Distributed Web Search Efficiency by Truncating Results

Christopher T. Fallen
Arctic Region Supercomputing Center
University of Alaska Fairbanks
Fairbanks, AK USA
fallen@arsc.edu

Gregory B. Newby
Arctic Region Supercomputing Center
University of Alaska Fairbanks
Fairbanks, AK USA
newby@arsc.edu

ABSTRACT

A large set of Web documents (the TREC GOV2 collection) comes from many separate Internet hosts, such as `www.nih.gov` and `travel.state.gov`. There is considerable variability in the number of Web pages (i.e., documents) from each host. In this paper, we present and evaluate a method for setting a maximum number of "hits" that may be presented for each web host. Federated search environments are increasingly common components of digital libraries and in these environments, the benefit of such a maximum is that it can reduce the number of possibly relevant documents presented by each subcollection, without hurting early precision measures such as P@20. Derivation of a maximum number, which is proportional to the subcollection size but not sensitive to different search topics, is made possible by an analysis of patterns of relevance judgment across approximately 17,000 web hosts in GOV2.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; H.3.7 [**Digital Libraries**]: Collection

General Terms

Algorithms, Experimentation, Human Factors, Performance

Keywords

World Wide Web, Domain Name, Host Name, Information Retrieval, Relevance Ranking, Filtering, Merging, Ranked Distributions

1. INTRODUCTION

This paper presents and evaluates a method for improving the efficiency of merging search results from multiple data sources. Specifically, we will seek a simple formula for estimating the maximum number of results to present from each data source, without hurting IR performance. To derive the method for choosing the maximum number of results, we will analyze the distribution of the numbers of relevant and non-relevant documents in each collection, using relevance judgments from the TREC GOV2 corpus.

The general framework for merging search results from multiple data sources is described in our work on Grid Information Retrieval (GIR) in [9], also online at `www.gir-wg.org`. There are several motivations for our work on GIR:

1. Not all potentially searchable data are accessible to monolithic search engines such as Google and Yahoo. This grey web, or dark web (see [26, 21]), includes items behind firewalls or passwords, as well as content not readily amenable to general-purpose search engines.

2. Researchers, particularly those in interdisciplinary studies, often need to search multiple digital library sources simultaneously and efficiently. Tools to identify high-quality sources and exclude query results from large but low-quality sources will increase the productivity of digital library searches.

3. Different IR systems might be better able to meet particular types of information needs. For example, while general-purpose monolithic search engines tend to work with relatively unstructured HTML, there are more structured data types (such as XML, per [19]) as well as users with more structured search topics, which might benefit from specialized IR systems.

4. As evidenced by the large number of metasearch engines, there is a perceived need to federate search results from multiple IR systems. (See, for example, [22, 1].) Yet, most metasearch engines have no effective means of providing relevance rankings which span multiple result sets. As such, they are not much improved from the days of WAIS [14].

5. In a federated search environment, we will likely be able to allocate larger computational resources to a topic by searching only a subset of available collections. Similarly, we might be able to divide and conquer a large collection into a set of subcollections.

6. Federated search applications in "Enterprise" environments need to scale to multiple extended enterprises distributed worldwide through partnerships and internal business relationships. Collections of information, often measured in petabytes, are segregated by geography, purpose, legal regulation, and time frame, all

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
JCDL'07, June 18–23, 2007, Vancouver, British Columbia, Canada.
Copyright 2007 ACM 978-1-59593-644-8/07/0006 ...$5.00.

the while remaining accessible and searchable to all authorized users [20].

7. Because some search engines might outperform others, we want to be able to identify the best possible merging technique to integrate result sets from each.

We start with the goal of presenting a combined set of results from multiple collections, with high early precision. In other words, it will be necessary to perform relevance ranking across a new larger set of documents returned from some number of smaller collections. For the past few years, we have used the TREC GOV2 collection, divided into several hundred smaller collections (described further below). More recently for TREC 2006, we have looked at each of the over 17,000 Internet hosts as a separate subcollection.

This paper does not focus on the merging technique for a unified set of relevance rankings we have employed in the past (described briefly below and in our TREC papers, informed by the work of [4]). Rather, it looks at domain characteristics of the GOV2 collection, and seeks to gain efficiency in merging by limiting the number of search results from each subcollection.

2. RESULT SET MERGING EXPERIMENT IN THE TREC 2006 TERABYTE TRACK

To study result-set merging, or the multisearch problem, for the 2006 TREC Terabyte Track we simulated a large-scale distributed IR application by using the open-source IR application Amberfish [18] to index and search each of the 17,000 host names in GOV2 independently. In this paper "domain" and "host" are used interchangeably and both refer to the fully qualified domain name. Amberfish was first used in the 2004 TREC Terabyte Track [17]. The result sets for each topic from each Web domain name were saved to disk. We used the GNU `sort` application (the "sort merge") on the relevance scores in the domain result sets for a topic as a baseline merge application to merge the results into a single ranked list. A "log merge" application used for comparison and described below renormalizes the relevance scores in each result set before sorting on the normalized relevance scores.

