RMIT University at INEX 2005: Ad hoc Track

Jovan Pehcevski, James A. Thom, and S. M. M. Tahaghoghi

School of Computer Science and Information Technology, RMIT University
Melbourne, Australia
{jovanp, jat, saied}@cs.rmit.edu.au

Abstract. Different scenarios of XML retrieval are analysed in the INEX 2005 ad hoc track, which reflect different query interpretations and user behaviours that may be observed during XML retrieval. The RMIT University group's participation in the INEX 2005 ad hoc track investigates these XML retrieval scenarios. Our runs follow a hybrid XML retrieval approach that combines three information retrieval models with two ways of identifying the appropriate element granularity and two XML-specific heuristics to rank the final answers. We observe different behaviours when applying our hybrid approach to the different retrieval scenarios, suggesting that the optimal retrieval parameters are highly dependent on the nature of the XML retrieval task. Importantly, we show that using structural hints in content only topics is a useful feature that leads to more precise search, but only when level of overlap among the retrieved elements is considered by the evaluation metric.

1 Introduction

Of the seven tracks at INEX 2005 — each exploring different applications of XML retrieval — our RMIT University group participated in four: ad hoc, interactive, multimedia [3], and heterogeneous. In this paper, we discuss our participation in the ad hoc track.

Two types of topics are explored in the ad hoc track: Content Only and Structure (`CO+S`) and Content And Structure (`CAS`). A `CO+S` topic is a request that typically ignores the document structure by only specifying plain query terms. However, there may be cases where adding structural hints to the query results in more precise search. Some `CO+S` topics therefore express the same information need by either ignoring or including the structural hints (we call the latter `+S topics`). Figure 1 shows a snippet of `CO+S` topic 203 that was proposed by our group, where two topic fields — `title` and `castitle` — are used to represent the two interpretations. A `CAS` topic is a request that contains references to the document structure and explicitly specifies the type of the returned answer elements (the target element) and the type of the contained elements of the search context (the support elements).

Within the INEX 2005 ad hoc track there are three XML retrieval sub-tasks: the `CO`, the `+S`, and the `CAS` sub-task, reflecting the three types of topics used. Three retrieval strategies are explored in the `CO+S` sub-tasks: `Focussed`,

```
<inex_topic topic_id="203" query_type="CO+S" ct_no="5">
   <title> code signing verification </title>
   <castitle> //sec[about(., code signing verification)] </castitle>
   <description> Find documents or document components, most probably
    sections, that describe the approach of code signing and verification.
   </description>
   <narrative> I am working in a company that authenticates a wide range of
    web database applications from different software vendors. [...]
   </narrative>
</inex_topic>
```

Fig. 1. A snippet of the INEX 2005 `CO+S` topic 203

`Thorough`, and `FetchBrowse`, which model different aspects of the XML retrieval task. Four retrieval strategies are explored in the `CAS` sub-task: `SS`, `SV`, `VS`, and `VV`, which correspond to the way target and support elements are interpreted [7].

The system we use in the ad hoc track follows a *hybrid* XML retrieval approach, combining information retrieval features from Zettair[1] (a full-text search engine) with XML-specific retrieval features from eXist[2] (a native XML database). The hybrid approach can be seen as a "fetch and browse" [2] XML retrieval approach, since full articles estimated as likely to be relevant to a query are first retrieved by Zettair (the *fetch* phase), and then the most specific elements within these articles are extracted by eXist (the *browse* phase) [9].

To calculate the similarity score of an article to a query (represented by terms that appear in the `title` part of an INEX topic), a similarity measure is used by Zettair. Three similarity measures are currently implemented, each based on one of the following information retrieval models: the vector-space model, the probabilistic model, and the language model. For the *fetch* phase of our hybrid system, we investigate which information retrieval model yields the best effectiveness for full article retrieval.

To identify the appropriate granularity of elements to return as answers, we use a retrieval module that utilises the structural information in the eXist list of extracted elements. For the *browse* phase of our hybrid system, we investigate which combination of the two ways of identifying element answers and the two XML-specific heuristics for ranking the answers yields the best effectiveness for element retrieval.

[1] `http://www.seg.rmit.edu.au/zettair/`
[2] `http://exist-db.org/`

2 Hybrid XML Retrieval

In this section, we describe the three information retrieval models implemented in Zettair, the two algorithms for identifying the CREs, and the two heuristics for ranking the CREs, all of which are used by our hybrid system.

2.1 Information Retrieval Models

The *similarity* of a document to a query, denoted as $S_{q,d}$, indicates how closely the content of the document matches that of the query. To calculate the query-document similarity, statistical information about the distribution of the query terms — within both the document and the collection as a whole — is often necessary. These term statistics are subsequently utilised by the similarity measure. Following the notation and definitions of Zobel and Moffat [14], we define the basic term statistics as:

- q, a query;
- t, a query term;
- d, a document;
- $N_{\mathcal{D}}$, the number of all the documents in the collection;
- For each term t:
  - $f_{d,t}$, the frequency of t in the document d;
  - $N_{\mathcal{D}_t}$, the number of documents containing the term t (irrespective of the term frequency in each document); and
  - $f_{q,t}$, the frequency of t in query q.
- For each document d:
  - $f_d = |d|$, the document length approximation.
- For the query q:
  - $f_q = |q|$, the query length.

