Robust, Web and Terabyte Retrieval with Hummingbird SearchServer$^{\mathrm{TM}}$ at TREC 2004

Stephen Tomlinson
Hummingbird
Ottawa, Ontario, Canada
stephen.tomlinson@hummingbird.com
http://www.hummingbird.com/

February 6, 2005

Abstract

Hummingbird participated in 3 tracks of TREC 2004: the ad hoc task of the Robust Retrieval Track (find at least one relevant document in the first 10 rows from 1.9GB of news and government data), the mixed navigational and distillation task of the Web Track (find the home or named page or key resource pages in 1.2 million pages (18GB) from the .GOV domain), and the ad hoc task of the Terabyte Track (find all the relevant documents with high precision from 25.2 million pages (426GB) from the .GOV domain). In the robustness task, SearchServer found a relevant document in the first 10 rows for 46 of the 49 new short (Title-only) topics. In the web task, SearchServer returned a desired page in the first 10 rows for more than 75% of the 225 queries. In the terabyte task, SearchServer found a relevant document in the first 10 rows for 45 of the 49 short topics.

1 Introduction

Hummingbird SearchServer[1] is a toolkit for developing enterprise search and retrieval applications. The SearchServer kernel is also embedded in other Hummingbird products for the enterprise.

SearchServer works in Unicode internally [3] and supports most of the world's major character sets and languages. The major conferences in text retrieval experimentation (TREC [6], CLEF [1] and NTCIR [4]) have provided opportunities to objectively investigate SearchServer's support for more than a dozen languages.

This paper looks at experimental work with SearchServer (experimental 6.0 builds) for robust retrieval (robustness of ad hoc search across topics), mixed web navigation and distillation (find the one page the user wanted, i.e. a known-item search task, or find key resource pages for broad topics), and terabyte retrieval (ad hoc search on terabyte scales).

2 Robust Retrieval

For the TREC 2004 Robust Retrieval Track, there were 50 new topics and 200 old topics. The collection to be searched was the same as last year: a subset of the news and government data of TREC Disks 4 and 5 (FBIS, Federal Register 94, Financial Times, LA Times). It consisted of 528,155 documents totaling 1,997,002,586 bytes (1.9 GB). The average document size was 3781 bytes, though some documents were hundreds of kilobytes.

While the general objective was to find all the relevant documents for the topic and return them at the top of the list, participants were asked to focus not just on mean average precision but on at least one other

[1] SearchServer$^{\mathrm{TM}}$, SearchSQL$^{\mathrm{TM}}$and Intuitive Searching$^{\mathrm{TM}}$ are trademarks of Hummingbird Ltd. All other copyrights, trademarks and tradenames are the property of their respective owners.

measure indicative of "robustness" across topics, such as the Success@10 measure (percentage of topics for which at least one relevant was retrieved in the first 10 rows).

Each topic contained a "Title" (subject of the topic, e.g. "killer bee attacks"), "Description" (a one-sentence specification of the information need, e.g. "Identify instances of attacks on humans by Africanized (killer) bees.") and "Narrative" (more detailed guidelines for what a relevant document should or should not contain, e.g. "Relevant documents must cite a specific instance of a human attacked by killer bees. Documents that note migration patterns or report attacks on other animals are not relevant unless they also cite an attack on a human.").

It turned out one of the new topics had no relevant documents, leaving 249 topics in total. For these, there were on average 70 relevant documents per topic (low 3, high 448, median 41).

For the submitted runs due in August 2004, it was requested that at least one run just use the Title field as the basis of the query and at least one run just use the Description field.

More information on this task is expected to be in the track overview paper of the TREC proceedings.

2.1 Indexing

Our indexing approach was mostly the same as last year [12]. We used a SearchServer index which supported both exact matching (after some Unicode-based normalizations, such as decompositions and conversion to upper-case) and matching of inflections based on English lexical stemming (i.e. stemming based on a dictionary or lexicon for the language). For example, in English, "baby", "babied", "babies", "baby's" and "babying" all have "baby" as a stem. Some stop words were excluded from indexing (e.g. "the", "by" and "of" in English); we used a smaller stopfile than last year.

2.2 Searching

Unlike previous years, this year we experimented with SearchServer's CONTAINS predicate (instead of the IS_ABOUT predicate); for short boolean-OR queries, CONTAINS should produce the same ranking as IS_ABOUT. Our test application specified SearchSQL to perform a boolean-OR or boolean-AND of the query words. For example, for topic 430 whose Title was "killer bee attacks", a corresponding SearchSQL query with a boolean-OR of the Title words would be:

```
SELECT RELEVANCE('2:3') AS REL, DOCNO
FROM ROBUST04
WHERE FT_TEXT CONTAINS 'killer'|'bee'|'attacks'
ORDER BY REL DESC;
```

For a boolean-AND, the CONTAINS list would be changed from `'killer'|'bee'|'attacks'` to `'killer'&'bee'&'attacks'`, and only documents containing all of the specified words (or an inflection of each of them) would be retrieved.

Most aspects of SearchServer's '2:3' relevance value calculation are the same as described last year [12]. Briefly, SearchServer dampens the term frequency and adjusts for document length in a manner similar to Okapi [5] and dampens the inverse document frequency using an approximation of the logarithm. When doing morphological searching (i.e. when SET TERM_GENERATOR 'word!ftelp/inflect' was previously specified), these calculations are based on the stems of the terms (roughly speaking).

This year's experimental SearchServer version contains an enhancement for handling multiple stemming interpretations (e.g. in English, "axes" has both "axe" and "axis" as stems). For each document, only the interpretation that produces the highest score for the document is used in the relevance calculation (but all interpretations are still used for matching and search term highlighting). Sometimes this enhancement causes the original query form of the word to get more weight than some of its inflections (and it never gets less weight). This enhancement typically makes little difference for English (an exception is topic 379 (mainstreaming) discussed below). More details were in our CLEF paper this year [10].

2.3 Relevance Options Compared

For ranking, SearchServer has several options, including 5 different RELEVANCE_METHOD settings ('2:1', '2:2', '2:3', '2:4', '2:5') and a RELEVANCE_DLEN_IMP setting for controlling document length normalization (scale 0 to 1000). There is also a TERM_GENERATOR option to control whether or not inflections of query terms are matched.

Using the full set of 249 topics, we looked at whether the impacts of SearchServer's ranking and matching options for English ad hoc search were the same for boolean-AND queries as for boolean-OR queries. For these tests, just the Title field of the topic was used (typically 3 words).

For each approach (boolean-OR and boolean-AND), there was a "baseline" run using the options expected to score highest based on past experience (SET RELEVANCE_METHOD '2:3', SET RELEVANCE_DLEN_IMP 500, SET TERM_GENERATOR 'word!ftelp/inflect'). The other diagnostic runs differed from the baseline as follows:

- "2:1": The run used relevance method '2:1', "hits count", i.e. a simple count of all of the matches in a document (so repeated matches count multiple times). Note that document length normalization is ignored by this method.

- "2:2": The run used relevance method '2:2', "terms count", i.e. a count of the number of query terms matched (so if the query contains 3 words, the maximum score for a document is 3). Since inflections were enabled, a query term would count if any inflection of it was matched (but different inflections of the same word would not count for more than 1). Document length normalization is ignored by this method.

- "2:3": This is the baseline run using relevance method '2:3', "terms ordered". A formula which incorporates term frequency, inverse document frequency and document length normalization is applied as described earlier (Section 2.2).

- "2:3 with no stemming": Inflections were disabled for this run (i.e. SET TERM_GENERATOR '').

- "2:3 with no dlen": Document length normalization was disabled for this run (i.e. SET RELEVANCE_DLEN_IMP 0).

- "2:4": The run used relevance method '2:4', "critical terms ordered", which squares the importance of inverse document frequency compared to '2:3' (i.e. less common terms get even stronger weight).

- "2:5": The run used relevance method '2:5', "consistent terms ordered", which is an experimental new relevance method that is the same as '2:3' except that it does not include inverse document frequency (all the query terms are treated as being equally important).