The log merge application used in our TREC experiment [7] is a variant of the logistic regression method described in [10, 4, 23]. The goal of the renormalization step is to assign parameters p and q to each result set $\{(x_i, y_i, d_i)\}$, where d_i is the ith ranked document, $x_i = \log i$, and y_i is a monotonically decreasing sequence of relevance scores, so that the curve

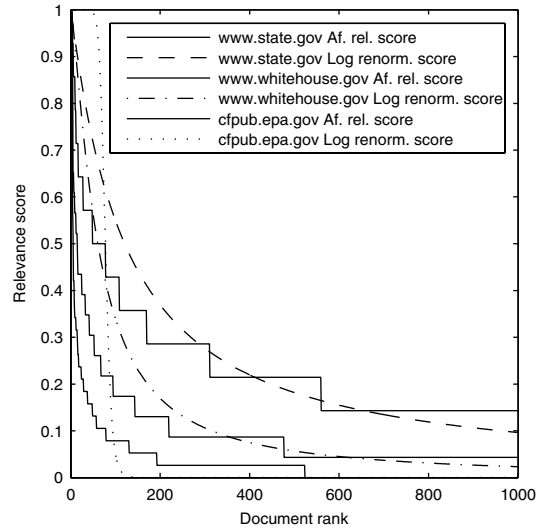
$$f_{p,q}(x) = \frac{e^{p+qx}}{1 + e^{p+qx}}$$

best fits the relevance scores of the result set in the least squares sense of minimizing the sum of the squares of the residuals. The implementation of this log merge result-set merging algorithm actually solves a simpler equation derived by taking the logarithm of the equation above to get an equation

$$(\log \circ f_{p,q})(x) = p + qx - \log(1 + e^{p+qx})$$

that very nearly describes a line over the IR data of interest. Then a standard linear least squares routine will yield estimates for the slope and intercept of the best-fit line and hence, the values of the parameters p and q. New relevance scores are then assigned to each document in the result-set by evaluating the function $f_{p,q}$ at each (log) rank and the process is repeated for all domain result-sets for a topic before re-ordering the entire set of results on the new relevance scores. The effects of this renormalization curve are shown in Figure 1 for the result sets from an arbitrary selection of Web hosts retrieved by Amberfish for TREC Terabyte Topic 804: `ban on human cloning`. Renormalization by the method described above effectively boosts the relevance scores of documents near the top of a result set and damps the relevance scores of the documents near the end of the list, the effect of the boosting or damping is typically greater for short result sets than long result sets.

Figure 1: Relevance scores and Log-normalized relevance scores vs. document rank for search results from selected domain names on TREC Topic 804



Note that the original relevance scores output by the version of Amberfish used in TREC 2006 are all normalized so the top document from every result set has the same score. In addition, many of the result sets retrieved from the individual Web hosts in GOV2 are very short; many result sets had only one document and this is either a boon or a bust, depending on the perspective taken, when the end goal is to fit a line to the data as it is with the log merge algorithm. These factors combined to effectively poison our results for the baseline "sort merge" experiment and the "log merge" experiment: many documents from the 1-hit result sets were assigned the top relevance score of 1.0 in the sort merge experiment and those documents in the log merge experiment were assigned a relevance score of 0.5. The fundamental problem with this approach, and large-scale distributed information retrieval in general, is that without a means to relevance rank entire collections, the only obvious way to compare the relevance to a query of two isolated documents from different collections is to examine the documents themselves. Additional factors in our approach, like neglecting to apply some standard IR techniques like stop-words [8] or thesaurus-based query expansion [15], solidified our "below median" ranking among the 2006 TREC Terabyte Track

participants. However, useful measurements of how truncating the domain result-sets before merging affects the quality of the final merged results may still be obtained by examining the best-performing topics within our 2006 TREC Terabyte results. The total number of documents contained in a Web host will be the parameter that determines how many documents from the domain result set to keep in the final ranked list.

3. VARIATIONS IN HOST SIZE IN THE GOV2 TEST COLLECTION

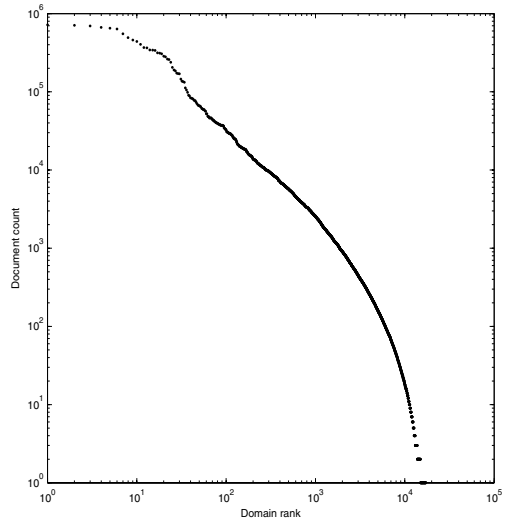
As described in [5], the GOV2 document corpus was first distributed by CSIRO in Australia in 2004 as part of the Terabyte Track in the Thirteenth Annual Text REtreival Conference (TREC) held at the National Institute of Standards and Technology (NIST). Each year from 2004 to 2006 NIST distributed 50 topics to Terabyte Track conference participants [5, 6, 3]. The GOV2 corpus is approximately 426GB of marked-up text divided into 25 million documents collected from publicly crawlable Web pages within the `.gov` Web domains in early 2004 [25]. This collection represents a significant proportion of the Web pages in the top-level `.gov` domains during the crawling process and contains text and HTML documents as well as the text extracted from PDF, Word, and postscript files [5]. A unique identification string is assigned to each document as is meta information including the full uniform resource locater (URL) or Web address.

