An Evaluation Measure for Distributed Information Retrieval Systems

Hans Friedrich Witschel, Florian Holz, Gregor Heinrich, and Sven Teresniak

University of Leipzig, Germany
{witschel|holz|heinrich|teresniak}@informatik.uni-leipzig.de

Abstract. This paper is concerned with the evaluation of distributed and peer-to-peer information retrieval systems. A new measure is introduced that compares results of a distributed retrieval system to those of a centralised system, fully exploiting the ranking of the latter as an indicator of gradual relevance. Problems with existing evaluation approaches are verified experimentally.

1 Introduction

One of the core requirements when creating an evaluation testbed for either distributed information retrieval (DIR) or peer-to-peer information retrieval (P2PIR) is a realistic distribution of documents onto databases or peers. A common method in DIR is to use TREC ad hoc test collections and distribute them according to source and date (e.g. [3]), with the advantage that human relevance judgments are available. For evaluation of P2PIR, with typically small and semantically more homogeneous collections, this approach is unrealistic.

In P2PIR, distribution of documents is either done in a way that springs naturally from the collection, e.g. via author information [2], built-in categories [1] or domains of web pages [4], or it is established in less natural ways via clustering [6] or even randomly [5]. Since generally these collections lack queries and relevance judgments, queries are either constructed from the documents [4,2,1] or taken from query logs matching the collection [6,8].

Although with both methods a large number of queries can be created, the need remains to assess query results for relevance. This challenge is approached in this article.

2 Related Work

Various approximations of relevance have been studied in P2PIR: assuming documents containing all query keywords to be relevant [2], using "approximate descriptions of relevant material" [1] or comparing results of distributed algorithms to results of a centralised system [6,4,9,8].

The last approach assumes that a distributed system will rarely be more effective than a centralised one. Although some studies (e.g. [7]) show that the

C. Macdonald et al. (Eds.): ECIR 2008, LNCS 4956, pp. 607–611, 2008.

contrary cannot be ruled out, most other studies agree with this (e.g. [4]). In the following, we will therefore concentrate on the last approach.

It is realised by either considering all documents returned by the centralised system (i.e. those with score > 0) relevant [6] – resulting in what is sometimes called *relative recall* (RR) – or just the N most highly ranked documents [4,9,8]. In the latter case, precision at k documents is used as an evaluation measure – we will call it $\mathrm{P}_N@k$ in the rest of this work, denoting its dependence on N.

3 Average Ranked Relative Recall

Considering all documents with score > 0 relevant is clearly not what we want to approximate. Assuming the top N documents to be relevant is simple, but also not sufficient since we do not know how to choose N as the number of relevant documents generally depends on the query.

However, the choice of N may influence the evaluation results: consider for example a scenario where the centralised system returns a ranked list $(d_1, d_2, ..., d_{15})$ and two distributed systems A and B, where A returns $(d_1, d_2, d_3, d_4, d_{15})$ and B returns $(d_6, d_7, d_8, d_9, d_{10})$.

This results in a $\mathrm{P}_5@5$ of 0 for system B and of 0.8 for system A, i.e. the evaluation predicts system A to perform better than system B. However, $\mathrm{P}_{10}@5$ is also 0.8 for A, but 1.0 for B, thus reversing our evaluation result.

In [8], N is chosen equal to k, in [4,9], values of 50 and 100 are used without further justification. Besides the problem of choosing N, this set-based approach also neglects the ranking of the centralised system within the first N documents.

Therefore, we propose *average ranked relative recall* (ARRR), a new evaluation measure that exploits the ranking of the centralised system as an indicator of gradual relevance and does not treat all of its returned documents (or the top N) as equally relevant.

Let $C = (c_1, ..., c_m) \in T^m$ be the ranking of the centralised system, where T is the set of all documents. We assume that the user has specified how many of the top-ranked documents should be retrieved; we call this value k. It plays a similar role as the k in precision at k documents (P@k) commonly used in the IR literature. Further, let $D = (d_1, ..., d_n) \in T^n$ be the ranking returned by the distributed system with $n \leq k$ (we have $n < k$ only if the distributed system retrieves less than k documents in total).