Table 1 lists the mean scores of the diagnostic runs (see the glossary at the end of the paper for definitions of the measures). It appears that either '2:3' or '2:4' is a good choice on average. For boolean-AND, '2:5' (which treats each term as equally important) also appears to be a reasonable choice, though it scores a little lower (on average) for boolean-OR queries. The simple hits count strategy ('2:1'), which is relatively poor with boolean-OR (Success@10 just 39%), is reasonably successful with boolean-AND (Success@10 of 76%). Where they differ, the results for boolean-AND seem to agree more with reports from the field than the results for boolean-OR. Perhaps some of the differences between past TREC results and field preferences has been from users preferring boolean-AND queries.

Table 2 shows more details of how each OR run differs from the baseline OR run and how each AND run differs from the baseline AND run in the average precision measure (see Section 6.1 for an explanation of the table columns). Tables 3 and 4 do the same for the Success@1 and Success@10 measures respectively. The followng subsections look at these differences in more detail.

Table 1: Scores of Diagnostic Robust Retrieval Runs, Short (Title-only) Queries

Run	MAP	Success@1	Success@5	Success@10
OR: “2:3” (normal idf)	0.255	141/249 (57%)	214/249 (86%)	227/249 (91%)
OR: “2:4” (idf squared)	0.250	144/249 (58%)	210/249 (84%)	223/249 (90%)
OR: “2:3 with no stemming”	0.228	132/249 (53%)	213/249 (86%)	223/249 (90%)
OR: “2:5” (no idf)	0.220	137/249 (55%)	209/249 (84%)	224/249 (90%)
OR: “2:3 with no dlen”	0.215	120/249 (48%)	194/249 (78%)	220/249 (88%)
OR: “2:2” (terms count)	0.114	46/249 (18%)	122/249 (49%)	151/249 (61%)
OR: “2:1” (hits count)	0.048	23/249 (9%)	74/249 (30%)	97/249 (39%)
AND: “2:3” (normal idf)	0.179	138/249 (55%)	205/249 (82%)	215/249 (86%)
AND: “2:4” (idf squared)	0.177	142/249 (57%)	203/249 (82%)	214/249 (86%)
AND: “2:5” (no idf)	0.176	135/249 (54%)	201/249 (81%)	215/249 (86%)
AND: “2:3 with no dlen”	0.153	116/249 (47%)	186/249 (75%)	207/249 (83%)
AND: “2:3 with no stemming”	0.142	128/249 (51%)	196/249 (79%)	204/249 (82%)
AND: “2:1” (hits count)	0.132	90/249 (36%)	156/249 (63%)	188/249 (76%)
AND: “2:2” (terms count)	0.092	44/249 (18%)	120/249 (48%)	147/249 (59%)

Table 2: Relevance Methods Compared to ‘2:3’ Baseline, Average Precision Measure

Expt	MAP Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
OR: 2:4	−0.005	(−0.011, 0.001)	90-133-26	−0.18 (311), −0.15 (665), 0.15 (444)
OR: no stem	−0.027	(−0.039,−0.016)	95-137-17	−0.63 (328), −0.46 (326), 0.21 (648)
OR: 2:5	−0.034	(−0.046,−0.023)	87-157-5	−0.46 (366), −0.46 (302), 0.13 (621)
OR: no dlen	−0.039	(−0.050,−0.029)	62-184-3	−0.65 (679), −0.38 (601), 0.21 (338)
OR: 2:2	−0.140	(−0.158,−0.123)	23-226-0	−0.72 (679), −0.59 (677), 0.20 (615)
OR: 2:1	−0.206	(−0.231,−0.183)	15-233-1	−0.91 (679), −0.84 (410), 0.10 (325)
AND: 2:4	−0.002	(−0.004, 0.000)	67-96-86	−0.07 (339), −0.06 (334), 0.05 (379)
AND: 2:5	−0.002	(−0.007, 0.001)	79-110-60	−0.26 (622), −0.25 (379), 0.11 (339)
AND: no dlen	−0.026	(−0.036,−0.017)	62-149-38	−0.65 (679), −0.38 (601), 0.18 (338)
AND: no stem	−0.037	(−0.051,−0.025)	59-133-57	−0.75 (679), −0.74 (328), 0.14 (648)
AND: 2:1	−0.047	(−0.060,−0.035)	45-166-38	−0.81 (679), −0.53 (601), 0.21 (361)
AND: 2:2	−0.087	(−0.104,−0.071)	28-191-30	−0.72 (679), −0.58 (677), 0.22 (615)

Table 3: Relevance Methods Compared to ‘2:3’ Baseline, Success@1 Measure

Expt	S@1 Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
OR: 2:4	0.012	(−0.017, 0.041)	8-5-236	1.00 (332), 1.00 (670), −1.00 (684)
OR: 2:5	−0.016	(−0.057, 0.025)	11-15-223	−1.00 (303), −1.00 (435), 1.00 (438)
OR: no stem	−0.036	(−0.081, 0.009)	11-20-218	−1.00 (639), −1.00 (306), 1.00 (632)
OR: no dlen	−0.084	(−0.145,−0.024)	20-41-188	−1.00 (331), −1.00 (374), 1.00 (387)
OR: 2:2	−0.382	(−0.454,−0.309)	12-107-130	−1.00 (303), −1.00 (440), 1.00 (612)
OR: 2:1	−0.474	(−0.543,−0.405)	8-126-115	−1.00 (425), −1.00 (317), 1.00 (413)
AND: 2:4	0.016	(−0.009, 0.041)	7-3-239	1.00 (332), 1.00 (445), −1.00 (318)
AND: 2:5	−0.012	(−0.049, 0.025)	9-12-228	−1.00 (303), −1.00 (307), 1.00 (447)
AND: no stem	−0.040	(−0.085, 0.005)	11-21-217	−1.00 (636), −1.00 (306), 1.00 (632)
AND: no dlen	−0.088	(−0.149,−0.028)	19-41-189	−1.00 (374), −1.00 (385), 1.00 (384)
AND: 2:1	−0.193	(−0.262,−0.124)	17-65-167	−1.00 (318), −1.00 (357), 1.00 (413)
AND: 2:2	−0.378	(−0.446,−0.305)	11-105-133	−1.00 (303), −1.00 (440), 1.00 (429)

Table 4: Relevance Methods Compared to ‘2:3’ Baseline, Success@10 Measure

Expt	S@10 Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
OR: 2:5	−0.012	(−0.041, 0.013)	4-7-238	−1.00 (670), −1.00 (433), 1.00 (356)
OR: 2:4	−0.016	(−0.037, 0.001)	1-5-243	−1.00 (422), −1.00 (638), 1.00 (688)
OR: no stem	−0.016	(−0.045, 0.013)	5-9-235	−1.00 (328), −1.00 (605), 1.00 (344)
OR: no dlen	−0.028	(−0.065, 0.005)	6-13-230	−1.00 (700), −1.00 (699), 1.00 (356)
OR: 2:2	−0.305	(−0.366,−0.244)	2-78-169	−1.00 (434), −1.00 (699), 1.00 (690)
OR: 2:1	−0.522	(−0.587,−0.457)	3-133-113	−1.00 (700), −1.00 (699), 1.00 (371)
AND: 2:5	0.000	(−0.017, 0.017)	2-2-245	1.00 (688), 1.00 (631), −1.00 (401)
AND: 2:4	−0.004	(−0.021, 0.009)	1-2-246	−1.00 (638), −1.00 (401), 1.00 (688)
AND: no dlen	−0.032	(−0.065, 0.001)	4-12-233	−1.00 (700), −1.00 (699), 1.00 (688)
AND: no stem	−0.044	(−0.081,−0.012)	4-15-230	−1.00 (625), −1.00 (401), 1.00 (343)
AND: 2:1	−0.108	(−0.157,−0.060)	6-33-210	−1.00 (700), −1.00 (699), 1.00 (688)
AND: 2:2	−0.273	(−0.330,−0.216)	1-69-179	−1.00 (434), −1.00 (699), 1.00 (690)

2.3.1 Impact of Inverse Document Frequency

Topic 366 (commercial cyanide uses): Table 2 shows that ‘2:5’ scored 46 points lower than ‘2:3’ for this topic when boolean-OR was used. Relevant documents typically used the uncommon word “cyanide”, and ‘2:3’ gave such documents high scores from the word’s high inverse document frequency, while the common words “commercial” and “use” had low inverse document frequencies and hence little impact. ‘2:5’ had no preference for the “cyanide” term and retrieved a lot of documents with just lots of occurrences of “commercial” and “use”, so it scored a lot lower on this topic. Boolean-AND does poorly on this topic with either relevance method because few of the relevant documents actually contained an inflection of “commercial”. In practice, a user might realize this and change the query to just “cyanide” and get good results with either method.

