Assessment of WWW-Based Ranking Systems for Smaller Web Sites

OLA ÅGREN

Department of Computing Science, Umeå University, SE-901 87 Umeå, SWEDEN
ola.agren@cs.umu.se

Abstract. A comparison between a number of search engines from three different families (HITS, PageRank, and Propagation of Trust) is presented for a small web server with respect to perceived relevance. A total of 307 individual tests have been done and the results from these were disseminated to the algorithms, and then handled using confidence intervals, Kolmogorov-Smirnov and ANOVA. We show that the results can be grouped according to algorithm family, and also that the algorithms (or at least families) can be partially ordered in order of relevance.

Keywords: Assessment, search engines, HITS, PageRank, Propagation of Trust, and eigenvectors

(Received January 26, 2006 / Accepted March 14, 2006)

1 Introduction

Finding the required information on the WWW is not a trivial task. Currently used search engines will usually give good advice on pages to look at, but are there more personalised tools that can be used instead? We will in this work compare the relative strength of some of the algorithms usable in a user defined personalisation environment, in order to find out how they behave over smaller networks (such as a single web server).

Two things that these algorithms have in common is that they operate on a connection matrix (or adjacency matrix), and they all use a set of pages that are known to be about a specific topic (called *known pages*) as the starting point. The algorithms belong to three different families:

Hypertext-Induced Topic Selection (HITS [9]) This family of algorithms does not work on the entire connection matrix, but will instead use a subset of this matrix called H. It includes the known pages together with pages pointing at one or more pages among the known pages as well as the pages pointed out by them. Each page in the entire set is given a start value in two categories, "hub" (denoting an important link page) and "authority" (denoting a page with valuable information on the given subject). These values are adjusted by iteration and normalisation over the simultaneous equations given in Eq. (1).

$$h_i = \sum_{(i,j)\in E} a_j \qquad a_j = \sum_{(i,j)\in E} h_i \qquad (1)$$

The basic idea behind HITS is to use the inherent strength of the connection matrix. The starting point is to give a value to all pages in the known set, and then calculate the final result by propagating these values first in the forward direction of H (giving a partial result of the authority pages), then back into the hub value until the calculations are stable.

The two algorithms of the HITS family that we will use are the original **HITS** [9] algorithm as well as the **Randomized HITS** [12] algorithm, where some of the hub/authority value given to each page is dissapated to all pages in the set. There are other versions in this family, including:

Subspace HITS [12], where all stable eigenvectors found are multiplied with their relative eigenvalue strength and these are then superpositioned,

Clever [6], where the connection matrix is slightly changed by weighting according to the number of incoming/outgoing hyperlinks as well as whether the pages resided on the same site or not[1],

[1] A later version of Clever breaks up pages with a vast amount of outgoing links into micro-pages, each with its own fine-grained hub

MHITS [11], where the connection matrix is generated using web logs as well as more than one link away from the original starting page,

BHITS [2], where two things are used in order to make HITS more stable; outliers are filtered out and the weight of links between two servers are one divided by the number of links, and

Stochastic Approach to Link Structure Analysis (SALSA) [10], where H and H' are updated in order to get stochastic matrices (by dividing each value in a row with the number of values in the row). The main reason for doing this is to get a sound and stable system, but the final outcome is that for systems without dangling nodes (or weights) we get a result directly related to the in- (for authority value) and out-degree (for hub value). This can be computed much faster with other techniques.

Clever and BHITS were ruled out since all pages resided on one web server and Subspace HITS were removed since our search engine framework was unfortunately not able to support them fully. No web logs were available, thus ruling out MHITS, and SALSA did not give sufficiently fine-grained results for weighted disseminations. This left us with HITS and Randomized HITS from this family.

PageRank This is the main algorithm of Google [3], and is used to give a query independent importance number (called a *rank value*) to each web page according to the structure of hyper links between web pages.

PageRank uses a random surfing model over the Internet. This means that it models the behaviour of a web surfer that follows one randomly chosen link in the current page, every once in a while this web surfer gets bored with the current chain of pages and skips to a random page on the Internet (called the *damping factor*). Each visit to a page would in theory indicate that a page gets slightly more interesting than before. Rather than letting a simulator mark each visited page a number of times, there are much more effective ways of simulating and calculating these values.

The probability that the web surfer will visit page w_j is given in Eq. (2), using $(1-\mu) < 1$ as the dampening factor, the graph $G = (V, E)$ where V is the set of pages and E is the set of hyper links, $n = |V|$, and $d(w_i)$ is the out-degree of page w_i.

$$PR(w_j) = \frac{1-\mu}{n} + \mu \sum_{(i,j)\in E} \frac{PR(w_i)}{d(w_i)} \qquad (2)$$

This is the same thing as using a connection matrix where each column sums to 1 modified by adding the dampening factor as can be seen in Eq. (3).

$$P = \left[\frac{1-\mu}{n}\right]_{n\times n} + \mu M \qquad (3)$$

The PageRank is the dominating eigenvector of P: $P\pi = \pi, \pi \geqslant 0, ||\pi||_1 = 1$. This means that the i-th entry of π is the probability that a surfer visits page i, or the PageRank of page i.

