Relevancy in Schema Agnostic Environment

Muhammad Ali Norozi
Department of Computer and Information Science
Norwegian University of Science and Technology
Sem Sælands vei 7-9
NO-7491 Trondheim, Norway
+47 7359 3440
mnorozi@idi.ntnu.no

ABSTRACT

Relevance is an important component in full-text search and often distinguishes the implementations. Relevancy is used to score matching documents and rank them according to the users intent. One of the reasons of the high popularity of Google is its good relevancy originally based on the PageRank algorithm.

The emergence of *semi-structured* data as a standard for data representation opened up new areas which could be related to both the database and information retrieval communities. Although the information retrieval and database viewpoints were, until quite recently irreconcilable, semi-structured retrieval helped to bridge the gap. This work is about exploring relevancy in semi-structured retrieval both in isolation and as bridge between database and information retrieval communities.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*semi-structured retrieval*

General Terms

Algorithms, Linear Algebra, Design, Evaluation, Ranking

Keywords

XML retrieval, Semi-structured retrieval, Semi-structured ranking, Schema agnostic search

1. INTRODUCTION

Users on the web are expanding from being active information *consumers* to becoming active information *producers* [18, 17], which leads to an unprecedented growth of information. With such a boom in information retrieval and the information explosion, there is an ever-increasing demand for accessibility, coverage, quick responses, and relevant results from relatively vague and loose queries. The huge collection of information inherently entails a loss of performance and efficiency, because it takes time to process (index, cluster, etc), retrieve (query) and keep the information up-to-date in the huge repositories. Thus there is a growing concern about the usability and the interaction time between the user and Information Retrieval (IR) systems. A trade-off between the quality of the results and query response time is mostly considered as an option. In such challenging settings a user query must yield meaningful, manageable and most importantly "*relevant*" set of results from IR systems.

A central topic in the iAD (Information Access Disruptions [10]) project is to index data collections where there is a big variety of data, both unstructured data (text documents), structured data (e.g. database records) and semi-structured data (e.g. XML or HTML data), all present at the same time. In such a setting we want to focus on *relevancy* in a *schema agnostic systems*. Here "schema agnostic" only means that the queries need not to use the schema information, while the search engine can possibly use the schema (meta)-/information to come up with a set of relevant documents. Relevancy calculations should take into account structure information whenever it exists and use this to improve ranking of results without or with minimal prior knowledge of the schema of the data. Schema can be automatically recognized by analysing data sources or the data itself. Important research questions are to find metrics which are giving a perceived better result for the end user and how to score various data with schema relative to each other. Variance and resulting inaccuracies in the document structures, vocabulary and document content dictate ranked retrieval as the only meaningful search paradigm.

To calculate relevancy many different metrics are being used. Examples on metrics are $tf \times idf$ score [2, 17], static weight for document, proximity of terms, freshness of document. The metrics are weighted relative to each other and combined into a final score. Different users can have different weightings based on their preferences. Structured and semi-structured documents will make it possible to find new metric such as, weighting of scopes and coexistence of terms in scopes.

In a collection of documents of different types it is difficult to combine the scores. In a document with no structure there will be no score from structure metrics. These documents therefore might get less score than documents with structure. To compensate for this we can boost score for these documents but how much should this be boosted? Can we use the inherent structure of document (extracted automatically or semi-automatically) to increase the visibility of

JCDL'11, June 12-17, 2011, Ottawa, ON, Canada.

contents and concepts ingrained in the document?

As in legacy search engines (library systems or digital libraries) and web search, schema agnostic search environment must also define a measure of relevance or merit for each response. In such an environment (schema agnostic) relevancy subsystem must efficiently generate only a few responses that have the greatest relevance scores in that particular setting.

Different ways of calculating relevancy scores will be researched in schema agnostic environment (XML or semi-structured dataset) as a main focus of this study.

