H-0228 (H0406-001) June 14, 2004
Computer Science

IBM Research Report

Scaling IR-System Evaluation Using Term Relevance Sets

Einat Amitay, David Carmel, Ronny Lempel, Aya Soffer
IBM Research Division
Haifa Research Laboratory
Mt. Carmel 31905
Haifa, Israel

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Scaling IR-System Evaluation using Term Relevance Sets

Einat Amitay, David Carmel, Ronny Lempel, Aya Soffer
{einat,carmel,rlempel,ayas}@il.ibm.com

IBM Haifa Research Lab, Haifa 31905, ISRAEL

ABSTRACT

This paper describes an evaluation method based on Term Relevance Sets (*Trels*) that measures an IR system's quality by examining the content of the retrieved results rather than by looking for pre-specified relevant pages. Trels consist of a list of terms believed to be relevant for a particular query as well as a list of irrelevant terms. The proposed method does not involve any document relevance judgments, and as such is not adversely affected by changes to the underlying collection. Therefore, it can better scale to very large, dynamic collections such as the Web. Moreover, this method can evaluate a system's effectiveness on an updatable "live" collection, or on collections derived from different data sources. Our experiments show that the proposed method is very highly correlated with official TREC measures.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

IR-system evaluation, Web search evaluation, Trels

1. INTRODUCTION

The evaluation of information retrieval (IR) systems is the process of assessing how well a system meets the information needs of its users. IR research has a well established tradition of comparing the relative effectiveness of different retrieval approaches. The predominant paradigm for system evaluation, first introduced in the Cranfield experiments [6], requires a test collection consisting of a fixed set of documents, a set of topics, and a set of relevance judgments (called Qrels) created by human assessors who mark the documents deemed relevant to each topic. Each retrieval system to be evaluated produces a ranked list of documents for each topic in the test collection. The quality of the system for a single topic is computed as a function of the ranks of the marked documents within the ranked list produced by the system. The effectiveness of the system as a whole is then computed by averaging the scores across the entire set of topics.

The main difficulty of the Cranfield approach is the necessity of complete relevance judgments, i.e., for each topic, all relevant documents in the collection must be discovered and marked. While this may be realistic for small collection, it is certainly infeasible for large ones. Voorhees [19] estimates that nine person months are needed to judge one topic for a collection of 800,000 documents. In order to overcome this difficulty, a *pooling technique* can be used to create a subset of documents to be judged for a topic. Each document in the pool is judged by a human assessor and documents not in the pool are assumed to be irrelevant. Zobel [22] showed that pooling, based on the top 100 results of each participant, can provide a reliable evaluation methodology. Many studies have demonstrated the reliability of evaluation based on Qrels and pooling [3].

1.1 Problems with Qrels-Based Evaluation

Qrels-based evaluation of IR systems is widely used for research purposes by the Text Retrieval Conference (TREC) and by many others. However, several problems with this method make it impractical for very large and dynamic collections. Indeed, the scalability of TREC's evaluation methods has been recently addressed in the SIGIR workshop on evaluation methodologies for terabytes-scale text collection [17]. Moreover, Qrels are in particular inappropriate for many tasks that are present when developing commercial IR systems for intranets and enterprises. Such systems need to be tested on many different collections, derived from different data sources and content types, with perhaps multiple score flavors. Once such systems are deployed, Qrels are again problematic when needing to continuously monitor search quality on "live", updatable indices. Many commercial search engines often use human testers to evaluate the performance of their systems. Following are several examples of the problems with Qrels-based evaluation:

Web indices. Web data is extremely dynamic - millions of pages are created and deleted every day, and the content of existing pages changes quite frequently. A recent study found that within a single week, the content in over 20% of Web pages will undergo some non-trivial change [8]. Consequently, snapshots of the Web taken several weeks apart may give rise to very different indices. Thus, Qrels may rapidly become stale, with new pages replacing old ones as

SIGIR'04, July 25–29, 2004, Sheffield, South Yorkshire, UK.

the most appropriate online resources for queries. In other cases, even though the "old" expected results remain good resources, search engines will not retrieve them in response to queries. Instead, the engines will return near-duplicate pages, that have equivalent content but different URLs than the Qrels (replicated content is prevalent on the Web [2]).

Search over mail and news collection. These are collections where IR systems may want to factor the dates associated with documents into their ranking formulae, biasing the results toward the more recent and fresh news articles or mail documents. Qrels-based approaches inevitably fail when recency is factored into the scores, since documents that existed while the Qrels were assembled will now be ranked less favorably, with fresher documents (that were not available when the Qrels were assembled) getting the advantage. Thus, by design, date-based ranking flavors will achieve low scores in Qrels-based evaluations.

Stand-alone evaluation. In a competitive market, where rival companies cannot share collections with each other but rather must perform quality evaluations in stand-alone mode with only their system at hand, pooling of results is not a viable option and so assembling a comprehensive set of Qrels is a daunting task. This is particularly true over collections of non-hyperlinked documents, where one cannot follow links from mediocre documents to relevant ones.