The version that we will use in this work is **Topic-Sensitive PageRank** [8]. It uses the same general ideas and algorithm as the normal PageRank, except that skipping will be to one of the known pages; the dampening factor is only added (and scaled accordingly) if the corresponding page is known to be about that particular subject.

Propagation of Trust This set of algorithms builds on the algorithm found in [1]. The main idea is that the trust of known pages are distributed (and diminished by $1/\xi$, where $\xi > 1$) over each outgoing link, until the value is too small to make a difference any more. We will use three different versions of this algorithm in this work:

Basic Propagation of Trust (ProT) Given an initial score $\varpi(j, 0) = 1$ (100%) for pages that are on-topic and zero otherwise, and using k as the iteration count as well as setting ξ to be a value just over the dominant eigenvalue of the corresponding connection matrix we can apply the algorithm in Eq. (4).

The final answer is given after normalisation of the k:th ϖ vector.

Superpositioned Singleton Propagation of Trust (S^{2}ProT) This algorithm is a much faster replacement for ProT. For each page among the known pages we calculate a singleton (or basic) vector using ProT, and then superposition these vectors to form the final answer. Calculating a singleton is usually much faster than using the set of known pages directly in ProT.

value [4]. These micro-pages are not seen as entirely separate entities, and a secondary aggregate hub/authority value can be calculated for them as well.

$$\varpi(j,k) = \frac{1}{\xi} \sum_{(i,j)\in E} \varpi(i,k-1) + \begin{cases} \varpi(j,k-1) & j \text{ is on-topic} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Hybrid Superpositioned Singleton Propagation of Trust (HyS^{2}ProT) This is a hybrid version of S^{2}ProT where each outgoing value is further decreased with the number of outgoing links (i.e. the out-degree of a page) in the same manner as for PageRank.

1.1 Hypotheses

We have two main hypotheses regarding smaller input sets (see Appendix A for a description of the input data set):

1. These algorithms will for this data set give top five results that are disparate when comparing different families of algorithms with each other.

2. The Propagation of Trust family gives better sets of top five results (with regards to relevancy) than any of the others for this data set.

2 Methods and Materials

In order to find out whether the algorithms in question yields roughly equally relevant results the following experiment was conducted: The top five results from applying the algorithms was gathered and combined for a number of keywords (see Section 2.1), yielding a total of between nine and 20 links per keyword. These links were manually examined for relevancy with regards to the corresponding keyword by those participating in the experiment, yielding a total of 307 tests of the ten keywords. The relevances were then propagated back to the corresponding algorithms as per Section 2.2 in order to find out how relevant the mean result of each algorithm are, both per question and in total. The total relevancy of each algorithm were then compared using confidence intervals (see Section 2.3) and Kolmogorov-Smirnov (see Section 2.4) to show whether there are differences between each algorithm in terms of relevancy.

2.1 Keyword Selection Criterias

The first criteria was that at least eight pages had to contain a keyword in order for it to be eligible for inclusion in this experiment. The reason for this is that there should be a basis for variability of the top lists of each algorithm. One keyword that appears in only one page was still kept here, since variation between HITS and Randomized HITS was apparent with this keyword.

The second criteria was that combining the results from the algorithms should yield at least nine unique links in the resulting set. Once more, this rule also stems from the variability of the top lists.

The third criteria was that the pages had to be present during the experiment. Some pages have been removed since the database was gathered, thus preventing some otherwise fitting keywords from being used.

The fourth criteria was that no student web page should be included in the top lists, so that no single individual of the student body should feel singled out.

The fifth criteria was that at most 20 links should be given when combining the results of the algorithms. The reason for this is that the workload of those participating in the experiment should be reasonably small.

The sixth and final criteria was that not all pages given by an algorithm should have exactly the same weight attributed to them, in order to use the weights to find differences between the data sets.

These criteria yielded a list of a few hundred suitable candidates for inclusion in the experiment. The final selection was done using two methods – direct selection of some keywords that we felt were well defined (in this case ‘aagren’, ‘jubo’, ‘kompilatorteknik’, and ‘ola’), while the others were selected at random.

2.2 Dissemination of Result to Algorithms

The relevance checking was done using blind reviews (i.e. no reference was given as to what or which algorithm(s) produced the link in the top five positions) on a five-graded scale. The first grade indicated that the reviewer was unable to say anything about the relevancy of the page regarding the current keyword, and these values were ignored in all calculations. The second grade corresponded to a complete lack of relevancy, i.e. 0. The third grade was indicative of some relevance, i.e. $\frac{1}{3}$. The fourth grade indicated a moderate amount of relevance, i.e. $\frac{2}{3}$. The fifth and final grade indicated that the page was very relevant, corresponding to 1 (or 100% relevance). This scaling will lead to an underestimation of the true relevance of the pages, but we are interested in the relative rather than exact relevance here.

2.2.1 Original Dissemination

The pages given in the top lists for each algorithm shows which pages should be included in each dissemination.

The values corresponding to each grade were summed up and then divided by the number of grades that did not belong to the first grade, thereby forming a mean relevancy for that keyword and algorithm combination according to that reviewer.