Next, we introduce a function m_D that, for a pair of documents, returns 1 if the first document is ranked ahead of the second within the set D, else 0:

$$\mathrm{m}_D : T^2 \rightarrow \{0, 1\}$$

$$\mathrm{m}_D(c_j, c_i) = \begin{cases} 1 \text{ if } \exists\, d_q \in D : d_q = c_j \;\wedge\; \exists\, d_p \in D : d_p = c_i \;\wedge\; q \leq p \\ 0 \text{ else} \end{cases}$$

With this new function, we define

$$\mathrm{ARRR}@k(D, C) = \frac{1}{\min(k, m)} \sum_{i=1}^{m} \mathrm{m}_D(c_i, d_n) \frac{\sum_{j=1}^{i} \mathrm{m}_D(c_j, c_i)}{i} \qquad (1)$$

This measure can be determined by the following algorithm: (1) For each document d_i in ranking D, starting from the top: (a) Mark d_i within ranking C if present, and (b) determine the portion of documents marked so far between the top of C and position j of d_i. (2) Sum up the values obtained for all d_i and divide by k or by m if the centralised system finds less than k documents.

Step 1b corresponds to the recall for the distributed system, considering relevant the first j documents in C. If j is large, i.e. d_i is ranked low in C, the "notion" of relevance becomes looser and it is less likely to achieve good recall. Obviously, ARRR@k becomes 1 iff $D = (c_1, ..., c_k)$, i.e. if the distributed system retrieves exactly the k highest-ranked documents found by the centralised system and ranks them in the same way the centralised system does. On the other hand, ARRR@k becomes small if the distributed system ranks documents highly within its first k documents that have low ranks in C.

As an example, we consider $C = (c_1, c_2, c_3, c_4, c_5, c_6)$. Now let us assume that system A returns $D_\mathrm{A} = (c_1, c_3, c_4)$ and system B returns $D_\mathrm{B} = (c_3, c_4, c_1)$. This yields ARRR@5 $= \frac{1}{5}(1 + \frac{2}{3} + \frac{3}{4}) = 0.48$ for system A and ARRR@5 $= \frac{1}{5}(\frac{1}{3} + \frac{2}{4} + 1) = 0.37$, penalising B for its "bad" ranking. If $D_\mathrm{B} = (c_3, c_4, c_1, c_5)$, we get ARRR@5 $= \frac{1}{5}(\frac{1}{3} + \frac{2}{4} + 1 + \frac{4}{5}) = 0.53$, showing that higher recall can compensate for suboptimal ranking.

4 Experimental Results

Experiments were performed with two IR test collections that provide human relevance judgments:

- Ohsumed: 348,566 medical abstracts, annotated with an average of 10.6 so-called MeSH (Medical Subject Headings) terms each. Each of the 14,596 MeSH terms in the collection was treated as a peer and every abstract was assigned to all peers corresponding to its MeSH terms.
- GIRT: 151,318 German abstracts from the social sciences, annotated with an average of 10.2 controlled terms each which were identified with peers as above, resulting in a total of 7,151 peers.

We will illustrate the flaws of the existing evaluation measures with the following example scenario: for each collection, we consider two peer selection strategies, (1) a variant of the CORI resource selection algorithm [3] and (2) a strategy we call "by-size" that ranks peers by the number of documents that they possess.

Both strategies are applied to ranking peers for the queries. The first 100 peers from the ranking are then visited according to the ranking and after each peer is visited, the quality of the results that have so far been retrieved is assessed using all of the evaluation measures discussed above. All peers use the BM25 retrieval function to rank documents locally; idf values are sampled globally (cf. [10]) so that document scores are comparable across all peers.

Fig. 1 shows the effectiveness of the two strategies and different measures as a function of the number of peers visited using Ohsumed. The curves using GIRT qualitatively resemble Fig. 1 but are not shown to preserve space.

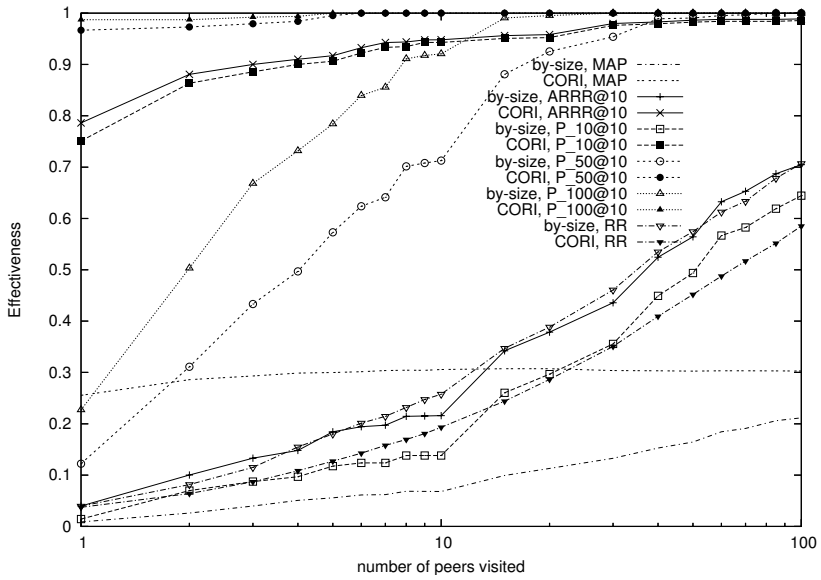

Fig. 1. Effectiveness of "by-size" and CORI as a function number of peers visited in terms of MAP, P_{10}@10, P_{50}@10, P_{100}@10, ARRR@10 and RR for Ohsumed. Note: absolute values of measures may not be compared directly, we need to concentrate on the general shape of the curves.