Indices covering only part of the document space. In many large-scale search systems, the index does not cover the entire document space, yet Qrels are distilled from the entire collection, not necessarily from its searchable (indexed) subset. Web-based Qrels, for example, may contain pages that were not crawled by some search engines, as even the largest engines cover only a fraction of the Web [11]. In such cases, Qrels-based quality evaluations cannot distinguish between imprecision (problems with the ranking formula as applied to the indexed content) and bad coverage (non-comprehensive set of documents in the index). In the Web-IR domain, where crawl policies impact the quality of the resulting index [5, 13], it is important to identify whether low quality stems from insufficient coverage or from an inappropriate score function.

1.2 Our Approach

In order to overcome the deficiencies of Qrels-based approaches we propose a method that measures an engine's quality by dynamically examining the content of the retrieved results rather than by looking for pre-specified relevant pages. Instead of counting the number of relevant documents in the result set, we look for occurrences of a pre-specified list of terms relevant and irrelevant to these queries in the result documents.

Specifically, the input to our evaluation system is a set of queries; each query is associated with two term relevance sets (*Trels*). The first set, *onTopic*, contains terms that are likely to appear in most relevant documents. The second, *offTopic*, contains terms that are unlikely to occur within relevant documents. Given a set of ranked documents returned by an examined system for a given query, our evaluation method scores every result d by considering the appearances of *onTopic* and *offTopic* terms in d. The score of all documents is aggregated into a score for the returned result set as a whole. This aggregation may take into account both the rank of each document d and the score it was assigned by the search system. As with Qrels-based evaluation, the Trels-based scores for all topics are averaged to produce the overall evaluation of the IR system.

The proposed method does not involve any document relevance judgments, and as such is much less sensitive to changes to the underlying collection. Furthermore, the evaluation method is not bound to any specific document collection nor restricted to any point in time, and can thus be used over different collections at different times. We show that there is a strong correlation between an engine's (or equivalently, a ranking function's) quality as measured by our method, and its quality as measured by the official TREC measures. In particular, we demonstrate that our evaluation scheme clearly separates the best (and worst) retrieval systems from the middle of the pack.

The rest of the paper is organized as follows: Section 2 reviews related work on IR evaluation methodologies. Section 3 formally defines Trels-based evaluation, and highlights its advantages. Section 4 describes our experimental setup, and reports the results of our experiments comparing evaluation by Trels to evaluation by Qrels. Conclusions and directions for future research are presented in Section 5.

2. RELATED WORK

Interestingly, an evaluation approach similar to ours was already proposed in 1965 [15]. O'Connor devised a method to validate manual indexing of a collection of 10,000 documents from a pharmaceutical research center. He examined two queries: "toxicity" and "penicillin" by characterizing for each query positive and negative evidence in a way which very much resembles our on/off-topic terms. O'Connor used those lists to test the correctness of the manual indexing system by looking for occurrences of the appropriate terms in the documents indexed under "toxicity" and "penicillin". His results were excellent - between 80% and 100% (depending on the sample) of his predictions regarding indexing accuracy were indeed correct.

Several recent studies have dealt with automatic evaluation without relevance judgments in order to bypass the difficulties related to manual assessment. Soboroff et al. [16] suggested an evaluation methodology which does not require relevance judgments at all. Their method avoids manual assessments by randomly selecting a subset of documents from the pool of each topic, indicating them as (pseudo) relevant to that topic. While the ranking of the IR systems based on this method correlated positively with official TREC ranking, the performance of the best systems and the worst systems was not predicted well.

In a somewhat similar approach, Nuray and Can [14] also avoid the need for relevance judgments by automatically considering a subset of each topic's pool as pseudo-relevant documents. However, instead of selecting that subset randomly, all documents in a pool are ranked according to their similarity to the query and the top ranked documents are considered as (pseudo) relevant. They also show a positive correlation of their ranking with the official TREC ranking. The correlation is similar to the one achieved by the random selection method described in [16].

Wu and Crestani [21] suggested another evaluation measure without relevance judgments. In their "reference count" approach, each document in the ranked list returned by the

evaluated system is scored according to the number of its references in the result lists of all other participants. The final score for the system is the sum of all document scores in its result list. Intuitively, a system with a higher reference count is likely to have more relevant documents than other systems with a lower reference count. While this is true in general, systems using original retrieval methods which are expected to return unique relevant documents not retrieved by other systems might be penalized by this approach.

There have been several studies that evaluate Web search systems using standard TREC methodologies. Hawking et al. [10] evaluated Web search systems using a frozen sample of Web pages to overcome the dynamic nature of the Web (the VLC2 snapshot containing 100GB of Web pages). Using 50 TREC informational topics, they measured the precision at the top 20 results returned by some commercial Web search engines. The amount of manual assessment is thus restricted to 20 documents multiplied by the number of queries and the number of evaluated systems. However, such an evaluation is hard to reproduce. In addition, it is not clear whether the VLC2 is indeed a representative sample of the entire Web, and similarly whether the TREC topics are representative of "real" Web queries.

