How to Improve Retrieval Effectiveness on the Web?

João Ferreira
Instituto Superior de Engenharia de Lisboa
jferreira@deetc.isel.ipl.pt

Alberto Rodrigues da Silva
INESC-ID, Instituto Superior Técnico
alberto.silva@acm.org

José Delgado
Instituto Superior Técnico
Jose.Delgado@tagus.ist.utl.pt

ABSTRACT

We explore the question of combining link analysis, content analysis and classification-based techniques to improve retrieval performance on the Web. We show the potential of combination and that relatively simple implementation of combination does improve the retrieval performance.

KEYWORDS

Combination, Retrieval Information, Classification

1 Introduction

How do we find information on the Web? It is an old question far from being solved. Web information is distributed, decentralized and huge in size. The Web can be viewed as one big virtual document collection. The findings from traditional Information Retrieval (IR) research (traditional IR means text-based approaches), however, may not always be applicable in the Web setting. The Web document collection, massive in size and diverse in content, context, format, purpose and quality, challenges the validity of previous research findings based on relatively small and homogeneous test collections. Also, some traditional IR approaches may be applicable in theory, but may not be possible or practical to implement in a Web IR system. For instance, the size, distribution and dynamic nature of information on the Web make it difficult, if not impossible, to construct a complete and up-to-date data representation required for an ideal IR system. In addition, conventional evaluation measures, such as precision, recall, and even relevance, may no longer be applicable to Web IR, where a test collection representative of dynamic and diverse Web data is all but impossible to construct.

To further complicate the matter, information seeking on the Web is quite diverse in characteristics and unpredictable in nature. Web searchers come from all kinds of reasons motivated by all types of information need. The wide range of experience, knowledge, motivation, need, and purpose of Web searchers means that searchers can express wide ranges of information needs in a wide variety of ways with various criteria for satisfying their needs.

At the same time, the Web is rich with new types of information not present in most previous test collections. Hyperlinks, usage statistics, document markup tags and bodies of topic hierarchies such as Yahoo present an opportunity to leverage the Web-specific document characteristics in novel approaches that go beyond the term-based retrieval framework of traditional IR.

This paper explores the question *of combining link analysis, content analysis, and classification-based techniques to improve retrieval performance* and is divided in 7 sections. Firstly, we introduce the problem, secondly we describe individual retrieval systems, thirdly we talk about combination formulae, fourthly we present our results, fifth we discuss results and finally we present our conclusions.

In Proceedings of the E-Society 2004 Conference (E-Society2004).
© IADIS 2004.

2 Basic Retrieval Systems

The two main elements of any combination experiments are *components* and *formulae*. Components are the individual components to be combined, which can be sources of evidence, retrieval methods, or both (i.e. retrieval methods that leverage sources of evidence). *Formulae* refer to methods of combining components, which can be applied at retrieval time.

As the source for these experiences we use the WT10g collection [1], which is a ten-gigabyte subset of the 1997 Web crawl by the Internet Archive, consists of 1.7 million Web documents, 100 TREC queries (topics 451-550), and official NIST relevance judgments. The WT10g collection also includes the connectivity data, which provides lists of inlinks and outlinks of all documents in the collection.

As classification systems for the Web lack an ideal Web directory, we use Yahoo <http://yahoo.com> due to its size and popularity. Yahoo is the largest and the most widely used Web directory, and consists of 14 top categories with over 645,000 sub-categories that contain almost 3 million Web pages, which are classified and annotated by over 150 professional Yahoo cataloguers.