As an example, consider the top-list P containing the five pages A, B, C, D and E. These were given the grades A - grade one, B - grade two, $\ldots$, E - grade five. This means that the mean relevancy for top-list P from this grader was

$$\frac{0 + 1/3 + 2/3 + 1}{4} = 0.5 \text{ or } 50\%.$$

2.2.2 Weighted Dissemination

The algorithms supply not only the list of pages, but also gives a weight $\in (0, 1]$ for each page. For all pages with grades higher than the first, add up both the product of page weight and the corresponding page relevancy and the sum of the weights. The final number is given by dividing the sum of products with sum of weights. The rationale here is that the higher weight attributed to them by an algorithm, the more important that page should be for the final score.

We can continue the example above by saying that the weight corresponding to page A is $a = 0.6$, weight of page B is $b = 0.7$, $\ldots$, and weight of page E is $e = 1.0$. Given the same grading as in the previous section the weighted relevancy would be

$$\frac{0.7 \times 0 + 0.8/3 + 0.9 \times 2/3 + 1}{0.7 + 0.8 + 0.9 + 1} \approx 0.5490/54.90\%.$$

2.3 Confidence Interval Comparisons

The mean result can be used to rank the algorithms according to relevancy. Moreover, by forming a confidence interval around this mean it is possible to show whether the results from the disseminations are disparate.

2.4 Kolmogorov-Smirnov Comparisons

One of the most widely used goodness-of-fit tests available is the Kolmogorov-Smirnov. It uses the maximum difference D in y values between two curves plotted in a cumulative fraction plot, i.e. going in discrete steps of 1/#steps from 0 to 1 from left to right. This difference is then compared to a number that depends on both the chosen α level (in our case 0.001) and the number of samples in the set. The number of samples to use in the comparison is calculated from the original number of samples for each input set (n_1 and n_2, respectively):

$$n = \frac{n_1 \times n_2}{n_1 + n_2}$$

3 Results

3.1 Confidence Interval Comparisons

3.1.1 Original

Both the 95% confidence intervals plotted in Figure 1 as well as the 99.9% confidence intervals given in Table 1 leads to the same result; we can divide the algorithms into four groups. The top group contains HyS^{2}ProT, S^{2}ProT and ProT, with each confidence interval encompassing the mean value of the others. The second group contains only one algorithm, Topic-Sensitive PageRank. The third group contains HITS Authority and Randomized HITS Authority, and (Randomized) HITS Hub is the algorithm that is the sole member of the last group.

Table 1: The mean relevance values and their 99.9% confidence intervals, given in descending order.

Algorithm	Mean	99.9% conf. inter.
HyS^{2}ProT	0.4823	(0.4383,0.5263)
S^{2}ProT	0.4797	(0.4325,0.5270)
ProT	0.4416	(0.3872,0.4959)
Topic-Sensitive PageRank	0.3462	(0.3073,0.3851)
HITS Authority	0.2723	(0.2210,0.3237)
Randomized HITS Authority	0.2465	(0.2057,0.2873)
(Randomized) HITS Hub	0.1719	(0.1344,0.2094)

3.1.2 Weighted

The same four groups with overlapping confidence intervals can be found in the weighted result set as well as can be seen in Figure 1 and Table 2. The first group consists of HyS^{2}ProT, S^{2}ProT and ProT, the second of Topic-Sensitive PageRank, the third of both versions of HITS Authority and the final group contains (Randomized) HITS Hub.

Table 2: The mean weighted relevance values and their 99.9% confidence intervals, given in descending order.

Algorithm	Mean	99.9% conf. inter.
ProT	0.5654	(0.5070,0.6237)
HyS^{2}ProT	0.5582	(0.5114,0.6050)
S^{2}ProT	0.5540	(0.5033,0.6047)
Topic-Sensitive PageRank	0.3783	(0.3312,0.4253)
HITS Authority	0.2761	(0.2248,0.3273)
Randomized HITS Authority	0.2761	(0.2248,0.3273)
(Randomized) HITS Hub	0.2034	(0.1569,0.2500)