Beitzel et al. [1] construct a large number of queries and Qrels for automatic evaluation. Queries are chosen from a search engine's query log, and "pseudo-relevant" documents are extracted from the open directory project (ODP)[1]. The assumption is that a category is relevant to a given query if the editor-entered title of that category exactly matches the query. While this approach is fully automatic (ignoring the intensive work done by ODP to construct the taxonomy), and can use a huge number of queries per evaluation, it can only be used for general Web search systems who repeatedly crawl the ODP collection which is dynamically updated. Moreover, since documents marked as relevant by ODP can vanish, the evaluation is not fully reproducible.

Fagin et al. [7], motivated by the Web search scenario where users are mostly interested in the precision of the top results, suggested a distance measure between two top-k lists. Such a measure can be used for comparison of the top results returned by different search systems, or different variants of the same system, without any need of relevance judgment. However, it cannot provide a global quality measure as Qrels do.

3. TRELS-BASED EVALUATION

As outlined in the introduction, the Trels-based evaluation measures an engine's quality by examining the content of the retrieved results rather than comparing them to prespecified relevant pages. Specifically, the method evaluates the content of results by looking for occurrences of a prespecified list of terms believed to be relevant and irrelevant to the query.

The input to our method is a set of queries; each query is associated with *Trels* (Term RELevance Sets), which consist of two sets of terms:

- *onTopic*: contains terms related to the query that are **likely** to appear in relevant documents.
- *offTopic*: contains terms related to the query but **unlikely** to occur within relevant documents.

[1]http://dmoz.org

Intuitively, the appearance of *onTopic* terms in a retrieved document d boosts our confidence that d is a relevant result, while *offTopic* terms appearing in d indicate that d could be irrelevant. Consequently systems whose top ranked results contain many of the *onTopic* terms and few *offTopic* terms are overall more effective in finding high quality documents than others.

Terms may be keywords, phrases, or lexical affinities – pairs of related terms found in close proximity to each other, in a window of small size [12]. Trels may also contain *definitive patterns* [18] as already used by the Question-Answering community for the definition of relevant passages extracted from search results[2]. As a first example, consider the following query (based on the title of TREC topic 419) with the associated sets of terms:

query: "recycle, automobile tires"

onTopic: "rubberized asphalt", "door mats", playground

offTopic: traction, air-pressure, paper, plastic, glass

Rubberized asphalt and door mats are important artifacts of tire recycling; old tires are often reused at playgrounds. The appearance of those terms within documents retrieved by the above query are good predictors for relevant documents. On the other hand, traction and air-pressure are qualities usually discussed in the context of tires that are new or still in use, whereas paper, plastic and glass are other materials that are widely recycled. The presence of these terms would indicate that a retrieved document is either irrelevant or at least not focused on the query at hand.

3.1 Evaluation Measures

The input to the evaluating system consists of a number of triplets of the form (q, *onTopic*, *offTopic*) where q denotes the query and *onTopic* and *offTopic* denote the onTopic and offTopic sets of terms. Given such a triplet and the ranked set of documents D_q returned by the examined system for the query q, our evaluation method scores every result $d \in D_q$ by considering the appearances of *onTopic* and *offTopic* terms in d. The score of all documents $d \in D_q$ is aggregated into a score for the result set D_q as a whole. This aggregation may take into account both the ranks and relevance scores of the individual documents in D_q. Aggregating the result set scores for all queries yields the score of the retrieval system.

We experimented with two evaluation schemes. In the *Basic scheme*, the evaluation score of document d, $tScore(d, q)$, is the weighted difference between the numbers of *onTopic* and *offTopic* terms appearing in d. Formally,

$$tScore_{basic}(d,q) = |t \in onTopic \cap d| - \beta \times |t \in offTopic \cap d| \quad (1)$$

Note that according to this definition, the $tScore$ of a document might be negative (for $\beta > 0$).

In the second scheme, called the *Similarity scheme*, the score of a document is determined by the difference between the cosine similarities of the document to the term vectors induced from the *onTopic* set and the *offTopic* set of terms. Thus a document is scored according to its similarity to

[2]The Trels used in the experiments reported in this paper do not contain definitive patterns

the onTopic vector of terms, and penalized according to its similarity to the offTopic vector of terms:

$$tScore_{sim}(d,q) = cos(onTopic,d) - \beta\cos(offTopic,d) \quad (2)$$

An important difference between the two schemes is the resources that are required to implement them. The basic scheme requires no special IR expertise, as $tScore_{basic}(d,q)$ is easily computed with little programming effort. The similarity scheme is more complex, requiring some IR machinery in order to measure the cosine similarity between the examined results and the term vectors induced from the Trels. One possible implementation relies on a search engine, dedicated for the evaluation, that evaluates queries derived from the onTopic and offTopic term vectors. The *tScore* of any document will be determined by its scores, as returned by the search engine, for the derived onTopic and offTopic queries[3].