The GOV2 collection can be divided or *partitioned* into disjoint subsets of documents corresponding to `.gov` Web domains by parsing the URL of each document to extract the host name. When partitioned by host name, the GOV2 collection can be searched in a manner similar to a federated search of a large digital library containing thousands of document subcollections of a wide range of sizes. Consequently inferences from and search methods developed for the GOV2 collection partitioned by host name may be useful to digital libraries with a large heterogeneous set of document subcollections.

Let the *size* of a Web domain name in GOV2 be the number of documents addressed under the fully qualified domain name. Then there are approximately 17,000 Web domain names in GOV2 and the mean domain size is about 1,500 pages with a standard deviation of about 18,000 pages. Similar to term-frequency distributions in large text collections, the standard deviation of domain sizes is comparable to, and in this case an order of magnitude larger than, the mean domain size.

If the sizes of the GOV2 hosts are plotted against the hosts ranked in order of decreasing size to form the rank-size plot in Figure 2, the size distribution of GOV2 Web hosts is approximately a power law as indicated by the linear relation on the log-log scale. Although the rank-ordered size distribution is technically distinct from a probabilistic size distribution, ranked distributions are useful empirical characterizations of complex systems like the Internet. This is because ranked distributions are not sensitive to individual realizations such as the ranked Web domain sizes from `.gov` in 2004 of a statistical ensemble that may be very broad like the collection of all Web domain sizes in the past, present, and future [11].

Figure 2: Rank-size plot of Web host size vs. Web host ranks in GOV2



The empirical observation readily apparent in Figure 2 is that *a few Web domains are very large and very many Web domains are almost trivially small.* In other words, a significant fraction of the Web pages in GOV2 are contained in the largest hosts and yet the vast majority of hosts are small. The top ten largest domain names account for more than 20% of the total number of pages in GOV2 and more than 20% of the Web hosts in GOV2 contain three or fewer pages each. The largest, smallest, and a selection of mid-sized domains are listed in Table 1.

An analogue to the well-known family of Zipf's laws like the "80-20 rule", the Pareto principle, Bradford's law, or Lotka's law, appears to proportionally relate the frequency of occurrence of host-sizes to a power of the host size as in Figure 3 or, alternately, proportionally relate the size of a host to a power of the size-rank of the host as in Figure 2. A specific "80-20" rule that applies to the GOV2 host sizes is: the largest 10% of the GOV2 hosts contain almost 93% of all the documents in GOV2. Evidence that the distribution of relevant documents among hosts over an aggregate of query topics also follows a Zipfian distribution is presented in Section 4 and the similarity to the host size distribution described above is the intuitive basis behind a pragmatic conjecture described in Section 5 that relates the *number of documents relevant to a given topic* contained in a host that retrieves documents for that topic to the *size of the host.*

These observations are also consistent with the probabilistic size-distribution of Internet Web hosts crawled by Alexa and Infoseek in the late 1990s [12] so the results reported here are likely to extend to large distributions of Web domains outside of GOV2. Given this disparity in size, practical IR systems might be able to benefit by somehow using host selection or host size as a way to improve search efficiency or IR performance. These prospects are explored below.

Table 1: Document counts of the largest, selected mid-sized, and smallest Web hosts in GOV2

Large		Medium		Small	
717,321	ghr.nlm.nih.gov	1,018	bluebook.state.or.us	1	adm.state.ky.us
709,105	nih-library.nih.gov	1,017	www.dma.state.mn.us	1	admin.monroecounty.gov
694,505	wcca.wicourts.gov	1,016	www.mclean.gov	1	ad02dwvchr.er.usgs.gov
665,987	cdaw.gsfc.nasa.gov	1,015	www.ofcm.gov	1	acousticaltest.grc.nasa.gov
650,208	catalog.tempe.gov	1,012	ag.state.nv.us	1	acmb.larc.nasa.gov
637,313	www.catalog.kpl.gov	1,010	clinicalstudies.info.nih.gov	1	aces.calpers.ca.gov
551,123	edc.usgs.gov	1,008	telerobotics.jpl.nasa.gov	1	acb700.tc.faa.gov
492,416	www.fs.usda.gov	1,005	femaweb2.fema.gov	1	abcprod.mstc.state.ms.us
459,329	gis.ca.gov	1,005	daac.ornl.gov	1	aasis.state.ar.us
441,201	www.csm.ornl.gov	1,003	www.obi.state.or.us	1	aaa.lanl.gov
403,648	www.fgdc.gov	1,003	www.fire.ca.gov	1	a841-dotweb01.nyc.gov
367,371	www.archives.gov	1,003	www.dir.texas.gov	1	a069-webapps3.nyc.gov
363,942	www-oss.fnal.gov	1,003	southwest.fws.gov	1	2002.cancer.gov
342,746	www.census.gov	1,002	www.ofa.noaa.gov	1	2001.nci.nih.gov
340,608	www.ssa.gov	1,002	historytogo.utah.gov	1	100.webstaging2.cit.nih.gov

4. DISTRIBUTION OF THE RELEVANCE JUDGMENTS AMONG WEB HOSTS

Each TREC Terabyte topic consists of an identifying number, a title, a description, and a narrative that encapsulates possible search keywords, an idea of the information desired, and a criteria for determining the relevance of a document to the topic. Participants could submit several runs of ranked result sets for the annual set of 50 topics. About 100 of the top documents from each of up to two runs were added to the judgment pool. The method of pooling to estimate the performance of an IR system on a large test collection is described in [13]. Assessors assign $(topic, document)$ pairs in the judgment pool to relevance scores of 0, 1, or 2 to not relevant, relevant, or highly relevant pairs. The information contained in the topic is used by the assessor to determine relevance [6]. For the purpose of discussion, a *qrel* or *relevance judgment* will be defined as the triplet $(topic, document, relevance\ score)$. The distribution of relevance judgments according to relevance score is summarized in Table 2.