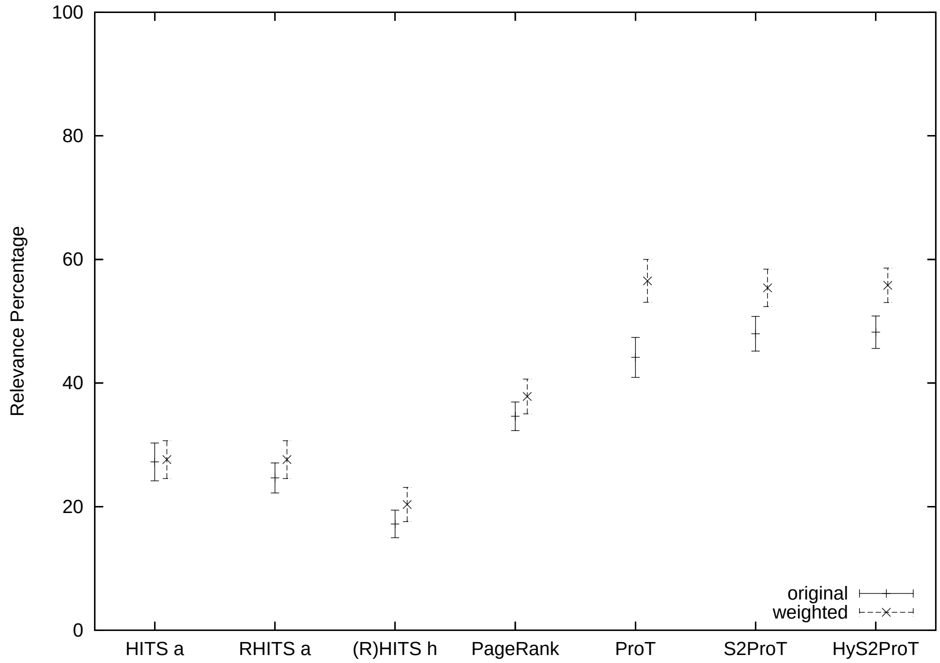

Figure 1: This graph shows the mean relevance values as well as 95% confidence intervals per algorithm, both unweighted and weighted.

3.2 Kolmogorov-Smirnov Comparisons

For each combination of result lists, we put up the following hypothesis:

H_0 : The distribution of the two lists are equal.

H_1 : The distribution of the two lists are not equal.

The Kolmogorov-Smirnov test is then applied to the combination in order to either reject or accept H_0.

3.2.1 Original

The results from these comparisons is that ProT, S^2ProT and HyS^2ProT have almost identical distribution, as does HITS Authority and both Randomized HITS Authority and (Randomized) HITS Hub. All other combinations are disparate at the 99.9% certainty level. The corresponding cumulative fraction plot can be seen in Figure 2. For full results see Appendix D.

3.2.2 Weighted

The results from these comparisons is that HITS Authority and Randomized HITS Authority have almost identical distribution, and S^2ProT has a distribution that is very close to both ProT and HyS^2ProT (while these two are disparate). All other combinations are disparate at the 99.9% certainty level. The corresponding cumulative fraction plot can be seen in Figure 3. For full results see Appendix D.

4 Discussion and Conclusions

Looking back at the two main hypotheses posed in the introduction (Section 1.1) we can see that both of them have been shown to be true with high significance:

1. The mean values from each algorithm of each family does neither appear in the confidence intervals of another (Section 3.1), nor will Kolmogorov-Smirnov retain H_0 of comparisons between families (Section 3.2) for both original and weighted values.

2. The values presented by the confidence intervals in Section 3.1 shows that a distinct relevance order can be seen among the algorithm families. The order is that Propagation of Trust yields better result than Topic-Sensitive PageRank, that in turn yields better result than HITS. This is also visible in Figure 2 where this order (remember that a lower curve corresponds to better relevancy) can be clearly seen. This is also true for the largest part of Figure 3, even though some crossing of the graphs can be seen.

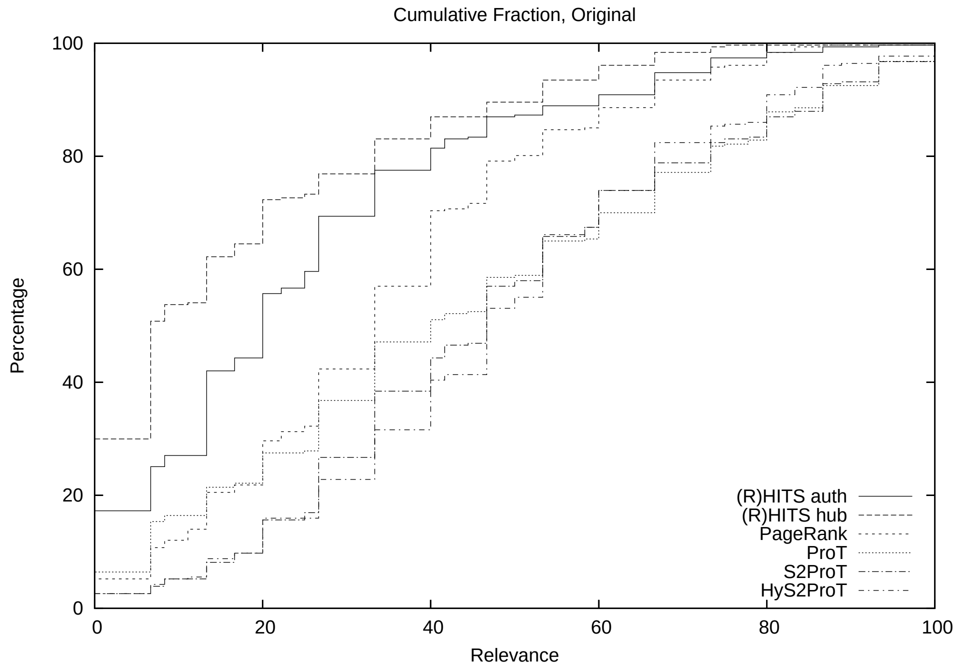

Figure 2: Cumulative fraction of answers that is at a certain level or lower.