We also denote the following sets:

- $\mathcal{D}$, the set of all the documents in the collection;
- $\mathcal{D}_t$, the set of documents containing term t;
- $\mathcal{T}_d$, the set of distinct terms in the document d;
- $\mathcal{T}_q$, the set of distinct terms in the query, and $\mathcal{T}_{q,d} = \mathcal{T}_q \cap \mathcal{T}_d$.

Vector-Space Model In the vector-space model, both the document and the query are representations of n-dimensional vectors, where n is the number of distinct terms observed in the document collection. The best-known technique for computing similarity under the vector-space model is the cosine measure, where the similarity between a document and the query is computed as the cosine of the angle between their vectors.

Zettair uses pivoted cosine document length normalisation [10] to compute the query-document similarity under the vector-space model:

$$S_{q,d} = \frac{1}{W_D \times W_q} \times \sum_{t \in \mathcal{T}_{q,d}} (1 + \log_e f_{d,t}) \times \log_e \left(1 + \frac{N_{\mathcal{D}}}{N_{\mathcal{D}_t}}\right)$$

where $W_D = \left((1.0 - s) + s \times \frac{W_d}{W_{AL}}\right)$ represents the pivoted document length normalisation, and W_q is the query length representation. The parameter s represents the *slope* (we use the value of 0.25), whereas W_d and W_{AL} represent the document length (usually taken as f_d) and the average document length (over all documents in $\mathcal{D}$), respectively.

Probabilistic Model Probabilistic models of information retrieval are based on the principle that documents should be ranked by decreasing probability of their relevance to the expressed information need. Zettair uses the Okapi BM25 probabilistic model developed by Sparck Jones et al. [11], which has proved highly successful in a wide range of experiments:

$$S_{q,d} = \sum_{t \in \mathcal{T}_{q,d}} w_t \times \frac{(k_1 + 1) f_{d,t}}{K + f_{d,t}} \times \frac{(k_3 + 1) f_{q,t}}{k_3 + f_{q,t}}$$

where $w_t = \log_e \left(\frac{N_{\mathcal{D}} - N_{\mathcal{D}_t} + 0.5}{N_{\mathcal{D}_t} + 0.5}\right)$ is a representation of inverse document frequency, $K = k_1 \times \left[(1 - b) + \frac{b \cdot W_d}{W_{AL}}\right]$, and k_1, b and k_3 are constants, in the range 1.2 to 1.5 (we use 1.2), 0.6 to 0.75 (we use 0.75), and 1,000 (effectively infinite), respectively. W_d and W_{AL} represent the document length and the average document length.

Language Model Language models are probability distributions that aim to capture the statistical regularities of natural language use. In information retrieval, language modelling involves estimating the likelihood that both the document and the query could have been generated by the same language model. Zettair uses a query likelihood approach with Dirichlet smoothing [13]:

$$S_{q,d} = f_q \times \log \lambda_d + \sum_{t \in \mathcal{T}_{q,d}} \log \left(\frac{N_{\mathcal{D}} \times f_{d,t}}{\mu \times N_{\mathcal{D}_t}} + 1\right)$$

where μ is a smoothing parameter (we use the value of 2,000), while λ_d is calculated as: $\lambda_d = \mu / (\mu + f_d)$.

2.2 Identifying the Appropriate Element Granularity

For each INEX topic (`CO`, `+S`, or `CAS`), a topic translation module is first used to automatically translate the underlying information need into a Zettair query. A list of (up to) 500 `article` elements — presented in descending order of

Table 1. eXist list of matching elements for INEX 2005 CO topic 203 and article `co/2000/r7108`. The elements in the list are generated by using an eXist `OR` query

Article	Matching element
co/2000/r7108	/article[1]/bdy[1]/sec[1]/ip1[1]
co/2000/r7108	/article[1]/bdy[1]/sec[1]/p[1]
co/2000/r7108	/article[1]/bdy[1]/sec[2]/st[1]
co/2000/r7108	/article[1]/bdy[1]/sec[2]/p[2]
co/2000/r7108	/article[1]/bdy[1]/sec[2]/p[3]
co/2000/r7108	/article[1]/bdy[1]/sec[2]/p[4]
co/2000/r7108	/article[1]/bdy[1]/sec[4]/p[1]
co/2000/r7108	/article[1]/bdy[1]/sec[6]/ip1[1]
co/2000/r7108	/article[1]/bm[1]/app[1]/p[2]
co/2000/r7108	/article[1]/bm[1]/app[1]/p[3]
co/2000/r7108	/article[1]/bm[1]/app[1]/p[4]

estimated likelihood of relevance — is then returned as a resulting answer list for the INEX topic[3].