From the perspective of distributed information retrieval, the set of relevance judgments for a topic can be partitioned by the document field into disjoint subsets corresponding to Web domain names. By assuming that the set of the top documents retrieved by the monolithic IR systems used in the TREC Terabyte track is about the same set of documents that would be returned by local IR search nodes at each Web host, the number of relevance judgments binned by Web domain name and topic can be used to estimate the number of domains that would respond to a given topic. Table 3 lists the number of hosts that contain the set of relevance judgments partitioned by relevance score. Many hosts that return relevant documents also return non-relevant documents so the total number of hosts responding to a query is about the same as the number of hosts retrieving non-relevant documents.

By the rank-size plot in Figure 4, hundreds of hosts in GOV2 contain non-relevant qrels for each topic and generally only tens of hosts contain relevant qrels. Using *a priori* statistical or experience-based knowledge like term-document frequencies or past performance of similar queries to identify the Web domain names most likely to retrieve documents relevant to a particular topic before transmitting

Figure 3: Zipf plot of the host-size frequency vs. host size for the hosts in GOV2

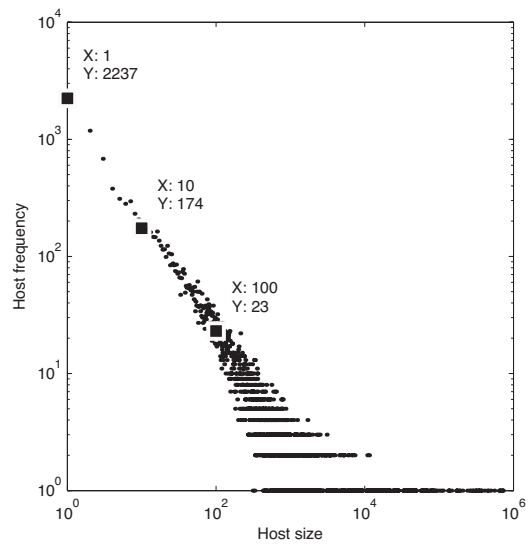


Table 2: qrel counts by relevance score

Relevance score	Mean	Median	Stdv.
Not relevant	108,434	728	335
Relevant	22,566	151	136
Very Relevant	4,351	29	49
Total	135,351	908	342

Table 3: The total number of Web hosts containing the entire set of qrels partitioned by relevance score

Relevance score	Total
Not relevant	6109
Relevant	2553
Very relevant	970
Any	6353

the search query to remote search nodes is one way to reduce the bandwidth used during the distributed search. However, this is not viable in practice because *a priori* knowledge of which hosts contain documents that are relevant to a topic is not easily available and most hosts do not contain relevant documents for any particular topic.

Over the union of all the TREC Terabyte Track topics, only 2704 hosts out of the 6353 hosts represented in the judgment pool retrieved at least one relevant document. As with the distribution of documents among hosts described in Section 3, the distribution of relevant or very relevant qrels among hosts in the judgment pool appears to follow a Zipfian distribution as illustrated in Figure 5. Quantitatively, 20% of the hosts *in the judgment pool that retrieved a relevant document for any topic* contain more than 78% of all the relevant and very relevant qrels. And more than 80% of the hosts that contain relevant (or very relevant) qrels for any TREC topic contain 10 or fewer relevant qrels for all topics. The qualitative similarity between the distribution of topic-aggregated relevant qrels and the distribution of documents among hosts is the premise of a model of relevance, described in detail below, postulating that the number of relevant documents retrieved from a host that contains relevant documents for a topic is approximately, over many topics and hosts, proportional to the size of the host. This model may then be used to improve the efficiency of a distributed search.

Before examining the relation, if any, between host size and the number of relevant documents retrieved on a topic-by-topic basis, first consider the number of relevance judgments in each Web domain after summing over all TREC topics as before. Taking the sum over the topics of the number of non-relevant, relevant, and very relevant qrels contained in each Web host yields, to a first order approximation, an upper bound on the number of relevant and non-relevant documents likely to be found in a domain that returns results for a typical TREC topic using a typical TREC IR system. Pearson correlation coefficients and p-values of the correlation between the summed relevance judgments from each domain name to the size of each domain are listed in Table 4. At least in the topic-aggregate, the number of relevance judgments in each Web host is linearly correlated with the size of the host and the correlation coefficient is relevance-dependent. This may not be surprising given that for any particular topic, a large host that retrieves any documents at all is by the number of pages available simply more likely to contain more relevant and non-relevant documents than a small host. However, this somewhat trivial observation may be used to conjecture that the number of relevant documents contained in a set of Web hosts for a particular topic may have an approximate but simple functional dependence on the size of the hosts.

