy the need - or perhaps not.

With multiple clause queries, there are many possible interpretations. In the CAS experiment at INEX 2005, the strict and vague interpretations are applied to both the target element, and the support elements. This gives four interpretations written XYCAS where X is the target element and Y is the

support element, and either X or Y can be S for strict or V for vague. Those interpretations are:

- **VVCAS:** The target element constraint is vague and the support element constraints are vague. This is the information retrieval view of the topic.
- **SVCAS:** The target element constraint is strict, but the support element constraints are vague.
- **VSCAS:** The target element constraint is vague, but the support element constraints are followed strictly.
- **SSCAS:** Both the target element constraint and the support element constraint are followed strictly. This is the database view.

3 Document Collection

The document collection used in the experiments was the INEX IEEE document collection version 1.8. This collection contains 16,819 documents taken from IEEE transactions and magazines published between 1995 and 2004. The total size of the source XML is 705MB. This is the latest version of the INEX collection at publication date.

4 Data Acquisition

This section discusses the acquisition of the queries from the participants and the verification that they are representative of previous years. It also discusses the acquisition of the judgments and the construction of the different judgment sets.

4.1 Query Acquisition

The document collection was distributed to the participating organizations. They were then each asked to submit one CAS topic along with any associated single clause child-topics (should they exist). These topics then went through a selection process in which queries were parsed for syntactic correctness, semantic correctness, consistency, and validated against their child topics. A total of 17 queries passed this selection process.

Table 1. A Breakdown of the complexity of INEX 2005 CAS topics shows that they are representative of previous years

Clauses	**1**	**2**	**3**	**4+**
2003	7 (23%)	12 (40%)	6 (20%)	5 (17%)
2004	4 (12%)	22 (65%)	4 (12%)	4 (12%)
2005	3 (18%)	12 (71%)	2 (12%)	0 (00%)

The breakdown of CAS topic complexity (excluding child-topics) for each of INEX 2003, 2004, and 2005 is given in Table 1. From visual inspection it can be seen that the breakdown in 2005 is representative of previous years, most queries contain two clauses with approximately the same number of three and one clause topics. In 2005 there were no topics with more than 3 clauses.

4.2 Child Topics

Table 2. The 17 topics and the topic numbers of their children

Parent	Children	Parent	Children	Parent	Children
244	245, 246	258	259, 281	270	
247	248, 249, 276	260		275	274, 273
250	251, 252	261	262, 263	280	277, 278, 279
253	254, 255	264	282, 283	284	266, 285
256	272, 271	265	267, 268	288	242, 243
257		269	286, 287		

Each topic and child topic was given a unique identifier (stored in the topic_id attribute of the inex_topic tag). Table 2 shows which topics are parent topics and which topics are their children. Topic 258, for example, has topics 259 and 281 as children whereas topic 260 is a single clause query and has no children.

It may appear at the onset that these child topics can be used as part of the evaluation giving a total of 47 topics. This, however, is not the case. The guidelines for topic development [7] identifies that for evaluation purposes queries must be diverse, independent, and representative. Using both the parent and the children topics for computing performance violates the independence requirement - and weights evaluation in favor of longer topics (which have more children).

Using just the child topics, and discarding the parents, violates the requirement that topics are representative. In Table 1, the breakdown of topics from previous years is shown. Most topics have two clauses, whereas child topics (by definition) have only one. The children, without their parents, are not representative.

4.3 Judgment Acquisition

The topics and child-topics were distributed to the participants. Each participating group was invited to submit up to two runs for each CAS task. At least one was required for VVCAS. A run consisted of at most 1,500 ranked results for each parent and child topic. There were no restrictions on which part of the topic was used to generate the query - participants were permitted to use the narrative, or description, or castitle if they so chose.

These results were then pooled in a similar manner to that used at TREC (and shown to be robust there by Zobel [15]). The details of the INEX pooling method are give by Piwowarski and Lalmas [6] and a discussion of the robustness is provided by Woodley and Geva [14].