To retrieve *elements* rather than full articles, a second topic translation module is used to formulate the eXist query. Depending on the topic type, either terms alone, or both terms and structural query constraints from the INEX topic are used to formulate the eXist query. We use the eXist `OR` query operator to generate the element answer list. The answer list contains (up to) 1,500 matching elements, which are taken from articles that appear *highest* in the ranked list of articles previously returned by Zettair.

Consider the eXist answer list shown in Table 1. It shows matching elements for the CO topic 203 after the eXist `OR` query operator is used (each matching element in the list therefore contains *one* or *more* query terms). The matching elements in the eXist answer list represent most specific (leaf) elements, and eXist correctly presents these elements in document order.

To effectively utilise the information contained in the resulting list of matching elements, we use a retrieval module capable of identifying the appropriate *granularity* of elements to return as answers, which we refer to as *Coherent Retrieval Elements* (CREs) [9]. To identify the CREs, our module first sequentially processes the list of matching elements, starting from the first element down to the last. For each pair of matching elements, their *most specific ancestor* is chosen to represent an answer element (a CRE). We denote these answer elements as `oCRE` elements.

The rationale behind choosing only `oCRE` elements as answers stems from the expectation that these elements are likely to provide better context for the contained textual information than that provided by each of their descendent leaf elements. However, it is often the case that relevance judgements for INEX

[3] We retrieve (up to) 500 rather than 1,500 articles because roughly that number of articles is used to generate the pool of retrieved articles for relevance judgements.

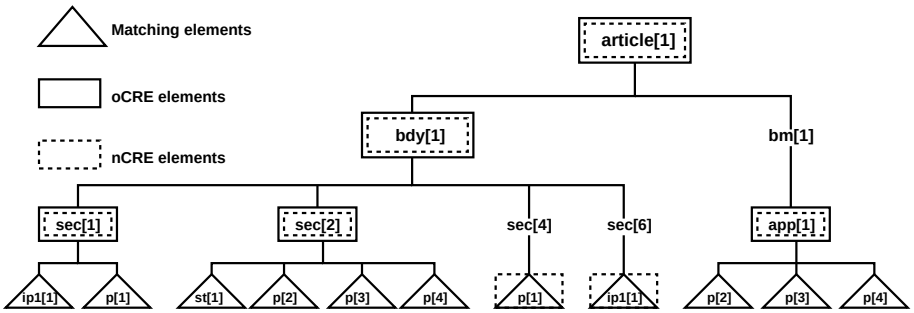

Fig. 2. Identifying appropriate element granularity: Matching, `oCRE`, and `nCRE` elements for INEX 2005 topic 203 and article `co/2000/r7108`

topics contain very specific answer elements [4, 9]. Therefore, the problem with only presenting the `oCRE` elements as answers is that in most cases the matching (and thus very specific) elements are *not included* in the final answer list. To cater for this, our retrieval module supports a second, alternative algorithm for identifying the CREs. The difference from the original `oCRE` algorithm is that, after sequentially processing all the pairs of matching elements, those matching elements whose *immediate parents* are not identified as CREs are also included in the final list of answers. We expect these newly included matching elements to allow for more focussed retrieval. We denote these answer elements as `nCRE` elements.

Figure 2 shows a tree representation of the eXist list of matching elements, as previously shown in Table 1. The matching elements appear within the triangle boxes, the `oCRE` elements appear within the solid square boxes, while the `nCRE` elements appear within dashed square boxes. Once the CREs are identified, we use heuristics to *rank* and present the answer elements according to their *estimated likelihood of relevance*.

2.3 Ranking the Answer Elements

In whole document retrieval, Anh and Moffat [1] present an empirical analysis which reveals that, to maximise query effectiveness, it is very important that answer documents contain most of the query terms. To explore the validity of the above finding for XML retrieval, we consider the following ranking heuristics in our CRE module:

1. The number of distinct query terms that appear in a CRE — more distinct query term appearances (`T`) or fewer distinct query term appearances (`t`);
2. The length of the absolute path of the CRE, taken from the root element — longer path (`P`) or shorter path (`p`); and
3. The frequency of all the query terms in a CRE — more frequent (`F`) or less frequent (`f`).

Table 2. Rank orderings of retrieved `nCRE` elements using two ranking heuristic combinations (`TPF` and `PTF2`) for article `co/2000/r7108`. The query used is "code signing verification", which represents the `title` part of the INEX 2005 topic 203

Rank	`TPF` ordering	`PTF2` ordering
1	/article[1]/bdy[1]/sec[2]	/article[1]/bdy[1]/sec[6]/ip1[1]
2	/article[1]/bdy[1]	/article[1]/bdy[1]/sec[2]
3	/article[1]	/article[1]/bm[1]/app[1]
4	/article[1]/bdy[1]/sec[6]/ip1[1]	/article[1]/bdy[1]/sec[1]
5	/article[1]/bm[1]/app[1]	/article[1]/bdy[1]
6	/article[1]/bdy[1]/sec[1]	/article[1]
7	/article[1]/bdy[1]/sec[4]/p[1]	/article[1]/bdy[1]/sec[4]/p[1]

Preliminary experiments using the INEX 2004 test collection show that two heuristic combinations — `TPF` and a modification of `PTF` — perform better than others in the case where more *specific* elements are target of retrieval. The two heuristic combinations can be interpreted as follows.

