Aggregation-based Structured Text Retrieval

Theodora Tsikrika
Centrum voor Wiskunde en Informatica
Kruislaan 413
1098 SJ Amsterdam
The Netherlands
Theodora.Tsikrika@cwi.nl

SYNONYMS

None.

DEFINITION

Text retrieval is concerned with the retrieval of documents in response to user queries. This is achieved by (i) representing documents and queries with indexing features that provide a characterisation of their information content, and (ii) defining a function that uses these representations to perform retrieval. Structured text retrieval introduces a finer-grained retrieval paradigm that supports the representation and subsequent retrieval of the individual document components defined by the document's logical structure. *Aggregation-based structured text retrieval* defines (i) the representation of each document component as the aggregation of the representation of its own information content and the representations of information content of its structurally related components, and (ii) retrieval of document components based on these (aggregated) representations.

The aim of aggregation-based approaches is to improve retrieval effectiveness by capturing and exploiting the interrelations among the components of semi-structured text documents. The representation of each component's own information content is generated at indexing time. The recursive aggregation of these representations, which takes place at the level of their indexing features, leads to the generation, either at indexing or at query time, of the representations of those components that are structurally related with other components.

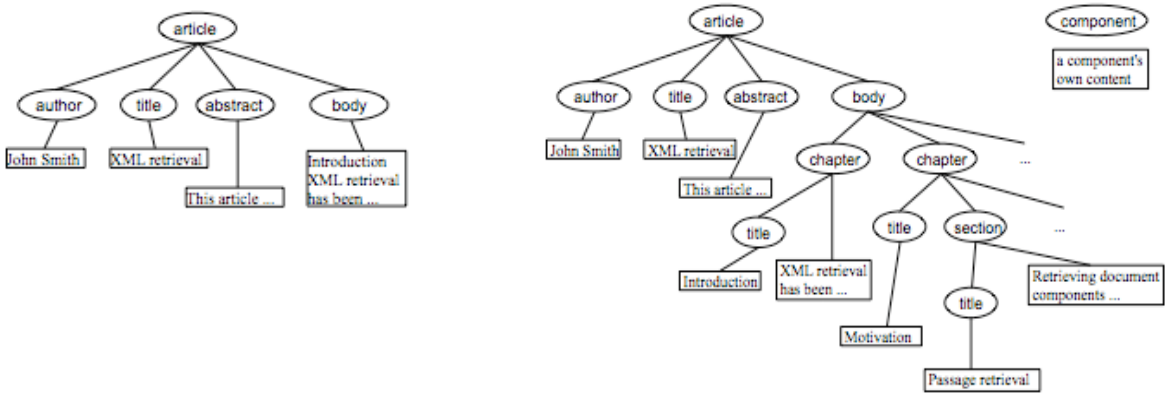
Aggregation can be defined in numerous ways; it is typically defined so that it enables retrieval to focus on those document components more specific to the query or to each document's best entry points, i.e., document components that contain relevant information and from which users can browse to further relevant components.

HISTORICAL BACKGROUND

A well-established Information Retrieval (IR) technique for improving the effectiveness of text retrieval (i.e., retrieval at the document level) has been the generation and subsequent combination of multiple representations for each document [3]. To apply this useful technique to the text retrieval of semi-structured text documents, the typical approach has been to exploit their logical structure and consider that the individual representations of their components can act as the different representations to be combined [11]. This definition of the representation of a semi-structured text document as the combination of the representations of its components was also based on the intuitive idea that the information content of each document consists of the information content of its sub-parts [2, 6].

As the above description suggests, these combination-based approaches, despite restricting retrieval only at the document level, assign representations not only to documents, but also to individual document components. To generate these representations, semi-structured text documents can simply be viewed as series of non-overlapping components (Figure 1(a)), such as title, author, abstract, body, etc. [13]. The proliferation of SGML and XML documents, however, has led to the consideration of hierarchical components (Figure 1(b)), and their interrelated representations [1]. For these (disjoint or nested) document components, the combination of their representations

can take place (i) directly at the level of their indexing features, which typically correspond to terms and their statistics (e.g., [13]), or (ii) at the level of retrieval scores computed independently for each component (e.g., [15]). Overall, these combination-based approaches have proven effective for the text retrieval of semi-structured text documents [15, 11, 13].



