Journal of Computer Science 2 (8): 638-645, 2006
ISSN 1549-3636
© 2006 Science Publications

Improving the Ranking Capability of the Hyperlink Based Search Engines Using Heuristic Approach

[1]Haider A. Ramadhan, [1]Khalil Shihab and [2]Jafar H. Ali
[1]Computer Science Department, Sultan Qaboos University, Oman
[2]Department of Information Systems, Kuwait University, Kuwait

Abstract: To evaluate the informative content of a Web page, the Web structure has to be carefully analyzed. Hyperlink analysis, which is capable of measuring the potential information contained in a Web page with respect to the Web space, is gaining more attention. The links to and from Web pages are an important resource that has largely gone unused in existing search engines. Web pages differ from general text in that they posse's external and internal structure. The Web links between documents can provide useful information in finding pages for a given set of topics. Making use of the Web link information would allow the construction of more powerful tools for answering user queries. Google has been among the first search engines to utilize hyper links in page ranking. Still two main flaws in Google need to be tackled. First, all the backlinks to a page are assigned equal weights. Second, less content rich pages, such as intermediate and transient pages, are not differentiated from more content rich pages. To overcome these pitfalls, this paper proposes a heuristic based solution to differentiate the significance of various backlinks by assigning a different weight factor to them depending on their location in the directory tree of the Web space.

Key words: Ranking capability, web link information, search engine, heuristic based solution

INTRODUCTION

The Web is growing rapidly and as an important new medium for communication, it provides a tremendous amount of information related to a wide range of topics, hence continues to create new challenges for information retrieval. A search engine provides users with a mean to search for valuable information on the Web. Traditionally, Web search engines, which rely on keyword matching and frequency, visit the Web sites, fetch pages and analyze text information to build indexes. Typically, a user will be willing to look at only a few of these pages, usually the first 10-20 results[1,3,4,12]. Hence, relevancy of results has become among the main issues which need to be seriously addressed.

With the explosive growth in the amount of Internet information, the number of documents in the indices has been increasing by many orders of magnitude. In particular, the results returned for a query may contain several thousand, or even million, relevant Web pages. One of the problems of text-based search engines is that many Web pages among the returned results are low quality matches. It is also common practice for some developers to attempt to gain attention by taking measures meant to mislead automated search engines. This can include the additional of spurious keywords to trick a search service into listing a page as rating highly in a popular subject. How to select the highest quality Web pages for placement at the top of the return list is the main concern of search engine design.

Another problem for those designing search engines is that most users are not experts in information retrieval. The Web users asking the question may not have enough experience to format their query correctly. It is not always intuitively easy to formulate queries which can narrow the search to the precise area. Furthermore, regular users generally do not understand the search mechanisms. The document indices constructed by search engines are designed to be general and applicable to all[1]. If a user tries to narrow his or her search by including all senses as a key search term, it often results in irrelevant information being presented. On the other hand, if a user is skilled enough to formulate an appropriate query, most search engine will retrieve pages with adequate recall (relevancy percentage of the relevant pages retrieved among all possible relevant pages), but with poor precision (the ratio of relevant pages to the total number of pages retrieved). These disadvantages indicate that the performance of current search engines is far from satisfactory.

The main problem with most approaches is that they almost invariably evaluate a Web page in terms of its text information alone. They fail to take into account the Web structure, in particular the hyperlinks. With the exception of Google[7], this pitfall applies to most of the existing search engines, such as Altavista and Lycos[1]. The Internet structure of the hyperlink environment can be a rich source of information about the content of the

Corresponding Author: Haider A. Ramadhan, Computer Science Department, Sultan Qaboos University, Oman

environment. Analyzing the hyperlink structure of Web pages gives a way to improve the behavior of text-based search engines providing an effective method that can locate not only a set of relevant pages, but also relevant pages of the highest quality.

In order to evaluate the informative content of a Web page, the Web structure has to be carefully analyzed. Hyperlink analysis, which is capable of measuring the potential information contained in a Web page with respect to the Web space, is gaining more attention. The links to and from Web pages are an important resource that has relatively gone unused in existing search engines. Web pages differ from general text in that they posses external and internal structure. The Web types and links between documents can be useful information in finding pages for a given set of topics. Making use of the Web link information will allow the construction of more powerful tools for answering user queries. Generally speaking, in this approach, the search engine computes both the sum of incoming hyper links to the page (back links) and the sum of outgoing links. The relevancy of a page is then measured by normalizing these two sums. This approach was successfully implemented by the designers of the Google search engine[2,3,4,6,7].

The intuition behind this approach is that a page has high rank if it is referred to by many highly ranked pages. In particular, the creation of a hyper link by the author of a page represents an implicit endorsement of the page being linked or pointed to, hence by mining the collective judgment contained in the set of such endorsements, web users can gain a richer understanding of the relevance and quality of web pages and their content. Thus by counting links from all pages equally and by normalizing the number of links on a page, the ranking and the relevancy of a page can be more objectively measured.