Regardless of the selected measure, the scores of all documents for a particular query are aggregated into a score for the returned result set D_q as a whole using the following formula. Let $d_i \in D_q$ denote the i'th ranked document in the ordered set D_q, and let $|D_q| = n$. The aggregation is defined by the weighted average of the documents' *tScore*, where the contribution of each $d_i \in D_q$ is inversely proportional to its rank i:

$$tScore(D_q) = \frac{\sum_{i=1}^{n} \frac{1}{i} tScore(d_i,q)}{\sum_{i=1}^{n} \frac{1}{i}} \quad (3)$$

An alternative measure is the average *tScore* of the top-k results (k=10,20,100) which might be interesting for comparing systems based on their top results:

$$tScore@k(D_q) = \frac{1}{k}\sum_{i=1}^{k} tScore(d_i,q) \quad (4)$$

As usual, the final score of system S is the average score over the entire set of queries Q.

$$tScore[@k](S) = \frac{1}{|Q|}\sum_{q\in Q} tScore[@k](D_q) \quad (5)$$

As mentioned earlier, we expect that relevant results for a query would contain *onTopic* terms with high frequency, while *offTopic* terms would tend to appear in less relevant documents, if at all. Note, however, that relevant documents will usually not contain all of our *onTopic* terms, and will occasionally contain an *offTopic* term. Individual terms are almost never absolute indicators of relevance or lack thereof. The power of the evaluation lies in the aggregation of term appearances across many documents returned for many queries.

Obviously, the Trels methodology does not involve any document relevance judgments, and as such is not adversely affected by changes to the underlying collection. Furthermore, it is not bound to any specific collection nor restricted to any point in time, and can thus be used continuously (at different times) over different collections. Returning to the "recycle, automobile tires" example, over any snapshot of the Web (irrespective of size) taken anytime in the last few years, relevant documents should be rich with *onTopic* terms and rarely contain *offTopic* terms.

To summarize, given a set of evaluation queries and associated sets of *onTopic* and *offTopic* terms. Let S be a system to evaluate, the evaluation is carried out as follows:

1. Let D_q be the ranked set of results returned by S for query q in the set.
2. Compute the *tScore* for each document d in D_q using Equation 1 or Equation 2.
3. Aggregate the scores of all documents in D_q to the *tScore* of D_q using Equation 3 or Equation 4.
4. Average the tScores for all queries in the evaluation set to get the final tScore of S using Equation 5.

3.2 Constructing Trels and suitable queries

Constructing Trels consists of defining the *onTopic* and *offTopic* sets of terms pertaining the query in question. The set of *onTopic* terms can be collected in several manners:

- One may browse through documents that are highly relevant to the query, extracting the terms that distinctly identify the subject of the query. In other words, the extracted terms are those which caused the documents to be considered relevant in the first place. To arrive at such relevant documents, an IR system (e.g., a Web search engine) could be used.
- In cases where the query is well-understood by an expert, the expert may produce some of the *onTopic* terms without resorting to an IR system.
- There are types of queries for which natural *onTopic* terms exist, such as acronyms and the words for which they stand. For example, "international business machines" is an *onTopic* phrase-term for the query "IBM", and vice versa.

We aim to measure if an IR system truely answered the information need behind the query, or rather simply returned documents that contain the query terms. Therefore, while we allow the original query terms to appear in phrases or lexical affinities in the onTopic set, we exclude from the onTopic set the individual query terms (and their linguistic derivatives). Essentially, we grade systems on terms or compounds they have not been exposed to.

At first glance, one may consider the task of defining Trels just as challenging as identifying Qrels; note, however, that *onTopic* terms do not require the scanning of a large and comprehensive set of relevant documents. Definitive *onTopic* terms are usually picked up after scanning a handful of documents. In our experiments we employed mainly the first method. It took on average two person hours per query, looking at no more than 40 pages, to define Trels for TREC topics on which we were by no means subject experts.

In order to associate a concise set of *offTopic* terms with a query, part of the query must be very general or ambiguous, with the other part narrowing its scope or disambiguating its meaning. We have already mentioned the query "recycle automobile tires", where the term "recycle" is quite general, and the second part of the query is required to narrow

[3]Note that the similarity score for a document-query pair returned by most search engines is not the real cosine (or relevance probability), but rather a partial score for ranking purposes. Therefore the on-Score and the off-Score returned by the search engine must be normalized in order to be aggregated into the final *tScore*.

its scope. Another example is the query "Cuba, sugar, exports" (TREC topic 414), where the scope of every pair of words is narrowed by the third word: we are not looking for other Cuban exports (e.g., cigars), we are not interested in sugar-exporting countries such as Brazil, and have no need to know about other aspects of sugar in Cuba (e.g., Cuba's internal consumption of sugar). In TREC topic 447, "Stirling engine", "Stirling" is an ambiguous word, disambiguated by the term "engine". While an IR system might return a document discussing some mechanical issue in Stirling, Scotland, most people would consider such a document as irrelevant for that query. Thus, our process for producing *offTopic* terms consisted of:

- Submitting "partial" queries (without some disambiguating terms) to an IR system.
- Identifying documents returned for the partial queries, that are irrelevant to (or not focused on) the original query.
- Identifying repeating terms in the irrelevant documents that we do not expect to be commonly found in documents that are relevant to the query.

