Applying the Divergence From Randomness Approach for Content-Only Search in XML Documents

Mohammad Abolhassani and Norbert Fuhr

Institute of Informatics and Interactive Systems, University of Duisburg-Essen,
47048 Duisburg, Germany
{mohasani,fuhr}@is.informatik.uni-duisburg.de

Abstract Content-only retrieval of XML documents deals with the problem of locating the smallest XML elements that satisfy the query. In this paper, we investigate the application of a specific language model for this task, namely Amati's approach of divergence from randomness. First, we investigate different ways for applying this model without modification by redefining the concept of an (atomic) document for the XML setting. However, this approach yields a retrieval quality lower than the best method known before. We improved the retrieval quality through extending the basic model by an additional factor that refers to the hierarchical structure of XML documents.[1]

1 Introduction

As XML document collections become more and more available, there is a growing need for retrieval methods exploiting the specific features of this type of documents. Since XML documents contain explicit information about their logical structure, XML retrieval methods should take into account the structural properties of the documents to be retrieved. One of the two tracks of INEX (initiative for the evaluation of XML retrieval [6]) deals with *content-only queries*, where only the requested content is specified. Instead of retrieving whole documents, the IR system should aim at selecting document components that fulfil the information need. Following the FERMI model [3], these components should be the deepest components in the document structure, i. e. most specific, while remaining exhaustive to the information need.

Whereas classical IR models have treated documents as atomic units, XML markup implies a tree-like structure of documents. Content-only queries now search for subtrees of minimum size that are relevant to the query. In order to address this problem, most approaches are based on the notion of the so-called *index nodes* (or index elements): Given the XML markup, not every XML element should be considered as a possible answer, e.g. because the element is too fine-grained or it is missing important elements,

[1] The work presented in this paper is founded by the German Research Foundation (DFG), as part of the CLASSIX project.

like a section body without the section title. So first the set of index nodes has to be defined in some way, e.g. based on the DTD, or document-specific by applying some heuristics. Now there are two possible approaches for addressing the retrieval task:

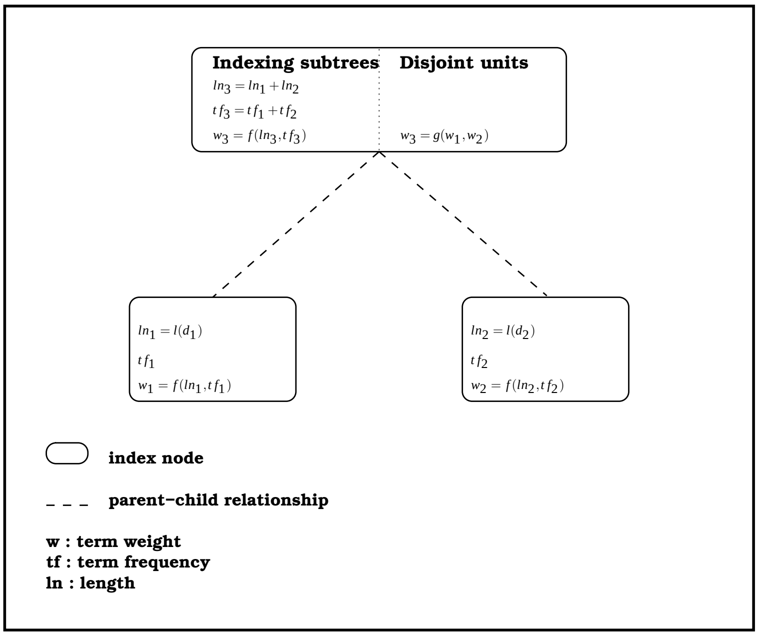

Figure 1. Approaches for computing the indexing weights of inner nodes

Indexing subtrees: The complete text of any index node is treated like an atomic document, and some standard indexing method is applied. Due to the hierarchical structure, index nodes may be contained within each other. In contrast, most indexing methods assume that the collection consists of disjoint text blocks, and so care has to be taken in order to avoid violation of this assumption. Furthermore, the overlapping leads to some redundancy in the index file. Piwowarski et al. propose a Bayesian network approach, where the retrieval weight of an index node also depends on the weights of those nodes in which it is contained [11]. Grabs and Schek apply this idea when the query also involves structural conditions, regarding as collection only those XML elements which are fulfilling the structural conditions [9].