(a) Disjoint document components

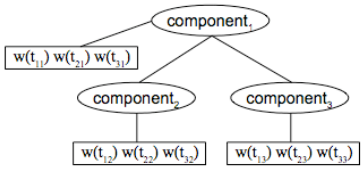
(b) Nested document components

Figure 1: Two views on the logical structure of a semi-structured text document.

Following the recent shift towards the structured text retrieval paradigm [2], which supports the retrieval of document components (including whole documents), it was only natural to try to adapt these combination-based approaches to this new requirement for retrieval at the sub-document level. Here, the focus is on each document component: its representation corresponds to the combination of its own representation with the representations of its structurally related components, and its retrieval is based on this combined representation. Similarly to the case of combination-based approaches for text retrieval, two strands of research can be identified: (i) approaches that operate at the level of the components' indexing features (e.g., [12]), referred to as *aggregation-based structured text retrieval* (described in this entry), and (ii) approaches that operate at the level of retrieval scores computed independently for each component (e.g., [14]), referred to as propagation-based structured text retrieval.
Figure 2(b) illustrates the premise of aggregation- and propagated-based approaches for the simple semi-structured text document depicted in Figure 2(a). Since these approaches share some of their underlying motivations and assumptions, there has been a cross-fertilisation of ideas between the two. This also implies that this entry is closely related to the entry on propagation-based structured text retrieval.

SCIENTIFIC FUNDAMENTALS

Structured text retrieval supports, in principle, the representation and subsequent retrieval of document components of any granularity; in practice, however, it is desirable to take into account only document components that users would find informative in response to their queries [2, 1, 4, 6]. Such document components are referred to as indexing units and are usually chosen (manually or automatically) with respect to the requirements of each application. Once the indexing units have been determined, each can be assigned a representation of its information content, and, hence, become individually retrievable.
Aggregation-based structured text retrieval approaches distinguish two types of indexing units: *atomic* and *composite*. Atomic components correspond to indexing units that cannot be further decomposed, i.e., the leaf components in Figure 1(b). The representation of an atomic component is generated by considering only its own information content. Composite components, on the other hand, i.e., the non-leaf nodes in Figure 1(b), correspond to indexing units which are related to other components, e.g., consist of sub-components. In addition to its own information content, a composite component is also dependent on the information content of its structurally related components. Therefore, its representation can be derived via the **aggregation** of the representation of its own information content with the representations of the information content of its structurally related components; this aggregation takes place at the level of their indexing features. Given the



(a) structured text document: $w()$ an indexing function of term t_i in component j.

AGGREGATION: retrieval_score' ($component_1$) = f($w(t_1)$, $w(t_2)$, $w(t_3)$)

where:
$w(t_1) = w(t_{11}) \oplus w(t_{12}) \oplus w(t_{13})$
$w(t_2) = w(t_{21}) \oplus w(t_{22}) \oplus w(t_{23})$
$w(t_3) = w(t_{31}) \oplus w(t_{32}) \oplus w(t_{33})$

PROPAGATION: retrieval_score' ($component_1$) = retrieval_score ($component_1$) $\oplus$ retrieval_score ($component_2$) $\oplus$ retrieval_score ($component_3$)

where:
retrieval_score ($component_1$) = f($w(t_{11})$, $w(t_{21})$, $w(t_{31})$)
retrieval_score ($component_2$) = f($w(t_{12})$, $w(t_{22})$, $w(t_{32})$)
retrieval_score ($component_3$) = f($w(t_{13})$, $w(t_{23})$, $w(t_{33})$)

(b) Aggregation vs. Propagation: $f()$ a retrieval function and $\oplus$ an aggregation operator.

Figure 2: Simple example illustrating the differences between aggregation- and propagation-based approaches.

representations of atomic components and of composite components' own information content, aggregation-based approaches recursively generate the aggregated representations of composite components and, based on them, perform retrieval of document components of varying granularity.

In summary, each aggregation-based approach needs to define the following: (1) the representation of each component's own information content, (2) the aggregated representations of composite components, and (3) the retrieval function that uses these representations. Although these three steps are clearly interdependent, the major issues addressed in each step need to be outlined first, before proceeding with the description of the key aggregation-based approaches in the field of structured text retrieval.

