Relevance Evaluation of Search Engines' Query Results

Longzhuang Li, Yi Shang, and Wei Zhang
Dept. of Computer Engr. and Computer Sci.
University of Missouri-Columbia
ll059@mizzou.edu, shangy@missouri.edu, wz206@mizzou.edu

ABSTRACT
In this paper, we apply four popular relevance scoring methods to evaluate six major search engines' query results based on a large number of sample queries. The methods include vector space model, Okapi similarity measurement, cover density ranking, and a three-level scoring method. Our experimental results show that search engine Google is always the best, whereas the relative performance of the other search engines varies depending on the relevance scoring method.

Keywords
Relevance Measurement, Search Engine Precision

1. INTRODUCTION
Search engines are designed to quickly find useful information on the Web. With thousands of search engines available and each with different indexing/ranking method and different coverage, it is important to know which search engine returns the most relevance Web pages, or has the highest precision on the top hits. In the past, precision of search engines has been evaluated based on general subject queries and some specific query domains, both manually and automatically. The benefit of manual evaluation is the accuracy with respect to user's expection. The drawback is that it is subjective and time-consuming. Automatic evaluation is much better in adapting to the fast changing Web and search engines, as well as the large amount of information on the Web.

Many relevance scoring methods have been developed to evaluate the relevance of hits returned by search engines. In this paper, we present our experimental results of applying four popular relevance scoring methods to evaluate six major search engines' query results. These methods include vector space model, Okapi similarity measurement[2], cover density ranking[1], and a three-level scoring method. The first three methods have been widely studied in the traditional information retrieval (IR) field. In adapting them to the Web, we estimate their key parameters either based on previous work or through sampling. The last method were developed by us to mimic commonly used manual evaluation methods. Since the performance of search engines may vary from one query domain to another, we used two large query sets derived from two query domains.

2. RELEVANCE EVALUATION
In this section, we present the four relevance scoring methods.

2.1 Vector Space Model (VSM)
VSM has been widely used in the traditional IR field. Most search engines also use similarity measures based on this model to rank Web documents. This method is based on the number of occurance of query terms in the document.

To apply VSM to the Web, we need to estimate a key parameter, the inverse document frequency (IDF). IDF is calculated based on the total number of documents on the Web and the number of documents containing a query term. There are three ways to come up with an estimation: (1) making a simple assumption such as the IRF is a constant for all documents, (2) sampling the Web, and (3) using information from search engines, or/and experts' estimation. In our experiment, we used a combination of the second and the third approaches.

The Web is estimated to contain around 1 billion indexable pages by Inktomi and the NEC Research Institute in February, 2000. Thus, we set the total number of documents on the Web to 1,000,000,000. To estimate the number of documents containing a certain query term, we use information obtained from multiple search engines as follows: (1) submit the term to the search engines and get the number of documents containing the term from each of them; (2) normalize the number of each search engine based on the relative size of the search engine to that of the whole Web. The sizes of the six search engines, *AltaVista, Fast, Google, Go, iWon*, and *NorthernLight*, used in our experiments are 340, 340, 560, 50, 500, and 270 million documents, respectively, as reported in previous studies [3]; (3) take the median of the normalized numbers of the search engines as the final result.

2.2 Okapi Similarity Measurement (Okapi)
Okapi is another popular method in the IR field. To apply the Okapi method to the Web, we need to know the values of the following parameters: (a) the total number of documents on the Web, (b) the number of documents containing a query term, (c) the average length of Web documents. In our experiment, the first two parameters can be estimated in the way described in the last section. The last parameter is estimated as the average length of the top 20 hits from six search engines based on all the queries (3109 queries) in the two query sets. The average length of Web documents was estimated to be 10,939 bytes after removing all the HTML tags and Java scripts.

2.3 Cover Density Ranking (CDR)
CDR is developed to meet user's expectation better – a document containing most or all of the query terms should

be ranked higher than a document containing fewer terms, regardless of the frequency of term occurrence. To adapt CDR to the Web, we need to find out how many distinct query terms a document has and rank documents with more distinct terms higher. Our version of CDR method computes the relevance scores of documents in two steps: (1) Documents are scored according the regular CDR method. Each document belongs to a coordination level group and has a score within that group. (2) The scores are normalized to range (0, 1] for documents containing only one term, to range (1, 2] for documents containing two different terms, and so on, so forth.

2.4 Three-Level Scoring Method (TLS)

The TLS method is developed to mimic commonly used manual evaluation methods. The method consists of two steps:

1. Given a query phrase q with n terms and a Web page x, a raw score is calculated as:

$$A(q,x) = \frac{t_n \cdot k^{n-1} + t_{n-1} \cdot k^{n-2} + \cdots + t_1}{k^{n-1}} \quad (1)$$

where k is a constant, corresponding to the weight for longer sub-phrases; $t_i, 1 \leq i \leq n$, is the number of occurrence of the sub-phrases of length i, i.e., containing i terms.