Disjoint units: The document is split into disjoint units, such that the text of each index node is the union of one or more of these disjoint parts. Then standard indexing methods can be applied to the disjoint units, by treating them like atomic documents, where the collection is made up of the units of the documents in the collection. For retrieval, indexing weights for nodes consisting of several units must be aggregated in some way; this makes the retrieval process more complex. Ogilvie and Callan describe a language model following this approach, where the language models of a 'higher level' node is computed as the weighted sum of the language models of its units [10].

Figure 1 illustrates the differences between the two approaches for an example document: The subtree method first collects word statistics (like e.g. document length ln, within-document frequency tf) for the complete text contained in the subtree, and then computes the indexing weight w based on these statistics. In contrast, the disjoint units method first computes indexing weights for the leaf nodes, whereas the weights for the inner nodes are derived from the combination of the weights in the leaf nodes.

Fuhr and Grossjohann describe an augmentation model based on the disjoint units approach [4]. Here indexing weights of units are propagated to the index nodes containing these units. However, when propagating from one index node to the next comprising node, the indexing weights are downweighted by multiplying them with a so-called augmentation factor. The experimental evaluation within INEX [8] showed that this approach leads to top performance among the participating systems. However, the augmentation model makes no assumptions about the underlying indexing model. For the INEX runs, we used the BM25 indexing formula.

In this paper, we present a new model for content-only retrieval which combines the subtree approach with language models. As starting point, we chose Amati's framework of retrieval, called *Divergence From Randomness (DFR)* [1, 2]. We investigate several possibilities for applying this approach to XML retrieval, and combine it also with ideas from the augmentation approach.

The remainder of this paper is structured as follows: First in Section 2 we give a brief survey into Amati's model. Then we investigate the application of this approach to XML retrieval in Section 3. Finally, in Section 4, we give a summary and an outlook on our future work.

2 Divergence From Randomness

Amati and Rijsbergen introduce a framework for deriving probabilistic models of IR [1]. These models are non-parametric models of IR as obtained in the *language model* approach. The term weighting models are derived by measuring the divergence of the actual term distribution from that obtained under a random process.

There are two basic assumptions underlying this approach:

1. Words which bring little information are randomly distributed on the whole set of documents. One can provide different basic probabilistic models, with probability distribution $Prob_1$, that define the notion of *randomness in the context of IR*.
2. If one restrict statistics to the set of all documents in which a term occurs, the “elite” set, then one can derive a new probability $Prob_2$ of the occurrence of the word within a document with respect to its elite set.

Based on these ideas, the weighting formula for a term in a document is the product of the following two factors:

1. $Prob_1$ is used for measuring the *information content* of the term in a document, and $(-\log_2 Prob_1)$ gives the corresponding amount of information.
2. $Prob_2$ is used for measuring the *information gain* of the term with respect to its 'elite' set (the set of all documents in which the term occurs). The less the term is expected in a document with respect to its frequency in the elite set, measured by the counter-probability $(1 - Prob_2)$, the more the amount of information is gained with this term.

Now the weight of a term in a document is defined as

$$w = (1 - Prob_2) \cdot (-\log_2 Prob_1) = Inf_2 \cdot Inf_1 \tag{1}$$

For computing the two probabilities, the following parameters are used:

N number of documents in the collection,
tf term frequency within the document (since different normalisations are applied to the term frequency, we use tf_1 and tf_2 in the following formulas),
n size of the elite set of the term,
F term frequency in elite set.

Furthermore, let $\lambda = F/N$ in the following.