5. RELEVANCE VARIABILITY AMONG HOSTS AND TOPICS

A simple proportional model of relevance is described below and will be used to estimate the number of relevant documents likely to be found in a host that retrieves documents for a topic. This estimate can be used as an upper bound for the number of results retrieved from any particular domain name and hence reduce the bandwidth required in a large-scale distributed Web search application. Later we

Table 4: Correlation coefficients and p-values of topic-summed qrel counts in each Web host and Web host size

Relevance score	ρ	p
Not relevant	0.432	10^{-276}
Relevant	0.275	10^{-45}
Very relevant	0.178	10^{-8}
Relevant or Very Relevant	0.267	10^{-45}

demonstrate that truncating the result sets from each Web host has an insignificant effect on the first-page precision from a simulation of a distributed IR application.

The degree to which an individual topic and Web domain name affects the variance of the number of relevance judgments contained in each domain was investigated with a 2-way analysis of variance (ANOVA) test on a domain-by-topic table of the number of relevance judgments for each of the three relevance scores. The 2-way ANOVA test compares the means of the number of relevance judgments grouped by topic and the means of the judgments grouped by Web domain name to the overall mean by measuring how much of the total variance is distributed among and within the topic means and domain means. There are no interaction effects to consider because only one measurement, the total number of relevance judgments of a given relevance score, is available for each $(topic, host)$ pair. The p-values from the results of the 2-way ANOVA test for each relevance score individually and in union are null to within machine precision. The ANOVA assumption of the normality of the differences from the mean is not satisfied by this data but the low p-values indicate that both the variations in Web hosts and the variations in topics both account for variation in the number of relevance judgments, as expected.

This test can be modified to account for a simple model dependence of the number of relevance judgments in a Web host, presupposed to contain relevant documents, on the size of the host. For instance, if the topic t is fixed, D is the set of documents in a given Web domain name, and $N_r(D,t)$ is the number of qrels in domain D with relevance score r, then the simplest model dependence is that the number of relevance judgments is proportional to the domain name size with a relevance-score dependent constant of proportionality.

$$N_r(D,t) = C_r(t)|D|$$

Instead of applying the ANOVA test to tables of the qrel counts $N_r(D,t)$, now apply the test to a table of the *measured* proportionality constants,

$$C_r(t) = \frac{N_r(D,t)}{|D|},$$

which will fluctuate with the choice of domain name and topic. A low p-value in the variability in domain means will indicate that the simple proportional model dependence of qrel counts on domain name size is not adequate to describe the data for any particular topic.

The p-values from the 2-way ANOVA test on the fractions of the hosts that have been judged for relevance, i.e. the values $N_r(D,t)/|D|$ for each topic and domain, are listed in Table 5. The p-value for the variation among domain means is still significant over the non-relevant qrels but is no longer significant among the relevant qrels. Therefore

Figure 4: Scatter and rank-size plot of the number of domain names containing not relevant, relevant, and very relevant qrels for each TREC topic

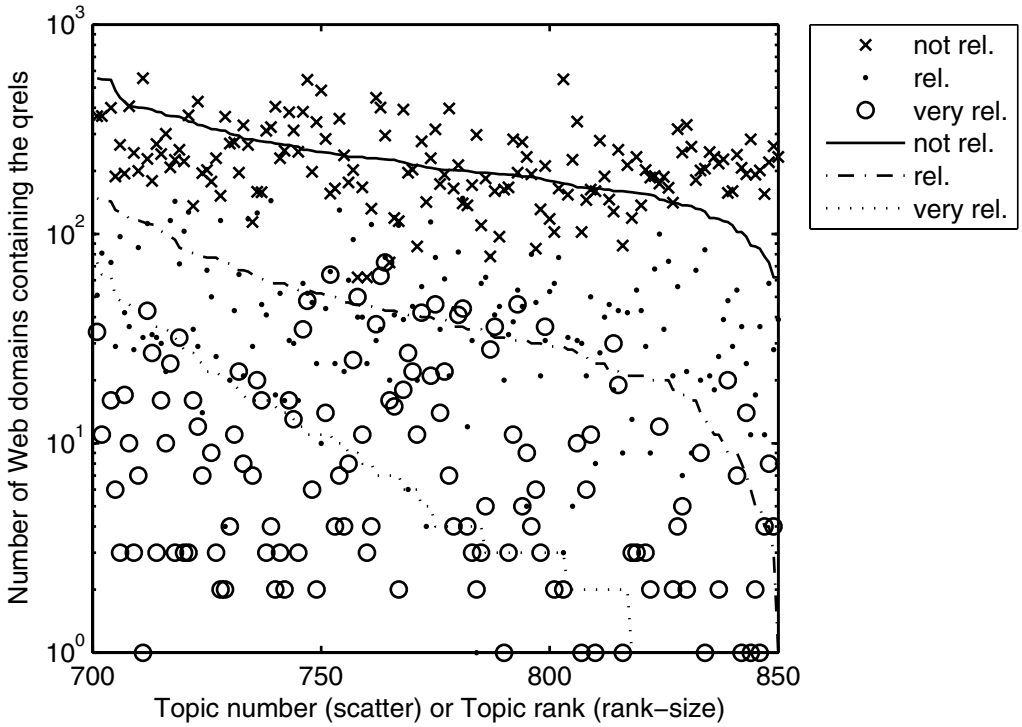


Table 5: 2-Way ANOVA p-values on the domain-size fractional domain relevances arranged by topic

Relevance score	Topic p-val.	Domain p-val.
Not relevant	0	2.7×10^{-9}
Relevant	0	0.73
Very relevant	0.059	0.48
Relevant or very relevant	0	0.74
All	0	1.7×10^{-6}

the number of relevant qrels per topic in a Web host is approximately proportional to the size of the host in the sense that the variation in the number of relevant qrels for a fixed topic among Web domains is significantly smaller than the variations in the number of relevant qrels for a given domain name among topics.