The Hub lists of HITS and Randomized HITS are identical in the unweighted version, while the weighted version shows some minor differences (D = 0.0749). A slightly larger difference can be seen when looking at the authority scores, with difference in distribution of (D = 0.07818) or $\alpha = 0.2890$.

There is a slightly more complex situation among the algorithms of the Propagation of Trust family. While looking at the original comparisons at the high significance level we are unable to reject that each mean value *could* come from one of the other algorithms. Looking at Figure 2 and Figure 3 gives a clear indication that there *is* in fact some minor differences between the algorithms. The only way to show this is to increase α, and the α required to show that the distributions are disparate can be seen in Table 3.

Table 3: This table shows the α that must be chosen in order to show that the distributions from the algorithms are disparate.

	Original		Weighted	
	S^2ProT	HyS^2ProT	S^2ProT	HyS^2ProT
ProT	0.0074	0.0024	0.0988	0.0004
S^2ProT		0.4526		0.0528

The results given in Section 3.1 has been confirmed by using ANOVA tests, where only tests between algorithms of the same family have p-values of 0.001 or higher. The relevant tests are between:

- HITS Authority and Randomized HITS Authority ($f = 1.6811, p = 0.1953$),
- ProT and S^2ProT ($f = 3.0072, p = 0.0834$),
- ProT and HyS^2ProT ($f = 3.6335, p = 0.0571$), and, finally,
- S^2ProT and HyS^2ProT ($f = 0.0169, p = 0.8966$).

Our conclusion of this experiment is that not only does the algorithms in the Propagation of Trust family yield good results even for smaller databases, they give better results than the competition. The main reason for the lower results of Topic-Sensitive PageRank is probably the relative lack of links, the more links (and pages) the better it seems to be working. There are on the other hand two reasons for the lower than expected results from HITS:

- The first reason is that some pages that came from the hub lists do not talk about a subject directly but have lots of links to pages that does.
- The second reason is that HITS suffer from mutually reinforcing relationships between pages among the included pages as well as topic drift, where a

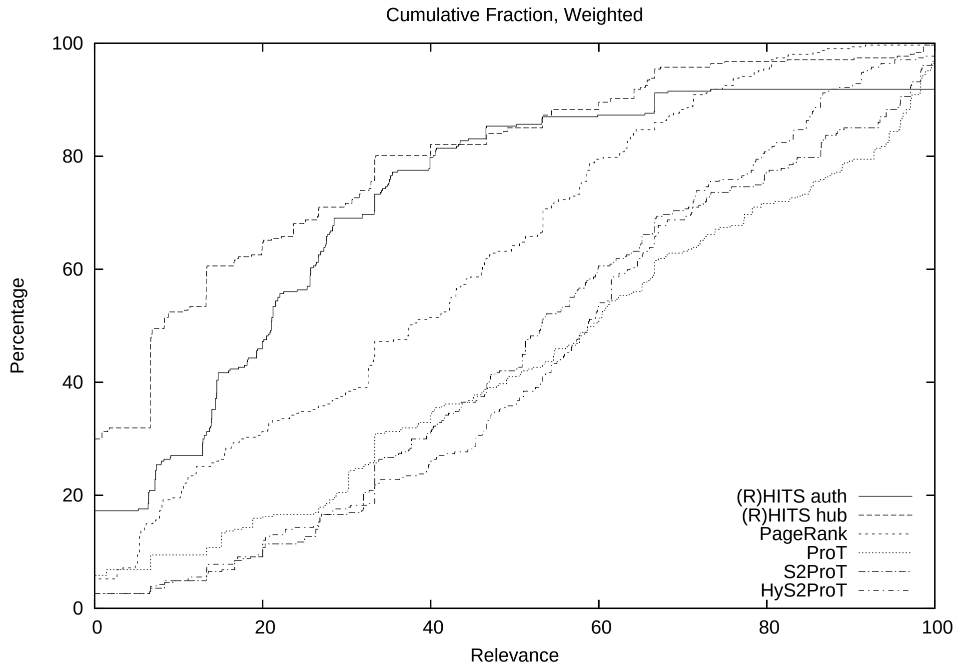

Figure 3: Weighted cumulative fraction of answers that is at a certain level or lower.

tight-knit community of pages can take over as the most important pages for a query.

The scaling could be improved on an intuitive level by using a more sensible scale (such as ignore, 0%, 50%, 75% and 100%) if more exact relevance number were required. We have opted to continue with this scaling, since rescaling would not affect the final result.

One thing that could be done to get even more information per keyword is to look at more than 5 links per list. This method have the drawback that the number of links to process for those participating in the experiment increases almost linearily, so that an increase from 5 to 10 links per algorithm yields roughly twice as many links to check.

Another test that should be done is to look at a much larger database, preferrably the entire Internet. Since this data is not available at this time this is hardly feasible, even though we do have much larger databases to work on (such as the entire web structure at Umeå University). It would however be much harder to choose keywords to use, since even more criterias (such as pages on more than one web server) could be applied.

Hiding the true souce of each link rather than comparing each list directly was first seen in [7] (comparing Clever and Yahoo), since they found a distinct problem in their earlier comparisons that showed the entire result lists from each search engine/algorithm [5]. One set of the included result lists in the older test contained annotations and one-line summaries, thus yielding better information for the classifier to use when assessing relevancy. We must agree that using blind examinations for relevancy yields an objectively better result and should be used in future studies.