1. **Representing each component's own information content**: In the field of text retrieval, the issue of representing documents with indexing features that provide a characterisation of their information content has been extensively studied in the context of several IR retrieval models (e.g., Boolean, vector space, probabilistic, language models, etc.). For text documents, these indexing features typically correspond to term statistics. Retrieval functions produce a ranking in response to a user's query, by taking into account the statistics of query terms together with each document's length. The term statistics most commonly used correspond to the *term frequency* $tf(t,d)$ of term t in document d and to the *document frequency* $df(t,C)$ of term t in the document collection C, leading to standard $tf \times idf$ weighting schemes.
Structured text retrieval approaches need to generate representations for all components corresponding to indexing units. Since these components are nested, it is not straightforward to adapt these term statistics (particularly *document frequency*) at the component level [10]. Aggregation-based approaches, on the other hand, directly generate representations only for components that have their own information content, while the representations of the remaining components are obtained via the aggregation process. Therefore, the first step is to generate the representations of atomic components and of the composite components' own information content, i.e., the content not contained in any of their structurally related components. This simplifies the process, since only *disjoint* units need to be represented [6], as illustrated in Figure 3 where the dashed boxes enclose the components to be represented (cf. [5]).
Text retrieval approaches usually consider that the information content of a document corresponds only to its *textual content*, and possibly its metadata (also referred to as *attributes*). In addition to that,

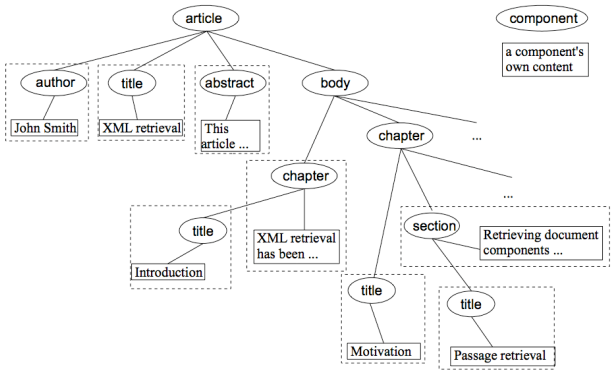


Figure 3: Representing the components that contain their own information.

structured text retrieval approaches also aim at representing the information encoded in the logical structure of documents. Representing this *structural information*, i.e., the interrelations among the documents and their components, enables retrieval in response to both content-only queries and content-and-structure queries.
Aggregation-based approaches that only represent the textual content typically adapt standard representation formalisms widely employed in text retrieval approaches to their requirements for representation at the component level (e.g.,[11, 9]). Those that consider richer representations of information content, apply more expressive formalisms (e.g., various logics [2, 4]).

2. **Aggregating the representations**: The concept underlying aggregation-based approaches is that of *augmentation* [4]: the information content of a document component can be *augmented* with that of its structurally related components. Given the already generated representations (i.e., the representations of atomic components and of composite components' own information content), the augmentation of composite components is performed by the aggregation process.
The first step in the aggregation process is the identification of the structurally related components of each composite component. Three basic types of structural relationships can be distinguished (Figure 4): hierarchical (h), sequential (s), and links (l). Hierarchical connections express the *composition* relationship among components, and induce the tree representing the logical structure of a semi-structured document. Sequential connections capture the order imposed by the document's author(s), whereas links to components of the same or different documents reference (internal or external) sources that offer similar information. In principle, all these types of structural relationships between components can be taken into account by the aggregation process (and some aggregation-based approaches are generic enough to accommodate them, e.g., [7]); in practice, however, the hierarchical structural relations are the only ones usually considered. This leads to the aggregated representations of composite components being recursively generated in an ascending manner.

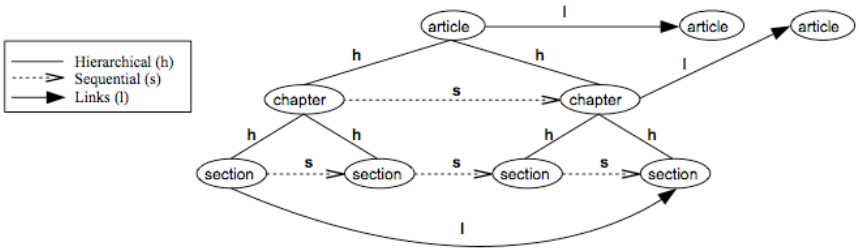


Figure 4: Different types of structural relationships between the components of a semi-structured text document.