The text-based retrieval component is based on a Vector Space Model (VSM) using the SMART length-normalized term weights as implemented in OpenFts <http://openfts.sourceforge.net/>. For implementation details, see [2]. From text, tags are removed; stop words and weights are based on *Lnu* document term weight (1):

$$d_{ik} = \frac{(1+\log(f_{ik})) \big/ (1+\log(avg\_f_i))}{(1.0 - slope) * p_i + slope * t} \qquad (1)$$

with the slope of 0.3 for document terms [3], where f_{ik} is the number of times term k appears in document i; avg_f_i is the average in-document frequency for document i; t is the number of unique terms in the collection and p_i is the average number of unique terms in a document i. The formula for *ltc* query term weight is:

$$q_k = \frac{(\log(f_k)+1) * idf_k}{\sqrt{\sum_{j=1}^{t}\left[(\log(f_j)+1) * idf_j\right]^2}} \qquad (2)$$

where f_k is the number of times term k appears in the query; and idf_k is the inverse document frequency [4] of term k. The denominator is a document length normalization factor, which compensates for the length variation in queries. Documents were ranked in decreasing order of the inner product of document and query vectors,

$$\mathbf{q}^{\mathrm{T}}\mathbf{d}_i = \sum_{k=1}^{t} q_k d_{ik} \qquad (3)$$

For feedback, we use the top ten positive and top two negative weighted terms from the top three ranked documents of the initial retrieval results. These terms were used to expand the initial query in a pseudo-feedback retrieval process based on the adaptive linear model. The basic approach of the adaptive linear model, which is based on the concept of preference relations from decision theory [5], is to find a solution vector that will rank a more-preferred document before a less-preferred one [6]. The solution vector is arrived at via an error-correction procedure, which begins with a starting vector $\mathbf{q}_{(0)}$ and repeats the cycle of "error-correction" until a vector is found that ranks documents according to the preference order estimation based on relevance feedback [7]. The error-correction cycle i is defined by

$$\mathbf{q}_{(i+1)} = \mathbf{q}_{(i)} + \alpha\mathbf{b} \qquad (4)$$

where α is a constant, and $\mathbf{b}$ is the *difference vector* resulting from subtracting a less-preferred document vector from a more preferred one [8].

We tested 36 VSM based on the combination of four parameters (notation;p/m/l;c/t/d;0/1;0/1): query length (small(p), medium(m), large(l)), term sources (body (c), header (t), document all (d)), phrase use (1-yes;0-no) and feedback use (1-yes;0-no).

The HITS system's algorithm was modified by adopting a couple of improvements from other HITS-based approaches. As implemented in the ARC algorithm [2], the root set was expanded by 2 links instead of 1 link (i.e. expand *S* by all pages that are 2 link distance away from *S*). All intrahost links and stoplist URLs were eliminated from the hub and authority score computations. Stoplist URLs, defined as Web pages with very high indegree, were selected from the list of URLs with indegree greater than 500. Also, the edge weights by [9], which essentially normalize the contribution of authorship by dividing the contribution of each page by the number of pages created by the same author, was used to modify the HITS formulae as follows:

$$a(p) = \sum_{q \to p} h(q) \times auth\_wt(q,p) \qquad (5)$$

$$h(p) = \sum_{p \to q} a(q) \times hub\_wt(p,q) \qquad (6)$$

In the formulae above, *auth_wt(q,p)* is 1/*m* for page *q*, whose host has *m* documents pointing to *p*, and *hub_wt(p,q)* is 1/*n* for page *q*, which is pointed to by *n* documents from the host of *p*. To compute the edge weights of the modified HITS algorithm as well as to eliminate intrahost links, one must first establish a definition of a host to identify the page authorship (i.e. documents belonging to a given host are created by the same author). Though host identification heuristics employing link analysis might be ideal, we opted for simplistic host definitions based on URL lengths. Short host form was arrived at by truncating the document URL at the first occurrence of a slash mark (i.e. '/'), and long host form from the latest occurrence. We tested 6 Hits systems based on 2 parameters:

host definition (short (p), long (l)); seed set from VSM systems ((p) Vpc10, (m)Vmc10, (l)Vlc10).

The Web Directory search was implemented based on the Term Match (TM) method. TM takes a simpler approach of finding categories in which query terms occur by extending the typical category search implementation of Web directory services.

