EXTIRP: Baseline Retrieval from Wikipedia

Miro Lehtonen[1] and Antoine Doucet[1,2]

[1] Department of Computer Science
P.O. Box 68 (Gustaf Hällströmin katu 2b)
FI–00014 University of Helsinki
Finland
{Miro.Lehtonen,Antoine.Doucet}@cs.helsinki.fi
[2] IRISA-INRIA
Campus de Beaulieu
F-35042 Rennes Cedex
France
Antoine.Doucet@irisa.fr

Abstract. The Wikipedia XML documents are considered an interesting challenge to any XML retrieval system that is capable of indexing and retrieving XML without prior knowledge of the structure. Although the structure of the Wikipedia XML documents is highly irregular and thus unpredictable, EXTIRP manages to handle all the well-formed XML documents without problems. Whether the high flexibility of EXTIRP also implies high performance concerning the quality of IR has so far been a question without definite answers. The initial results do not confirm any positive answers, but instead, they tempt us to define some requirements for the XML documents that EXTIRP is expected to index. The most interesting question stemming from our results is about the line between high-quality XML markup which aids accurate IR and noisy "XML spam" that misleads flexible XML search engines.

1 Introduction

The experimental XML retrieval system of University of Helsinki — EXTIRP [1] — needed only slight modification when adapted to indexing and retrieving information from the Wikipedia document collection. Application of the existing methods to a new set of documents was especially interesting: EXTIRP has previously been tested on the IEEE article collection only, although it can handle documents of arbitrary document types. The previous test results could be explained by alleged fine-tuning to a single document collection because we were not able to show how EXTIRP worked on other collections. Therefore, the Wikipedia documents added another valuable dimension to the testing history of EXTIRP.

Partly because of our low resources and partly because of our desire to keep our system from 2005 pristine in that there was no tuning one way or another, we did not analyse the Wiki documents before they were indexed and queried for the official submissions. We also have left out many of the characteristic features that have been part of EXTIRP during its short history. These features include query expansion, intra-document reference analysis, as well as weighting schemes for titles and inline elements. The remaining system has come close to a baseline retrieval model based on the vector space model and cosine similarity.

This article is organised as follows. The anatomy of EXTIRP is described in Section 2. The selection of the index units is explained in Section 3. The results are analysed in Section 4 and finally conclusions are drawn in Section 5.

2 Background

EXTIRP scans through the document collection and selects disjoint fragments of XML to be indexed as atomic units. Typical fragments include XML elements marking sections, subsections, and paragraphs. Examples and more details about the selection algorithm are included in Section 3. The disjoint fragments are treated as traditional documents which are independent of each other. The pros include that the traditional IR methods apply, so we use the vector space model with a weighting scheme based on the tf*idf. The biggest of the cons is that the size of the indexed fragments is static, and if bigger or smaller answers are more appropriate for some query, the fragments have to be either divided further or combined into bigger fragments.

Two separate inverted indices are built for the fragments. A *word index* is created after punctuation and stopwords are removed and the remaining words are stemmed with the Porter algorithm [2]. The *phrase index* is based on Maximal Frequent Sequences (MFS) [3]. Maximal phrases of two or more words are stored in the phrase index if they occur in seven or more fragments. The threshold of seven comes from the computational complexity of the algorithm. Although lower values for the threshold produce more MFSs, the computation itself would take too long to be practical. More details concerning the configuration of the phrase index are included in the PhD thesis of Antoine Doucet [4].

When processing the queries, we compute the cosine similarity between the document and the base term vectors which results in a `Word_RSV` value. In a similar fashion, each fragment vector gets a similarity score `MFS_RSV` for phrase similarity. These two scores are aggregated into a single RSV so that the aggregated RSV = α * `Word_RSV` + β * `MFS_RSV`, where α is the number of distinct query terms and β is the number of distinct query terms in the query phrases.