2. MOTIVATION

```
<proceedings>
  <inproceedings>
    <author>Serge Abiteboul</author>
    <title>Querying Semi-structured data</title>
  </inproceedings>
  <inproceedings>
    <author>Mounia Lalmas</author>
    <title>XML Retrieval</title>
  </inproceedings>
  <inproceedings>
    <author>Anja Theobald, Gerhard Weikum</author>
    <title>Adding Relevance to XML</title>
  </inproceedings>
</proceedings>
```

Figure 1: Find papers by author "Lalmas" on the topic of "semi-structured data"

It is becoming increasingly popular to publish data on the Web in the form of semi-structured documents which is useful both for data exchange and data semantics. The representation in Figure 1 if retrieved using the existing Text-based or conventional search engine based on traditional IR techniques, have two main drawbacks when it comes to searching for semi-structured documents:

- It is not possible to pose queries that explicitly refer to structure of the documents, e.g., twig like queries.
- Search engines return references (i.e. links) to documents and not specific fragments thereof. This is problematic, since large semi-structured documents may contain thousands of elements storing many pieces of information that are not necessarily related to each other.

Since a reference to whole semi-structured document is usually not a useful answer, the granularity of the search should be refined. The concept of the *logical document* [23] instead of just document comes here; the users are now not interested in the document but the most specific part of the document i.e., the logical document.

Figure 1 shows an example scenario, the overall document will be returned as relevant by a text-based search engine. But the document is not relevant to the posed query. In this query the user is interested in the documents authored by "Lalmas". The above document should not be retrieved at all by a structure-aware search engine. The set of answers in the set of retrieved documents should be semantically related, i.e., the set of the answer nodes are meaningful fragment of the semi-structured documents. For example, a paper and an author should be in the answer set only if the paper was written by this author.

The retrieval must not only return the most specific part of the documents but also it should take into account the degree of *relevance* of the retrieved document fragments with the posed query. And based on that the documents should appear in the ranked outcomes. The document in Figure 1 should appear lower down in the ranked outcomes for the given query.

Structure provides both *context* and *semantics* to the content as seen from the motivation scenario discussed above. This context and semantics should possibly be used to boost or reduce the documents relevancy scores. And without using the structural information, the search outcomes would simply be irrelevant or misleading to the posed queries.

Secondly, from the structure the importance of content in different parts of document could be learned. Text or set of keywords lying in body of a document could be less important than keywords lying in the title.

3. STATE OF THE ART

Semi-structured retrieval research is an interdisciplinary field of study. Both its inception and its implication crosses traditional boundaries from information retrieval community [13, 21, 12] to relational databases [8, 28] and at the same time having a wide range implications across digital libraries communities [4]. In the following, we provide an overview of existing approaches towards relevancy in semi-structured data or schema-agnostic environment, and an overview of why they are not significant enough to answer the research questions raised (Section 5) in this study.

Broadly, there are two major approaches towards semi-structured retrieval problem. Hence, we look into the existing work from these two perspectives:

- The relational approach towards semi-structured retrieval
- The native approach - conventional information retrieval approach

The relational approaches tend to make use of techniques from already mature area, the relational databases. Instead of considering the semi-structured retrieval problem in isolation, it is considered in the relational databases space. The benefit of adopting such an approach is that you don't have to reinvent the wheels. There are tools and techniques which have been in use for years and a lot of work have already been done on them over the years. So the answer to the semi-structured retrieval is to mold it in the relational databases space, and hence by doing that we retain the efficiency and strength of an already mature area [8]. Specifically, the relational approach directly utilizes relational databases to represent and retrieve semi-structured data, which enables to use all important capabilities of relational databases. XPath and XQuery, developed by W3C Consortium inspired by relational approaches but not necessarily using the capabilities of relational databases, address the problem of semi-structured retrieval. They are not suitable for our purpose mainly because (a) they require a thorough knowledge of

schema or structure beforehand (b) complicated to translate the user query into an XQuery (c) syntax of XQuery is by far more complicated than syntax of standard IR system and (d) nominal mechanism for ranking.

Apart from limitations discussed before, it is not always as easy to adopt to an existing and mature framework. The inherent peculiarities of the semi-structured retrieval prohibits making use of some of the main strengths of the relational databases. The saving due to significantly reducing system re-engineering costs in the semi-structured environment is less than reinventing the wheel in the specialized storage and query processing systems tailored for semi-structured settings. Hence the native approach is to consider the semi-structured retrieval in its own particular settings, from scratch to further improve semi-structured retrieval.