The first phase of the TM method, which produces a ranked list of categories for a query, matches query terms to terms in the Yahoo sitemap files (i.e. category labels, Yahoo site titles and descriptions, URLs) to find a set of matching nodes in the classification hierarchy and generates a ranked category list in the following manner:

1. For each matching category, (i) compute *tfc* (number of unique query terms in the category label); (ii) compute *tfs* (number of unique query terms in the site title and description) in all its sites; (iii) compute *pms* (proportion of sites with query terms in the category).

2. Rank the matching categories in the descending order of *tfc*, *tfs*, and *pms*.

Note that categories ranked via sorting by multiple variables in such an order that the terms in category labels, which are likely to be highly "powerful", are given precedence over terms in site titles or descriptions. This ranking approach is similar to how Yahoo ranks its search results except that it combines the category and site match results while collapsing the site match results to their parent categories.

The second phase of the TM method is to expand query vector (the class centroid in the TM method) that is built from the best matching categories to produce a ranked list of the WT10g documents. The expanded query vector of the TM method is a vector of selected category terms with normalized term-category association weights. The parameters tested for the TM systems are the number of top categories used, the WT10g term index and terms for pseudo-feedback. The combination of the parameters (3 top categories (1/2/3), 4 WT10g term index (body text, no phrase (*1*) body text, phrase (2) body+header, no phrase (3) body+header, phrase (4), 2 for feedback use(1-yes,0-no)) resulted in 24 TM systems.

3 How to Improve IR

We explored the question of combining text-, link- and classification-based retrieval methods for the purpose of improving IR performance on the Web. In order to investigate the effects of various evidence source parameters, 36 text-based systems based on the Vector Space Model, 6 link-based systems using the HITS algorithm and 24 classification-based systems using Yahoo category term matching approach were implemented to produce 66 sets of retrieval results for each of the 100 WT10g topics. Combinations are performed based on two main formulae, the Similarity Merge (SM) and Weighted Rank Sum (WRS) formula.

SM, originally introduced by Fox and Shaw [10] and refined by Lee [11,12], computes the combination score of a document by the sum of normalized component scores boosted by the retrieval overlap

$$CS = (\sum NS_i) * \frac{olp}{m(i)} \qquad (7)$$

where:*CS*=combination score of a document; NS_i =normalized score of a document by system *I*; *olp*=number of systems that retrieved a given document; *m*(*i*)= number of systems in a method to which system *i* belongs. The normalized document score, NS_i, is computed by Lee's min-max formula [11,12], where S_i is the retrieval score of a given document and S_{max} and S_{min} are the maximum and minimum document scores by system *i*.

$$NS_i = (S_i - S_{min}) / (S_{max} - S_{min}) \qquad (8)$$

The Weighted Sum (WS) formula attempts to compensate for the deficiencies of the SM formula by weighting the contributions of combination components according to their relative strengths and computing the combined estimate of a document's relevance to a query (i.e. retrieval status value) by the weighted sum of individual estimates.

Weighted Rank Sum **(WRS)** formula, which uses rank-based scores (e.g. 1/rank) in place of document scores of WS formula, was tested:

$$CS = \sum(w_i * RS_i) \qquad (9)$$

where: w_i = weight of system *i*, RS_i = rank-based score of a document by system *i*. Although WRS formula aims to weight the contributions of individual combination components to the retrieval outcome by their relative strength, it does not explicitly differentiate between overlapped and non-overlapped instances.

Overlap Weighted Rank Sum **(OWRS)**, attempts to leverage overlap while compensating for the differences among combination component systems by weighting rank-based scores by overlap partitions. For the *Rank-Overlap Weighted Rank Sum* (**ROWRS)** we replace w_i of eq. 9, for w_{ikj} = weight of system *i* in overlap partition *k* at rank *j*, which needs a performance estimate at a given rank, overall average precision is not appropriate. Instead, three rank-based measures, namely Precision (P), Effectiveness (F), and Success / Failure (sf) at each rank, were used to compute the weights in three versions of the ROWRS formula. The *F*-value, which is a Dice-coefficient of similarity for the set of documents retrieved and the set of documents relevant to a query, augments the precision value with consideration of recall at a given rank [13]:

$$F = \frac{2r}{n + N_r} = \frac{2}{{}^{1}\!/_{R} + {}^{1}\!/_{P}} \qquad (10)$$

where: *r* = number of relevant documents retrieved; *n* = number of documents retrieved; *Nr* = total number of relevant documents; R = recall (*r*/*Nr*); *P* = precision (*r*/*n*). For both WRS and OWRS formulae, three variations that amplify the contribution of the top performing system were investigated. These variations, in an increasing order of emphasis for the top system, are Top System Pivot 1 (st1), Top System Pivot 2 (st2), and Overlap Boost (olpboost). The basic idea here is to supplement the result of the best performing systems with combination by using a weighting combination function that amplifies the rank-based score of a document retrieved by the top systems while dampening the contributions from worse performing systems. A generalized form of *st1*, *st2*, and *olpboost* can be expressed as eq. 9 where w_i is changed to $w_{kj}(L_i)$ = weighting function of system group L_i in overlap partition *k* at rank *j*.

4 Results

First, we will discuss the results of single systems by combinations (§4.1) component methods, which will be the baseline for combinations runs. Internal combination method results will be reviewed next (§4.2), followed by inter-method combination results. Intra-method combination refers to combining systems within a given method (e.g. VSM system 1 and VSM system 2), whereas inter-method combination refers to combining systems across methods. In both intra- and inter-method combination, results by the Similarity Merge (SM) and Weighted Rank Sum (WRS) combination formulae were examined. While combination runs discussed in these two sections resulted from a general combination approach that produced many possible combination component combinations of interest in a given combination domain, the next section, called "top system combination", examines the results of combining a handful of "best" systems from each method. In top system combination, the results of WRS formula variations were compared to the baseline formula of SM.

4.1 Individual Systems

The best performing VSM system, measured by average precision, was vlc10 (long query, body text, phrase, and no feedback). The best HITS system was hpm (short host, seed set system of vmc10) for topics 451-500 and hpl (short host, seed set system of vlc10) for topics 501-500. The best TM system, which differed over topic sets as HITS did, was t221 (top 2 categories, body text, phrase, no feedback) for topics 451-500 and 501-550.

In general, the most influential system parameter appears to be the query length. It is interesting to note that VSM and HITS systems benefit from longer queries, whereas TM systems perform better with shorter queries. Host definition, which determines the elimination of intrahost links and computation of link edge weights, seems to be a crucial parameter for HITS systems.

We used the following combinations:

- Internal Systems (combination of internal parameters of each system).
- External Systems (combination of different systems).
- Top Systems (On top systems (simple, internal and external systems) identified we tested different formulae combinations). In each of the four possible combinations of the three methods, combination was conducted in a similar manner as the internal system combination to investigate the general combination tendencies of cross-method combination rather than to focus specifically on potentially advantageous system combinations.

4.2 Internal systems

Comparing the best performances of SM and WRS combination with the best baseline system reveals some interesting patterns of interplay between the combination formula and the retrieval method. In both VSM and TM combination, WRS closely shadows the baseline system while SM falls below the baseline performance. In HITS combination, however, SM results are the best by all performance measures while WRS seems to overtake and surpass the baseline performance at lower ranks. In VSM combination, the best WRS system achieves a higher average precision than the best baseline system in topics 501-550, although the difference is only marginal (less than 1%). R-Precision of WRS is also slightly better than baseline in both topic groups. In fact, WRS and baseline results are almost identical in TM combination. SM combination, on the other hand, significantly degrades the baseline performance in TM combination, while falling slightly below the baseline performance in VSM combination.

Notation for figures 1 and 2: F means parameter combination; a at end means "combination by SM formula"; b at end means "combination through WRS formula"; RRN = Total Number of Relevant documents; avgP =average precision averaged over queries; optF = optimum F (Optimum *F*, which is the maximum of *F*-value (Equation 10) over all ranks);R-P = R-Precision (R-Precision compensates for precision's insensitivity to the size of the relevant document pool by computing precision at rank R, where R is the total number of relevant documents for a query); P@*k* = Precision at rank *k*.