As probability distribution for estimating $Prob_1$, three different probabilistic models are regarded in [1]; using various approximations, this finally leads to seven different formulas. In this paper, we use only two of them:

D The approximation of the binomial model with the divergence:

$$Inf_1 = tf_1 \cdot \log_2 \frac{tf_1}{\lambda} + \left(\lambda + \frac{1}{12tf_1} - tf_1\right) \cdot \log_2 e + 0.5 \log_2(2\pi \cdot tf_1) \tag{2}$$

G The Geometric as limiting form of the Bose-Einstein model:

$$Inf_1 = -log_2 \frac{1}{1+\lambda} - tf_1 \cdot \log_2 \frac{\lambda}{1+\lambda} \tag{3}$$

For the parameter $Inf_2 = (1 - Prob_2)$ (which is also called *first normalisation*), $Prob_2$ is defined as the probability of observing another occurrence of the term in the document, given that we have seen already tf occurrences. For this purpose, Amati regards two approaches:

L Based on Laplace's law of succession, he gets

$$Inf_2 = \frac{1}{tf_2 + 1} \tag{4}$$

B Regarding the ratio of two Bernoulli processes yields

$$Inf_2 = \frac{F+1}{n \cdot (tf_2 + 1)} \tag{5}$$

These parameters do not yet consider the length of the document to be indexed. For the relationship between document length and term frequency, Amati regards two alternative hypotheses concerning the the density function $\rho(l)$ of the term frequency in the document (where c is a constant to be chosen):

H1 The distribution of a term is uniform in the document. The term frequency density $\rho(l)$ is constant; that is $\rho(l) = c$.
H2 The term frequency density $\rho(l)$ is a decreasing function of the length; that is $\rho(l) = c/l$.

In this paper, we also regard the generalisation of these two assumptions:

$$\rho(l) = c \cdot l^{\beta} \tag{6}$$

where β is a parameter to be chosen (we get H1 with $\beta = 0$ and H2 with $\beta = -1$)

In order to consider length normalisation, Amati maps the tf frequency onto a normalised frequency tfn computed in the following way: Let $l(d)$ denote the length of document d and avl is the average length of a document in the collection. Then tfn is defined as:

$$tfn = \int_{l(d)}^{l(d)+avl} \rho(l) dl \tag{7}$$

This approach yields $tfn = tf \cdot \frac{avl}{l(d)}$ for H1 and $tfn = tf \cdot \log_2(1 + \frac{avl}{l(d)})$ for H2.

For considering these normalisations, Amati sets $tf_1 = tf_2 = tfn$ in formulas 2–5 and then computes the term weight according to eqn 1.

For retrieval, the query term weight qtf is set to the number of occurrences of the term in the query. Then a linear retrieval function is applied:

$$R(q,d) = \sum_{t \in q} qtf \cdot Inf_2(tf_2) \cdot Inf_1(tf_1) \tag{8}$$

3 Applying Divergence From Randomness to XML Documents

3.1 Test Setting

For our experiments, we used the INEX test collection [6]. The document collection is made up of the full-texts, marked up in XML, of 12 107 articles of the IEEE Computer Society's publications from 12 magazines and 6 transactions, covering the period of 1995–2002, and totalling 494 megabytes in size. Although the collection is relatively small compared with TREC, it has a suitably complex XML structure (192 different content models in DTD) and contains scientific articles of varying length. On average an article contains 1 532 XML nodes, where the average depth of a node is 6.9 (a more detailed summary can be found in [5]). For our experiments, we defined four levels of index nodes for this DTD, where the following XML elements formed the roots of index nodes: article, section, ss1, ss2 (the latter two elements denote two levels of subsections).

As queries, we used the 24 content-only queries from INEX 2002 for which relevance judgements are available. Figure 2 shows the DTD of the queries applied. As query terms, we considered all single words from the topic title, the description and the keywords section.

For relevance assessments, INEX uses a two-dimensional, multi-valued scale for judging about relevance and coverage of an answer element. In our evaluations described below, recall and precision figures are based on a mapping of this scale onto a one-dimensional, binary scale where only the combination 'fully relevant'/'exact coverage' is treated as relevant and all other combinations as non-relevant. For each query, we considered the top ranking 1000 answer elements in evaluation.[2].