With `TPF`, the CREs are first sorted in a descending order according to the number of distinct query terms a CRE contains (the more distinct query terms it contains, the higher its rank). Next, if two CREs contain the same number of distinct query terms, the one with the longer length of its absolute path is ranked higher. Last, if the lengths of the two absolute paths are the same, the CRE with more frequent query term appearances is ranked higher than the CRE where query terms appear less frequently. The ranked list of CREs obtained by using the `TPF` ranking heuristic for article `co/2000/r7108` and the INEX 2005 topic 203 is shown in Table 2.

The table shows that when the `TPF` heuristic is used, less specific CREs tend to be preferred over more specific ones. To produce more specific CREs early in the ranking, the `PTF` ranking heuristic could be used. With `PTF`, the CREs are first sorted in a descending order according to the length of the absolute path of a CRE (where the longer CRE path results in a higher rank). Next, if the lengths of the two absolute paths are the same, the CRE that contains a larger number of distinct query terms is ranked higher. Last, if it also happens that the two CREs contain the same number of distinct query terms, the CRE with more frequent query term appearances is ranked higher. However, our experiments on the INEX 2004 test collection demonstrate that the system performance degrades when using the `PTF` ranking heuristic, since most highly ranked (and thus very specific) elements typically contain only one query term. We therefore use a modification of this heuristic in our retrieval module to ensure that all CREs that contain exactly one query term are moved to the end of the ranked list (where ties are broken by the `F` heuristic). We denote this modified heuristic combination as `PTF2`. The ranked list of CREs obtained by using the `PTF2` ranking heuristic for article `co/2000/r7108` and the INEX 2005 topic 203 is also shown in Table 2.

3 Experiments and Results

In this section, we present results of experiments that evaluate the performance of our INEX 2005 runs for each retrieval strategy in both the `CO+S` and `CAS` sub-tasks. A description of each of our submitted runs is provided in Table 3.

3.1 Evaluation Metrics

A new set of metrics is adopted in INEX 2005, which belong to the eXtended Cumulated Gain (XCG) family of metrics [6]. We use the following two official INEX 2005 metrics [5] to measure the retrieval effectiveness of our runs:

1. `nxCG`, with the `nxCG[r]` measure. The `genLifted` quantisation function is used with `nxCG` with the following values for the rank `r`: 10, 25, and 50. We choose this because with `genLifted` quantisation all the relevant elements — including the so-called *too small* elements — are considered during evaluation (which is not the case with `gen` quantisation) [5]. The three values for the rank `r` are officially reported on the INEX 2005 web site.
2. `ep/gr`, with the `MAep` measure. Both `strict` and `genLifted` quantisations are used with `ep/gr`.

In addition to the above metrics, we also report values obtained with `HiXEval`: an alternative evaluation metric for XML retrieval that is solely based on the amount of highlighted relevant information for an INEX 2005 topic. The reported values are: `P@r`, or the proportion of relevant information to all the information retrieved at a rank `r`; and `MAP`, the mean average precision calculated at natural recall levels [8].

3.2 `CO+S` Sub-Task

`Thorough` Retrieval Strategy The evaluation results of our INEX 2005 `CO+S` runs for this strategy are shown in Table 4. Here, the level of overlap among retrieved elements is not considered. Several observations can be drawn from these results.

First, when comparing the two algorithms on how well they identify answer elements, results for the `CO` runs obtained from the three metrics show that better overall performance is achieved with the `oCRE` algorithm than with `nCRE`. This finding suggests that, for the `Thorough` retrieval strategy, systems capable of only retrieving contextual answers are better rewarded than systems that additionally retrieve more specific elements as answers. Second, with the `nCRE` algorithm for identifying answer elements, the `TPF` ranking heuristic — which first presents those answers that contain most of the distinct query terms, irrespective of how specific these answers are — is consistently better than the `PTF2` ranking heuristic that presents more specific answers first. Finally, when comparing each `CO` run with its corresponding `+S` run, the obtained results show that using structural hints from `+S` topics does not result in better overall performance, although runs using the `nCRE` algorithm seem to benefit from the structural hints at ten or fewer elements returned.