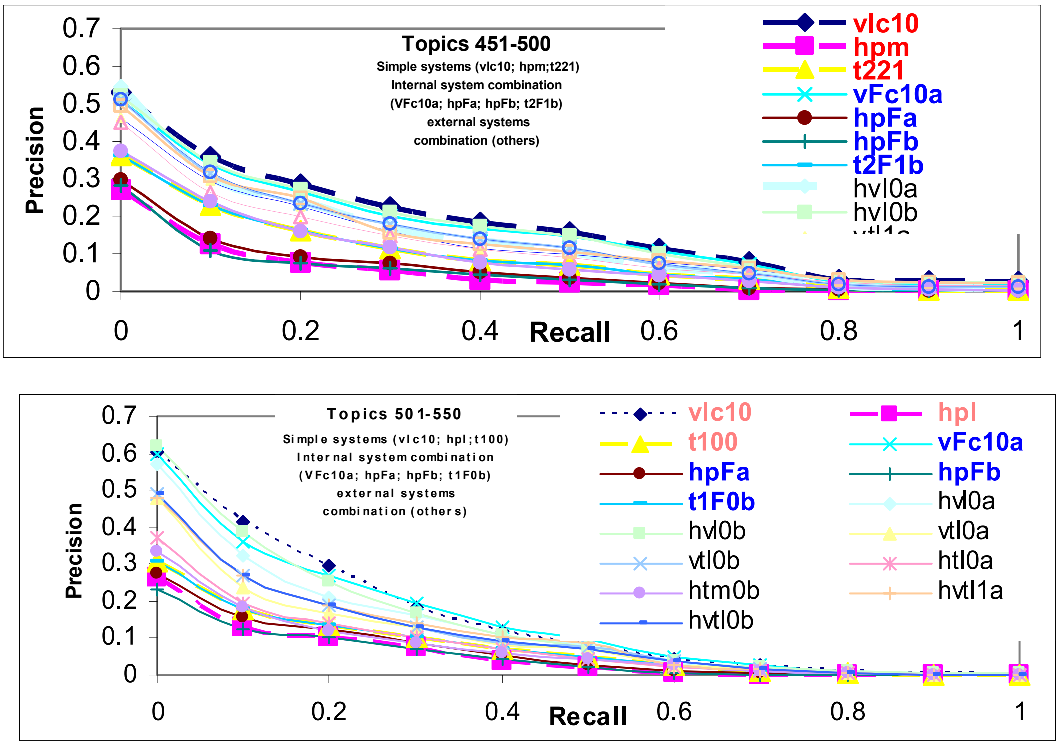

Figure 1. Results of best simple systems and internal and external combination. RRN on secondary axis.

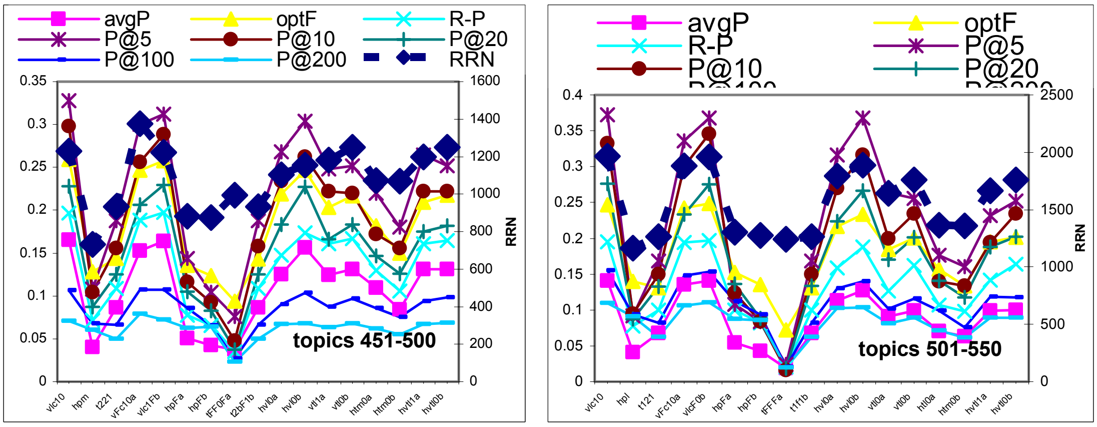

Figure 2. Recall-Precision Curve of best simple systems, internal and external combination .