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!-- An inex_topic has 4 parts: title, description, narrative and
keywords, and 3 attributes: the official INEX topic-id, query type
(CO=content-only, CAS=content-and-structure),
and ct-no (candidate topic number) -->

<!ELEMENT inex_topic  (title,description,narrative,keywords)>
<!ATTLIST inex_topic
  topic_id    CDATA  #REQUIRED
  query_type  CDATA  #REQUIRED
  ct_no       CDATA  #REQUIRED
>
<!ELEMENT title (#PCDATA)>
<!ELEMENT description   (#PCDATA)>
<!ELEMENT narrative     (#PCDATA)>
<!ELEMENT keywords      (#PCDATA)>
```

Figure 2. DTD of the INEX 2002 queries

3.2 Direct Application of Amati's Model

In Section 2, we have described the basic model along with a subset of the weighting functions proposed by Amati. Given that we have two different formulas for computing Inf_1 as well as two different ways for computing Inf_2, we have four basic weighting formulas which we are considering in the following.

In a first round of experiments, we tried to apply Amati's model without major changes. However, whereas Amati's model was defined for a set of atomic documents, CO retrieval is searching for so-called *index nodes*, i.e. XML elements that are meaningful units for being returned as retrieval answer.

As starting point, we assumed that the complete collection consists of the concatenation of all XML documents. When we regard a single index node, we assume that the complete collection consists of documents having the same size as our current node.

[2] The official INEX 2002 evaluation was based on the top 100 elements only — for details of the INEX evaluation, see [7].

Table 1. Results from direct application vs. augmentation approach

document length	Dynamic		Fixed	
	B Norm.	L Norm.	B Norm.	L Norm.
Bernoulli	0.0109	0.0356	0.0640	0.0717
Bose-Einstein	0.0214	0.0338	0.0468	0.0606
Augmentation	0.1120			

Table 2. Results from 2nd normalisation with four different values for β

	$\beta = 0$		$\beta = -0.75$		$\beta = -0.80$		$\beta = -1$	
	B Norm.	L Norm.	B Norm.	L Norm.	B Norm.	L Norm.	B Norm.	L Norm.
Bernoulli	0.0391	0.0586	0.0799	0.1026	0.0768	0.1005	0.0640	0.0900
Bose-Einstein	0.0376	0.0609	0.0453	0.0653	0.0448	0.0654	0.0376	0.0651

Let L denote the total length of the collection and $l(d)$ the length of the current node (as above), then we compute the number of hypothetical documents as $N = L/l(d)$.

Table 1 shows the experimental results. The first two result columns show the average precision values for this setting when applying the four different weighting functions. We assume that the poor performance is due to the fact that the weights derived from different document lengths are not comparable.

As an alternative method, we computed the average size of an index node. The two last columns in Table 1 show a much better retrieval quality for this case.

In the subsequent experiments, we focused on the second approach. By referring to the average size of an index node we were also able to apply document length normalisation according to Equation 5. In conformance with H1 and H2 (explained in Section 2) we tried the values 0 and -1 for β. The two first and two last (result) columns of Table 2 show the corresponding results. The results show that length normalisation with $\beta = -1$ improves retrieval quality in most cases. These results were also in conformance with Amati's findings that $\beta = -1$ gives better results than $\beta = 0$.

Subsequently we tried some other values for β. The four middle (result) columns of Table 2 show the corresponding results for $\beta = -0.75$ and $\beta = -0.80$, with which we got better results.

Overall, using a fixed average document length, and length normalisation, gave better results than those achieved in the first round. However, the resulting retrieval quality was still lower than that of the augmentation approach (see Table 1). Thus, in order to arrive at a better retrieval quality, we investigated other ways than straightforward application of Amati's model.