Acknowledgements

Many thanks to Leif Nilsson, Helena Lewandowska, and Mårten Forsmark for valuable comments and suggestions that were useful for improving the quality of this paper, as well as to the approximately 80 participants of the assessment.

References

[1] Ågren, O. Propagation of meta data over the world wide web. In *Proceedings of the International Conference on Internet Computing (IC '03)*, volume 2, pages 670–676, Las Vegas, Nevada, USA, June 23-26 2003.

[2] Bharat, K. and Henzinger, M. R. Improved algorithms for topic distillation in a hyperlinked environment. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference*

on Research and development in information retrieval, pages 104–111, New York, NY, USA, 1998. ACM Press.

[3] Brin, S. and Page, L. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[4] Chakrabarti, S. Integrating the document object model with hyperlinks for enhanced topic distillation and information extraction. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pages 211–220, New York, NY, USA, 2001. ACM Press.

[5] Chakrabarti, S., Dom, B., Raghavan, P., Rajagopalan, S., Gibson, D., and Kleinberg, J. Automatic resource compilation by analyzing hyperlink structure and associated text. In *WWW7: Proceedings of the seventh international conference on World Wide Web 7*, pages 65–74, Amsterdam, The Netherlands, 1998. Elsevier Science Publishers B. V.

[6] Chakrabarti, S., Dom, B. E., and Indyk, P. Enhanced hypertext categorization using hyperlinks. In Haas, L. M. and Tiwary, A., editors, *Proceedings of SIGMOD-98, ACM International Conference on Management of Data*, pages 307–318, Seattle, US, 1998. ACM Press, New York, US.

[7] Chakrabarti, S., Dom, B. E., Kumar, S. R., Raghavan, P., Rajagopalan, S., Tomkins, A., Gibson, D., and Kleinberg, J. Mining the web's link structure. *Computer*, 32(8):60–67, 1999.

[8] Haveliwala, T. H. Topic-sensitive PageRank. In *Proceedings of the eleventh international conference on World Wide Web*, pages 517–526. ACM Press, 2002.

[9] Kleinberg, J. Authoritative sources in a hyperlinked environment. In *Proc. of ACM-SIAM Symposium on Discrete Algorithms, 1998*, pages 668–677, 1998.

[10] Lempel, R. and Moran, S. SALSA: The stochastic approach for link-structure analysis. *ACM Trans. Inf. Syst.*, 19(2):131–160, 2001.

[11] Miller, J. C., Rae, G., Schaefer, F., Ward, L. A., LoFaro, T., and Farahat, A. Modifications of Kleinberg's HITS algorithm using matrix exponentiation and web log records. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 444–445, New Orleans, Louisiana, United States, 2001. ACM Press.

[12] Ng, A. Y., Zheng, A. X., and Jordan, M. Stable algorithms for link analysis. In *Proceedings of the Twenty-fourth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Sept. 2001.

A Test Database

The database used in this experiment contains a subset of all the pages available at www.cs.umu.se, the web site of the Department of Computing Science, Umeå University. This database was collected in January 2003.

The database contains 7312 pages, of which 2728 are HTML pages with outgoing links. There are a total of 22970 hyperlinks, yielding an average of approximately 8.42 outgoing hyperlinks per HTML page and just over 3.14 incoming links per page.

A total of 57823 keywords are present at least once in the entire set of pages. These keywords are present in on average 9.87 pages, for a total of 570870 page occurrences. 33854 of these keywords appear in the same set of pages as another keyword, so that only 23969 unique ranking lists are required for the entire set per algorithm.

Looking at the web site as a undirected graph we find that it contains 130 components, with 6885 pages in the largest component.

B Keywords

The keywords used in the assessment can be seen in Table 4.

C Confidence Intervals per Keyword

The 95% confidence intervals for each keyword given in Appendix B (and Table 4) can be seen in Figure 4 and Figure 5.

D Full Kolmogorov-Smirnov Results from Comparisons

Table 5 contains all results from applying Kolmogorov-Smirnovs tests on each of the 21 possible combinations of algorithms (counting the results from HITS Hub and Randomized HITS Hub values as equal). These values are given for both the original and for the weighted Kolmogorov-Smirnov tests.

Table 4: The keywords used in the tests

Keyword	English	#pages in DB	#pages in test	#tests
aagren	Ågren[a]	227	17	34
choklad	chocolate	15	17	33
exempelrapport	sample report	1[b]	9	34
jubo	Jürgen Börstler[a]	110	19	30
kallin	Kallin[a]	161	19	37
kompilatorteknik	compiler construction/techniques	23	18	31
konstant	constant	18	16	26
matrismultiplik	matrix multiplic[c]	8	16	19
ola	Ola[a]	251	17	37
relation	relation	17	14	26
			Sum:	307

[a] Proper name.

[b] Breaks the first selection criteria, but was included since all other criterias were met and it manifested a real difference between the authority lists of HITS and Randomized HITS.

[c] This keyword has been truncated by stemming.