Table 3. List of the 26 CO+S and CAS runs submitted by our RMIT University group to the INEX 2005 ad hoc track

Run ID	Topic Type	Topic Interpretation	Similarity Measure	Answer Elements	Ranking Heuristic	Overlap Allowed
CO+S.Thorough						
nCRE-CO-PTF2	CO	CO	Okapi	nCRE	PTF2	Yes
nCRE-+S-PTF2	+S	SS	Okapi	nCRE	PTF2	Yes
nCRE-CO-TPF	CO	CO	Okapi	nCRE	TPF	Yes
nCRE-+S-TPF	+S	SS	Okapi	nCRE	TPF	Yes
oCRE-CO-PTF2	CO	CO	Okapi	oCRE	PTF2	Yes
oCRE-+S-PTF2	+S	SS	Okapi	oCRE	PTF2	Yes
CO+S.Focussed						
nCRE-CO-PTF2-NO	CO	CO	Okapi	nCRE	PTF2	No
nCRE-+S-PTF2-NO	+S	SS	Okapi	nCRE	PTF2	No
nCRE-CO-TPF-NO	CO	CO	Okapi	nCRE	TPF	No
nCRE-+S-TPF-NO	+S	SS	Okapi	nCRE	TPF	No
oCRE-CO-PTF2-NO	CO	CO	Okapi	oCRE	PTF2	No
oCRE-+S-PTF2-NO	+S	SS	Okapi	oCRE	PTF2	No
CO+S.FetchBrowse						
Okapi-CO-PTF2	CO	CO	Okapi	nCRE	PTF2	Yes
Okapi-+S-PTF2	+S	SS	Okapi	nCRE	PTF2	Yes
PCosine-CO-PTF2	CO	CO	PCosine	nCRE	PTF2	Yes
PCosine-+S-PTF2	+S	SS	PCosine	nCRE	PTF2	Yes
Dirichlet-CO-PTF2	CO	CO	Dirichlet	nCRE	PTF2	Yes
Dirichlet-+S-PTF2	+S	SS	Dirichlet	nCRE	PTF2	Yes
SSCAS						
SS-PTF2	CAS	SS	Okapi	—	PTF2	Yes
SS-TPF	CAS	SS	Okapi	—	TPF	Yes
SVCAS						
SV-PTF2	CAS	SV	Okapi	—	PTF2	Yes
SV-TPF	CAS	SV	Okapi	—	TPF	Yes
VSCAS						
nCRE-VS-PTF2	CAS	VS	Okapi	nCRE	PTF2	Yes
nCRE-VS-TPF	CAS	VS	Okapi	nCRE	TPF	Yes
VVCAS						
nCRE-VV-PTF2	CAS	VV	Okapi	nCRE	PTF2	Yes
nCRE-VV-TPF	CAS	VV	Okapi	nCRE	TPF	Yes

Table 4. Evaluation results of our INEX 2005 CO and +S runs for the Thorough retrieval strategy, obtained with nxCG, ep/gr, and HiXEval, using the genLifted quantisation function with nxCG. The three metrics do not consider the amount of overlap between retrieved elements (setting: off). For each evaluation measure, the best performing CO run (the first of each pair of runs) is shown in bold

	nxCG[rank]			**ep/gr** (MAep)		**HiXEval**			
Run	10	25	50	genLifted	strict	P@10	P@25	P@50	MAP
nCRE-CO-PTF2	0.200	0.212	0.193	**0.019**	0.008	0.256	0.254	**0.216**	0.072
nCRE-+S-PTF2	0.211	0.158	0.145	0.014	0.008	0.245	0.181	0.158	0.050
nCRE-CO-TPF	0.218	0.226	0.193	**0.019**	0.009	0.262	**0.265**	**0.216**	0.073
nCRE-+S-TPF	0.224	0.166	0.1145	0.014	0.009	0.263	0.191	0.159	0.051
oCRE-CO-PTF2	**0.220**	**0.227**	**0.196**	**0.019**	**0.010**	**0.263**	0.258	0.204	**0.083**
oCRE-+S-PTF2	0.210	0.166	0.139	0.012	0.009	0.240	0.191	0.145	0.053

Focussed **Retrieval Strategy** Table 5 shows evaluation results of our INEX 2005 CO+S runs for the Focussed retrieval strategy. Contrary to the Thorough retrieval strategy, in this case the amount of overlap between retrieved elements is considered by all the metrics. To filter overlap, we use a top-down filtering approach where elements that either contain or are contained by any element residing higher in the ranked list are removed from the resulting answer list.

When comparing our two algorithms on how well they identify answer elements, we observe that with HiXEval the oCRE algorithm overall performs better than the nCRE algorithm. However, the results obtained using MAep with both quantisations show the opposite. This suggests that the most specific elements that are retained as answers by nCRE bring additional user gain in the Focussed retrieval strategy. With the nCRE algorithm for identifying answer elements, we observe that with all but two evaluation measures the PTF2 ranking heuristic performs better than TPF.

For each of the three non-overlapping CO runs, the results obtained with ep/gr show that using structural hints from the +S topics results in increased overall retrieval performance. With HiXEval, however, this improvement is only visible when measuring the overall performance of the nCRE runs, which suggests that structural hints are not useful for runs that contain non-overlapping and *contextual* elements. The nature of the XML retrieval task, therefore, seems to influence how structural hints in the INEX +S topics should be interpreted. More precisely, using structural hints from the INEX +S topics seems to be more useful for Focussed than for the Thorough retrieval strategy.

FetchBrowse **Retrieval Strategy** The evaluation methodology for this strategy is different than the ones that were used for the previous two strategies since two separate evaluation results are calculated: an article-level result and an element-level result [5].