Topic 622 (price fixing): ‘2:5’ scored 26 points lower than ‘2:3’ for this topic when using boolean-AND (as per Table 2). SearchServer’s lexical stemmer produced two stems for “fixing” (“fixing” and “fix”, presumably because the inflectional stem depends on whether “fixing” is a noun or a verb). The recent enhancement for alternative stems in effect weighted terms which share the “fixing” stem higher because that stem was less common (i.e. produced a higher inverse document frequency), and that was helpful for this topic. The ‘2:5’ method disabled inverse document frequency so this benefit was lost. In practice, a user could workaround the issue with ‘2:5’ by just disabling inflections for this query or by searching for “price fixing” as a phrase.

Topic 379 (mainstreaming): ‘2:5’ scored 25 points lower than ‘2:3’ for this topic when using boolean-AND (as per Table 2). The reason was similar to what happened in the “fixing” query. SearchServer’s lexical stemmer produced two stems for “mainstreaming” (“mainstreaming” and “mainstream”). ‘2:5’ lost the benefit of inverse document frequency in effect preferring the original query form. The ‘2:4’ method, which gives even more weight to inverse document frequency than ‘2:3’, scored 5 points higher than ‘2:3’ for this topic. In practice, a user could workaround the issue with ‘2:5’ by just disabling inflections for this query.

Topic 339 (Alzheimer’s Drug Treatment): For this topic ‘2:5’ scored 11 points higher than ‘2:3’, and ‘2:4’ scored 7 points lower than ‘2:3’, when using boolean-AND (as per Table 2). The methods which used inverse document frequency favored documents with lots of occurrences of “Alzheimer’s” and did not give as much weight to whether there were enough occurences of “drugs” or “treatment” for the document to be considered on topic. This topic is a case for which using inverse document frequency hurt the results of a boolean-AND query.

Overall, it appears inverse document frequency is useful for “noisy” queries because it often helps the important terms to stand out, but if the query is well-constructed (e.g. specifies only terms that really need to match, specifies only related inflections) then inverse document frequency might not be of much value and can sometimes be detrimental.

2.3.2 Impact of Document Length Normalization

For both boolean-AND and boolean-OR, enabling document length normalization was of statistically significant benefit for both mean average precision and Success@1 (as per the confidence intervals of the "no dlen" rows of Tables 2 and 3). Here we focus on the extreme differences for boolean-AND in Table 2:

Topic 679 (opening adoption records): This topic scored 65 points lower when not enabling document length normalization. The query words were fairly common (e.g. one can "adopt" new technologies, not just children) and the collection had a lot of long documents (100-700KB) which contained inflections of all 3 query words. The shorter documents (1-7KB) containing all 3 words were often relevant.

Topic 338 (Risk of Aspirin): For this topic, the score was 18 points higher without document length normalization. One of the relevant documents was fairly long (160KB), and there were just 4 relevants in total.

Overall, it appears that document length normalization is a helpful technique on average when the collection has a lot of long documents, and it probably won't make much difference if all the documents are short, so it seems safe to enable it by default (though note that looking up the document lengths may add modestly to search time in the current implementation).

2.3.3 Impact of Stemming

Enabling inflections from stemming produced statistically significant increases in mean average precision for both boolean-OR and boolean-AND (as per Table 2) and also for Success@10 when using boolean-AND (as per Table 4).

Topic 328 (Pope Beatifications): This topic benefited from stemming because most relevants just used the singular form "Beatification". One relevant actually just used a derivation ("beatified", different part of speech) instead of an inflection and so was still not matched in the boolean-AND case.

Topic 648 (family leave law): This topic was hurt by stemming because it was better not to match inflections of "leave" such as "left" or "leaving". Note that SearchServer supports controlling inflections on a per-term basis (not just on the per-query basis investigated in this experiment).

Overall, for English, while enabling inflections may be a better default on average, it's advisable for the application to let the user control inflections on a per-query-word basis.

2.3.4 Simple Hits Count

In Table 2, the topics on which the '2:1' method scored lower were mostly the same as those for the "no dlen" row. The simple count of all the hits appears to be particularly vulnerable to long documents.

In the boolean-OR case, '2:1' was particularly unsuitable for this task because a lot of occurrences of just one of the terms can dominate ('2:1' was the one method for which boolean-OR queries scored lower in mean average precision than boolean-AND). For boolean-AND, '2:1' was reasonably effective on average (though still less effective than the '2:3' method which dampens extra hits from the same term). '2:1' has the advantage of being easier for users to understand, so it's not surprising that some applications use it.

2.3.5 Simple Terms Count

The '2:2' method should never score lower with boolean-OR than boolean-AND. All results which satisfy the boolean-AND will have the same '2:2' score (which is the number of query words minus stop words). With '2:2', a boolean-OR will start with all the same results as boolean-AND and then follow with the results of fewer matching terms.

Topic 677 (Leaning Tower of Pisa): The '2:2' method gives the same score to any document mentioning the tower, regardless of how often the tower is mentioned. Documents with more references are more likely to be on topic, so '2:3' was more effective.

Topic 615 (timber exports Asia): The '2:2' method scored higher than '2:3' for this topic from what seems to be a chance result. In the boolean-AND case, '2:2' in effect returned the matches in the order they were cataloged. LA Times articles were inserted first, and those matches happened to have most of the relevants for this topic.

Overall, it's not surprising that we seldom hear of the '2:2' method being used in practice.

Table 5: Scores of Submitted Robust Retrieval Runs

Run	MAP	Success@1	Success@5	Success@10	τ_{249}
humR04t5e1	0.298	31/49 (63%)	42/49 (86%)	43/49 (88%)	0.33
humR04t5	0.286	32/49 (65%)	44/49 (90%)	45/49 (92%)	0.30
humR04t1	0.266	31/49 (63%)	44/49 (90%)	46/49 (94%)	0.35
humR04t1m	0.254	28/49 (57%)	44/49 (90%)	46/49 (94%)	0.29
humR04t1i	0.228	30/49 (61%)	43/49 (88%)	44/49 (90%)	0.23
humR04d4e5	0.320	31/49 (63%)	41/49 (84%)	41/49 (84%)	0.42
humR04d4	0.299	35/49 (71%)	43/49 (88%)	46/49 (94%)	0.42
humR04d5	0.281	31/49 (63%)	46/49 (94%)	47/49 (96%)	0.43
humR04d5m	0.261	27/49 (55%)	42/49 (86%)	44/49 (90%)	0.40
humR04d5i	0.163	23/49 (47%)	37/49 (76%)	44/49 (90%)	0.26

2.4 Submitted Runs

Table 5 lists the MAP, S1, S5 and S10 scores of the runs submitted in August 2004 over just the 49 new topics because some of the decisions on the parameters used were based on the older 200 topics. (The τ_{249} correlation measure is based on all 249 topics as explained in Section 2.4.1.)

humR04t5 was the same as the diagnostic '2:3' boolean-OR baseline run (including inflections and a document length normalization setting of 500). Possibly there were differences from an older version of SearchServer being used.

humR04t1 was the same as humR04t5 except that document length normalization was set to just 100 (which produced higher Success@n scores on the old 200 topics, but this did not carry over to the new topics).

humR04t1i was the same as humR04t1 except that the '2:5' relevance method was used (no inverse document frequency). As expected for boolean-OR, mean average precision was significantly lower. The Success@n scores were still respectable.

humR04t1m was the same as humR04t1 except that inflections from stemming were disabled.

humR04t5e1 used the top row of humR04t1 as an expansion query (because it had a high Success@1 score on the old 200 topics) and merged with the result of humR04t5 (highest MAP on the old 200 topics). Compared to humR04t5, the MAP score was a little higher and the Success@n scores were a little lower, but these results were not statistically significant.

humR04d5 was the same as humR04t5 except that the Description field was used instead of the Title field. Note that for descriptions we automatically removed "query stop words" such as "find", "relevant" and "document" before giving the query to SearchServer (based on looking at some older topic lists). The mean scores were not significantly different when using the description instead of the title.

humR04d5i was the same as humR04d5 except that the '2:5' relevance method was used (no inverse document frequency). Mean average precision dropped significantly. Inverse document frequency may be more important for longer boolean-OR queries because it tends to help the more important terms to stand out. (All of the submitted runs used boolean-OR.)

humR04d5m was the same as humR04d5 except that inflections from stemming were disabled. The drop in mean average precision was statistically significant.

humR04d4 was the same as humR04d5 except that the '2:4' relevance method was used (squared importance of inverse document frequency). The increase in mean average precision was statistically significant.

humR04d4e5 used the top row of humR04d5 as an expansion query (because it had a higher Success@1 score on the old 200 topics than humR04d4, but this did not carry over) and merged the result with humR04d4 (highest MAP on the old 200 topics). While mean average precision went up, the decline in the Success@10 score (compared to humR04d4) was statistically significant. Blind expansion seems likely to help just when help is least needed, i.e. when there already is some success at the top of the list.