The pool identifies which documents and elements the search engines considered relevant to the query. Using a graphical interface (the 2005 version of X-Rai [5, 6]) to the document collection, the original author of the query (where possible) was asked to identify which elements of which documents in the judgment pool were, in fact, relevant to the information need. Assessors first highlighted relevant passages from the text, and then they assigned relevance values to all elements in this region on a three points scale: highly exhaustive, partly exhaustive, or too small. This assessment was performed for the parent topics in isolation of the child topics - and not necessarily by the same assessor.

As a topic may contain many different interpretations of the information need (for example the description and the castitle) all judgments were made with reference to the description contained in the topic narrative.

4.4 CAS Relevance Assessments

Table 3. Topics assessed by more than one assessor. Listed for each set against each topic is the pool-id of the assessments

Topic	Pool	
	Set-a (official)	**Set-b (other)**
261	350	362
244	354	358
250	356	369
258	289	360

In a separate experiment the consistency of the judgments is being measured across multiple assessors. This is done by asking two or more judges to assess the same topic, without knowledge of the other's decisions. Of the CAS topics, those listed in Table 3 were multiple-judged.

The consequence of this multiple assessment process is that there is no single set of relevance assessments. Inline with INEX 2004, the assessments are divided into two groups: set-a, and set-b (see Pehcevski *et al.* [4] and Trotman [9] for a discussion of the 2004 results of this experiment). The INEX 2005 assignment was made based on proportion of completion at the date the first relevance assessments were released. Those judgments that, from visual inspection, appeared most complete were assigned to set-a, while the other was assigned to set-b. In this way set-a, the set used to generate the official results, was most complete and therefore most reliable.

Internal to X-Rai (the online assessment tool), each assessment of each topic by each judge is given an internal identifier - the pool id. Table 3 also shows which pool ids were assigned to which judgment set.

4.5 CAS Relevance Sets

From set-a, four sets of judgments were generated, one for each of the four CAS interpretations - each derived from the same initial set of judgments.

- **VVCAS:** The assessments as done by the assessors (against the narrative). These assessments are unmodified from those collected by INEX from the assessor.
- **SVCAS:** Those VVCAS judgments that strictly satisfy the target element constraint. This set of judgments was computed by taking the VVCAS judgments and removing all judgments that did not satisfy the target element constraint. This was done by a simple matching process. All those elements that were, indeed, the target element were included and those that were not were removed. Topic 260 is an exception. In this case the target element is specified as //bdy//*. To satisfy this constraint all descendants of //bdy (excluding //bdy) are considered to strictly comply.
- **VSCAS:** A relevant element is not required to satisfy the target constraint, however the document must satisfy all other constraints specified in the query. That is, for a multiple clause topic, an element is relevant only if it comes from a document in which the child-topics are strictly adhered to. In all except two topics, given the conjuncton of documents relevant to the child topics, this is any relevant element from the VVCAS set that comes from this conjunction. In one exception (topic 247), this conjunction is replaced with a disjunction. In the other exception (topic 250) there are (presently) no judgments as the assessment task has not been completed.
- **SSCAS:** Those VSCAS judgments that satisfy the target element constraint. These are computed from the VSCAS judgments in the same way that SVCAS judgments are computed from VVCAS judgments - strict conformance to the target element.

The guidelines for topic development [7] identify groups of tags that are equivalent. For example, for historic paper publishing reasons the sec, ss1, ss2 and ss3 tags are all used to identify sections of documents in the collection. The strict conformance to a given structural constraint occurs with reference to the equivalence list - //article//bdy//ss1 strictly conforms to //article//sec.

These relevance sets are considered to be the full recall base for each interpretations of CAS. Different metrics and quantizatons functions could further reduce the relevance sets. For example, in the case of struct quantization only those elements that conform to the interpretation of CAS and further conform to the interpretation of strict are considered relevant.

5 Measurement

The official metric used to report the performance of a system at INEX 2005 is MAep, the mean average nxCG rank at 1500 elements. This measure is described by Kazai and Lalmas [3]. The results (produced using xcgeval) for the INEX 2005 CAS task are available from INEX. There were 99 runs submitted to the CAS tasks, of which 25 were SSCAS, 23 SVCAS, 23 VSCAS, and 28 VVCAS[1].