3 Selective Indexing

The selection of indexed fragments is based on two parameters: fragment size (min and max) and the proportion of Text and Element nodes (T/E measure) [5]. The algorithm starts from the document root and traverses the document tree in document order. The following steps are then iterated:

1. If the element is too big, move on to the next node and start over (from 1).
2. If the content looks like structured data ($T/E<1.0$), move on to the next node and start from 1.

3. If the element is too small, skip the subtree, move on to the next node and start from 1.
4. Index the element as an atomic unit, skip the subtree, move on to the next node and start from 1.

The resulting fragment collection does not cover the whole document collection. For example, parts of the documents that consist mostly of elements are discarded. Previous experiments on IEEE articles have shown that the algorithm works: it reduces the index size and improves retrieval precision. When tested with the article collection, bibliographic and other data were successfully excluded from the full-text index [6]. Therefore, the Wikipedia XML documents were an interesting challenge for our algorithm.

Figure 1 shows the document with the lowest T/E value in the Wikipedia XML collection. The nested `cadre` elements are there either because of a faulty conversion from the Wiki format into XML or because of inconsistency in the source data. Because of the extra elements, the text content of this document was not included in the full-text index of EXTIRP, and thus it could not be retrieved, regardless of the query. However, the nested structures created with the proliferating XML elements are highly artificial. Therefore, it is questionable to exclude text content from the full-text index because of such artificial structures.

Our observations raise an interesting question: What is the validity of this evaluation at INEX, where the test documents can only be used in the evaluation because the structure is completely useless elsewhere?

4 Results

The results from INEX 2005 showed that the official evaluation metrics [7] do not favour systems like EXTIRP because there is no reward for returning "too small" answers. The 2005 version of EXTIRP could not adjust the granularity of the answers according to the query, but the granularity came directly from the indexed document fragments [8]. The 2006 version of EXTIRP comes with the same drawback even though some "near misses" are rewarded.

In 2006, we only submitted one run for the CO.Focused task. The fragment size in the index was limited to the range of 150–7,000 characters of text content. Only the title and the keyword part of the queries were considered. The overall poor results according to the official metrics are shown in Table 1.

The poor overall performance has several possible explanations. First, all answers shorter than 500 bytes of XML markup were discarded because of the assumption that short answers are not worth retrieving. This assumption no longer holds as the official metrics reward short answers, too, as long as they are relevant to the query. We also observe that the rankings with the filtered assessments are systematically better than those with the original assessments that include very short relevant answers.

Second, EXTIRP has always had a better performance with the strict quantisation of the assessments than with the generalised one which was the only

```
             </cadre>
            </cadre>
           </cadre>
          </cadre>
         </cadre>
        </cadre>
       </cadre>
      </cadre>
     </cadre>
    </cadre>
   </cadre>
  </cadre>
 </cadre>
</cadre>
1987 "Recent Acquisitions", Tel Aviv Museum of Art
<cadre>
 <cadre>
  <cadre>
   <cadre>
    <cadre>
     <cadre>
      <cadre>
       <cadre>
        <cadre>
         <cadre>
          <cadre>
           <cadre>
            <cadre>
             <cadre>
              <cadre>
               <cadre>
                <cadre>
                 <cadre>
                  <cadre>
                   <cadre>
"Recent Acquisitions", Isreal Museum, Jerusalem, Isreal "20 Years of Occupation",
Bograshov Gallery, Tel Aviv, Israel (catelogue) "New Israeli Realism", traveling
exhibition, Omanut La'am (catelogue) "Artisrael, the 1980's", Leonard Perlson Gallery,
New YorkCity, "Haifa, Portrait of the City", Haifa Museum, Haifa Israel (catelogue)
"Fresh Paint", Tel Aviv Museum of Art (catelogue)
                   </cadre>
                  </cadre>
                 </cadre>
                </cadre>
               </cadre>
              </cadre>
             </cadre>
            </cadre>
           </cadre>
          </cadre>
         </cadre>
        </cadre>
       </cadre>
      </cadre>
     </cadre>
    </cadre>
   </cadre>
  </cadre>
 </cadre>
1989 International Biennale of Graphic Art, Ljublijana,
Yugoslavia (catelogue)
<cadre>
 <cadre>
  <cadre>
```

Fig. 1. An excerpt from the document `'3125748.xml'` with a T/E value of 0.14

Table 1. The submission "UHel_Run1" measured with nxCG, generalised quantisation. A total of 85 submissions are included in the ranking.

