Query Refinement by Relevance Feedback in an XML Retrieval System

Hanglin Pan, Anja Theobald, Ralf Schenkel

Max-Planck-Institute for Computer Science
D-66123 Saarbrücken, Germany
{pan,atb,schenkel} @mpi-sb.mpg.de

1   Introduction

In recent years, ranked retrieval systems for heterogeneous XML data with both structural search conditions and keyword conditions have been developed for digital libraries, federations of scientific data repositories, and hopefully portions of the ultimate Web. These systems, such as XXL [2], are based on pre-defined similarity measures for atomic conditions (using index structures on contents, paths and ontological relationships) and then use rank aggregation techniques to produce ranked results lists. An ontology can play a positive role for term expansion [2], by improving the average precision and recall in the INEX 2003 benchmark [3].

Due to the users' lack of information on the structure and terminology of the underlying diverse data sources, and the complexity of the (powerful) query language, users can often not avoid posing overly broad or overly narrow initial queries, thus getting either too many or too few results. For the user, it is more appropriate and easier to provide relevance judgments on the best results of an initial query execution, and then refine the query, either interactively or automatically by the system. This calls for applying relevance feedback technology in the new area of XML retrieval [1].

The key question is how to appropriately generate a refined query based on a user's feedback in order to obtain more relevant results among the *top-k* result list. Our demonstration will show an approach for extracting user information needs by relevance feedback, maintaining more intelligent personal ontologies, clarifying uncertainties, reweighting atomic conditions, expanding query, and automatically generating a refined query for the XML retrieval system XXL.

2   Stages of the Retrieval Process

a. Query Decomposition and Weight Initialization : A query is composed of weighted (i.e., differently important) atomic conditions, for example, XML element content constrains, XML element name (tag) constrains, path pattern constrains, ontology similarity constrains, variable constrains, search space constrains, and output constrains. In the XXL system, each atomic condition has an initial weight. If some constrains are uncertain, we specify them by the operator '$\sim$'. Concrete examples are shown in the poster.

b. Retrieval with Ontology based Similarity Computation : Content index and path index structures are pre-computed and used for the relevance score evaluation of result item candidates. The global ontology index is built beforehand as a table of concepts from WordNet, and frequency-based correlations of concepts are computed statistically using large web crawls. To enable efficient query refinement in the following feedback iterations, we have a set of strategies to maintain a query-specified personal ontology which is automatically generated from fragments of global ontology. This is the source for query term expansion, as well as ontological similarity computations.

c. Result Navigation and Feedback Capturing : The retrieved ranking list is visualized in a user-friendly way supporting zoom plus focus. Features like group selection and re-ranking are supported in our system, which can capture richer feedback at various levels, i.e., content, path and overall level.

d. Strategy Selection for Query Reweighting and Query Expansion : The strategy selection module will choose an appropriate rank aggregation function over atomic conditions for overall score computation. After each feedback iteration, tuning functions (such as minimum weight, average weight algorithm [4]), are used to derive the relative importance among all atomic conditions, and to update the personal ontology [1].

e. Adaptable Query Reformulation : Our system is adaptable using reweighting and expansion techniques. The open architecture allows us easily add new rank aggregation functions, reweighting strategies, or expansion strategies.

3   Demonstration

The INEX 2003 benchmark [3] consists of a set of content-and-structure queries and content-only queries over 12117 journal articles. For each query, there is a result pool. Each document in a result set has a relevance assessment score provided by human experts. We run our method on this data set to show the improvement of average precision and recall using relevance feedback with up to four iterations. Our baseline is using only ontology-based expansion [2]. We show the comparison between different strategies of rank aggregation, query reweighting and expansion. We also show our approach to refine structural XML queries based on relevance feedback.

References

1. Hanglin Pan. Relevance Feedback in XML Retrieval. In: Proceedings of the *ICDE/EDBT* Ph.D. Workshop, Boston, March 2004, pages 193-202, 2004.
2. Ralf Schenkel, Anja Theobald and Gerhard Weikum. XXL@INEX2003. In: Proceedings of the 2003 INEX Workshop, Dagstuhl Castle, Germany, December 15-17, 2003.
3. Norbert Fuhr, Mounia Lalmas. Initiative for the evaluation of XML retrieval (INEX), 2003. `http://inex.is.informatik.uni-duisburg.de:2003/`.
4. Michael Ortega-Binderberger, Kaushik Chakrabarti, and Sharad Mehrotra. An approach to integrating query refinement in SQL. In *Extending Database Technology, EDBT*, pages 15-33, 2002.