Of the 17 topics used for evaluation (the parent topics of Table 2) judgments currently exist for only 10 topics - at the time of writing the assessment task had not been completed for the other 7 topics. Of those 10 topics, only 7 have any elements that strictly conform to their child topic structural constraint. The comparison of systems herein is based only on these topics.

Table 4. Number of relevant elements for each topic using generalised quantization

Topic	SSCAS	SVCAS	VSCAS	VVCAS
253	0	23	0	156
256	492	724	1431	2101
257	96	96	711	711
260	5159	5159	5264	5264
261	0	59	0	4437
264	6	40	155	1272
265	0	40	0	211
270	35	35	850	850
275	111	183	12870	16965
284	2	111	326	14265

In Table 4 and Table 5 the number of relevant elements for each topic of each task is shown. The judgments for strict quantization are highly sparse - for the SSCAS task, there are only 4 topics with highly specific and highly exhaustive judgments. It does not seem reasonable to draw any conclusions from only 4 topics so the remainder of this analysis applies to only the generalized quantization of results.

By taking all runs submitted to any CAS task and correlating the performance on one task to the performance on another (say, VVCAS with SSCAS), it is possible to see if a search engine designed for one interpretation also performs well on other interpretations (and therefore if there is any material difference in the tasks). That is, if a search engine is designed to answer in one way but the user expects results in another, does it matter? Taking all the CAS runs (including the "unofficial" runs) the IBM Haifa Research Lab run VVCAS_no_phrase_no_tags submitted to the VVCAS task performs best using the VVCAS judgments (with a MAep score of 0.1314), but if the user need included a strict interpretation of

[1] Submissons version 1 and judgments version 7 are used throughout

Table 5. Number of relevant elements for each topic using strict quantization

Topic	SSCAS	SVCAS	VSCAS	VVCAS
253	0	0	0	11
256	139	162	198	228
257	0	0	0	0
260	66	66	66	66
261	0	0	0	2
264	0	0	12	44
265	0	0	0	1
270	1	1	2	2
275	18	22	330	424
284	0	5	4	196

the topic (it was evaluated using the SSCAS judgments) then it is at position 50 with a score of 0.0681.

By comparing the performance of runs submitted to each task it is possible to determine if one task is inherently easier, or harder, than the others. With a harder task there is more room for improvement - further investigation into this task might result in improvements all-round.

5.1 Do the Judgment Sets Correlate?

Table 6. Pearson's product moment correlation coefficient between each CAS task

	SSCAS	SVCAS	VSCAS	VVCAS
SSCAS	1.0000	0.8934	0.4033	0.3803
SVCAS	0.8934	1.0000	0.3409	0.3768
VSCAS	0.4033	0.3409	1.0000	0.9611
VVCAS	0.3803	0.3768	0.9611	1.0000

Table 6 shows the Pearson's product moment correlation coefficient computed for all runs when scored at each task. Scores close to 1 show a positive correlation, those close to -1 a negative correlation and those at 0 show no correlation.

It is clear from the table that VVCAS and VSCAS are strongly correlated. A search strategy that performs well at one task performs well at the other. SSCAS and SVCAS, both with a strict interpretation of the target element are less strongly correlated. There is little correlation between a strict interpretation of the target element and a vague interpretation of the target element (SVCAS and VSCAS, for example).

Figure 1 shows this correlation for the vague target element tasks. There is a cluster of best-scoring runs at the top-right of the graph. They are runs that have performed well at both VVCAS and VSCAS. These four runs are those from IBM Haifa Research Lab. Although different runs perform best on the VVCAS and VSCAS task, both "best" runs were submitted to the VVCAS task - providing further evidence of the correlation of the two tasks.

Figure 2 shows the same for the strict target element tasks. The cluster is not seen. The best performing run measuring on the SVCAS task was submitted to the SSCAS task (again IBM Haifa Research Lab). These same runs were only bettered by the four from the University of Tampere when measured for the SSCAS task. Although Tampere produced runs that performed well at the SSCAS task and not at the SVCAS task, IBM Haifa Research Lab produced runs that performed well at both tasks. Again further evidence of the correlation of the two tasks.