3.3 Considering the Hierarchical Structure of XML Documents

In order to consider the hierarchical structure of our documents, we investigated different ways for incorporating structural parameters within the weighting formula. Considering the basic ideas, as described in Section 2, the most appropriate way seemed the modification of the Inf_2 parameter, which refers to the 'elite' set. Therefore, we computed Inf_1 as above, by performing document length normalisation with respect to the average size of an index node.

For computing Inf_2, we also applied document length normalisation first, thus yielding a normalised term frequency tfn. Then we investigated several methods for 'normalising' this factor with respect to the hierarchical document structure; we call this process *third normalisation*. For this purpose, we introduced an additional parameter $h(d)$ specifying the height (or level) of an index node relative to the root node (which has $h = 1$).

Using the level information, we first tried several heuristic formulas like $tf_2 = tfn \cdot h(d)^{\alpha}$ and $tf_2 = tfn \cdot h(d)^{-\alpha}$, which, however, did not result in any improvements. Finally, we came up with the following formula:

$$tf_2 = tfn \cdot (h(d)/\alpha) \tag{9}$$

Here α is a constant to be chosen, for which we tried several values. However, the experiments showed that the choice of α is not critical.

Table 3. Average precision for the Bose-Einstein L Norm combination with various values of α

α	2	4	9	16	20	32	64	96	104	116	128
prec.	0.0726	0.0865	0.0989	0.1059	0.1077	0.1083	0.1089	0.1094	0.1087	0.1081	0.1077

Table 3 shows the results for the combination of Bose-Einstein and Laplace normalisation, for which we got significant improvements. This variant also gave better results in Amati's experiments.

The significant benefits through the third normalisation are also confirmed by the recall-precision curves shown in Figure 3, where we compare our results (DFR) with and without third normalisation to that of the augmentation approach.

In INEX 2003[3] we used the best configuration according to our experimental results, i.e. Bose-Einstein and L Normalisation with the parameters $\alpha = 96$ and $\beta = -0.80$. This submission ranked high among all submissions. The average precision achieved was 0.0906, while we got 0.1010 through our "augmentation" method with 0.2 as "augmentation factor". Figure 4 shows the recall-precision curves for these two submissions, confirming again that DFR with third normalisation performs almost as well as the augmentation approach.

In order to explain the effect of third normalisation, let us consider the weighting formula for Inf_2 again; we are using the Laplace version of this formula, which yields:

$$Inf_2 = \frac{1}{tfn + 1} \tag{10}$$

Using third normalisation, we now have instead:

$$Inf_2 = \frac{1}{\frac{h(d)}{\alpha} \cdot tfn + 1} \tag{11}$$

[3] http://inex.is.informatik.uni-duisburg.de:2003/

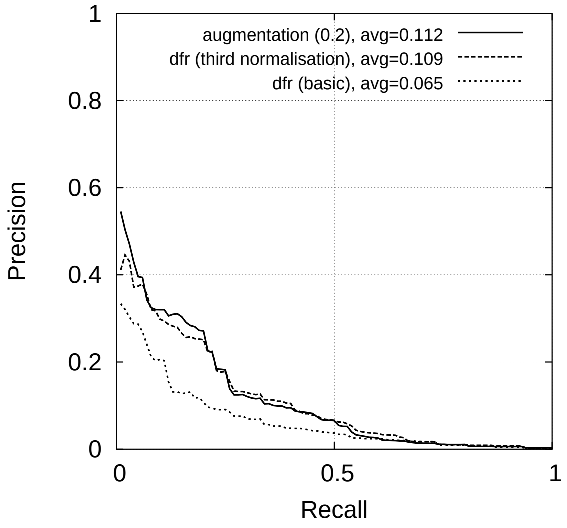

Figure 3. Recall-Precision curves for the best approaches (INEX 2002)

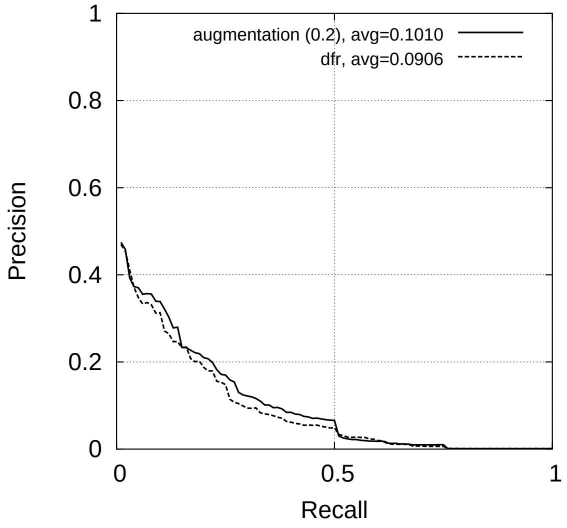

Figure 4. Recall-Precision curves for the INEX 2003 queries

In the latter formula, α controls the influence of the level parameter; for $\alpha = 1$ and $h(d) = 1$, we would get the same results as before.

As described by Amati, Inf_2 measures the 'risk' or (potential) gain if the term is accepted as a descriptor of the document. In both formulas, the gain increases as tf decreases. However, there are two major differences:

1. In general, third normalisation yields a higher gain, since we got the best retrieval performance for values of the constant α which are much higher than those of the level $h(d)$.
2. The risk/gain is higher for smaller levels. This observation conforms to the general goal of the CO queries of the INEX task, where the most specific answers (i.e. those with higher levels) should be preferred. Thus, if the system returns a lower level element, the risk is higher.