XRANK [9] proposed by Guo et.al generalizes the idea initially proposed by Page and Brin [19]. Like Google's PageRank, XRank consider the dataset as a Tree or Graph (see Figure 2). Unlike PageRank which consider one-size fit all approach, XRANK advocates that the data tree has different type of edges namely containment edge and hyperlink edge. Random Surfer in the XRANK instead of following just the hyperlinks also visits the containment edges (CE) (elements, sub-elements), hyperlink-edges (HE) and reverse containment edges (CE^{-1}) (sub-elements, elements). Like PageRank, XRANK is also calculated offline independent of any query. Equation 1 taken from [9] summarizes the random surfer model of XRANK.

$$e(v) = \frac{1-d_1-d_2-d_3}{N_d \times N_{de}(v)} + d_1 \sum_{(u,v)\in HE} \frac{e(u)}{N_h(u)} + d_2 \sum_{(u,v)\in CE} \frac{e(u)}{N_c(u)} + d_3 \sum_{(u,v)\in CE^{-1}} e(u) \quad (1)$$

$e(v)$ is ElemRank of v

Structural information is mainly used in the calculation of the *ElemRank*, and to answer the queries that have structural dependencies (i.e., structural queries). A more in-depth study is required to observe experimentally the effects of structure on the quality of *relevancy* calculation. In case of PageRank, it was claimed based on intuition and based on the theoretical interpretation of *Markov chain model*, to mimic the users' behaviour on the Web, but in case of XRANK, the random surfer model given in Equation 1, also resembles users' behaviour? There still is room to explore the structural benefits ingrained in the semi-structured retrieval, a more active use of structure throughout retrieval processes.

Just like XRANK, ObjectRank [3] inspired by Google's PageRank, more actively utilizes the link structure of the semi-structured data. It calculates both global (PageRank) and keyword-specific ObjectRank of each node in the *authority transfer schema / data graph*. Unlike XRANK, ObjectRank is a relational approach and applicable only on Databases.

The work in XRANK, ObjectRank and other graph based methods correspond to the study of Link Analysis Ranking [5, 17, 27]. The motivation is based on intuition as a semi-structured document form a tree structure in most of the cases and a complete graph with cycles in specific cases (as can be seen in Figure 2).

XSEarch [6] on the other hand instead of using the link structure of the Tree representation of the semi-structured

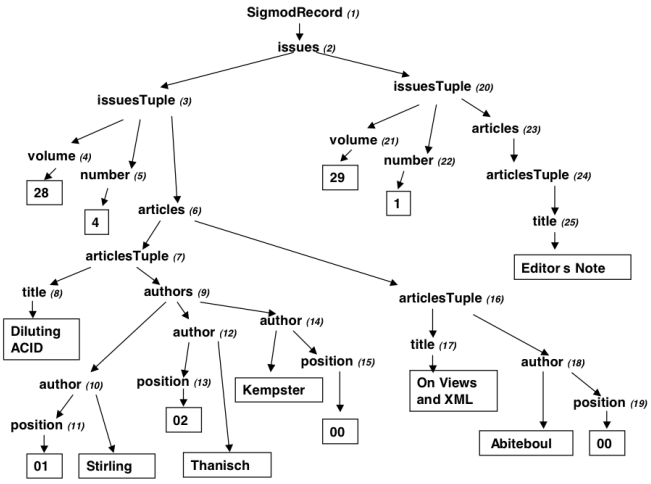

Figure 2: Semi-structured data, XML, is represented mostly as a Tree structure (Figure taken from [6])

data, uses extended *Vector Space Model* for retrieval and ranking, and the same kind of approach was employed by Schlieder and Meuss [23]. They make use of the *interconnection relationship* among the XML elements to use the structure in retrieval and ranking. By doing that, they tried to answer the question that under what conditions elements of a XML document are semantically related.

Again, relevancy scoring is not effected directly by the structural constraints. Rather the structural elements gets user-defined weights (manual process), instead of structural elements lying in the heart of relevancy scoring. Under what conditions the semantic constructs in the document, i.e., the structural elements could be used automatically or semi-automatically to purify the relevancy scoring?