Figure 3 shows the performance of SSCAS against VVCAS. It is clear from this figure that those runs that perform well at one task do not perform well at the other. It appears, from visual inspection, that they are average performers at each other's tasks.

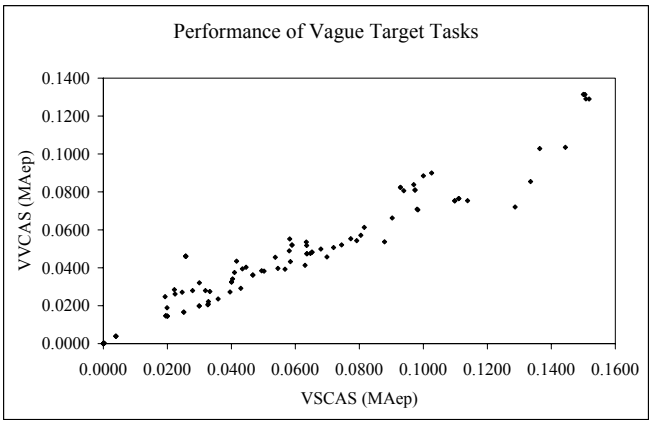

Fig. 1. Plot of performance of all submitted runs using VVCAS and VSCAS shows a strong correlation of one to the other

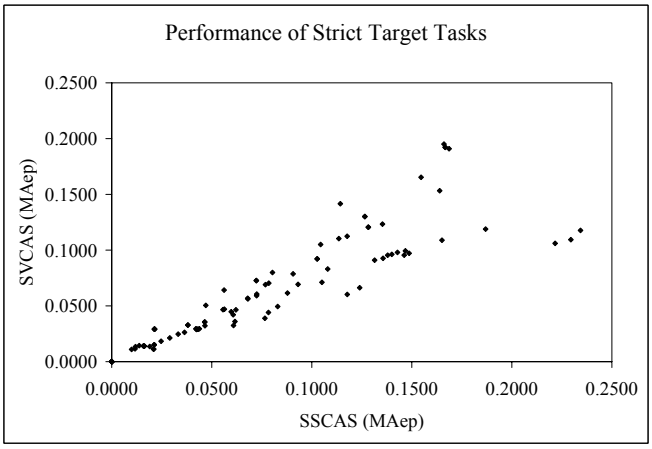

Fig. 2. Plot of performance of all submitted runs using SVCAS and SSCAS shows a strong correlation of one to the other

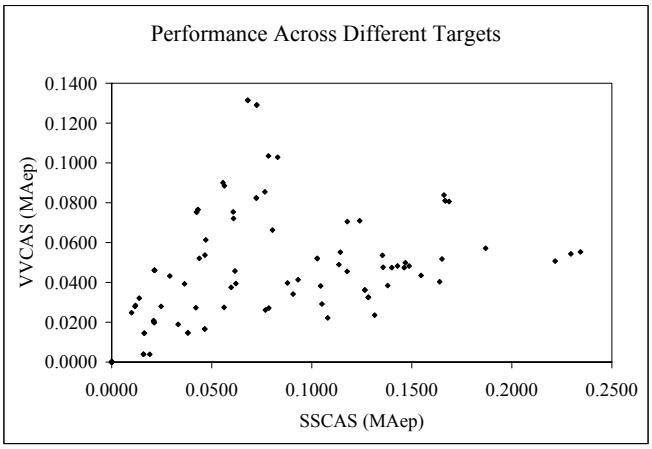

Fig. 3. Plot of performance of all submitted runs using VVCAS and SSCAS shows little correlation of one to the other

Table 7. Mean performance of top 21 runs from each task

	SSCAS	SVCAS	VSCAS	VVCAS
Mean	0.1285	0.0832	0.0628	0.0690
Std Dev	0.0510	0.0484	0.0439	0.0310
Best	0.2343	0.1922	0.1508	0.1314
Worst	0.0381	0.0292	0.0039	0.0208

5.2 Tasks Complexity

For each task the performance (MAep) of each of the top performaing 21 runs submitted to that task was computed. This number was chosen because different numbers of runs were submitted to each task, and for all tasks there were at least 21 runs with a non-zero score. Table 7 presents the performance of the best run, the mean computed over all runs submitted to the task, and the worst run submited to each task. It can be seen that the best performing run was submitted to the SSCAS task, and for that task the average run performs better that the average run from other tasks. From this we deduce that the SSCAS task is easier than the other tasks. This task may be easiest because the required structural constraints are specified explicitly in the query and the search engine can use this as a filter to remove known non-relevant elements from the result list.