Interestingly enough, SM combination sometimes hurt early precision but retrieves more total number of relevant documents in VSM and TM combination (topics 451-500 of VSM combination, topics 451-500 of TM combination). One possible explanation is that SM in some situations retrieves more relevant documents at lower ranks, which the recall-precision graphs of TM combination runs seem to suggest. One possible explanation for this phenomenon may be that the combined solution space of HITS systems is much larger than that of the best individual HITS system, while the best system dominates the combined solutions space in VSM and WD methods.

4.3 External systems

Observations from internal systems combination mostly held true in external systems combination, although combination seemed to degrade the best single system performance more in inter-method combination than in intra-method combination. In all but the HITS-TM method combination, the baseline systems act as upper and lower bound performance thresholds and combination results fall nicely between them. There is, however, a distinct difference in the level of combination results, which is nicely illustrated in the recall-precison graphs. The VSM-HITS combination system results tend to be closer to the upper bound baseline, while VSM-TM results fall towards the middle. VSM-HITS-TM results fall towards the middle of the upper and higher of the two lower bound baseline results (i.e. VSM and TM).

4.4 Top systems

Results show (figure 3) all top three combination systems retrieved fewer relevant documents and had higher precision at 200 than the baseline, which suggests that the gain in performance came from boosting the ranking of relevant documents at earlier ranks, a document in top systems.

The naming convention for combination system is the combination system name = v/h/t, where:

- v for VSM systems with query and phrase parameters, F – all combined, or F2 vlc00 and vlc10 combined
- h for HITS systems with host, query and phrase parameters combined, or F all parameters
- t for TM systems, F all combined, F2 t101 and t121 combined, F3 t10.1, t121, t201 and t221 combined or F4 t121, t221 and t321 combined
- Formulae tested were: WRS **(a);** OWRS: *no pivot* **(b0)**, st1 **(b1)**, st2 **(b2)**, *olpboost* **(b3);** - ROWRS-*sf*: *no pivot* **(c0)**, st1 **(c1)**, st2 **(c2)**, *olpboost* **(c3);** ROWRS-*F*: *no pivot* **(d0)** and - ROWRS-*P*: *no pivot* **(e0)**

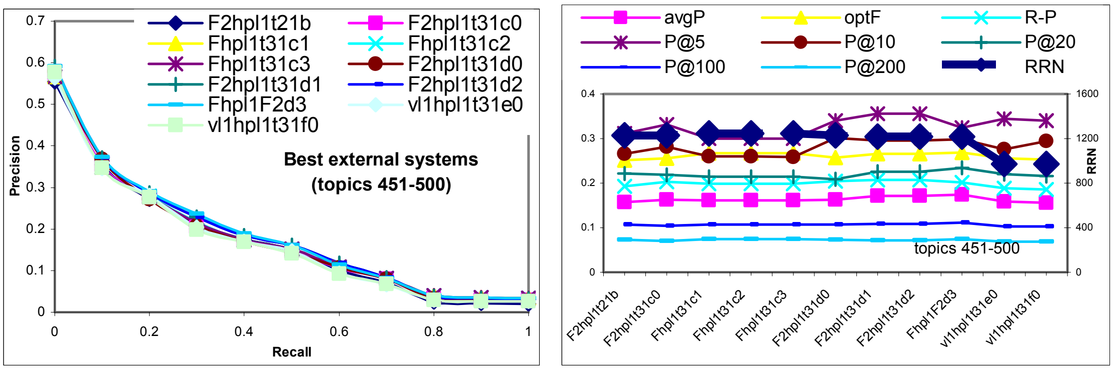

Figure 3. Recall-Precision Curve Results of best systems with different combination formulae for topics 451-500. For topics 501-550 graphics is equivalent.