A variation of $tf \times idf$ is used for relevancy scoring in XSEarch, where tf correspond to number of occurrences of a query term in a fragment and *Inverse Leaf Frequency* ilf: number of leaves containing a query term divided by number of leaves in the corpus (the data tree), see Equation 2. The $tf \times ilf$ score together with the interconnection relationship measure (calculated based on how close the elements are in the relationship tree) are used to determine the ranking of the answer.

$$tf(k,n_l) := \frac{occ(k,n_l)}{max\{occ(k',n_l)|k' \in words(n_l)\}} \quad (2)$$

$$ilf(k) := log\left(1 + \frac{|N|}{\{|n' \in N|k \in words(n')|\}}\right)$$

XXL [24, 25] was mainly proposed to introduce active support for ranked retrieval. In addition, ontological information or relationship has also been integrated as a basis for effective similarity search. In the same line XPRES [29, 30] extends the classical probabilistic model, that exploits the semantic of different text part given in semi-structured document. Like XSEearch, XPRES extends the classical weighting measure $tf \times idf$ and call it $tf \times ief$ (*ief: Inverse Element Frequency*).

BM25F-based [22, 16] XML retrieval has recently been introduced to score individual XML elements [11]. In this

approach each XML element is scored as if it were an independent document. This method ignores hierarchy i.e., the parent-child relationships (which usually contains the contextual information), but rather focuses on the elements independently.

To sum up, this section has presented some of the methods that employ structure in the document to somehow improve or purify the retrieval. They have formed a good basis and background for this study and at the same time provided the prospects for possible future work. We believe that a more active and exclusive use of structure in the semi-structured documents would be a worthy contribution in the field of semi-structured retrieval.

4. ISSUES AND CHALLENGES

The main challenge in the schema-agnostic environment (as discussed from existing work) is that: how the implicit and explicit structure of the document helps to improve the semantics of the retrieval, i.e., improved relevancy? The other challenges as also identified by [1, 3, 15] are:

- The structure is irregular, inconsistent and possibly inaccurate, the same piece of information can be structured in different ways.
- The structure is implicit and is part of the data.
- The schema could be very large. And it keeps on evolving rapidly, and hence the distinction between the schema and data keeps on blurring.
- Differentiating between semantically meaningful constructs and semantically meaningless.
- The two dimensional view of the proximity.
    1. Result specificity: more specific results higher than less specific results. One dimension of result proximity.
    2. Keyword proximity: another dimension of result proximity.
- Users require the most specific answer (part of the document only) instead of the whole document as the answer.
- What constitute an *indexing unit*?
- Partial matching elements that do not meet the structural constraints perfectly should be ranked lower and should not be omitted from search outcomes.

In this context, returning a set of *relevant* and notion of ranking at the finest granularity of semi-structured documents (e.g., in case of XML, it is XML element), is a challenging task in itself. Few of the above challenges have already been addressed in existing work as identified in the Section 3, but there combination as a whole would be interesting future work and core of this study. From these challenges and issues we have formulated the research questions for this research.

5. RESEARCH QUESTIONS

The overall research can be stated as a number of research questions:

1. How semantics in the document i.e., the structure, could possibly be used to understand the content in the document and possibly use it to improve retrieval?
2. How should the structure extracted from the semi-structured document be represented? Which type of index structures provide better or worse results?
3. The support for full-text (keyword) and structured query, using the structure to boost the relevancy scores of the documents in either cases.
4. How would the proximity be impacted by the structure in the documents? Does the parent-child relationship add to the conventional proximity measure?
5. How to accommodate variation in the data, as distinction between the schema and data is getting blurred?

The requirements and challenges described in the previous sections are represented in the research questions above. These questions are based on current knowledge of the area, hence it could as well be extended or further purified later based on increased understanding of the subject area.

6. EXPECTED RESULTS AND CONTRIBUTIONS

We would like to use capabilities of different algorithms utilizing different index structures in isolation and together with one another to see their impact on the overall retrieval in general and ranking in particular. At the moment, we are in the processs of implementing different index structures (Dewey inverted index [9] and its different flavours) and use them together with the existing state-of-art methods for example, XRANK's ranking algorithm ($ElemRank$, see Equation 1) and index structures (interconnection index) used in XSearch system. And by doing that we would like to measure the impact of different indexing schemes on the search outcomes. We would evaluate the effectiveness of our approach with the evaluation metrics and standard datasets from the INitiative for the Evaluation of XML Retrieval (INEX) [7, 26].

One of the previous contributions by the author [18] together with XRANK and / or ObjectRank could be a valuable contribution. As identified in Section 3 XSEarch lacks the capability to automatically weight the XML elements, together with the recent work by Liu et. al [14] could possibly be a positive contribution to XSEarch system. $BM25F$ could also possibly be improved by incorporating structural construct in the algorithm. At the moment $BM25F$ scores XML elements independently without considering the context surrounding it, i.e., the structure.