2.4.1 Predicting Topic Difficulty

For the submitted runs, the participants were asked to append a ranking of the system's confidence of how well it did on the topic. For each run, we just used the relevance value (i.e. the number returned by the SearchServer RELEVANCE() function) of the top-retrieved row as our basis for ranking the topics. A higher relevance value was considered to mean a higher confidence in the relevance of the document.

The organizers computed Kendall's tau (τ), a measure of rank correlation, between the predicted rank and the actual rank of the topics (based on the run's average precision score for each topic); see the track overview paper for more details. This value is denoted as τ_{249} in Table 5 to emphasize that it is based on all 249 topics (but we did no tuning for this measure on the older 200 topics). Kendall's τ can range from -1 to $+1$, where $+1$ would mean perfect agreement between the rankings, 0 is the expected value if the prediction was random, and -1 would mean an exactly opposite ranking. The positive values for τ_{249} (0.23 to 0.43 in Table 5) suggest that SearchServer's RELEVANCE() values may have some relative meaning across topics. Hence normalizing them (by dividing each value by the first row's value) may lose information.

The lowest τ_{249} scores (0.23 and 0.26) were for the runs which used the '2:5' relevance method (inverse document frequency replaced by a fixed constant). SearchServer's relevance value, roughly speaking, is the average inverse document frequency of the query terms, weighted by dampened term frequency of the matches. Intuitively, it makes sense that a system can be more confident of relevance when a rare term was found in a document.

For the other runs, the τ_{249} scores were higher for those using the (one-sentence) Description as the query (0.40 to 0.43) than those using the (short) Title (0.29 to 0.35).

Note that we have not computed confidence intervals for the τ scores (or for the differences between them), so we should be particularly cautious of conclusions based on them.

3 Web Retrieval

For the "mixed query" task of the TREC 2004 Web Track, there were 225 queries: 75 topic distillation queries, 75 home-page finding queries and 75 named page finding queries. But the queries were not labelled by their type (until after the submitted runs were due in September 2004).

The collection to be searched was the same .GOV collection as the previous two years. It consisted of pages downloaded from the .gov domain of the World Wide Web in early 2002. Uncompressed, it was 19,455,030,550 bytes (18.1 GB) and a total of 1,247,753 documents. The average document size was 15,592 bytes.

For topic distillation queries (e.g. "science"), the objective was to find home pages of sites in .GOV relevant to the topic (e.g. "www.nsf.gov" (National Science Foundation), "www.house.gov/science/welcome.htm" (House Committee on Science), etc.). This year, it turned out that there were on average 21 right answers per query (low 1, high 147, median 13).

For home-page finding queries (e.g. "Internal Revenue Service"), the objective was to find the home page of the named site (e.g. "www.irs.gov"). This year, 69 of the 75 queries had just one right answer (and none had more than four, presumably duplicates).

For named page finding queries (e.g. "passport application form"), the objective was to find the named page (which would not be a home page, e.g. "travel.state.gov/dsp11.pdf"). This year, 71 of the 75 queries had just one right answer (and none had more than three, presumably duplicates).

More information on this task is expected to be in the track overview paper of the TREC proceedings.

3.1 Indexing

The indexing approach was the same as the previous two years (described in detail in [9]) except that a newer version of the software was used which may have contained an updated English lexicon for stemming.

Briefly: in addition to full-text indexing, the custom text reader cTREC populated particular columns such as TITLE (if any), URL, URL_TYPE and URL_DEPTH. The URL_TYPE was set to ROOT, SUBROOT, PATH or FILE, based on the convention which worked well in TREC 2001 for the Twente/TNO group [13] on the entry page finding task (also known as the home page finding task). The URL_DEPTH was set to a term indicating the depth of the page in the site. Table 6 contains URL types and depths for

Table 6: Examples of URL Type and Depth Values

URL	Type	Depth	Depth Term
http://nasa.gov/	ROOT	1	URLDEPTHA
http://www.nasa.gov/	ROOT	1	URLDEPTHA
http://jpl.nasa.gov/	ROOT	2	URLDEPTHAB
http://fred.jpl.nasa.gov/	ROOT	3	URLDEPTHABC
http://nasa.gov/jpl/	SUBROOT	2	URLDEPTHAB
http://nasa.gov/jpl/fred/	PATH	3	URLDEPTHABC
http://nasa.gov/index.html	ROOT	1	URLDEPTHA
http://nasa.gov/fred.html	FILE	2	URLDEPTHAB

Table 7: Number of Pages of each URL Type and Depth (.GOV collection)

Type	#Pages	Depth	#Pages	Depth	#Pages
ROOT	6,906	1	635	6	269,949
SUBROOT	18,179	2	16,792	7	136,513
PATH	55,332	3	128,898	8	44,960
FILE	1,167,336	4	282,086	9	15,289
		5	344,694	10+	7,937

example URLs, and Table 7 shows the number of .GOV pages of each URL type and depth. The exact rules we used are given in [9].

3.2 Searching

humW04l: The submitted humW04l run was a plain content search including linguistic expansion from English inflectional stemming. This run was the analog of the baseline (‘2:3’) run of the Robust track, including document length normalization (SET RELEVANCE_DLEN_IMP 500). The IS_ABOUT predicate was used instead of the CONTAINS predicate (and hence the VECTOR_GENERATOR was set to enable inflections instead of the TERM_GENERATOR), but the relevance calculation was the same. This run used the same approach as the submitted humNP03l run of last year [12] (except that the software was newer and had minor updates for inflections).

humW04pl: The submitted humW04pl run was the same as humW04l except that it put additional weight on matches in the title, url, first heading and some meta tags, including extra weight on matching the query as a phrase in these fields. Below is an example SearchSQL query. The searches on the ALL_PROPS column (which contained a copy of the title, url, etc. as described in [9]) are the difference from the humW04l run. Note that the FT_TEXT column indexed the content and also all of the non-content fields except for the URL. This run used the same approach as the submitted humNP03pl of last year, and last year’s paper [12] explains some of the syntax in more detail:

```
SELECT RELEVANCE('2:3') AS REL, DOCNO
FROM GOV
WHERE
 (ALL_PROPS CONTAINS 'visiting pandas national zoo' WEIGHT 1) OR
 (ALL_PROPS IS_ABOUT 'visiting pandas national zoo' WEIGHT 1) OR
 (FT_TEXT IS_ABOUT 'visiting pandas national zoo' WEIGHT 10)
ORDER BY REL DESC;
```

humW04dpl: The submitted humW04dpl run was the same as humW04pl except that it put additional weight on urls of depth 4 or less (but not on the url type, though url types were still listed with weight 0 as a way to prevent urls of depth greater than 4 from being excluded). Less deep urls also received higher weight from inverse document frequency because they are less common as per Table 7. Below is an example WHERE clause:

```
WHERE
((ALL_PROPS CONTAINS 'visiting pandas national zoo' WEIGHT 1) OR
 (ALL_PROPS IS_ABOUT 'visiting pandas national zoo' WEIGHT 1) OR
 (FT_TEXT IS_ABOUT 'visiting pandas national zoo' WEIGHT 10)
) AND (
 (URL_TYPE CONTAINS 'ROOT' WEIGHT 0) OR
 (URL_TYPE CONTAINS 'SUBROOT' WEIGHT 0) OR
 (URL_TYPE CONTAINS 'PATH' WEIGHT 0) OR
 (URL_TYPE CONTAINS 'FILE' WEIGHT 0) OR
 (URL_DEPTH CONTAINS 'URLDEPTHA' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHAB' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHABC' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHABCD' WEIGHT 5) )
```

humW04rdpl: The submitted humW04rdpl run was the same as humW04dpl except that it put additional weight on the url type. This run used the same approach as the successful humTD03upl run of last year (u = r+d). Below is an example WHERE clause:

```
WHERE
((ALL_PROPS CONTAINS 'visiting pandas national zoo' WEIGHT 1) OR
 (ALL_PROPS IS_ABOUT 'visiting pandas national zoo' WEIGHT 1) OR
 (FT_TEXT IS_ABOUT 'visiting pandas national zoo' WEIGHT 10)
) AND (
 (URL_TYPE CONTAINS 'ROOT' WEIGHT 10) OR
 (URL_TYPE CONTAINS 'SUBROOT' WEIGHT 10) OR
 (URL_TYPE CONTAINS 'PATH' WEIGHT 10) OR
 (URL_TYPE CONTAINS 'FILE' WEIGHT 0) OR
 (URL_DEPTH CONTAINS 'URLDEPTHA' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHAB' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHABC' WEIGHT 5) OR
 (URL_DEPTH CONTAINS 'URLDEPTHABCD' WEIGHT 5) )
```

humW04dp: The submitted humW04dp run was the same as humW04dpl except that linguistic expansion (inflections) from English stemming was disabled by "SET VECTOR_GENERATOR ‘’ ".