2. Convert $A(q,x)$ to a three-level similarity score through thresholding, with 2 for relevant, 1 for partially relevant, and 0 for irrelevant:

$$sim(q,x) = \begin{cases} 2 & \text{if } A(q,x) \geq \Theta \\ 1 & \text{if } \Theta > A(q,x) \geq \alpha\Theta \\ 0 & \text{if } A(q,x) < \alpha\Theta \end{cases} \quad (2)$$

where Θ is a constant threshold, representing the requirement for being relevant; α is a value between 0 and 1, representing the requirement for being partially relevant.

In our experiment, we used the default setting, $\Theta = 1, \alpha = 0.1$, and $k = 10$. We found that the relative performance of search engines computed using the TLS was not very sensitive to the parameter values of TLS and the default setting generally worked well.

3. EXPERIMENTAL RESULTS

In the experiments, we evaluate the relevance scores of query results of six search engines, *AltaVista (AV), Fast (F), Google (G), Go (IS), iWon (W)*, and *NorthernLight (NL)*. Two sample query sets were used: (a) the TKDE set containing 1383 queries derived from the index terms of papers published in the IEEE Transactions on Knowledge and Data Engineering between January 1995 and June 2000, and (b) the TPDC set containing 1726 queries derived from the index terms of papers published in the IEEE Transactions on Parallel and Distributed Systems between January 1995 and February 2000. These queries are all very short, containing 2, 3, or 4 terms. Short queries are important in the Web application because (a) most queries submitted to search engines are short, consisting of three terms or less; (b) most technical phrases have no more than 5 terms; (c) few Web pages contain exact query phrases with more than 4 terms; and (d) single-term queries are not good at evaluating precision of search engines. Two search modes were used in the experiments: the default search mode and the exact phrase search mode.

We only considered the top 20 hits from each search engine. To compute the relevance score, we followed each hit to retrieve the corresponding Web document. Then the HTML document was converted to a ASCII file with all HTML tags and scripts removed. The same query was submitted to all search engines simultaneously. The search agent tried to follow each hit and gave up when the connection could not be established in one minute.

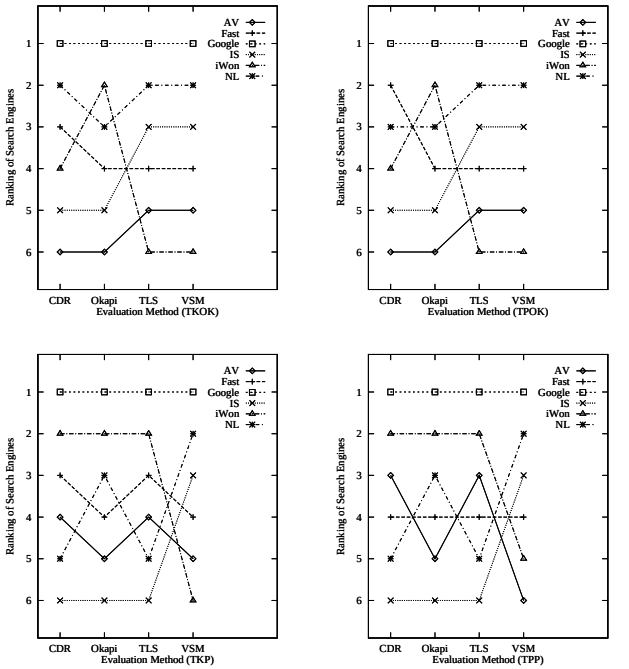

Figure 1: Ranking of the six search engines using the four scoring algorithms. TKOK (upper left) for default search+TKDE query set; TPOK (upper right) for default search+TPDC query set; TKP (lower left) for exact phrase search+TKDE query set; and TPP (lower right) for exact phrase search+TPDC query set.

Figure 1 shows the relative performance of the six search engines based on their average relevance scores computed using the four scoring methods, respectively. Rank 1 is the best and 6 is the worst. *Google* is always the best, whereas the ranking of other search engines varies with respect to the scoring method, the query set, and the search mode. *NorthernLight* is mostly second in the default search mode, whereas *iWon* is mostly second in the exact phrase search mode. *AltaVista* is mostly worst in the default search mode, whereas *Go* is mostly worst in the exact phrase search mode.

4. REFERENCES

[1] Charles L.A. Clarke, Gordon V. Cormack and Elizabeth A. Tudhope. Relevance Ranking for One to Three Term Queries. *Information Processing & Management*, 36:291-311, 2000.

[2] David Hawking, Peter Bailey and Nick Craswell. Acsys Trec-8 Experiments. In *Proceedings of the TREC-8*, 2000.

[3] Danny Sullivan. Search Engine Sizes. *http://searchenginewatch.com/reports/sizes.html*, July 2000.