The next step is to define the *aggregation operator* (or *aggregation function*). Since the aggregation of the textual content of related components is defined at the level of the indexing features of their representations, the aggregation function is highly dependent on the model (formalism) chosen to represent each component's own

content. This aggregation results in an (aggregated) representation modelled in the same formalism, and can be seen as being performed at two stages (although these are usually combined into one step): the aggregation of *index expressions* [2] (e.g., terms, conjunctions of terms, etc.), and of the *uncertainty* assigned to them (derived mainly by their statistics).
An aggregation function could also take into account: (i) *augmentation factors* [6], which capture the fact that the textual content of the structurally related components of a composite component is not included in that component's own content and has to be "propagated" in order to become part of it, (ii) *accessibility factors* [4], which specify how the representation of a component is influenced by that of its connected components (a measure of the contribution of, say, a section to its embedding chapter [2]), and (iii) the overall importance of a component in a document's structure [7] (e.g., it can be assumed that a title contains more informative content than a small sub-section [13]). Finally, the issue of the possible aggregation of the attributes assigned to related components needs to be addressed [2].
The above aggregation process can take place either at indexing time (*global aggregation*) or ar query time (*local aggregation*). Global aggregation is performed for all composite indexing units and considers all indexing features involved. Since this strategy does not scale well and can quickly become highly inefficient, local aggregation strategies are primarily used. These restrict the aggregation only to indexing features present in the query (i.e., query terms), and, starting from components retrieved in terms of their own information content, perform the aggregation only for these components' ancestors.

3. **Retrieval**: The retrieval function operates both on the representations of atomic components and on the aggregated representations of composite components. Its definition is highly dependent on the formalism employed in modelling these representations. In conjuction with the definition of the aggregation function, the retrieval function operationalises the notion of relevance for a structured text retrieval system. It can, therefore, determine whether retrieval focusses on those document components more specific to the query [2], or whether the aim is to support the users' browsing activities by identifying each document's best entry points [7] (i.e., document components that contain relevant information and from which users can browse to further relevant components).

Aggregation-based approaches
One of the most influential aggregation-based approaches has been developed by Chiaramella et al. [2] in the context of the FERMI project (`http://www.dcs.gla.ac.uk/fermi/`). Aiming at supporting the integration of IR, hypermedia, and database systems, the FERMI model introduced some of the founding principles of structured text retrieval (including the notion of retrieval focussed to the most specific components). It follows the logical view on IR, i.e., it models the retrieval process as inference, and it employs predicate logic as its underlying formalism. The model defines a generic representation of content, attributes, and structural information associated with the indexing units. This allows for rich querying capabilities, including support for both content-only queries and content-and-structured queries. The indexing features of semi-structured text documents can be defined in various ways, e.g., as sets of terms or as logical expressions of terms, while the semantics of the aggregation function depend on this definition. Retrieval can then be performed by a function of the specificity of each component with respect to the query.
The major limitation of the FERMI model is that it does not incorporate the uncertainty inhererent to the representations of content and structure. To address this issue, Lalmas [8] adapted the FERMI model by using propositional logic as its basis, and extended it by modelling the uncertain representation of the textual content of components (estimated by a $tf \times idf$ weighting scheme) using Dempster-Shafer's theory of evidence. The structural information is not explicitly captured by the formalism; therefore, the model does not provide support for content-and-structured queries. The aggregation is performed by Dempster's combination rule, while retrieval is based on the belief values of the query terms.
Fuhr, Gövert, and Rölleke [4] also extended the FERMI model using a combination of (a restricted form of) predicate logic with probabilistic inference. Their model captures the uncertainty in the representations of content, structure, and attributes. Aggregation of index expressions is based on a four-valued logic, allowing for the handling of incomplete information and of inconsistencies arising by the aggregation (e.g., when two components containing contradictory information are aggregated). Aggregation of term weights is performed according to

the rules of probability theory, typically by adopting term independence assumptions. This approach introduced the notion of *accessibility factor* being taken into account. Document components are retrieved based on the computed probabilities of query terms occurring in their (aggregated) representations.

Following its initial development [4], this logic-based probablistic aggregation model was investigated further by Fuhr and his colleagues [5, 6]. They experimented with modelling aggregation by different Boolean operators; for instance, they noted that, given terms propagating in the document tree in a bottom-up fashion, a probabilistic-OR function would always result in higher weights for components further up the hierarchy. As this would lead (in constrast to the objectives of specificity-oriented retrieval) to the more general components being always retrieved, they introduced the notion of *augmentation factors*. These could be used to "downweight" the weights of terms (estimated by a *tf* $\times$ *idf* scheme) that are aggregated in an ascending manner. The effectiveness of their approach has been assessed in the context of the INitiative for the Evaluation of XML retrieval (INEX) [6].