Table 5. Evaluation results of our INEX 2005 CO and +S runs for the Focussed retrieval strategy, obtained with nxCG, ep/gr, and HiXEval, using the genLifted quantisation function with nxCG. The three metrics do consider the amount of overlap between retrieved elements (setting: on). For each evaluation measure, the best performing CO run (the first of each pair of runs) is shown in bold

	nxCG[rank]			ep/gr (MAep)		HiXEval			
Run	10	25	50	genLifted	strict	P@10	P@25	P@50	MAP
nCRE-CO-PTF2-NO	**0.044**	0.041	0.048	**0.012**	**0.012**	0.264	0.240	**0.191**	0.104
nCRE-+S-PTF2-NO	0.044	0.041	0.060	0.014	0.014	0.252	0.193	0.146	0.112
nCRE-CO-TPF-NO	0.040	0.041	**0.054**	0.011	0.011	0.248	0.212	0.177	0.117
nCRE-+S-TPF-NO	0.040	0.039	0.067	0.012	0.012	0.249	0.168	0.136	0.120
oCRE-CO-PTF2-NO	0.031	**0.052**	0.050	0.011	0.011	**0.298**	**0.249**	0.189	**0.118**
oCRE-+S-PTF2-NO	0.021	0.045	0.060	0.013	0.013	0.256	0.188	0.137	0.112

By measuring the article-level results obtained from our three FetchBrowse runs, we aim to find which of the three information retrieval models implemented in Zettair yields the best performance for full article retrieval. The graph in Fig. 3 shows the results of this analysis. We observe that highest *effort-precision* at 0.3 or less *gain-recall* is achieved with the Okapi similarity measure, which also produces highest value for MAep among the three measures. Of the other two implemented measures, Dirichlet seems to perform better overall than PCosine. When compared with their corresponding +S runs, all the similarity measures except Dirichlet produce higher MAep values for +S runs than for runs that use plain text queries.

By measuring the element-level results obtained from our three FetchBrowse runs, we aim to investigate the extent to which each of the three information retrieval models influences the system performance for element retrieval. Table 6 shows results for the FetchBrowse retrieval strategy when elements are units of retrieval. The evaluation methodology implemented in the ep/gr metric for this strategy is explained by Kazai and Lalmas [5]. The two metrics, ep/gr and HiXEval, use both overlap settings (on,off). The evaluation measures used by HiXEval in this case are as follows: Prec, which measures precision at final rank for each article cluster, averaged over all clusters and then over all topics; and MAP, the mean average precision (at natural recall levels) for each article cluster, averaged over all clusters and then over all topics.

Results in Table 6 show that, for FetchBrowse element-level retrieval, the PCosine-CO-PTF2 run yields the highest retrieval performance among the three CO runs, irrespective of the metric *and* the overlap setting used. However, when the performance of each CO run is compared to that of its corresponding +S run, we observe that the overlap setting *does* have an impact on the measured comparison, but only when the HiXEval metric is used. When the amount of overlap between retrieved elements is not considered, the results obtained from both metrics show that the structural hints found in the +S topics are not useful. How-

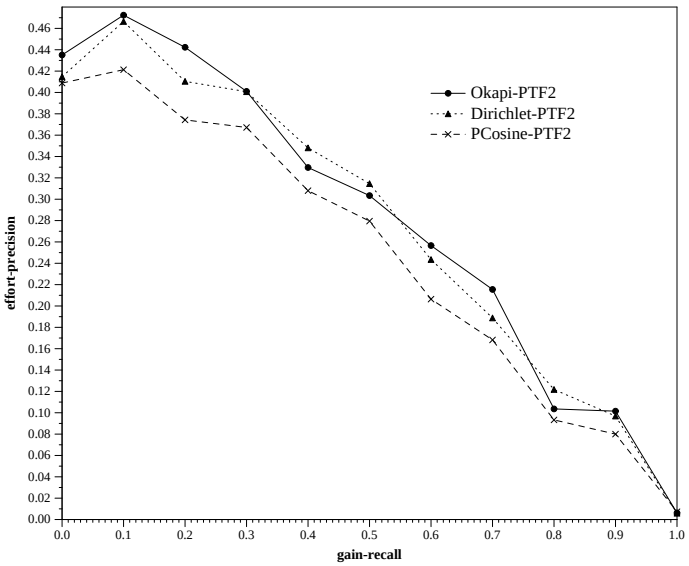

Fig. 3. Evaluation results of our INEX 2005 `CO` runs for the `FetchBrowse` article-level retrieval strategy, obtained by using the `genLifted` quantisation function in the `ep/gr` INEX 2005 metric

ever, with overlap considered (setting: `on`), the results obtained from the `HiXEval` metric show that using structural hints leads to more precise search, which is reflected in increased values for both `Prec` and `MAP`. This suggests that using structural hints from the INEX `+S` topics is a useful feature in the `FetchBrowse` retrieval strategy, provided that the level of overlap among retrieved elements is considered.