SearchServer's relevance value calculation was the same as described for the Robust track. When multiple predicates are combined, as was done for some of the web approaches, SearchServer currently does not normalize by query length. For example, the URL_TYPE clauses would have a lot less relative impact if the topic query contained 5 words instead of 1.

3.3 Results

Table 8 lists the scores of the 5 submitted runs (see the glossary section for definitions of the measures). The first group is the scores on the 75 topic distillation (TD) queries. The next two groups are for the 75 home-page finding (HP) and 75 named page finding (NP) queries respectively. The final group is the scores over all 225 queries.

The ‘p’ technique (extra weight for phrases in the Title and other properties plus extra weight for vector search on properties) increased the mean scores for all 3 query types (compare ‘pl’ to ‘l’). Tables 9, 10, 11 and 12 show that the increases from the ‘p’ technique were statistically significant for the mean reciprocal rank, mean average precision, Success@1 and Success@10 measures respectively. (See Section 6.1 for an explanation of the columns of Tables 9-12.)

Table 8: Scores of Submitted Web Track Runs

Run	MRR	Success@1	Success@5	Success@10	MAP
TD: humW04rdpl	0.553	28/75 (37%)	59/75 (79%)	68/75 (91%)	0.163
TD: humW04dpl	0.394	16/75 (21%)	45/75 (60%)	54/75 (72%)	0.109
TD: humW04dp	0.351	14/75 (19%)	43/75 (57%)	52/75 (69%)	0.098
TD: humW04pl	0.260	6/75 (8%)	35/75 (47%)	44/75 (59%)	0.079
TD: humW04l	0.131	2/75 (3%)	22/75 (29%)	27/75 (36%)	0.046
HP: humW04rdpl	0.479	28/75 (37%)	44/75 (59%)	52/75 (69%)	0.482
HP: humW04dpl	0.437	27/75 (36%)	38/75 (51%)	46/75 (61%)	0.438
HP: humW04dp	0.422	25/75 (33%)	38/75 (51%)	46/75 (61%)	0.421
HP: humW04pl	0.316	17/75 (23%)	29/75 (39%)	36/75 (48%)	0.318
HP: humW04l	0.187	10/75 (13%)	18/75 (24%)	22/75 (29%)	0.187
NP: humW04rdpl	0.484	27/75 (36%)	48/75 (64%)	56/75 (75%)	0.485
NP: humW04dpl	0.559	36/75 (48%)	47/75 (63%)	57/75 (76%)	0.559
NP: humW04dp	0.554	35/75 (47%)	49/75 (65%)	61/75 (81%)	0.555
NP: humW04pl	0.569	36/75 (48%)	50/75 (67%)	57/75 (76%)	0.570
NP: humW04l	0.466	28/75 (37%)	42/75 (56%)	49/75 (65%)	0.466
ALL: humW04rdpl	0.505	83/225 (37%)	151/225 (67%)	176/225 (78%)	0.376
ALL: humW04dpl	0.463	79/225 (35%)	130/225 (58%)	157/225 (70%)	0.368
ALL: humW04dp	0.443	74/225 (33%)	130/225 (58%)	159/225 (71%)	0.358
ALL: humW04pl	0.382	59/225 (26%)	114/225 (51%)	137/225 (61%)	0.322
ALL: humW04l	0.261	40/225 (18%)	82/225 (36%)	98/225 (44%)	0.233

Table 9: Impact of Submitted Web Techniques on Reciprocal Rank

Expt	MRR Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
r-TD	0.159	(0.066, 0.253)	38-20-17	0.99 (111), 0.96 (50), −0.80 (191)
d-TD	0.134	(0.068, 0.205)	45-15-15	0.95 (63), 0.93 (41), −0.50 (101)
p-HP	0.130	(0.067, 0.198)	57-7-11	0.99 (118), 0.98 (67), −0.50 (217)
p-TD	0.129	(0.080, 0.184)	57-9-9	0.99 (177), 0.98 (12), −0.45 (79)
d-HP	0.121	(0.058, 0.189)	41-12-22	0.96 (224), 0.93 (9), −0.67 (51)
p-NP	0.103	(0.044, 0.169)	32-8-35	0.97 (208), 0.93 (39), −0.50 (28)
l-TD	0.043	(−0.013, 0.100)	17-23-35	1.00 (63), 0.89 (170), −0.99 (207)
r-HP	0.042	(−0.030, 0.114)	31-19-25	0.91 (54), 0.90 (161), −0.88 (78)
l-HP	0.015	(−0.010, 0.047)	14-23-38	0.75 (213), 0.67 (78), −0.17 (51)
l-NP	0.004	(−0.027, 0.035)	10-19-46	−0.67 (114), 0.50 (187), 0.50 (34)
d-NP	−0.011	(−0.041, 0.019)	12-16-47	−0.50 (150), −0.50 (208), 0.50 (28)
r-NP	−0.075	(−0.131,−0.022)	7-33-35	−0.83 (55), −0.67 (38), 0.67 (114)

The ‘d’ technique (modest extra weight for less deep urls) increased the mean scores for the 2 types which wanted home pages (TD and HP) and had little impact on the mean scores for the non-homepage type (NP) (compare ‘dpl’ to ‘pl’). The increases on the TD and HP types were statistically significant (as per Tables 9-12).

The ‘l’ technique (linguistic expansion from English inflectional stemming) modestly increased most of the mean scores (compare ‘dpl’ to ‘dp’). However, the only statistically significant impact was on mean average precision for distillation queries (as per Table 10).

The ‘r’ technique (strong extra weight for urls of root, subroot or path types) substantially increased the mean scores for topic distillation and also increased the mean scores for home-page finding, but decreased

Table 10: Impact of Submitted Web Techniques on Average Precision

Expt	MAP Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
r-TD	0.054	(0.034, 0.074)	57-17-1	0.29 (8), 0.28 (99), −0.16 (5)
p-TD	0.033	(0.022, 0.045)	66-8-1	0.21 (74), 0.18 (81), −0.05 (209)
d-TD	0.030	(0.020, 0.041)	59-15-1	0.18 (134), 0.16 (21), −0.03 (47)
l-TD	0.011	(0.000, 0.024)	32-31-12	0.35 (170), 0.11 (160), −0.14 (207)

Table 11: Impact of Submitted Web Techniques on Success@1

Expt	S@1 Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
r-TD	0.160	(0.026, 0.294)	19-7-49	1.00 (50), 1.00 (111), −1.00 (10)
d-HP	0.133	(0.053, 0.227)	11-1-63	1.00 (6), 1.00 (218), −1.00 (51)
d-TD	0.133	(0.039, 0.227)	12-2-61	1.00 (191), 1.00 (99), −1.00 (101)
p-NP	0.107	(0.026, 0.187)	9-1-65	1.00 (39), 1.00 (208), −1.00 (28)
p-HP	0.093	(0.013, 0.187)	9-2-64	1.00 (78), 1.00 (118), −1.00 (217)
p-TD	0.053	(0.013, 0.107)	4-0-71	1.00 (177), 1.00 (81), 0.00 (221)
l-HP	0.027	(−0.001, 0.067)	2-0-73	1.00 (78), 1.00 (213), 0.00 (224)
l-TD	0.027	(−0.041, 0.094)	4-2-69	1.00 (141), 1.00 (63), −1.00 (97)
l-NP	0.013	(−0.027, 0.067)	2-1-72	1.00 (34), 1.00 (187), −1.00 (114)
r-HP	0.013	(−0.094, 0.121)	8-7-60	1.00 (59), 1.00 (54), −1.00 (49)
d-NP	0.000	(−0.054, 0.054)	2-2-71	1.00 (187), 1.00 (28), −1.00 (150)
r-NP	−0.120	(−0.201,−0.039)	1-10-64	−1.00 (132), −1.00 (222), 1.00 (114)

most of the mean scores for named page finding (compare 'rdpl' to 'dpl'). The increases for topic distillation were statistically significant, but the increases for home-page finding were not. Some of the decreases for named page finding were statistically significant. (See Tables 9-12.)