The flip side of the discussion above is that *offTopic* terms cannot be easily associated with some types of queries (e.g., one-term queries). We argue that this is not a limitation of our method, since the space of "agreeable" queries is by far large enough for an effective evaluation suit. Furthermore, we contend that queries that contain an ambiguous term along with a disambiguating term, or a general term with a more precise modifier, are natural benchmarks for testing modern IR systems. Many such systems are faced with Web-like queries, that are typically very short (usually less than 3 terms per query), and often too broad or ambiguous to some extent. IR systems employ many heuristics when ranking documents, and our methodology naturally puts these heuristics to the test, by measuring how well the engine captures the information present in the query. Essentially, the ranking formula is evaluated by testing its ability to exploit a "small" refinement for one aspect of the query and differentiate that aspect from all other aspects. Indeed a successful deployment of Trels-based evaluation includes selecting test queries that exhibit this property.

The connection between query refinement and Trels can be seen from a different angle as well: the *onTopic* set consists of terms one might add to refine the query, whereas the *offTopic* set of terms might also be added in the refinement process, but with "minus", or "not" modifiers. A system with high query expansion capability can assist in Trels creation by recommending refining terms for the query as candidates to the Trels sets.

4. EXPERIMENTS WITH TRELS

To test the reliability of the Trels-based evaluation paradigm we measured the correlation between Qrels-based evaluation and Trels-based evaluation of the same systems. The Qrels-based measures (MAP and P@10) for a specific system were evaluated using the official TREC Qrels and the *trec_eval* program, while the Trels-based measures ($tScore$, $tScore@k$) were evaluated using a set of Trels, manually created by us, for the same TREC topics for which Qrels exist.

The correlation between the two measures was evaluated using the Pearson correlation coefficient and Kendall's$-\tau$[4].

4.1 Trels-based versus Qrels-based evaluation of TREC-8 participants

Our first experiment compared the ranking of 128 participants (Runs) of the TREC-8 ad-hock task [20] as measured by Qrels, to the ranking of the same participants as measured by Trels. Unlike an ideal scenario, in which an evaluator would define queries for which Trels are easily constructed, we were confined to use topics $401-450$ of that specific task. To simulate short, Web-like queries, we used the title of each topic as the query for which Trels will be built. However, not all 50 topics lend themselves to this approach. In particular, TREC Qrels are created to match the topic's narrative which, in some cases, is much more specific than the topic's title. For example, the title of topic 428 is "declining birth rates", but its narrative is: "To be relevant, a document will name a country *other than the U.S. or China* in which the birth rate fell from the rate of the previous year." Since no IR system can possibly be expected to return results that precisely reflect the narrative of topic 428 given its title alone, we removed topic 428 from consideration. Similar considerations eliminated an additional 22 topics, leaving us with 27 of the 50 topics[5].

After identifying the 27 TREC-8 topic titles to be used as queries, we proceeded to construct Trels for each one. This involved examining several relevant documents per query to identify *onTopic* terms, as well as looking for definitive *offTopic* terms. We avoided using TREC-8 documents for this purpose, since that would have resulted with our on/off-topic term sets being biased towards TREC-8 Qrels. In order not to taint our experiments, we collected Trels by submitting our 27 queries (and variations thereof) to Google[6] and examined the returned Web pages. Thus, our Trels were gleaned from an entirely different collection than TREC-8. Furthermore, these Trels were constructed in December 2003, 4 years after the TREC-8 topics were made available. Note that some of these 27 queries were hardly ideal for the Trels-based approach. For example, the Trels assigned to "creativity" (the title of topic 417) consisted of an empty *offTopic* set, as single-term queries cannot be assigned any natural off-topic terms. The constructed Trels contained on average 32.5 terms per onTopic set and 8.5 terms per offTopic set. Figure 1 shows the Trels for topic 414.

Each Run was evaluated twice – once using Qrels, and once using Trels based on Equation 2 with $\beta = 1$. Both evaluations were confined to the 27 topics for which Trels were created. Table 1 shows the correlation between the Qrels-based scores and the Trels-based scores of the 128 Runs as produced by TREC Qrels and by our Trels respectively. It also shows the similarity between the ranking of the runs, using Kendall's$-\tau$ measure. The table also provides the correlation and the ranking similarity between the Qrels-based

[4]The Kendall$-\tau$ correlation between two rankings is measured by the minimum number of pairwise adjacent swaps required to turn one ranking into the other. This number is normalized such that the similarity between two identical rankings is 1.0, and between two inverse rankings is -1.0.

[5]The TREC topics selected for Trels creation are: 401,402,407,409,410,411,412,414,415,417,418,419,421,422,423, 426,427,430,433,434,435,436,441,442,443,447,449.