3.3 CAS Sub-Task

Since 2003, there has been much debate among the INEX participants over how to interpret the structure component of a CAS topic. For instance, at INEX 2003 and 2004 there were two interpretations: SCAS, which allows for a strict interpretation of the target element; and VCAS, which allows for the target element to be interpreted vaguely. However, none of these interpretations consider how the *support* elements of the CAS topic should be interpreted. Consequently, four retrieval strategies were explored in the INEX 2005 `CAS` sub-task: `SS`, `SV`, `VS`, and `VV`, which represent the four possible combinations of interpreting both the target and support elements.

Trotman and Lalmas [12] perform an extensive analysis of all the INEX 2005 runs that were submitted for the `CAS` sub-task, which reveals that those retrieval strategies that share the same interpretation of the target element correlate. In this section, we confirm their findings with our own `CAS` runs, by submitting the

Table 6. Evaluation results of our INEX 2005 CO and +S runs for the FetchBrowse-Element retrieval strategy. The two metrics, ep/gr and HiXEval, use both overlap settings (on,off). For an evaluation measure and an overlap setting, the best performing CO run (the first of each pair of runs) is shown in bold

	overlap=off				overlap=on			
	ep/gr (MAep)		HiXEval		ep/gr (MAep)		HiXEval	
Run	genLifted	strict	Prec	MAP	genLifted	strict	Prec	MAP
Okapi-CO-PTF2	0.024	0.012	0.062	0.023	0.086	0.011	0.041	0.028
Okapi-+S-PTF2	0.014	0.007	0.060	0.018	0.062	0.008	0.047	0.030
PCosine-CO-PTF2	**0.025**	**0.013**	**0.066**	**0.024**	**0.090**	**0.012**	**0.043**	**0.029**
PCosine-+S-PTF2	0.014	0.008	0.063	0.019	0.065	0.009	0.048	0.030
Dirichlet-CO-PTF2	0.023	0.011	0.060	0.023	0.082	0.010	0.041	0.028
Dirichlet-+S-PTF2	0.013	0.006	0.060	0.017	0.058	0.007	0.048	0.030

two SS runs to the SV retrieval strategy (and vice versa), and by also submitting the two VS runs to the VV retrieval strategy (and vice versa).

Table 7 presents the results of our CAS runs for each of the four retrieval strategies using measures from three evaluation metrics, where the amount of overlap between retrieved elements is not considered (overlap setting: off). For the two retrieval strategies that strictly interpret the target element (SS and SV), we observe that — regardless of the evaluation measure or metric used — the best performing run for the SS strategy, when submitted to the SV strategy, again performs best. On the other hand, we observe similar (but not the identical) behaviour for the two retrieval strategies that allow for a vague interpretation of the target element (VS and VV). More precisely, with nxCG and ep/gr the best performing run in the VS strategy also performs best when submitted to the VV strategy, whereas with HiXEval this is only true with P@50 and MAP measures.

Table 7 also shows that the performance of our CAS runs that use the TPF ranking heuristic is consistently higher than that of runs using the PTF2 heuristic, regardless of the retrieval strategy, evaluation measure, or metric used.

4 Conclusions

In this paper we have reported on our participation in the INEX 2005 ad-hoc track. We have tested three information retrieval models, two ways of identifying the appropriate element granularity, and two XML-specific ranking heuristics under different retrieval strategies in both the CO+S and CAS sub-tasks.

For the CO sub-task, better overall performance seems to be achieved when our retrieval module uses only contextual answer elements (oCRE), and not when most specific answer elements (nCRE) are also used. Moreover, the obtained user cumulated gain seems to be higher when the retrieval module uses the ranking heuristic which first presents those answers that contain most of the distinct

Table 7. Evaluation results of our INEX 2005 CAS runs for the SS, SV, VS, and VV retrieval strategies, obtained with nxCG, ep/gr, and HiXEval, using the genLifted quantisation function with nxCG. The three metrics do not consider the amount of overlap between retrieved elements (setting: off). For an evaluation measure and a retrieval strategy, the best performing CAS run is shown in bold