7. GOALS ACHIEVED SO FAR AND FURTHER PLAN

As the research candidate is quite early stage of the study, the most of work performed so far is around specifying the

research questions, planing, taking courses, performing literature review and getting a field overview. It is expected to write a self-contained search engine from scratch, or maybe customize some of the opensource solutions available such as, *Lucene* [1] / *Solr*[2] or maybe *Nutch*[3]. Alternatively, *Terrier*[4] search engine could also be customized to fit the purpose of this study.

According to the plan and after the experimental setup, the candidate is suppose to start testing the initial ideas described in the last section and proceed with the research question 1 (Section 5).

8. THE METHODOLOGY

In this section we discuss the research design and methodology and its appropriateness for this study. The process of collecting, recording, and analysing data is quite crucial in this study, because of its innovative and technical nature. An account of the assumptions of the study have also been discussed briefly.

8.1 The research paradigm and its rationale

The study will be conducted within the quantitative paradigm. There are two major reasons for selecting the quantitative paradigm: firstly, this research demands an in-depth study. Secondly, we can explore the problem in different experimental settings possibly using the standard practices, metrics and evaluation framework from the state of the art, INEX, TREC and the likes.

8.2 Experimental Research

Experimental research is the demand of my research topic primarily because it is a collection of research designs which use the manipulation and controlled testing to understand causal processes. Generally one or more variables and heuristics are manipulated to determine the effect on a dependent variable, which could be thresholds, performance and throughput bottlenecks. Thus Experimental research is a systematic and scientific approach to research in which the researcher manipulates one or more variables, and controls and measures any change in other variables [20].

Generally the experimental research is used when:

- There is a time and performance priority in the causal relationships.
- There is a consistency in a causal relationship.
- The magnitude of the correlation is great.

In semi-structured or schema agnostic retrieval environment we will actively monitor the influence of different approaches towards retrieval in the state-of-the-art settings. Through the experimental research, we intend to empirically evaluate the feasibility of the different approaches chosen as described in previous sections, for the problem of semi-structured retrieval. Both the relational and the native approaches as described in Section 3 towards semi-structured retrieval will be experimentally compared and parametrized using state of the art evaluation techniques mainly relying on the metrics and dataset from INEX [7, 26].

[1]The Apache *Lucene*^{TM} project: http://lucene.apache.org/
[2]http://lucene.apache.org/solr/
[3]http://lucene.apache.org/nutch/
[4]http://terrier.org/

9. CONCLUSION AND IMPLICATIONS

In this paper an overview of the problems in the semi-structured retrieval or schema-agnostic search has been presented. In the state-of-the-art section a number of possible solutions have been discussed, along with their shortcomings. Given the existing work, we still think that there is a lot that need to be done in the semi-structured retrieval, specifically there is not much done in the ranking or relevancy in schema-agnostic environment. How to actively employ the structure in the relevancy subsystem? There is still enough room for introduction, innovation and improvements in the ranking of semi-structured dataset. We think that the research outcomes from this study will be quite beneficial in future. The preliminary research questions proposed aimed to develop a combination of methods to better answer the full-text and structured queries to semi-structured data.

The contributions from this study will be applicable to a wide variety of areas. For example the research on entity extraction or entity search is a direct application of this study. Also in multimedia retrieval mostly the documents are represented in semi-structured form and searching through myriad of them is a contemporary requirement and future need. In natural language processing it is usually worth to extract and use the latent-structures in the documents in order to detect important objects or features.

This Ph.D. study is expected to finish by August 2013, with a Ph.D. defence in October / November 2013.

10. REFERENCES

[1] S. Abiteboul. Querying semi-structured data. *Database Theory-ICDT'97*, pages 1–18, 1997.

[2] R. Baeza-Yates, B. Ribeiro-Neto, et al. *Modern information retrieval*. Addison-Wesley Harlow, England, 1999.

[3] A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 564–575. VLDB Endowment, 2004.

[4] D. Bamman, A. Babeu, and G. Crane. Transferring structural markup across translations using multilingual alignment and projection. In *Proceedings of the 10th annual joint conference on Digital libraries*, pages 11–20. ACM, 2010.