Myaeng et al. [11] also developed an aggregation-based approach based on probabilistic inference. They employ Bayesian networks as the underlying formalism for explicitly modelling the (hierarchical) structural relations between components. The document components are represented as nodes in the network and their relations as (directed) edges. They also capture the uncertainty associated with both textual content (again estimated by *tf* $\times$ *idf* term statistics) and structure. Aggregation is performed by probabilistic inference, and retrieval is based on the computed beliefs. Although this model allows for document component scoring, in its original publication [11] it is evaluated in the context of text retrieval at the document level.

Following the recent widespread application of statistical language models in the field of text retrieval, Ogilvie and Callan [12] adapted them to the requirements of structured text retrieval. To this end, each document component is modelled by a language model; a unigram language model estimates the probability of a term given some text. For atomic components, the language model is estimated by their own text by employing a maximum likelihhod estimate (MLE). For instance, the probability of term t given the language model θ_T of text T in a component can be estimated by: $P(t|\theta_T) = (1-\omega)P_{MLE}(t|\theta_T) + \omega P_{MLE}(t|\theta_{collection})$, where ω is a parameter controlling the amount of smoothing of the background collection model. For composite components $comp_i$, the aggregation of language models is modelled as a linear interpolation: $P(t|\theta'_{comp_i}) = \lambda^{c'}_{comp_i} P(t|\theta_{comp_i}) + \sum_{j \in children(comp_i)} \lambda^c_j P(t|\theta_j)$, where $\lambda^{c'}_{comp_i} + \sum_{j \in children(comp_i)} \lambda^c_j = 1$. These λs model the contribution of each language model (i.e., document component) in the aggregation, while their estimation is a non-trivial issue. Ranking is typically produced by estimating the probability that each component generated the query string (assuming an underlying multinomial model). The major advantage of the language modelling approach is that it provides guidance in performing the aggregation and in estimating the term weights.

A more recent research study has attempted to apply BM25 (one of the most successful text retrieval term weighting schemes) to structured text retrieval. Robertson et al. [13] initially adapted BM25 to semi-structured text documents with non-hierachical components (see Figure 1(a)), while investigating the effectiveness of retrieval at the document level. Next, they [9] adapted BM25 to deal with nested components (see Figure 1(b)), and evaluated it in the context of the INitiative for the Evaluation of XML retrieval (INEX).

A final note on these aggregation-based approaches is that most aim at focussing retrieval on those document components more specific to the query. However, there are approaches that aim at modelling the criteria determining what constitutes a best entry point. For instance, Kazai et al. [7] model aggregation as a fuzzy formalisation of linguistic quantifiers. This means that an indexing feature (term) is considered in an aggregated representation of a composite component, if it represents LQ of its structurally related components, where LQ a linguistic quantifier, such as "at least one", "all", "most", etc. By using these aggregated representations, the retrieval function determines that a component is relevant to a query if LQ of its structurally related components are relevant, in essence implementing different criteria of what can be regarded as a best entry point.

KEY APPLICATIONS

Aggregation-based approaches can be used in any application requiring retrieval according to the structured text retrieval paradigm. In addition, such approaches are also well suited to the retrieval of multimedia documents. These documents can be viewed as consisting of (disjoint or nested) components each containing one or more media. Aggregation can be performed by considering atomic components to only contain a single medium, leading to retrieval of components of varying granularity. This was recognised early in the field of structured text retrieval and some of the initial aggregation-based approaches, e.g., [2, 4], were developed for multimedia environments.

EXPERIMENTAL RESULTS

For most of the presented approaches, particularly for research conducted in the context of the INitiative for the Evaluation of XML retrieval (INEX), there is an accompanying experimental evaluation in the corresponding reference.

DATA SETS

A testbed for the evaluation of structured text retrieval approaches has been developed as part of the efforts of the INitiative for the Evaluation of XML retrieval (INEX) (`http://inex.is.informatik.uni-duisburg.de/`).

URL TO CODE

The aggregation-based approach developed in [12] has been implemented as part of the open source *Lemur* toolkit (for language modeling and IR), available at: `http://www.lemurproject.org/`.