[6]http://www.google.com/

query: "Cuba, sugar, exports"

onTopic: "million tons", "Pinar del Rio", "raw sugar", "sugar export", "sugar industry", "sugar mills", "sugar ministry", sugar*crop, Cuban*sugar, export*ton, prices*sugar, shipment*sugar, slump*sugar, sugar*agroindustry, sugar*company, sugar*factories, sugar*loading, sugar*ports, sugar*production, sugar*railways, sugar*revenue, sugarcane*cuba

offTopic: missile, coup, capitalist, "life expectancy", political system, tobacco, "Cold War", President, embargo, cigars

Figure 1: Trels for TREC topic 414. Phrases appear within quotes; lexical affinities are marked by '*'.

p@10 and MAP measures, for relative comparison. Figure 2 plots the relation between the tScores of the 128 Runs and the corresponding Qrels-based scores, p@10 and MAP.

	Correlation	Kendall's$-\tau$
p@10-tScore	0.951	0.734
MAP-tScore	0.938	0.746
p@10-tScore@10	0.944	0.732
MAP-tScore@10	0.875	0.711
p@10-tScore@100	0.909	0.675
MAP-tScore@100	0.883	0.682
p@10-MAP	**0.956**	**0.842**

Table 1: The correlation and the ranking similarity between P@10, MAP measures and tScores.

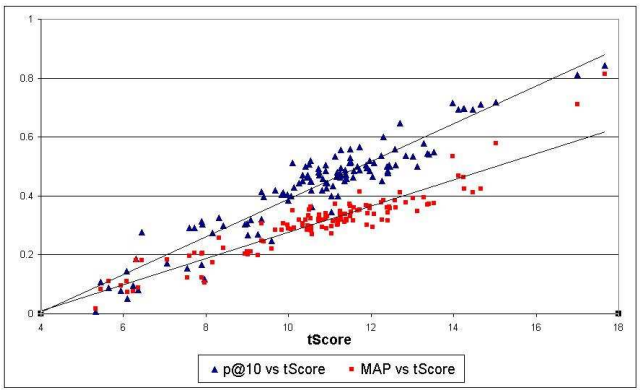

Figure 2: The relation between the tScores of TREC-8 Runs and their Qrels-based scores, p@10 and MAP.

The correlation between Qrels-based measures and Trels-based measures is extremely high. Basically, the tScores are almost as correlated with the Qrels-based scores as both Qrels-based scores are correlated with each other. As expected, MAP is less correlated with $tScore$@10 and $tScore$@100 since MAP depends on the entire set of results while these measures only depend on the top results. In general, according to these results, we can definitely say that Trels-based scores can highly predict the Qrels-based scores, thus Trels-based evaluation is as good as Qrels-based evaluation for measuring systems' effectiveness. Furthermore, these results indicate that Trels are good predictors for the best and worst systems alike.

4.2 Trels-based evaluation on different collections

In the second experiment we evaluated the effectiveness of Trels when invoked on yet another collection, the WT10g collection. We produced 10 variations of our home-grown search engine Juru [4] and compared the relative effectiveness of these variations by two means:

- Using the official Qrels for topics 501-550 of the WebTrack-10 ad-hock task [9]. Each variant of our system invoked topics 501-550 against the WT10g collection (using the topic's title as a query) and was scored using the official Qrels for these topics.

- Using the Trels created for the 27 topics extracted from the 50 topics of TREC-8 ad-hock task. Each variant invoked the 27 queries against the WT10G collection and was scored according to Equation 2 with $\beta = 0.5$.

Assuming that there is a "universal" ordering of the 10 variants by their quality as IR systems, we would expect both measurements to agree on that order. Table 2 shows the correlation and the ranking similarity between the Qrels-based measures and the Trels-based measure for the 10 runs.

	Correlation	Kendall's$-\tau$
p@10-tScore	0.993	0.944
MAP-tScore	0.960	0.922
p@10-MAP	**0.961**	**0.866**

Table 2: Correlation and ranking similarity between Qrels-based P@10,MAP and Trels-based tScore of the 10 Runs, measured over the WT10G collection.

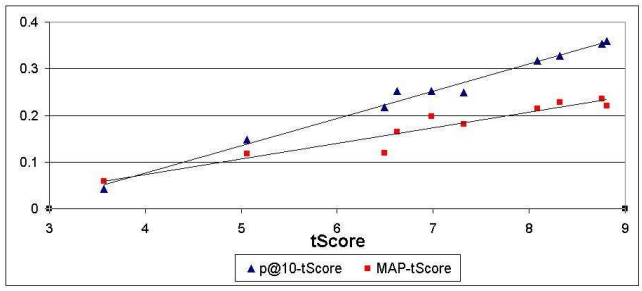

Figure 3: The relation between the tScores of 10 variations of our search engine and p@10, MAP, as measured over the WT10G collection.

Figure 3 plots the relation between the tScores of the 10 variants of our system, and their Qrels-based scores, p@10 and MAP. As in the previous experiment, the correlation between Qrels-based measures and Trels-based measures is extremely high. Furthermore, the rank similarity between the tScores and either of p@10 and MAP is higher than the rank similarity between p@10 and MAP. Overall, the Trels performed on the WT10g collection just as well as they did on the TREC-8 collection, even though their assembly process involved neither collection. In other words, this demonstrates that indeed a set of queries and associated Trels can be successfully used on different and unrelated collections.