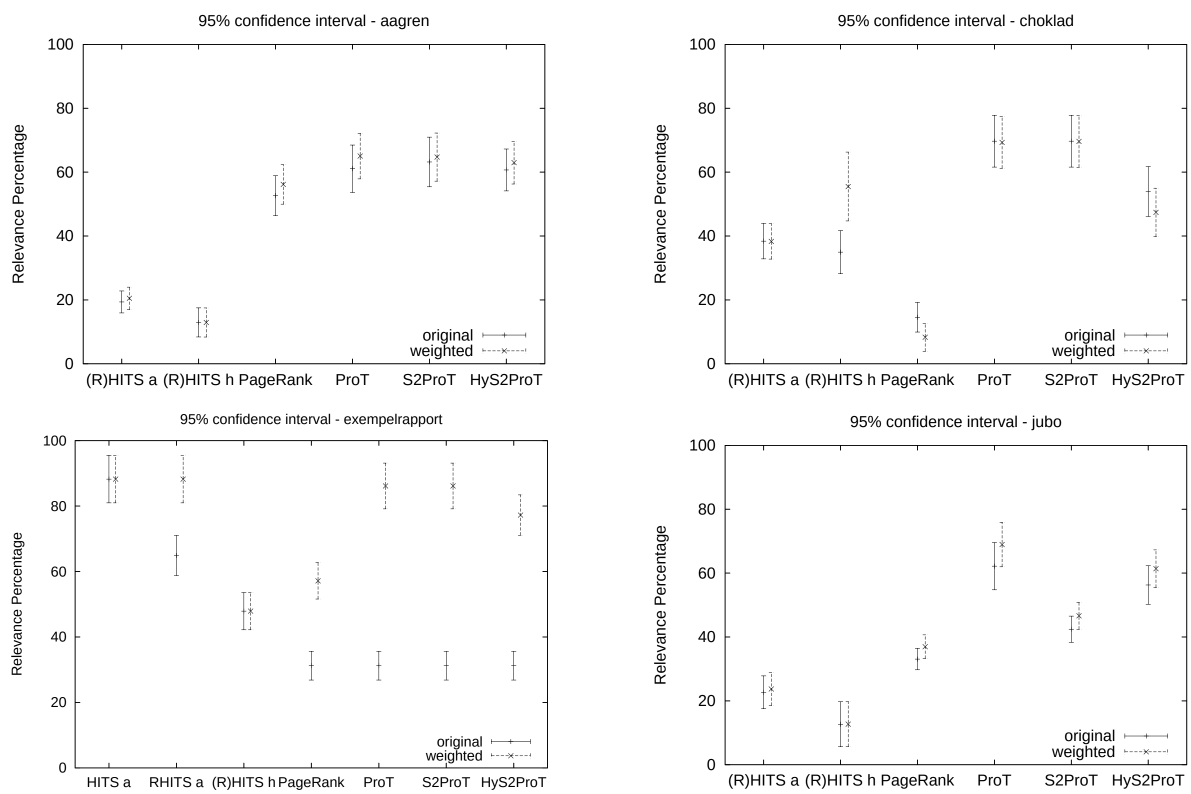

Figure 4: Mean relevance and 95% confidence intervals for the first four keywords.