The loss in the number of relevant documents retrieved can be attributed to ROWRS formula's tendency to eliminate uniquely retrieved documents from the result set. Even without the uniquely retrieved relevant documents, ROWRS outperforms OWRS regardless of top-system pivot variations. Results from external systems combinations are similar to the internal systems combinations. There is, however, a distinct difference in the level of combination results, which is nicely illustrated in the recall-precison graphs. The VSM-HITS combination system results tend to be closer to the upper bound baseline, while VSM-TM combination results fall towards the middle. VSM-HITS-TM combination results fall towards the middle of the upper and higher of the two lower bound baseline results (i.e. VSM and WD). Comparison of rank-based measures shows the success/failure measure to be superior to precision- or effectiveness-based measures for ROWRS. As for top-system pivot variations, the ROWRS formula seems to work best with the heaviest emphasis on the top system contribution (*olpboost*), in contrast with the OWRS formula that shows the best results without st1 and st2. The different effects of st1 and st2 between OWRS and ROWRS formulae may indicate the relationship between the rank and the relevance of a document in top systems.

The uneven distribution of relevant documents over ranks means that rank-based weighting is more likely to be effective than weighting based on performance estimates over all ranks, as evidenced by the superior results of systems that use ROWRS over OWRS formula. It is not immediately clear, however, why the st1 and st2 enhances performance when used with rank-based weights but hurt performance when applied evenly across ranks. It might be that top-system pivot and rank-based weight together boosts the top system contributions over all ranks can both help and hurt the performance contributions more when they are more beneficial, whereas indiscriminate boosting of top-system contributions over all ranks can both help and hurt the performance.

5 Results Discussion

In order to investigate the effects of various evidence source parameters, 36 text-based systems based on the Vector Space Model, 6 link-based systems using the HITS algorithm and 24 classification-based systems using Yahoo category term matching approach were implemented to produce 66 sets of retrieval results for each of the 100 WT10g topics. The retrieval results were then combined in a comprehensive manner within each method as well as across methods using a score-based and a rank-based combination formula. In addition, a handful of the best performing systems from each method were combined with variations of the rank-based combination formulae to explore the optimization of combination parameters. Analysis of results suggests that *query length and host definition are the most influential system parameters for retrieval performance*.

For VSM and HITS systems that use the VSM results as the seed documents, longer queries produced far better results than shorter queries, while shorter queries affected better results in TM systems. The host definition, which directly influences both the elimination of intrahost links and link weight computation of the HITS algorithm, turned out to be a crucial parameter for HITS systems, with the shorter definition is clearly superior to the longer definition.

For HITS systems, the quality of the seed document set, both in the number of relevant documents and the richness of link topology appeared to be vital for their effectiveness. Even the optimum HITS system, using the seed set of all known relevant documents, produced disappointing results due to many queries that produced only a small number of relevant documents and the possibly truncated and spurious link topology of WT10g. In fact, 85 out of 100 seed sets produced by the best VSM system were composed of 85% or more non-relevant documents, which severely handicapped the maximum performance threshold of HITS systems. Among the retrieval systems tested, VSM systems clearly outperformed other systems, with TM systems showing better results than HITS systems. In general, average precisions of VSM systems were roughly twice as good as TM systems and four times the average precisions of HITS systems. The differences in retrieval methods that affected different retrieval outcomes appeared to influence both internal and external combinations systems, where the system results were combined within and across retrieval methods. Interestingly, the only internal combination systems that enhanced the baseline performance of the best combination component results occurred with the worse performing HITS systems. Internal combination of VSM and TM systems behaved similarly in that combination detracted from the baseline performance although combining TM system results degraded baseline results much more severely than VSM combination when using the SM formula.

In HITS combination, the score-based SM formula produced better combination results than the rank-based WRS formula, which was opposite of the VSM and TM combination results. To investigate the possible reasons why combining HITS system results enhanced retrieval performance while combining VSM or TM system results degraded the baseline performance, we examined the degree of overlap in relevant documents in HITS systems in comparisons with VSM and TM systems and found that HITS systems retrieved much more diverse sets of relevant documents than VSM or TM systems and thus had the most to gain by combination.