[5] A. Borodin, G. Roberts, J. Rosenthal, and P. Tsaparas. Link analysis ranking: algorithms, theory, and experiments. *ACM Transactions on Internet Technology*, 5(1):231–297, 2005.

[6] S. Cohen, J. Mamou, Y. Kanza, and Y. Sagiv. XSEarch: A semantic search engine for XML. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, page 56. VLDB Endowment, 2003.

[7] L. Denoyer and P. Gallinari. The wikipedia xml corpus. In *ACM SIGIR Forum*, volume 40, pages 64–69. ACM, 2006.

[8] G. Gou and R. Chirkova. Efficiently Querying Large XML Data Repositories: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 19(10):1381–1403, Oct. 2007.

[9] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram. XRANK: Ranked keyword

search over XML documents. In *Proceedings of the 29th ACM SIGMOD international conference on Management of data*, page 27. ACM, 2003.

[10] iAD Team. iad information access disruptions centre. http://www.iad-centre.no/about.html.

[11] K. Itakura and C. Clarke. A framework for BM25F-based XML retrieval. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 843–844. ACM, 2010.

[12] J. Kim, X. Xue, and W. B. Croft. A Probabilistic Retrieval Model for Semistructured Data. *Collections*, pages 228–239, 2009.

[13] M. Lalmas. XML Retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 1(1):1–111, Jan. 2009.

[14] D. Liu, C. Wan, L. Chen, and X. Liu. Automatically weighting tags in XML collection. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1289–1292. ACM, 2010.

[15] S. Liu, Q. Zou, and W. Chu. Configurable indexing and ranking for XML information retrieval. *Proceedings of the 27th annual international conference on Research and development in information retrieval - SIGIR '04*, page 88, 2004.

[16] W. Lu, S. Robertson, and A. MacFarlane. Field-weighted XML retrieval based on BM25. *Advances in XML Information Retrieval and Evaluation*, pages 161–171, 2006.

[17] M. Norozi. Information Retrieval Models and Relevancy Ranking. Master's thesis, Centre of Mathematics for Application, University of Oslo, 2008.

[18] M. Norozi. Extrapolation to speed-up query-dependent link analysis ranking algorithms. In *Proceedings of the 8th International Conference on Frontiers of Information Technology - FIT '10*, page 2. ACM, 2010.

[19] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web, 1998.

[20] K. Peffers, T. Tuunanen, C. Gengler, M. Rossi, W. Hui, V. Virtanen, and J. Bragge. The Design Science research process: a model for producing and presenting information systems research. In *Proceedings of the First International Conference on Design Science Research in Information Systems and Technology (DESRIST 2006)*, pages 83–106, 2006.

[21] D. Petkova, W. B. Croft, and Y. Diao. Refining Keyword Queries for XML Retrieval by Combining Content and Structure The Problem of Adding Structure to Keyword Queries. pages 662–669, 2009.

[22] S. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49. ACM, 2004.

[23] T. Schlieder and H. Meuss. Querying and ranking XML documents. *Journal of the American Society for Information Science and Technology*, 53(6):489–503, 2002.

[24] A. Theobald and G. Weikum. Adding relevance to XML. *The World Wide Web and Databases*, pages 105–124, 2001.

[25] A. Theobald and G. Weikum. The Index-Based XXL Search Engine for Querying XML Data with Relevance Ranking. *Advances in Database Technology—EDBT 2002*, pages 311–340, 2002.

[26] A. Trotman, M. del Rocio Gomez Crisostomo, and M. Lalmas. Visualizing the Problems with the INEX Topics. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 826–826. ACM, 2009.

[27] P. Tsaparas. *Link Analysis Ranking*. PhD thesis, University of Toronto, 2004.

[28] J. Wang and J. X. Yu. Answering Tree Pattern Queries Using Views : a Revisit. In *Proceedings of the 14th International Conference on Extending Database Technology*, page 4. ACM, 2011.

[29] J. Wolff, H. Florke, and A. Cremers. XPRES: A ranking approach to retrieval on structured documents. Technical report, Citeseer, 1999.

[30] J. Wolff, H. Florke, and A. Cremers. Searching and browsing collections of structural information. In *adl*, page 141. Published by the IEEE Computer Society, 2000.