4.3 Several variations of Trels-based evaluation

In Section 3, we proposed two schemes for Trels-based evaluation: the basic scheme (Equation 1) and the similarity scheme (Equation 2). In order to compare the two, we measured the correlation and similarity between the rankings induced by both schemes while evaluating the results of the 128 participants of TREC-8. The correlation between the tScores of the 128 Runs as computed by the basic scheme and the tScores as computed by the similarity scheme is 0.991 and the ranking similarity as measured by Kendall's-τ is 0.92. The high correlation between the two schemes indicates that both can effectively be used for Trels-based evaluation. The choice of which score to use thus only depends on which flavor is easier to implement in a particular environment.

We also experimented with the relative weight between the scores for the onTopic query and offTopic query (the β value in Equation 2). Figure 4 shows the correlation between Trels-based evaluation and Qrels-based evaluation of the 128 participants of TREC-8 for values of β between -3 and 3.

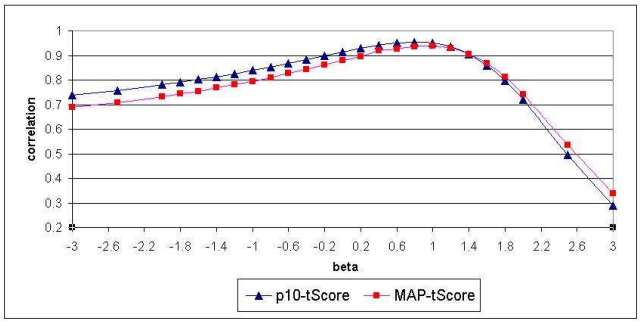

Figure 4: The correlation between Trels-based evaluation and Qrels-based evaluation as a function of the relative weights in Equation 2.

The results indicate the significance of the offTopic terms for Trels-based evaluation. The highest correlation with Qrels is obtained when the relative weight between onTopic terms and offTopic terms is equal ($\beta = 1.0$). Setting the weight of offTopic terms to high values decreases the correlation between Qrels and Trels; hence, it decreases the quality of Trels evaluation. Ignoring the offTopic terms by setting β to zero hurts the evaluation quality as well. Setting a negative weight to the offTopic terms is equivalent to treating them as onTopic terms – an action that deteriorates evaluation quality as expected. The moderate pace of decrease in correlation as β is lowered from 1.0 seems to be due to the fact that the average number of *offTopic* terms per query (8.5) was relatively small, with several queries even having empty *offTopic* sets.

5. DISCUSSION

This work describes an evaluation method based on Trels rather than Qrels. This method measures an IR system's quality by examining the content of the retrieved results rather than by looking for pre-specified relevant pages. The method does not involve any document relevance judgments, and as such is less sensitive to the underlying collection. Therefore, it can scale better to a huge and dynamic collection such as the Web, and can evaluate system's effectiveness on an updatable "live" collection, or on different collections derived from different data sources.

Several previous works have suggested evaluation methods without relevance judgment. The typical ranking similarity between these methods and Qrels-based methods is approximately 0.5 [16, 14]. We experimented with a set of Trels created for 27 TREC-8 topics. The Trels-based evaluation of TREC-8 participants was highly correlated with the Qrels-based evaluation (correlation is approximately 0.93 and Kendall's$-\tau$ ranking similarity is 0.73). Furthermore, the same set of Trels achieved extremely high correlation with Qrels-based scores on a different collection: we created ten variations of our search system, and ran them over the WT10G collection. The effectiveness of those 10 flavors, as measured by our set of 27 TREC-8 based Trels, agreed almost entirely with their effectiveness as measured by the Webtrack-10 topics and Qrels.

While accumulating a large set of appropriate Trels is not trivial, it is much less laborious than evaluating search results for the purpose of collecting Qrels. Furthermore, the efforts invested in assembling Trels scale better and last longer. In particular:

Trels are scale-free: as the collection grows, the pool of results for Qrels creation may also grow since different systems are expected to return more unique results. This, in turn, results in an increase of the amount of effort required to mark Qrels. On the other hand, quality Trels will evaluate systems effectiveness regardless of the size of the collection which yielded the results.

Trels are (almost) forever: they expire much slower than Qrels, if at all. As the experiments of the previous section show, evaluation by our Trels, which were assembled by examining Web documents returned by Google in December 2003, were highly correlated with evaluation by Qrels on the TREC collection from the early 90's and on the WT10g collection from 2001.

Trels are global: they can be developed independently of any specific collection, and then reused with many collections. Our experiments demonstrated that Trels gleaned from today's World Wide Web are suitable for evaluating systems over two different collections from several years ago.

Note that we do not claim that one single set of Trels fits all collections. Naturally, Biotech collections will require different Trels than collections of aerospace documents, and general Web-type Trels will differ yet from both. However, aerospace-flavored Trels should be able to evaluate search systems deployed throughout the aerospace industry.

This global nature of Trels is particularly useful for tuning the ranking formulas of intranet search systems. The developers of such systems usually have little or no access to the collections on which the systems will actually run. When composing queries and the corresponding Trels, they need to be familiar with the nature of the customer's business (eg. Biotech versus Aerospace), and may need to consult with their customers in the process. However, they need not be exposed to the client's confidential documents, as is entailed by Qrels-based approaches.