Assuming that the number of relevant qrels in any particular host is also the approximate number of relevant documents in that host, *then the simple proportional model above may be adequate to yield a rough upper bound of the number of documents relevant to a topic likely to be retrieved from a host that retrieves any documents at all* for that topic. If those relevant documents are assumed to be at the top of the ranked result set retrieved from the host, then measuring an average constant of proportionality $\langle C_r \rangle$ for relevant or highly relevant documents over the TREC Terabyte topics provides a natural search depth per domain beyond which the documents in the ranked result set are likely to be non-relevant. This result may apply to new topics not in the set of 149 topics covered in the TREC Terabyte Tracks. Other functional relationships on the number of relevance judgments on host size like quadratic and logarithmic were explored in a procedure similar to the one outlined above but no other relationships were found that effectively eliminated the fixed-topic variation among domain names.

6. ESTIMATING THE CONSTANT OF PROPORTIONALITY

Combined with an estimate of an appropriate host-search depth based on the constant $\langle C_r \rangle$ for relevant or very relevant documents over the set of TREC Terabyte topics, a ranked result set from each Web domain name that contains hits for some topic, and an algorithm to merge the result sets into a single ranked list, the effect of truncating the domain result sets according to the search depth can be measured using standard IR performance measures provided by the `trec_eval` program [2]. Throwing away results will generally not benefit recall-type measures so the measure used here is the TREC Precision-at-20 measure (P@20) [24] which roughly corresponds to the quality of the "first page" of results typically presented to a user. Using data from the best-performing topics in our TREC 2006 Terabyte Track experiment, we demonstrate that truncating the result sets from each Web domain name before merging the results into a single ranked list has an insignificant effect relative to the P@20 measure compared to merging the entire result sets.

First we estimated the constant $\langle C_r \rangle$ that relates the number of relevance judgments in a Web host to the size of the host by constructing topic-by-domain tables of the number

Figure 5: Zipf plot of the host frequency vs. the total number of contained relevant and very relevant qrels over all TREC topics

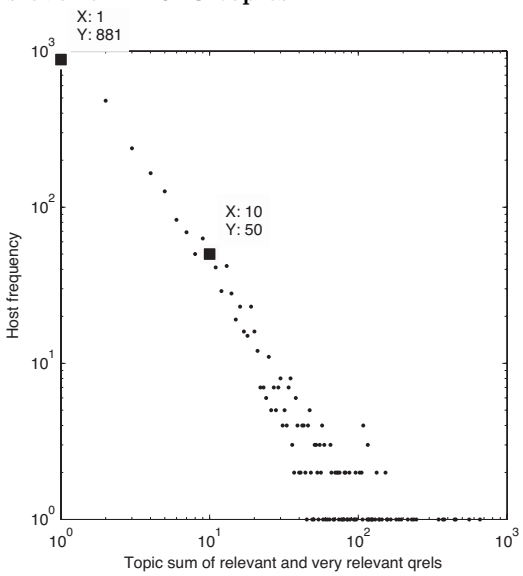


Table 6: Mean fraction of relevance judgments per domain name

Relevance score	Mean	Median	Stdv.
Not relevant	0.0071	0.0006	0.0550
Relevant	0.0060	0.0005	0.0354
Very relevant	0.0031	0.0004	0.0262
Relevant or very relevant	0.0061	0.0006	0.0360
All	0.0085	0.0006	0.0556

of relevant, very relevant, and non-relevant relevance judgments contained in each host, dividing each column corresponding to a host name by the size of that host, and then finally averaging over the nonzero elements in the table. Results from this process after averaging over topics are listed in Table 6. The large variance relative to the average values is due to the topic variability so the standard deviation itself may serve as a conservative upper bound for $\langle C_r \rangle$. To investigate the effect of truncating the result sets from domain name D after the top $\langle C_r \rangle |D|$ documents before merging, we chose to apply an aggressive value of $\langle C_r \rangle = 0.0005$ which will likely throw away some relevant documents from some domains. For example, if $\langle C_r \rangle = 0.0005$ then a maximum of five results will be retained in a result set from a domain with 10,000 documents.

The top performing topics from each of our sort merge and log merge experiments were used to investigate the effect of truncating the result sets before merging. Result sets from each host name D for each topic were truncated at the top $\langle C_r \rangle |D| = 0.0005|D|$ documents, rounding up to the next largest integer. The topics were sorted in descending order of result set P@20 performance from the log merge algorithm and the mean values are reported for the top ten topics in Table 7. Using the same topic ordering, the mean P@20 performance of the merged result sets derived from the truncated domain result sets was calculated and the p-value from a 1-way ANOVA test applied to the ten pairs of P@20 values. The process was repeated for the sort merge algorithm. The merge algorithms are described in detail in Section 2.