4 Conclusions and Outlook

XML retrieval is an important new area for the application of IR methods. Whereas little research has been performed on retrieval of structured documents in the past, the increasing availability of XML collections offers the opportunity for developing appropriate retrieval methods. Content-only retrieval of XML documents corresponds to the classic ad-hoc retrieval task of atomic documents, but with the additional constraint of locating the smallest XML elements that satisfy the query.

In this paper, we have investigated the application of a language model approach for content-only retrieval. We have shown that a straightforward application of language models is possible by appropriate redefinition of the concept of an (atomic) document for the XML setting. In this setting, the experimental results for the different weighting formulas are in line with Amati's findings for the TREC collection. However, the retrieval quality resulting from this direct application was lower than the best results from the first round of INEX.

By adopting ideas from the successful augmentation approach, we have extended Amati's model by a third normalisation component which takes into account the hierarchical structure of XML documents. This approach has improved results, thus leading to a retrieval quality comparable to that of the augmentation approach.

We view our work as a starting point for developing appropriate language models for XML retrieval. In this paper, we have considered only one specific model of this kind, for which we were able to provide an extension yielding a high retrieval quality. In the future, we will study also other language models and investigate different extensions for coping with XML retrieval.

References

[1] **Amati, G.; van Rijsbergen, C.** (2002). Probabilistic models of Information Retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems 20(4)*, pages 357–389.

[2] **Amati, G.** (2003). *Probability Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, University of Glasgow.

[3] **Chiaramella, Y.; Mulhem, P.; Fourel, F.** (1996). *A Model for Multimedia Information Retrieval*. Technical report, FERMI ESPRIT BRA 8134, University of Glasgow.

[4] **Fuhr, N.; Großjohann, K.** (2001). XIRQL: A Query Language for Information Retrieval in XML Documents. In: Croft, W.; Harper, D.; Kraft, D.; Zobel, J. (eds.): *Proceedings of the 24th Annual International Conference on Research and development in Information Retrieval*, pages 172–180. ACM, New York.

[5] **Fuhr, N.; Gövert, N.; Kazai, G.; Lalmas, M.** (2002). INEX: INitiative for the Evaluation of XML Retrieval. In: Baeza-Yates, R.; Fuhr, N.; Maarek, Y. S. (eds.): *Proceedings of the SIGIR 2002 Workshop on XML and Information Retrieval*. http://www.is.informatik.uni-duisburg.de/bib/xml/Fuhr_etal_02a.html.

[6] **Fuhr, N.; Gövert, N.; Kazai, G.; Lalmas, M. (eds.)** (2003). *INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the First INEX Workshop. Dagstuhl, Germany, December 8–11, 2002*, ERCIM Workshop Proceedings, Sophia Antipolis, France. ERCIM. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.

[7] **Gövert, N.; Kazai, G.** (2003). Overview of the INitiative for the Evaluation of XML retrieval (INEX) 2002. In [6], pages 1–17. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.

[8] **Gövert, N.; Fuhr, N.; Abolhassani, M.; Großjohann, K.** (2003). Content-oriented XML retrieval with HyREX. In [6], pages 26–32. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.

[9] **Grabs, T.; Schek, H.-J.** (2003). Flexible Information Retrieval from XML with PowerDB-XML. In [6], pages 141–148. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.

[10] **Ogilvie, P.; Callan, J.** (2003). Language Models and Structure Document Retrieval. In [6], pages 33–40. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.

[11] **Piwowarski, B.; Faure, G.-E.; Gallinari, P.** (2003). Bayesian Networks and INEX. In [6], pages 149–154. http://www.ercim.org/publication/ws-proceedings/INEX2002.pdf.