CROSS REFERENCE

Content-only queries

Content-and-structured queries

INitiative for the Evaluation of XML retrieval (INEX)

Indexing units

IR retrieval models

Logical structure

Propagation-based structured text retrieval

Relevance

Specificity

Structured text retrieval ???

Text retrieval

RECOMMENDED READING

Between 3 and 15 citations to important literature, e.g., in journals, conference proceedings, and websites.

[1] Y. Chiaramella. Information retrieval and structured documents. In M. Agosti, F. Crestani, and G. Pasi, editors, *Lectures on Information Retrieval, Third European Summer-School (ESSIR 2000), Revised Lectures*, volume 1980 of *Lecture Notes in Computer Science*, pages 286–309. Springer, 2001.

[2] Y. Chiaramella, P. Mulhem, and F. Fourel. A model for multimedia information retrieval. Technical Report FERMI, ESPRIT BRA 8134, University of Glasgow, 1996.

[3] W. B. Croft. Combining approaches to information retrieval. In W. B. Croft, editor, *Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval*, volume 7 of *The Information Retrieval Series*, pages 1–36. Kluwer Academic Publishers, 2000.

[4] N. Fuhr, N. Gövert, and T. Rölleke. DOLORES: A system for logic-based retrieval of multimedia objects. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 257–265. ACM Press, 1998.

[5] N. Fuhr and K. Großjohann. XIRQL: A query language for information retrieval in XML documents. In W. B. Croft, D. J. Harper, D. H. Kraft, and J. Zobel, editors, *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 172–180. ACM Press, 2001.

[6] N. Gövert, M. Abolhassani, N. Fuhr, and K. Großjohann. Content-oriented XML retrieval with HyREX. In N. Fuhr, N. Gövert, M. Lalmas, and G. Kazai, editors, *Proceedings of the 1st International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2002)*, pages 26–32, 2003.

[7] G. Kazai, M. Lalmas, and T. Rölleke. A model for the representation and focussed retrieval of structured documents based on fuzzy aggregation. In *Proceedings of the 8th International Symposium on String Processing and Information Retrieval (SPIRE 2001)*, pages 123–135. IEEE Computer Society Press, 2001.

[8] M. Lalmas. Dempster-Shafer's theory of evidence applied to structured documents: Modelling uncertainty. In N. J. Belkin, A. D. Narasimhalu, P. Willett, and W. Hersh, editors, *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 110–118. ACM Press, 1997.

[9] W. Lu, S.E. Robertson, and A. MacFarlane. Field-weighted XML retrieval based on BM25. In N. Fuhr, M. Lalmas, S. Malik, and G. Kazai, editors, *Advances in XML Information Retrieval and Evaluation, Proceedings of the 4th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2005), Revised Selected Papers*, volume 3977 of *Lecture Notes in Computer Science*, pages 161–171. Springer, 2006.

[10] Y. Mass and M. Mandelbrod. Retrieving the most relevant XML components. In N. Fuhr, M. Lalmas, and S. Malik, editors, *Proceedings of the 2nd International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2003)*, pages 53–58, 2004.

[11] S.-H. Myaeng, D.-H. Jang, M.-S. Kim, and Z.-C. Zhoo. A flexible model for retrieval of SGML documents. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 138–145. ACM Press, 1998.

[12] P. Ogilvie and J. Callan. Hierarchical language models for retrieval of XML components. In N. Fuhr, M. Lalmas, S. Malik, and Z. Szlávik, editors, *Advances in XML Information Retrieval and Evaluation, Proceedings of the 3rd International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2004), Revised Selected Papers*, volume 3493 of *Lecture Notes in Computer Science*, pages 224–237. Springer, 2005.

[13] S.E. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In D. Grossman, L. Gravano, C.-X. Zhai, O. Herzog, and D. A. Evans, editors, *Proceedings of the 13th ACM International Conference on Information and Knowledge Management (CIKM 2004)*, pages 42–49. ACM Press, 2004.

[14] K. Sauvagnat, M. Boughanem, and C. Chrisment. Searching XML documents using relevance propagation. In A. Apostolico and M. Melucci, editors, *Proceedings of the 11th International Symposium on String Processing and Information Retrieval (SPIRE 2004)*, volume 3246 of *Lecture Notes in Computer Science*, pages 242–254. Springer, 2004.

[15] R. Wilkinson. Effective retrieval of structured documents. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 311–317. Springer, 1994.