Table 7: Precision-at-20 performance and average length of merged result sets before and after truncating domain result sets

Method	Mean P@20	Mean num. retr.
Sort merge	0.3450	185
Sort merge, truncated	0.3500	115
ANOVA p-value	0.3434	0.0142
Log merge	0.3950	213
Log merge, truncated	0.3750	145
ANOVA p-value	0.2229	0.0093

The results indicate that any difference between the mean P@20 of the merged result sets derived from the truncated domain results and the entire domain results is not significant for either merge algorithm while the difference in the mean number of documents retrieved is significant. *Truncating the domain result sets reduces the total number of documents retrieved by more than 30% without affecting search performance.* While this result may not mean much in the context of a monolithic Web search, in a distributed Web search application the bandwidth required to retrieve result sets from a large number of search nodes is considerable and the procedure described here reduces the bandwidth required while still maintaining first-page precision.

7. DISCUSSION AND SUMMARY

This paper has presented three main findings of interest for the merging of IR results from multiple data sources, a necessary step in a typical federated collection search from a digital library. First, we demonstrated that most Internet hosts have no relevant documents for most topics. In fact, most of the hosts in the GOV2 collection had no relevant documents at all, across the 150 TREC topics using that collection. This finding is unsurprising, and seems consistent with common experiences with general-purpose Web search engines, where only a small number of possible domain names are represented in any given search result set.

The investigation of domain name characteristics, made possible through analysis of the approximately 17,000 Internet hosts in GOV2 along with 150 TREC topics, led us to consider whether domain characteristics could be a source of greater efficiency or higher IR performance. Our second finding was that the distribution of relevant documents over hosts was stable enough to allow a single constant to be applied to limit search results from any domain. Given the wide variety in domain sizes (from one document to over 700,000), non-overlapping sets of relevant and non-relevant documents from each of the 150 topics, and differing difficulty of the topics, it was surprising to find that the correlation between domain size and the number of relevant documents is statistically significant despite spanning many disparate topics. The practical implication of this finding is that a federated search system could specify an upper limit to the response set for a collection, based solely on the size of that collection. No knowledge of topics, search terms, or the relative sizes of other collections is required.

The third finding came from a numerical investigation of our chosen constant. For the TREC topics, truncating results yielded a 30% savings in the total number of "hits" returned from the collections individually, with no significant change in IR performance (P@20). This finding can give guidance to systems developers, as they seek to confirm results with different collections, and tune their constant for particular searchers' needs.

There are several limitations to the current work. As with most IR research based on TREC collections and relevance judgments, we cannot know the extent to which findings with the 150 topics for the GOV2 collection will apply to other collections or other topics. While stability was found across topics for the constant we looked at, there are likely to be other topic variations that could matter, such as topic length.

Another limitation is that the TREC 2006 results used were relatively poor. These results were employed as a baseline for the experiment with truncation on IR performance, but the results were not strong on IR performance compared to other systems at TREC. It might be that better-performing systems would have different experiences with result set truncation.

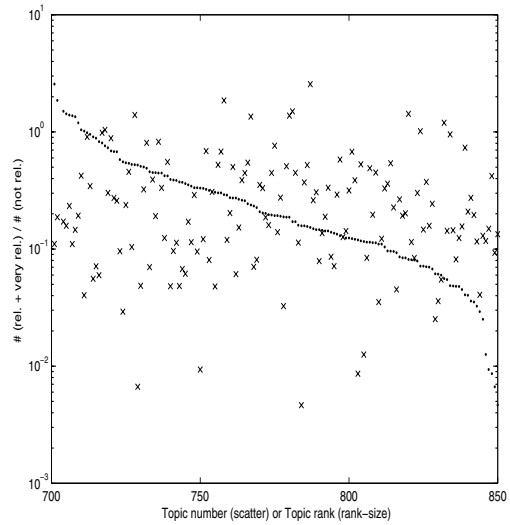
Despite these limitations, we believe the findings have promise. In a distributed search environment, the ability to limit the number of results sent by each collection is important for improved latency in merging all of the response sets. This paper also summarizes our research on merging multiple response sets, which is ongoing. Within the context of collections which, for whatever reason, are not available through monolithic search engines, we believe the GIR approach offers scalability and utility.

8. FUTURE WORK

The results in this paper are relatively weak in the sense that the results sets from our 2006 TREC Terabyte Track experiment used to investigate the effects of truncating the domain result sets were below the TREC median to start with so the effects of truncating results based on domain size on the best IR systems available is unknown. Furthermore, the GOV2 collection may not be an adequate representation of the Internet today so the results may not necessarily extend to a real-world environment. Additional testing of the domain result truncation technique is needed on state-of-the-art IR systems on additional large test collections. Alternate relevance score renormalization procedures for the purpose of result-set merging also need to be explored. For instance, the multiple nested ranker approach described by [16] could be used after collecting the top $\langle C_r \rangle |D|$ documents from each domain to rank the final merged list.

Variability of the relevance judgment distribution due to the topics themselves was not considered in this paper although some evidence (Figure 6 and Table 8) indicates that the ratio of the number of relevant qrels to the number of nonrelevant qrels is a convenient *a posteriori* estimate of topic difficulty. Even though the qrel counts used are aggregated over many IR systems, the ranked distribution shown is not likely to be sensitive to perturbations [11] in the IR systems used, the keywords chosen, or the GOV2 document corpus, so the rank-order of the topics can be used as a practical and robust method to determine the most and least difficult topics for current TREC IR systems. An *a priori* estimate of topic difficulty relative to the TREC topics

Figure 6: Scatter and rank-size plot of the ratio of the number of relevant or very relevant qrels to the number of non-relevant qrels for each topic



then may be derived using the performance of the previous TREC topics and the proportional model of relevance: the number and sizes of the domains retrieving documents for a topic yields an estimate for the total number of relevant documents and this can be compared to the number of relevant qrels for each TREC topic. One possible application to distributed information retrieval is that difficult topics could be sent to additional or more specialized search nodes that are not otherwise used for easy topics.