So the 'p', 'd' and 'l' techniques appear to be generally useful for web search, while the 'r' technique may be useful if queries for home pages are expected to be relatively frequent.

Overall, the web search task was apparently more challenging than the ad hoc task of the Robust track. The content search technique that produced a Success@10 of more than 90% in the Robust track had a

Table 12: Impact of Submitted Web Techniques on Success@10

Expt	S@10 Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
p-TD	0.227	(0.133, 0.334)	18-1-56	1.00 (99), 1.00 (189), −1.00 (79)
p-HP	0.187	(0.106, 0.281)	14-0-61	1.00 (78), 1.00 (118), 0.00 (224)
r-TD	0.187	(0.106, 0.281)	14-0-61	1.00 (107), 1.00 (111), 0.00 (221)
d-HP	0.133	(0.066, 0.214)	10-0-65	1.00 (117), 1.00 (49), 0.00 (118)
d-TD	0.133	(0.026, 0.241)	14-4-57	1.00 (79), 1.00 (160), −1.00 (57)
p-NP	0.107	(0.013, 0.201)	11-3-61	1.00 (44), 1.00 (120), −1.00 (88)
r-HP	0.080	(−0.014, 0.174)	10-4-61	1.00 (149), 1.00 (162), −1.00 (32)
l-TD	0.027	(−0.067, 0.121)	7-5-63	1.00 (63), 1.00 (58), −1.00 (157)
d-NP	0.000	(−0.067, 0.067)	3-3-69	1.00 (151), 1.00 (43), −1.00 (116)
l-HP	0.000	n/a	0-0-75	0.00 (117), 0.00 (7), 0.00 (224)
r-NP	−0.013	(−0.081, 0.054)	3-4-68	−1.00 (62), −1.00 (70), 1.00 (26)
l-NP	−0.053	(−0.121, 0.001)	1-5-69	−1.00 (26), −1.00 (27), 1.00 (151)

Success@10 of less than 50% in the web task. However, by using SearchServer's ability to take advantage of various kinds of structure in the data, Success@10 for the mixed web task was increased to more than 75% and for the broad topic (distillation) queries to more than 90%.

4 Terabyte Retrieval

The Terabyte Track was new to TREC this year. The collection to be searched was the GOV2 collection, a crawl of most of the .gov domain in early 2004. Once binaries (such as images) were removed, its size was less than half a terabyte. The GOV2 distribution was 457,165,206,582 bytes uncompressed (426 GB) and consisted of 25,205,179 documents. More than 90% of the documents were html, 8% were (extracted text from) pdf, and the rest were extracted text from other formats (plain text, msword, postscript, etc.). The collection was 86,594,814,080 bytes gzip-compressed and distributed on a hard drive. The hard drive contained 273 directories, each typically containing 100 .gz files of 3GB in size. Uncompressed, the average document size was 18,137 bytes.

Even though the data was like that of the Web Track, the task was more like that of the Robust Retrieval Track. There were 50 new topics, each with a Title, Description and Narrative, and the objective was to find all the relevant documents for the topic and return them at the top of the list. It turned out one of the topics had no relevant documents, leaving 49 topics in total. For these, there were on average 217 relevant documents per topic (low 7, high 617, median 167).

4.1 Indexing

The indexing approach was the same as described for the Web Track (i.e. not just the content was indexed but separate columns were created for the title, url, some meta fields, etc.).

For the diagnostic runs in January 2005, the entire collection was indexed in one SearchServer table. But for the submitted runs in September 2004, lack of resources led to the collection being indexed in 137 parts, and for the record this approach is described here. The 600MHz lab machine available at the time did not have enough disk space remaining for this task. There was a network server with enough disk space to hold the indexes but its processor was not available full-time for this task (and our experimental version of SearchServer was not installed on it). There was also a 2.4GHz desktop machine which was sometimes available. For the submitted runs, most of the indexing was done in pieces on the 2.4GHz machine, 2 directories at a time into separate tables. The tables (indexes) were moved to the network server when it was available. The result was 137 small tables, each indexing about 3GB of the collection. This approach was convenient for getting the data indexed but made searching less efficient. Most of the index was positioning information for supporting proximity searches, a feature that was not actually used for our submitted (or diagnostic) Terabyte runs. The on-disk structures also included other table files such as the catalog which stored the titles, urls, some meta fields, etc.

4.2 Submitted Runs

While the diagnostic runs in January 2005 simply searched one table, the submitted runs in September 2004 searched 137 small tables, and for the record that approach is described here. The search approaches were intended to be similar to those done for the Web Track. The FROM clause of the searches specified a union of the 137 small tables (GOV000 UNION GOV001 UNION ... GOV136). The experimental version of SearchServer did not calculate global inverse document frequencies for the terms. For each document, it would use the inverse document frequencies based on just the table containing the document. To make the relevance calculation more consistent with a one-table ranking, for most runs the test application looked up the number of occurrences of the terms and their inflections in each table (using the SEARCH_TERMS system table) so that a global inverse document frequency could be computed using the most common inflection. This value was then assigned to the term in the SearchSQL query using the WEIGHT clause. The '2:5' relevance method was used (instead of '2:3') so that the per-table inverse document frequencies were ignored (just the specified WEIGHT was used). This approach would not take advantage of the enhancement for multiple stemming interpretations described in the Robust section. It turned out that the

Table 13: Scores of Submitted Terabyte Track Runs

Run	MRR	Success@1	Success@5	Success@10	MAP
humT04l	0.673	27/49 (55%)	40/49 (82%)	44/49 (90%)	0.224
humT04vl	0.664	26/49 (53%)	43/49 (88%)	45/49 (92%)	0.221
humT04	0.637	24/49 (49%)	42/49 (86%)	42/49 (86%)	0.196
humT04dvl	0.606	22/49 (45%)	41/49 (84%)	45/49 (92%)	0.212
humT04l3	0.544	18/49 (37%)	39/49 (80%)	40/49 (82%)	0.155

Table 14: Impact of Submitted Terabyte Techniques on Reciprocal Rank

Expt	MRR Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
l	0.036	(−0.061, 0.131)	13-9-27	0.97 (716), −0.86 (706), −0.92 (714)
v	−0.009	(−0.081, 0.060)	9-9-31	−0.80 (727), −0.67 (735), 0.50 (739)
d	−0.058	(−0.112,−0.013)	5-14-30	−0.75 (747), −0.50 (722), 0.25 (720)
3	−0.129	(−0.243,−0.013)	12-22-15	−0.96 (740), −0.86 (735), 0.86 (706)

experimental lookup phase was not well-optimized and added a lot to the search times, particularly when the QUERY_TERM() function was used to look up the inflections. The runs retrieving the top-20 rows were done by the 2.4GHz machine reading the indexes via a Windows network drive. The searches also read a lot of positioning information which actually was not used to produce the results. Some of the submitted runs, which retrieved 10,000 rows per query, were done with the 600MHz machine.