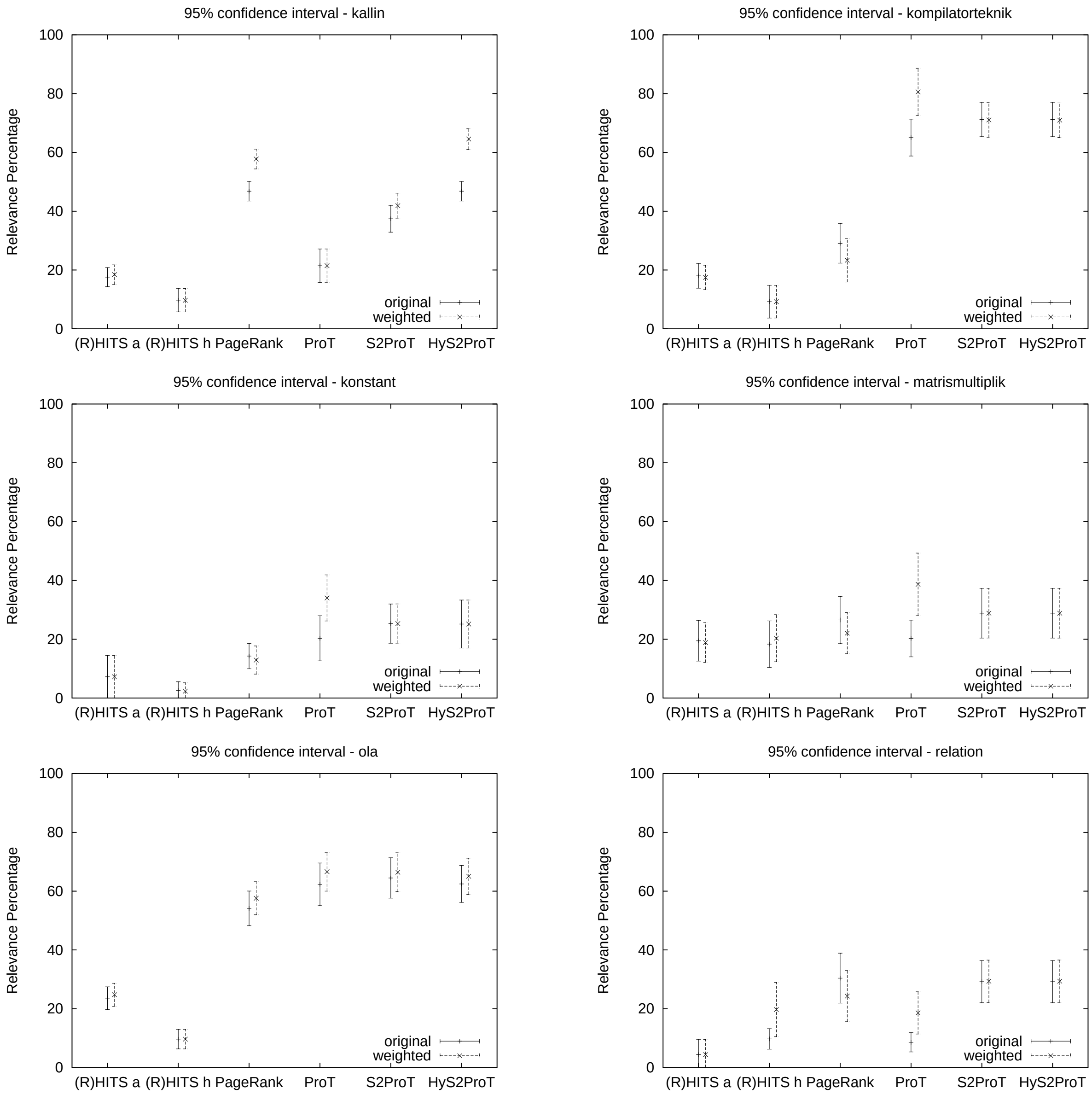

Figure 5: Mean relevance and 95% confidence intervals for the last six keywords.

Table 5: The full results from the Kolmogorov-Smirnovs tests. Both tables use a cut-off of 0.157 ($n = 154$, $\alpha = 0.001$) .

Original	Randomized HITS Authority	(Randomized) HITS Hub	Topic-Sensitive PageRank	ProT	S^2ProT	HyS^2ProT
HITS Authority	D = 0.0782 < 0.157, H_0 accepted	D = 0.2704 > 0.157, H_0 rejected	D = 0.2704 > 0.157, H_0 rejected	D = 0.3290 > 0.157, H_0 rejected	D = 0.4235 > 0.157, H_0 rejected	D = 0.4625 > 0.157, H_0 rejected
Randomized HITS Authority		D = 0.2704 > 0.157, H_0 rejected	D = 0.2736 > 0.157, H_0 rejected	D = 0.3322 > 0.157, H_0 rejected	D = 0.4267 > 0.157, H_0 rejected	D = 0.4658 > 0.157, H_0 rejected
(Randomized) HITS Hub			D = 0.4267 > 0.157, H_0 rejected	D = 0.4560 > 0.157, H_0 rejected	D = 0.5700 > 0.157, H_0 rejected	D = 0.5733 > 0.157, H_0 rejected
Topic-Sensitive PageRank				D = 0.2052 > 0.157, H_0 rejected	D = 0.2606 > 0.157, H_0 rejected	D = 0.3029 > 0.157, H_0 rejected
ProT					D = 0.1336 < 0.157, H_0 accepted	D = 0.1466 < 0.157, H_0 accepted
S^2ProT						D = 0.0684 < 0.157, H_0 accepted

Weighted	Randomized HITS Authority	(Randomized) HITS Hub	Topic-Sensitive PageRank	ProT	S^2ProT	HyS^2ProT
HITS Authority	D = 0.0782 < 0.157, H_0 accepted	D = 0.2932 > 0.157, H_0 rejected	D = 0.3225 > 0.157, H_0 rejected	D = 0.4951 > 0.157, H_0 rejected	D = 0.5244 > 0.157, H_0 rejected	D = 0.5505 > 0.157, H_0 rejected
Randomized HITS Authority		D = 0.2932 > 0.157, H_0 rejected	D = 0.3225 > 0.157, H_0 rejected	D = 0.4951 > 0.157, H_0 rejected	D = 0.5244 > 0.157, H_0 rejected	D = 0.5505 > 0.157, H_0 rejected
(Randomized) HITS Hub			D = 0.3550 > 0.157, H_0 rejected	D = 0.5342 > 0.157, H_0 rejected	D = 0.5700 > 0.157, H_0 rejected	D = 0.5831 > 0.157, H_0 rejected
Topic-Sensitive PageRank				D = 0.2899 > 0.157, H_0 rejected	D = 0.2410 > 0.157, H_0 rejected	D = 0.3062 > 0.157, H_0 rejected
ProT					D = 0.0977 < 0.157, H_0 accepted	D = 0.1629 > 0.157, H_0 rejected
S^2ProT						D = 0.1075 < 0.157, H_0 accepted