External systems combinations produced results in between upper and lower threshold performance levels determined by the baseline systems of the methods combined. As was the case in the internal combination systems, introduction of the TM system results into the combination pool degraded the performance level of the combined results in all external combination except in HITS-TM combination, where diverse solution spaces of HITS systems seemed to overpower the potential adverse influences of TM systems to produce the combination results that surpassed the baseline performance level. The combination of VSM and HITS systems, however, did not produce better results than the baseline, because the solution space of HITS systems, though diverse from one another and from the solution spaces of TM systems, had much overlap with those of VSM systems.

The different outcomes of SM and WRS combination formulae were also observed in internal and external combination results, although the SM formula results seemed to be more stable across methods than WRS formula results. In general, the WRS formula appeared to have an advantage over the SM formula, which worked better with HITS systems. Instead of optimizing the combination formula, we considered as potential causes for the different outcomes the main differences between SM and WRS formulae, which were SM's tendency to differentiate between documents in rank proximity, SM's heavier emphasis on the overlap count, and the WRS weighting of combination component contributions based on past performance.

In top system combination WRS formula variations tested, st1, st2 and the overlap boost, which emphasized progressively the contributions of overlapped top system documents showed the best results, which suggests that leveraging overlap in conjunction with the rankings of the best performing systems is an advantageous combination approach. The gain in performance by top system combination, however, was marginal at best, but also came at the cost of recall (i.e. the number of relevant document retrieved). The decrease in recall was due to the loss of uniquely retrieved relevant documents because top system combination formulae considered only documents that were retrieved by multiple systems.

6 Conclusions

This paper confirms the viability of combination for Web IR by not only determining the existence of the combination potential in the combined solution spaces of text-, link-, and classification-based retrieval methods but also by demonstrating that relatively simple implementation of combination does improve the retrieval performance.

7 References

[1] http://es.cmis.csiro.au/TRECWeb/access_to_ data.html.

[2] Ferreira, J. (2004). Ways of searching Information on the Web. PhD Thesis, IST, Portugal (in Portuguese).

[3] Buckley C. Singhal A. e Mitra M. (1997). Using query zoning and correlation within SMART: TREC 5. In E. M. Voorhees & D. K. Harman (Eds.).

[4] Buckley C. Singhal A. Mitra M. & Salton G. (1996). New retrieval approaches using SMART: TREC 4. In D. K. Harman (Ed.).

[5] Sparck Jones K. (1972). A statistical externospretation of term specificity and its application in retrieval. *Journal of Documentation 28* 11-21.

[6] Fishburn P. C. (1970). Utility *theory for decision making*. New York: John Wiley.

[7] Wong S. K. M. Yao Y. Y. & Bollmann P. (1988). Linear structure in information retrieval. Proceedings of the 11th A. I. ACM SIGIR Conference on Research and Development in Information Retrieval 219-232

[8] Wong S. K. M. Yao Y. Y. Salton G. & Buckley C. (1991). Evaluation of an adaptive linear model. JASIS, 42 723-730.

[9] Chakrabarti S. Dom B. Raghavan P. Rajagopalan S. Gibson D. & Kleinberg J. (1998b). Automatic resource list compilation by analyzing hyperlink structure and associated text. *Proceedings of the 7th International World Wide Web Conference.*

[10] Fox E. A. & Shaw J. A. (1994). Combination of multiple searches. In D. K. Harman (Ed.). TREC-2.

[11] Lee J. H. (1996). *Combining multiple evidence from different relevance feedback methods (Tech. Rep. No. IR-87).* Amherst: University of Massachusetts Center for Intelligent Information Retrieval.

[12] Lee J. H. (1997). Analyses of multiple evidence combination. Proceedings of the ACM SIGIR Conference on Research and Development in IR 267-276.

[13] Shaw W. M. Jr. (1986). On the foundation of evaluation. *JASIS,* 37 346-348.