The submitted runs all just used the Title field of the topic.

humT04l: This run was a plain content run (i.e. just searched FT_TEXT) and included inflections from English stemming. It was the analog of the humW04l run described in the Web section. The document length importance was set to 750 instead of 500 (just from an oversight).

humT04: This run was the same as humT04l except that inflections were not included. The experimental lookup phase described earlier had a lot less to do.

humT04vl: This run was the same as humT04l except that it put additional weight on matches in the title, url and some meta tags (i.e. the ALL_PROPS column). The weight for ALL_PROPS was one-tenth the weight used for FT_TEXT. A difference from the analogous web run (humW04pl) was that the phrase search on ALL_PROPS was omitted (in part because it would have been extra effort to get the test application to weight it in an analogous way, and in part because we verified last year ('v' and 'q' experiments in [12]) that the phrase search was of less importance for non-homepage searches anyway). Also, the humT04vl weighting for ALL_PROPS was in effect based on inverse document frequencies calculated from FT_TEXT because of the '2:5' workaround, unlike for humW04pl which used the '2:3' relevance method.

humT04dvl: This run was the same as humT04vl except that it put additional weight on less deep urls. The weights were analogous to those used in Web run humW04dpl but, as '2:5' was used, the SearchSQL needed to explicitly multiply in the weight from inverse document frequency.

humT04l3: This run was the same as humT04l except that it did not do the lookup phase to calculate

Table 15: Impact of Submitted Terabyte Techniques on Average Precision

Expt	MAP Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
l	0.027	(0.013, 0.042)	34-14-1	0.16 (733), 0.13 (738), −0.06 (714)
v	−0.003	(−0.008, 0.003)	21-28-0	0.08 (748), −0.04 (749), −0.05 (726)
d	−0.009	(−0.015,−0.004)	13-35-1	−0.10 (749), −0.04 (748), 0.02 (741)
3	−0.069	(−0.089,−0.049)	7-42-0	−0.22 (704), −0.19 (726), 0.03 (746)

Table 16: Scores of Diagnostic Terabyte Track Runs

Run	MRR	Success@1	Success@5	Success@10	MAP
OR: “2:4” (idf squared)	0.753	33/49 (67%)	43/49 (88%)	44/49 (90%)	0.243
OR: “2:3” (normal idf)	0.730	30/49 (61%)	43/49 (88%)	44/49 (90%)	0.259
OR: “2:5” (no idf)	0.677	27/49 (55%)	42/49 (86%)	45/49 (92%)	0.225
OR: “2:3 with no stemming”	0.675	26/49 (53%)	42/49 (86%)	43/49 (88%)	0.231
OR: “2:3 with no dlen”	0.590	21/49 (43%)	37/49 (76%)	44/49 (90%)	0.166
OR: “2:2” (terms count)	0.165	3/49 (6%)	13/49 (27%)	20/49 (41%)	0.044
OR: “2:1” (hits count)	0.083	1/49 (2%)	7/49 (14%)	9/49 (18%)	0.010
AND: “2:4” (idf squared)	0.790	35/49 (71%)	45/49 (92%)	46/49 (94%)	0.233
AND: “2:3” (normal idf)	0.748	31/49 (63%)	44/49 (90%)	46/49 (94%)	0.239
AND: “2:3 with no stemming”	0.710	28/49 (57%)	43/49 (88%)	44/49 (90%)	0.201
AND: “2:5” (no idf)	0.691	28/49 (57%)	42/49 (86%)	45/49 (92%)	0.227
AND: “2:3 with no dlen”	0.594	21/49 (43%)	37/49 (76%)	44/49 (90%)	0.156
AND: “2:1” (hits count)	0.396	13/49 (27%)	29/49 (59%)	35/49 (71%)	0.100
AND: “2:2” (terms count)	0.165	3/49 (6%)	13/49 (27%)	20/49 (41%)	0.044

global inverse document frequencies. It used the ‘2:3’ relevance method instead of ‘2:5’.

Table 13 lists the scores of the 5 submitted runs (see the glossary section for definitions of the measures). Note that the mean average precision (MAP) scores are based on just the first 1000 documents retrieved (as per the usual convention for MAP), even though the submissions included up to 10,000 rows per topic.

The ‘l’ technique (linguistic expansion from English inflectional stemming) modestly increased most of the mean scores (compare ‘humT04l’ to ‘humT04’). The increase in mean average precision was statistically significant (as per Table 15), like it was for the Robust diagnostic task and the Web Distillation subtask.

The ‘v’ technique (extra weight for matches in the title, url and some meta tags) made little difference to the mean scores (compare ‘humT04vl’ to ‘humT04l’). Note that the individual reciprocal rank scores could be impacted substantially in either direction (ranging from a decrease of 0.80 on topic 727 to an increase of 0.50 on topic 739 as per Table 14). The ‘v’ technique did not have the consistent beneficial impact of the analogous ‘p’ technique for the Web task, evidence that the Web and Terabyte tasks are quite different.

The ‘d’ technique (modest extra weight for less deep urls) decreased most of the mean scores (compare ‘humT04dvl’ to ‘humT04vl’). The decreases for mean reciprocal rank and mean average precision were statistically significant (as per Tables 14 and 15). This impact is the opposite of the ‘d’ technique for the Web task.

The ‘3’ technique (use local inverse document frequencies) decreased all of the mean scores (compare ‘humT04l3’ to ‘humT04l’). The decreases for mean reciprocal rank and mean average precision were statistically significant (as per Tables 14 and 15). This result suggests that for multi-table searches, the ‘2:5’ relevance method (which ignores inverse document frequency) should be considered.

Overall, the results of the submitted experiments suggest that the Terabyte task is much more like the Robust task than the Web task, as far as relevance-ranking is concerned. The Web task was searching for particular named pages or for home pages, and favoring the document title or less deep urls was helpful. The Terabyte task, though it is also searching web data, is looking for any document relevant to the topic (as in the Robust task), and the web techniques did not (on average) improve on a plain content search. The diagnostic runs focus on plain content search techniques.

4.3 Diagnostic Runs

For the diagnostic runs in January 2005, the GOV2 collection was indexed in one SearchServer table. Table 16 shows the same diagnostic runs as were done for the Robust task (Table 1) except that the document length importance setting (RELEVANCE_DLEN_IMP) was set to 250 instead of 500 (250 produced a little higher scores on this collection, perhaps because there were more long relevant documents). Also, the experimental

Table 17: Relevance Methods Compared to Terabyte ‘2:3’ Baseline, Reciprocal Rank Measure

Expt	MRR Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
OR: 2:4	0.023	(−0.031, 0.080)	9-7-33	0.50 (715), 0.50 (728), −0.50 (716)
OR: 2:5	−0.052	(−0.118, 0.010)	4-13-32	−0.75 (748), −0.67 (716), 0.67 (746)
OR: no stem	−0.055	(−0.131, 0.022)	5-14-30	−0.80 (716), −0.50 (733), 0.80 (714)
OR: no dlen	−0.139	(−0.239,−0.043)	10-18-21	−0.90 (704), −0.90 (701), 0.67 (746)
OR: 2:2	−0.564	(−0.674,−0.451)	2-43-4	−1.00 (723), −1.00 (749), 0.47 (741)
OR: 2:1	−0.646	(−0.745,−0.543)	0-48-1	−1.00 (708), −1.00 (701), 0.00 (710)
AND: 2:4	0.042	(−0.004, 0.093)	9-5-35	0.50 (715), 0.50 (731), −0.50 (710)
AND: no stem	−0.038	(−0.107, 0.032)	6-11-32	0.80 (714), −0.50 (722), −0.50 (720)
AND: 2:5	−0.057	(−0.109,−0.013)	4-11-34	−0.75 (748), −0.50 (720), 0.25 (743)
AND: no dlen	−0.155	(−0.248,−0.067)	9-18-22	−0.90 (704), −0.90 (701), 0.50 (728)
AND: 2:1	−0.352	(−0.463,−0.241)	5-32-12	−0.99 (701), −0.96 (732), 0.75 (743)
AND: 2:2	−0.583	(−0.690,−0.472)	2-43-4	−1.00 (723), −1.00 (749), 0.47 (741)

Table 18: Relevance Methods Compared to Terabyte ‘2:3’ Baseline, Average Precision Measure

Expt	MAP Diff	95% Conf	vs.	3 Extreme Diffs (Topic)
OR: 2:4	−0.016	(−0.026,−0.005)	11-36-2	−0.12 (707), −0.08 (733), 0.08 (705)
OR: no stem	−0.027	(−0.044,−0.012)	14-35-0	−0.21 (745), −0.19 (733), 0.09 (723)
OR: 2:5	−0.033	(−0.062,−0.010)	17-32-0	−0.49 (736), −0.26 (731), 0.09 (746)
OR: no dlen	−0.093	(−0.125,−0.059)	10-39-0	0.34 (745), −0.29 (728), −0.30 (710)
OR: 2:2	−0.215	(−0.264,−0.164)	3-46-0	−0.53 (710), −0.50 (739), 0.21 (745)
OR: 2:1	−0.249	(−0.301,−0.199)	0-49-0	−0.69 (709), −0.60 (737), −0.00 (729)
AND: 2:4	−0.006	(−0.011, 0.000)	19-30-0	−0.06 (707), −0.05 (723), 0.03 (701)
AND: 2:5	−0.011	(−0.022,−0.002)	19-30-0	−0.14 (749), −0.12 (732), 0.06 (707)
AND: no stem	−0.037	(−0.059,−0.018)	11-38-0	−0.35 (745), −0.23 (726), 0.09 (723)
AND: no dlen	−0.082	(−0.108,−0.058)	9-40-0	−0.30 (710), −0.26 (749), 0.11 (745)
AND: 2:1	−0.138	(−0.178,−0.101)	5-44-0	−0.49 (710), −0.43 (737), 0.08 (743)
AND: 2:2	−0.195	(−0.240,−0.152)	1-48-0	−0.53 (710), −0.51 (709), 0.00 (729)

version of SearchServer used for these diagnostic runs had an updated English inflection module.