As one would expect, the CORI strategy is clearly superior to the trivial "by-size" approach when analysed with MAP, ARRR@10 and P_{10}@10. However, the values of P_{50}@10 and P_{100}@10 for the "by-size" strategy catch up with those of CORI rather quickly and RR shows even higher values for by-size than for CORI from the first few peers on.

The results for P_{50}@10, P_{100}@10 and RR allow the conclusion that "by-size" is competitive with CORI after visiting a relatively small number of peers. However, such a conclusion apparently cannot be drawn from the other measures, especially MAP, which is based on human relevance judgments.

The measure P_{10}@10 behaves very similarly to ARRR@10, suggesting that both might be equally trustworthy. It also suggests that the set of the first 10 highest-ranked documents is a better approximation of the set of relevant documents than the first 50 or 100 documents or even the set of all documents with score > 0: since the probability of retrieving some of the N highest-ranked documents increases with N, we will overrate the effectiveness of a strategy as "by-size" (which retrieves *many* documents) at some point. In the case of RR, there is a very high probability of arbitrary documents being considered "relevant"; for Ohsumed, this probability is around 17.5% on average, which, of course, does not mean that the documents are really relevant for the user. Despite the good behaviour of P_{10}@10, it is still (at least) theoretically unpleasing that we do not know which choice of N is optimal.

In further experiments not shown here, we also detected that – when ranking retrieval runs using $P_N@k$ – the rank correlation of two run rankings with different values of N is generally high, but often below 1. This indicates that the problem described above does indeed arise in practice: different choices of N may result in different rankings of systems.

5 Conclusions

In this work, we have introduced a new measure – average ranked relative recall (ARRR) – for comparing the retrieval results of a distributed system against a centralised system. As opposed to previous work, this measure fully exploits the ranking of the centralised system as an indicator of gradual relevance. Experimental results confirm problems of existing approaches in ranking systems consistently (something which ARRR avoids by design) and show that – depending on the result set size N – the measures $P_N@k$ and relative recall may lead to wrong conclusions.

References

1. Akavipat, R., Wu, L.-S., Menczer, F., Maguitman, A.G.: Emerging semantic communities in peer web search. In: P2PIR 2006. Proceedings of the international workshop on Information retrieval in peer-to-peer networks, pp. 1–8 (2006)
2. Bawa, M., Manku, G.S., Raghavan, P.: SETS: search enhanced by topic segmentation. In: Proc. of SIGIR 2003, pp. 306–313 (2003)
3. Callan, J.P., Lu, Z., Croft, W.B.: Searching distributed collections with inference networks. In: Proc. of SIGIR 1995, pp. 21–28 (1995)
4. Lu, J., Callan, J.: Content-based retrieval in hybrid peer-to-peer networks. In: CIKM 2003. Proceedings of the twelfth international conference on Information and knowledge management, pp. 199–206 (2003)
5. Michel, S., Bender, M., Ntarmos, N., Triantafillou, P., Weikum, G., Zimmer, C.: Discovering and exploiting keyword and attribute-value co-occurrences to improve P2P routing indices. In: CIKM 2006. Proceedings of the 15th ACM international conference on Information and knowledge management, pp. 172–181 (2006)
6. Neumann, T., Bender, M., Michel, S., Weikum, G.: A reproducible benchmark for P2P retrieval. In: Proc. of First Int. Workshop on Performance and Evaluation of Data Management Systems, ExpDB (2006)
7. Powell, A.L., French, J.C., Callan, J., Connell, M., Viles, C.L.: The impact of database selection on distributed searching. In: Proc. of SIGIR 2000, pp. 232–239 (2000)
8. Puppin, D., Silvestri, F., Laforenza, D.: Query-driven document partitioning and collection selection. In: InfoScale 2006. Proceedings of the 1st international conference on Scalable information systems, pp. 34–41 (2006)
9. Shokouhi, M., Baillie, M., Azzopardi, L.: Updating collection representations for federated search. In: Proc. of SIGIR 2007, pp. 511–518 (2007)
10. Witschel, H.F.: Global term weights in distributed environments. Information Processing and Management (2007), DOI: doi:10.1016/j.ipm.2007.09.003