	Overlap on		Filtered assesments		Overlap off		Filtered assesments	
Cutoff	Rank	Score	Rank	Score	Rank	Score	Rank	Score
MAP5	67	0.2316	65	0.2299	70	0.2371	68	0.2371
MAP10	70	0.1818	69	0.1821	71	0.1898	69	0.1898
MAP25	73	0.1295	71	0.1315	73	0.1358	71	0.1358
MAP50	74	0.0960	73	0.1001	73	0.0990	72	0.0992

quantisation in 2006. Third, what EXTIRP assumes of the quality of XML is based on observations of real XML documents that are usable outside the context of INEX. For example, we assume that the XML structure of the documents is designed before any content is converted into that structure and that the XML documents have a real use case instead of only being test material for researchers.

Despite the relatively poor performance, the results do show some signs of stability in the performance of EXTIRP. Along the lines of the previous INEX results from 2004 and 2005, the relative ranking of EXTIRP decreases as the cutoff value increases. This supports the earlier observation that EXTIRP is more geared towards high precision tasks than high recall ones.

5 Conclusion

As a simple implementation of an XML retrieval system, the 2006 version of EXTIRP serves as a baseline that other more advanced implementations can be compared with. However, according to the official evaluation metric (XCG), the performance of this baseline is so poor that other metrics with better results are necessary for a meaningful comparison. In the future, we are hoping to have a fully implemented version of our system in order to see where it really stands. We also look forward to experimenting with more realistic document collections in order to increase the validity of the results.

References

1. Doucet, A., Aunimo, L., Lehtonen, M., Petit, R.: Accurate Retrieval of XML Document Fragments using EXTIRP. In: INEX, Workshop Proceedings, Schloss Dagstuhl, Germany, pp. 73–80 (2003)
2. Porter, M.F.: An algorithm for suffix stripping. Program 14, 130–137 (1980)
3. Ahonen-Myka, H.: Finding all frequent maximal sequences in text. In: Mladenic, D., Grobelnik, M., (eds.) Proceedings of the 16th International Conference on Machine Learning ICML-99 Workshop on Machine Learning in Text Data Analysis, Ljubljana, Slovenia, J. Stefan Institute, pp. 11–17 (1999)
4. Doucet, A.: Advanced Document Description, a Sequential Approach. PhD thesis, University of Helsinki (2005)
5. Lehtonen, M.: Preparing heterogeneous XML for full-text search. ACM Trans. Inf. Syst. 24, 455–474 (2006)

6. Lehtonen, M.: Indexing Heterogeneous XML for Full-Text Search. PhD thesis, University of Helsinki (2006)
7. Kazai, G., Lalmas, M.: INEX 2005 Evaluation Measures. In: Fuhr, N., Lalmas, M., Malik, S., Kazai, G. (eds.) INEX 2005. LNCS, vol. 3977, pp. 16–29. Springer, Heidelberg (2006)
8. Lehtonen, M.: When a few highly relevant answers are enough. [9] 296–305
9. Fuhr, N., Lalmas, M., Malik, S., Kazai, G., (eds.): Advances in XML Information Retrieval and Evaluation (Revised Selected Papers). In: Fuhr, N., Lalmas, M., Malik, S., Kazai, G. (eds.) INEX 2005. LNCS, vol. 3977, Springer, Heidelberg (2006)