The following directions are left for future work:

- Facilitating the creation of Trels: how to choose appropriate queries for evaluation, and how to pick the right on/off topic terms.
- Examining the robustness of Trels: how sensitive is the Trels-based evaluation to the specific set of Trels. Preliminary tests indicate that the Trels-based method is indeed robust. Dividing the set of Trels to two arbitrary sets resulted in very highly correlated evaluations as well. A related issue is studying just how many terms are usually needed for on/off topic sets.
- Examining the significance of a specific query for evaluation: A good query for evaluation differentiates between the evaluated systems according to their ability to retrieve good results. Queries in the evaluating set might be weighted according to their expected contribution for evaluation. Similarly, *onTopic* and *offTopic* terms might be weighted as more (or less) important indicators of topicality.

6. REFERENCES

[1] Steven M. Beitzel, Eric C. Jensen, Abdur Chowdhury, and David Grossman. Using titles and category names from editor-driven taxonomies for automatic evaluation. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 17–23. ACM Press, 2003.

[2] Andrei Z. Broder, Steve C. Glassman, and Mark S. Manasse. Syntactic clustering of the web. In *Proceedings of the sixth International World Wide Web Conference*, 1997.

[3] Chris Buckley and Ellen M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 33–40. ACM Press, 2000.

[4] David Carmel, Einat Amitay, Miki Herscovici, Yoelle S. Maarek, Yael Petruschka, and Aya Soffer. Juru at TREC 10 - Experiments with Index Pruning. In *Proceeding of Tenth Text REtrieval Conference (TREC-10)*. National Institute of Standards and Technology. NIST, 2001.

[5] Junghoo Cho, Hector García-Molina, and Lawrence Page. Efficient crawling through URL ordering. *Computer Networks and ISDN Systems*, 30(1–7):161–172, 1998.

[6] Cyril W. Clevedron. The significance of the cranfield tests on index languages. In *Proceedings of the Fourteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,*, pages 3–12, 1991.

[7] Ron Fagin, Ravi Kumar, and D. Sivakumar. Comparing top k lists. In *ACM-SIAM Symposium on Discrete Algorithms (SODA '03)*, pages 28–36, 2003.

[8] Dennis Fetterly, Mark Manasse, Marc Najork, and Janet Wiener. A large-scale study of the evolution of web pages. In *Proceedings of the sixth International World Wide Web Conference (WWW2003), Budapest, Hungary*, pages 669–678, 2003.

[9] D. Hawking and N. Craswell. Overview of the TREC-2001 Web Track. In E. M. Voorhees and D. K. Harman, editors, *Proceedings of the Tenth Text Retrieval Conference (TREC-10)*. National Institute of Standards and Technology. NIST, 2001.

[10] David Hawking, Nick Craswell, Paul Thistlewaite, and Donna Harman. Results and challenges in Web search evaluation. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1321–1330, 1999.

[11] Steve Lawrence and C. Lee Giles. Searching the world wide web. *Science*, 280, April 1998.

[12] Yoelle Maarek and Frank Smadja. Full text indexing based on lexical relations: An application: Software libraries. In *Proceedings of the Twelfth International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 198–206, Cambridge, MA, June 1989.

[13] Marc Najork and Janet L. Wiener. Breadth-First Crawling Yields High-Quality Pages. In *Proceedings of the tenth International World Wide Web Conference*, pages 114–118, Hong Kong, May 2001.

[14] Rabia Nuray and Fazli Can. Automatic ranking of retrieval systems in imperfect environments. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 379–380. ACM Press, 2003.

[15] John O'Connor. Automatic subject recognition in scientific papers: An empirical study. *Journal of the ACM*, 12(4):490–515, October 1965.

[16] Ian Soboroff, Charles Nicholas, and Patrick Cahan. Ranking retrieval systems without relevance judgments. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 66–73. ACM Press, 2001.

[17] Ian Soboroff, Ellen Voorhees, and Nick Craswell. Summary of the SIGIR 2003 workshop on defining evaluation methodologies for terabyte-scale test collections. In *SIGIR Forum*, volume 37 (2), 2003.

[18] M. M. Soubbotin. Patterns of potential answer expressions as clues to the right answers. In *Proceeding of Tenth Text REtrieval Conference (TREC-10)*. National Institute of Standards and Technology. NIST, 2001.

[19] Ellen Voorhees. The philosophy of information retrieval evaluation. In *Proceedings of the Second Workshop of the Cross-Language Evaluation Forum, (CLEF 2001)*, pages 355–370, 2001.

[20] Ellen M. Voorhees and Donna Harman. Overview of the eighth text retrieval conference (trec-8). *Information Processing and Management*, 36(1):3–35, 2000.

[21] Shengli Wu and Fabio Crestani. Methods for ranking information retrieval systems without relevance judgments. In *Proceedings of the 2003 ACM Symposium on Applied Computing (SAC)*, pages 811–816. ACM Press, 2003.

[22] Justin Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August, Melbourne, Australia*, pages 307–314, August 1998.