Unlike in the Robust section, mean reciprocal rank (MRR) of the first relevant retrieved is included in Table 16. (When the Robust tables were produced, we were using an older version of the trec_eval utility.) Although the ‘2:4’ method scored higher than ‘2:3’ in MRR in the Terabyte task (for both OR and AND), Table 17 shows the differences were not statistically significant.

Table 18 compares each run to its ‘2:3’ baseline in average precision. The results were very similar to the results in the Robust track in its analogous Table 2. One difference is that for the Terabyte task, the decrease in mean average precision when using the experimental ‘2:5’ method with boolean-AND was statistically significant, but the mean decrease was small (just 1 point) and the confidence interval was not wide (0 to 2 points). Table 18 shows the largest decrease was the 14 point drop on topic 749 (Puerto Rico state), for which it seems that not down-weighting the common word ‘state’ brought in a lot of long documents with relatively few references to ‘Puerto Rico’. In practice, a user might use a proximity query (‘Puerto Rico’ near ‘state’) to avoid this issue.

5 Conclusions

In the Robust ad hoc search task, we found that some conclusions were different for boolean-AND queries than for boolean-OR. A simple count of all the matches, which did poorly with boolean-OR (Success@10 of just 39%), was reasonably effective with boolean-AND (Success@10 of 76%). Weighting terms by inverse document frequency was of less value on average for boolean-AND queries than for boolean-OR. The results for boolean-AND seem to agree more with user impressions of the merits of the ranking schemes.

A technique that improved robustness was an enhancement for handling multiple stemming interpretations (we originally developed it for languages other than English). A few problem topics for stemming approaches (e.g. "mainstreaming", "price fixing") were found to be improved by this technique. The approach of selecting alternatives on a per-document basis may generalize to incorporating phrase groups, synonyms, etc. for unstructured queries, but further research is needed.

The Web Track showed that there are different search tasks which require different techniques. The content search technique that produced a Success@10 of more than 90% in the Robust task had a Success@10 of less than 50% in the Web task. The Web task was searching not for any relevant document, but for particular named pages or home pages. By using SearchServer's ability to assign extra weight to particular columns, such as columns containing the document title or the depth of a url, Success@10 for the mixed web task was increased to more than 75% and for the broad topic (distillation) queries to more than 90%.

In the Terabyte Track, we found that SearchServer's ad hoc search techniques successfully scaled and Success@10 exceeding 90% was achieved. Generally speaking, the conclusions of the Robust task carried over to the Terabyte task. Even though its data was like that of the Web task, the web techniques were not beneficial for the Terabyte task, presumably because its task definition was like that of the Robust task. It would be good if a standard test for known-item search was created for this data set.

6 Glossary

- "Precision" is the percentage of retrieved documents which are relevant.
- "Precision@n" is the precision after n documents have been retrieved.
- "Recall" is the percentage of relevant documents which have been retrieved.
- "Average precision" for a topic is the average of the precision after each relevant document is retrieved (using zero as the precision for relevant documents which are not retrieved). In this paper, it is based on the first 1000 retrieved documents for the topic. The score ranges from 0.0 (no relevants found) to 1.0 (all relevants found at the top of the list). Average precision takes into account both precision and recall, and it is very good for detecting retrieval differences because even small differences in the ranks of relevant documents affect the score.
- "Mean Average Precision" (MAP) is the mean of the average precision scores over all of the topics (i.e. all topics are weighted equally).
- "Reciprocal Rank" for a topic is one divided by the rank of the first row for which a desired page is found, or zero if a desired page was not found.
- "Mean Reciprocal Rank" (MRR) is the average of the reciprocal ranks over all the topics.
- "Success@n" is the percentage of topics for which at least one relevant document was returned in the first n rows. This measure hides a lot of retrieval differences (particularly in recall), but it may be an indicator of a user's impression of a method's robustness across topics. This paper lists Success@1, Success@5 and Success@10.

Note that Success@1 is the same as Precision@1, but for n>1, Success@n does not have the same definition as Precision@n. Success@1 seems likely to correlate well with MAP for recall-oriented methods.

6.1 Statistical Significance Tables

For tables comparing 2 runs (such as Table 2), the columns are as follows:

- "Expt" specifies the experiment. The table's heading may need to be consulted to know details such as the baseline run or the evaluation measure.
- "Diff" is the difference of the mean scores of the two runs being compared (the table or column heading says which evaluation measure is being compared).
- "95% Conf" is an approximate 95% confidence interval for the difference calculated using Efron's bootstrap percentile method [2] (using 100,000 iterations). If zero is not in the interval, the result is "statistically significant" (at the 5% level), i.e. the feature is unlikely to be of neutral impact on average, though if the average difference is small (e.g. <0.020) it may still be too minor to be considered "significant" in the magnitude sense.
- "vs." is the number of topics on which the experimental run scored higher, lower and tied (respectively) compared to the baseline run. These numbers should always add to the number of topics.
- "3 Extreme Diffs (Topic)" lists 3 of the individual topic differences, each followed by the topic number in brackets. The first difference is the largest for any topic (based on the absolute value). The third difference is the largest difference in the other direction (so the first and third differences give the range of differences observed in this experiment). The middle difference is the largest of the remaining differences (based on the absolute value).

See [8] for some comparisons of confidence intervals from the bootstrap percentile, Wilcoxon signed rank and standard error methods for both average precision and Precision@10.

References

[1] Cross-Language Evaluation Forum web site. http://www.clef-campaign.org/

[2] Bradley Efron and Robert J. Tibshirani. An Introduction to the Bootstrap. 1993. Chapman & Hall/CRC.

[3] Andrew Hodgson. Converting the Fulcrum Search Engine to Unicode. Sixteenth International Unicode Conference, 2000.

[4] NTCIR (NII-Test Collection for IR) Home Page. http://research.nii.ac.jp/∼ntcadm/index-en.html

[5] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford. Okapi at TREC-3. Proceedings of TREC-3, 1995.

[6] Text REtrieval Conference (TREC) Home Page. http://trec.nist.gov/

[7] Stephen Tomlinson and Tom Blackwell. Hummingbird's Fulcrum SearchServer at TREC-9. Proceedings of TREC-9, 2001.

[8] Stephen Tomlinson. Experiments in 8 European Languages with Hummingbird SearchServer^{TM} at CLEF 2002. Working Notes for the CLEF 2002 Workshop. http://clef.isti.cnr.it/workshop2002/WN/26.pdf

[9] Stephen Tomlinson. Experiments in Named Page Finding and Arabic Retrieval with Hummingbird SearchServer^{TM} at TREC 2002. Proceedings of TREC 2002.

[10] Stephen Tomlinson. Finnish, Portuguese and Russian Retrieval with Hummingbird SearchServer^{TM} at CLEF 2004. Working Notes for the CLEF 2004 Workshop.

[11] Stephen Tomlinson. Hummingbird SearchServer^{TM} at TREC 2001. Proceedings of TREC 2001.

[12] Stephen Tomlinson. Robust, Web and Genomic Retrieval with Hummingbird SearchServer^{TM} at TREC 2003. Proceedings of TREC 2003.

[13] Thijs Westerveld, Wessel Kraaij and Djoerd Hiemstra. Retrieving Web Pages using Content, Links, URLs and Anchors. Proceedings of TREC 2001.