Normally it is invalid to measure the performance of two different search engines by measuring the performance of one on one collecion and the second on a second collection (or set of topics, or judgments). In this experiment the document collection and topics are fixed, the judgments are derived from a single common source, and mean performance across several search engines is compared. We believe this comparison is sound.

6 Conclusions

The Pearson's correlation shows that there are only two different interpretations of the query, those with a strict interpretation of the target element and those with a vague interpretation of the target element (the database and the information retrieval views). It is possible to ignore the interpretation of the child elements and concentrate on only the target elements. In previous years, INEX has made a distinction between strict and vague conformance to the target element, but has disregarded conformance to child constraints (the so-called SCAS and VCAS tasks). This finding suggests the experiments of previous years did, indeed, make the correct distinction. Checking child constraints does not appear worthwhile for content-oriented XML retrieval.

The vague task has proven more difficult than the strict task. Strict conformance to the target element can be computed as a filter of a vague run - from those vague elements, remove all that do not conform to the target element con-

straint. The vague interpretation of CAS is a better place to concentrate research effort.

If the CAS task continues in future years, a single set of topics, without the child topics is all that is necessary for evaluation and participants should concentrate on the vague interpretation of topics.

References

1. Fuhr, N., Gövert, N., Kazai, G., and Lalmas, M. (2002). INEX: Initiative for the evaluation of XML retrieval. In *Proceedings of the ACM SIGIR 2000 Workshop on XML and Information Retrieval*.
2. Harman, D. (1993). Overview of the first TREC conference. In *Proceedings of the 16th ACM SIGIR Conference on Information Retrieval*, (pp. 36-47).
3. Kazai, G., and Lalmas, M. (2005). INEX 2005 evaluation metrics. In *Proceedings of INEX 2006*.
4. Pehcevski, J., Thom, J. A., and Vercoustre, A.-M. (2005). Users and assessors in the context of INEX: Are relevance dimensions relevant? In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology, Second Edition*, (pp. 47-62).
5. Piwowarski, B., and Lalmas, M. (2004). Interface pour l'évaluation de systèmes de recherche sur des documents XML. In *Proceedings of the Premiere COnference en Recherche d'Information et Applications (CORIA'04)*.
6. Piwowarski, B., and Lalmas, M. (2004). Providing consistent and exhaustive relevance assessments for XML retrieval evaluation. In *Proceedings of the 13th ACM conference on Information and knowledge management*, (pp. 361-370).
7. Sigurbjörnsson, B., Trotman, A., Geva, S., Lalmas, M., Larsen, B., and Malik, S. (2005). INEX 2005 guidelines for topic development. In *Proceedings of INEX 2005*.
8. Tombros, A., Larsen, B., and Malik, S. (2004). The interactive track at INEX 2004. In *Proceedings of INEX 2004*, (pp. 410-423).
9. Trotman, A. (2005). Wanted: Element retrieval users. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology, Second Edition*, (pp. 63-69).
10. Trotman, A., and Sigurbjörnsson, B. (2004). Narrowed Extended XPath I (NEXI). In *Proceedings of INEX 2004*, (pp. 16-40).
11. Trotman, A., and Sigurbjörnsson, B. (2004). NEXI, now and next. In *Proceedings of INEX 2004*, (pp. 41-53).
12. Voorhees, E. M. (2001). The philosophy of information retrieval evaluation. In *Proceedings of the The Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, (pp. 355-370).
13. Woodley, A., and Geva, S. (2004). NLPX at INEX 2004. In *Proceedings of INEX 2004*, (pp. 382-394).
14. Woodley, A., and Geva, S. (2005). Fine tuning INEX. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology, Second Edition*, (pp. 70-79).
15. Zobel, J. (1998). How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st ACM SIGIR Conference on Information Retrieval*, (pp. 307-314).