	nxCG[rank]			ep/gr (MAep)		HiXEval			
Run	10	25	50	genLifted	strict	P@10	P@25	P@50	MAP
SSCAS									
SS-PTF2	0.288	0.339	0.360	0.070	0.044	0.184	0.138	0.117	0.055
SS-TPF	**0.316**	**0.345**	**0.368**	**0.071**	0.045	**0.208**	**0.143**	**0.124**	**0.057**
SV-PTF2	0.194	0.177	0.197	0.052	0.062	0.150	0.126	0.114	0.052
SV-TPF	0.229	0.187	0.206	0.053	**0.063**	0.185	0.134	0.121	0.055
SVCAS									
SS-PTF2	0.214	0.191	0.229	**0.065**	0.066	0.207	0.154	0.133	0.061
SS-TPF	**0.243**	**0.195**	**0.236**	**0.065**	**0.068**	**0.237**	**0.159**	**0.140**	**0.062**
SV-PTF2	0.135	0.127	0.157	0.040	0.049	0.131	0.113	0.105	0.047
SV-TPF	0.169	0.138	0.164	0.041	0.051	0.169	0.122	0.112	0.049
VSCAS									
nCRE-VS-PTF2	0.129	0.125	0.101	0.008	0.004	0.124	0.113	0.091	0.029
nCRE-VS-TPF	**0.230**	**0.144**	**0.113**	0.009	0.005	**0.210**	**0.128**	0.101	0.032
nCRE-VV-PTF2	0.098	0.108	0.100	0.011	0.006	0.097	0.100	0.092	0.046
nCRE-VV-TPF	0.198	0.127	0.112	**0.012**	**0.007**	0.183	0.114	**0.102**	**0.049**
VVCAS									
nCRE-VS-PTF2	0.164	0.171	0.142	0.006	0.005	0.204	0.187	0.153	0.042
nCRE-VS-TPF	**0.265**	**0.188**	**0.152**	0.007	0.005	0.278	0.198	0.162	0.044
nCRE-VV-PTF2	0.109	0.135	0.138	0.009	0.007	0.168	0.191	0.175	0.058
nCRE-VV-TPF	0.249	0.183	0.149	**0.010**	**0.008**	**0.286**	**0.215**	**0.186**	**0.063**

query terms (TPF) than the ranking heuristic that presents more specific answers first (PTF2), although for the Focussed retrieval strategy the gain seems to be higher with PTF2. Using structural hints in the +S topics does not lead to more precise search; however, we have observed that structural hints improve both early and overall precision only for those retrieval strategies that do not allow retrieving overlapping elements. More specifically, Focussed retrieval strategy seems to benefit more from the structural hints than FetchBrowse, while there is no visible performance improvement for the Thorough retrieval strategy.

For the CAS sub-task we have observed that, regardless of the way the constraints in a CAS topic are interpreted, the TPF ranking heuristic produces consistently better performance than the PTF2 ranking heuristic. Importantly, for the CAS sub-task we have verified the previous finding by Trotman and Lalmas [12] that the structure component of an INEX CAS topic should only be interpreted in two different ways: one that allows for strict interpretation of the target element, and another that allows for its vague interpretation.

In the future, we plan to extend this work by implementing and experimenting with different combinations of information and data retrieval models in eXist to allow for more effective as well as more efficient XML retrieval.

References

1. V. N. Anh and A. Moffat. Impact transformation: Effective and efficient web retrieval. In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 3–10, Tampere, Finland, 2002.
2. Y. Chiaramella, P. Mulhem, and F. Fourel. A model for multimedia information retrieval. Technical report, FERMI ESPRIT BRA 8134, University of Glasgow, April 1996.
3. D.N.F. Awang Iskandar, J. Pehcevski, J. A. Thom, and S. M. M. Tahaghoghi. Combining image and structured text retrieval. In *INEX 2005 Workshop Pre-Proceedings, Dagstuhl, Germany, November 28–30, 2005*, pages 365–372, 2005.
4. K. Hatano, H. Kinutan, M. Watanabe, Y. Mori, M. Yoshikawa, and S. Uemura. Keyword-based XML fragment retrieval: Experimental evaluation based on INEX 2003 relevance assessments. In *Proceedings of the Second International Workshop of the INitiative of the Evaluation of XML Retrieval, INEX 2003, Dagstuhl Castle, Germany, December 15–17, 2003*, pages 81–88, 2004.
5. G. Kazai and M. Lalmas. INEX 2005 evaluation metrics. In *INEX 2005 Workshop Pre-Proceedings, Dagstuhl, Germany, November 28–30, 2005*, pages 401–406, 2005.
6. G. Kazai and M. Lalmas. Notes on what to measure in INEX. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology*, pages 22–38, Glasgow, UK, 2005.
7. M. Lalmas. INEX 2005 retrieval task and result submission specification. In *INEX 2005 Workshop Pre-Proceedings, Dagstuhl, Germany, November 28–30, 2005*, pages 385–390, 2005.
8. J. Pehcevski and J. A. Thom. HiXEval: Highlighting XML retrieval evaluation. In *INEX 2005 Workshop Pre-Proceedings, Dagstuhl, Germany, November 28–30, 2005*, pages 11–24, 2005.
9. J. Pehcevski, J. A. Thom, and A.-M. Vercoustre. Hybrid XML retrieval: Combining information retrieval and a native XML database. *Information Retrieval*, 8(4):571–600, 2005.
10. A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 21–29, Zurich, Switzerland, 1996.
11. K. Sparck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. Parts 1 and 2. *Information Processing and Management*, 36(6):779–840, 2000.
12. A. Trotman and M. Lalmas. The Interpretation of CAS. In *INEX 2005 Workshop Pre-Proceedings, Dagstuhl, Germany, November 28–30, 2005*, pages 40–53, 2005.
13. C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.
14. J. Zobel and A. Moffat. Exploring the similarity space. *ACM SIGIR Forum*, 32(1):18–34, 1998.