9. REFERENCES

[1] J. Barker. Meta-search engines. `http://www.lib.berkeley.edu/TeachingLib/Guides/Internet/MetaSearch.html`, 2006.

[2] C. Buckley. trec_eval IR evaluation package. Available at `http://trec.nist.gov/trec_eval/trec_eval_latest.tar.gz`, 2006.

[3] S. Büttcher, C. Clarke, and I. Soboroff. Overview of the TREC 2006 terabyte track. In *Proceedings of the Fifteenth Text REtrieval Conference*. National Institute of Standards and Technology, 2005.

[4] A. L. Calvé and J. Savoy. Database merging strategy based on logistic regression. *Information Processing & Management*, 36:341–359, 2000.

[5] C. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2004 terabyte track. In *Proceedings of the Thirteenth Text REtrieval Conference*. National Institute of Standards and Technology, 2004.

[6] C. Clarke, F. Scholer, and I. Soboroff. Overview of the TREC 2005 Terabyte Track. In *Proceedings of the Forteenth Text REtrieval Conference*. National Institute of Standards and Technology, 2005.

Table 8: TREC topic titles of the most and least difficult topics

Topic	Title
	Most difficult
784	mersenne primes
729	Whistle blower department of defense
803	May Day
750	John Edwards womens issues
805	Identity Theft Passport
829	Spanish Civil War support
724	Iran Contra
778	golden ratio
810	timeshare resales
830	model railroads
	Least difficult
787	Sunflower Cultivation
758	Embryonic stem cells
781	Squirrel control and protections
820	imported fire ants
728	flag display rules
780	Arable land
767	Pharmacist License requirements
832	labor union activity
718	deformed leopard frogs
824	Civil Air Patrol

[7] C. T. Fallen and G. B. Newby. Partitioning the GOV2 Corpus by Internet domain name: A result-set merging experiment. In *Proceedings of the Fifteenth Text REtrieval Conference*. National Institute of Standards and Technology, 2006.

[8] W. B. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures & Algorithms*. Prentice Hall, 1992.

[9] K. Gamiel, G. B. Newby, and N. Nassar. *Grid Information Retrieval Requirements (GFD.27)*. Global Grid Forum, Lamont, Illinois, 2003.

[10] F. Gey. Inferring probability of relevance using the method of logistic regression. In *Proceedings of the 17th International Conference of the ACM-SIGIR'94*, 1994.

[11] R. Günther, L. Levitin, B. Schapiro, and P. Wagner. Zipf's law and the effect of ranking on probability distributions. *International Journal of Theoretical Physics*, 35:395–417, 1996.

[12] B. A. Huberman and L. A. Adamic. Growth dynamics of the World-Wide Web. *Nature*, 401:131, 1999.

[13] K. S. Jones and C. van Rijsbergen. Report on the need for and provision of an "ideal" information retrieval test collection. Technical Report 5266, Computer Laboratory, University of Cambridge, 1975.

[14] B. Kahle and A. Medlar. An information system for corporate users: Wide area information servers. *Online*, 15(5):56–60, 1991.

[15] G. Kowalski. *Information Retrieval Systems: Theory and Implementation*. Kluwer Academic Publishers, 1997.

[16] I. Matveea, C. Burges, T. Burkard, A. Laucius, and L. Wong. High accuracy retrieval with multiple nested ranker. In *Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, 2006.

[17] N. Nassar. Amberfish at the TREC 2004 Terabyte Track. In *The Thirteenth Text REtrieval Conference Proceedings*, 2004.

[18] N. Nassar. Amberfish IR software from Etymon Systems Inc. Available at `http://www.etymon.com/tr.html`, 2005.

[19] G. B. Newby. Will XML and information retrieval make society transparent? In *XML 2000 Conference Proceedings*. Graphic Communications Association, 2000.

[20] R. Radhakrishnan. Information retrieval at boeing: Plans and successes. In *Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, 2006.

[21] S. Raghavan and H. Garcia-Molina. Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases (VLDB)*, pages 129–138, 2001.

[22] C. Sherman. Metacrawlers and metasearch engines. `http://searchenginewatch.com/showPage.html?page=2156241`, 2006.

[23] E. M. Voorhees. Siemens TREC-4 report: Further experiments with database merging. In *The Fourth Text REtreieval Conference*, 1995.

[24] E. M. Voorhees. Evaluation by highly relevant documents. In *Research and Development in Information Retrieval*, pages 74–82, 2001.

[25] E. M. Voorhees. Overview of the TREC 2004. In *Proceedings of the Thirteenth Text REtrieval Conference*. National Institute of Standards and Technology, 2004.

[26] Y. Zhou, J. Qin, G. Lai, E. Reid, and H. Chen. Exploring the dark side of the web: Collection and analysis of U.S. extremist online forums. In *Proceedings of the Intelligence and Security Informatics: IEEE International Conference on Intelligence and Security Informatics (ISI 2006)*, San Diego, CA, USA, May 2006.