Although being novel improvement over traditional text oriented search engines which heavily depend on recognizing weighted keywords found in a page, there are still two main problems associated with the way Google implemented link based ranking of pages. First, the system does not differentiates between various incoming links, hence all the back links are assigned equal weights regardless of the domain they come from or the level of the link directory in which they are found. Second, less content rich pages, such as intermediate and transient pages which mainly contain links and are designed to point users to other more relevant pages, are not differentiated from more content rich pages. This paper proposes an alternative solution to improve the ranking capability of Google. The solution uses a heuristic approach which is based on the application of differential weights to back links. The paper also discusses the design and implementation of an improved prototype link-based search engine.

The types of links a Web site may contain have been fully studied by researchers[1,9,12]. Hypertext links within a Web site can be upward in the file hierarchy, downward, or crosswise. The links pointing to other sites are referred to as outward links and can help identify the type of a Web page. For example, a page which contains many outward links typically is a topic index Web page, while a page which also contains many links but most of them downward is a institution homepage. In types of sites, such as Yahoo, most of the links are downward links to subcategories or outward links. Furthermore, we can infer other information about a page from the number of links to it and from it. For example, we might guess a page to be popular if it has more links toward it than from it. Furthermore, More incoming outward links should indicate high popularity of the page and hence would show explicit interest in the page from various different domains.

Ranking algorithms, when applied to the large number of results returned by the existing search engines such as Altavista and Excite[1], can then help users to select those of most valuable to them from the Web resources. In practice, given a Web page p and the user query q, the ranking algorithm will compute a score rank (p, q). The higher ranked (p, q) is, the more valuable a Web page p is likely to be for the query q.

Google, a search engine with a full text and hyperlink database, is designed to crawl and index the Web efficiently and return much more satisfying search results than existing systems[4]. It makes use of the link structure of the Web to calculate a quality ranking for each Web. The rank algorithm used by Google is PageRank[6,7,11]. PageRank extends the idea that the importance as quality of an academic publication can be evaluated by its citations to pages on the Web, which can be similarly be evaluated by counting back links.

To provide some insights into the method followed by the PageRank algorithm of Google, here is how the ranking value PR of a page A is measured:

$$PR(A)=(1-d)+d\ (PR(T1)/C(A)+\ \ldots+\ PR(Tn)/C(A))$$

Where T1…Tn are pages pointing to page A, hence representing backlinks. The parameter d is a damping factor which is scaled between 0 and 1 and C(A) is the number of links leaving page A, hence representing outgoing links. The rank of page A or PR(A) can be calculated using a simple iterative algorithm and corresponds to the principal eigenvector of the normalized link matrix of the Web. As shown by the formula, the rank metric recursively defines the relevance of page A to be the weighted sum of its backlinks. The PageRank method maintains a tree structured directory of all the backlinks. A backlink can be of the following three types: (1) descendant (downward link) of a page being ranked, (2) ancestor (upward link) of the page and (3) an outsider or outward (not in the same domain of the page being ranked). Figure 1 below shows this structure.

Google assigns a score of 1 to each and every backlink. This approach poses two implications (1) local and global interests in a page are weighted equally and (2) transient pages with less rich content receive higher ranking.

Besides Google, another ranking system called Clever[2] was designed to improve the performance of the current search engines by using the Hyperlink Induced Topic Search algorithm. It can work with any existing text-based search engine and rearrange the returned results by applying its ranking algorithm. The system classifies all relevant pages returned results for a given query into two different categories: authority pages that contain rich information and hub pages that collect all the authority pages together. The advantage of the Clever system over Google is that it considers not only the in-degree but also the out-degree of a Web site. However, few disadvantages of the system have been cited[2], which include (1) mutually reinforcing relationship, e.g. two pages within a web site or even from two different web sites could have a lot of links pointing to each other and hence increase the authority and hub scores of those pages, (2) automatically generated links by programs or machines negatively affect the ranking score since such links offer no value judgment on the pages they point to and (3) some pages contain links pointing to irrelevant pages.

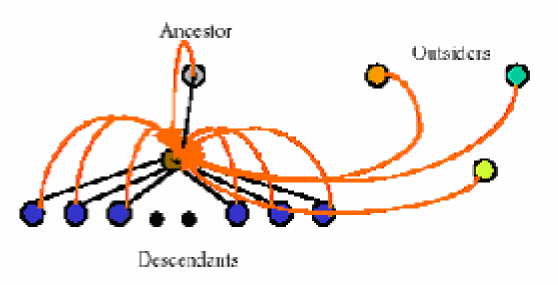


Fig. 1: Various backlinks

To get some more insights into these implications, we conducted a test which aimed at evaluating the relevancy of pages returned by Google in response to our queries. Results from the test have shown that some of the highest rankings were assigned to pages which are considered to be index pages of some archiving systems, or index pages containing links to some popular sites and hence are pointed at by many other pages (backlinks), but are not necessarily content rich pages which are of interest to us. Obviously, these pages are expected to have such high ranks since most pages in an archive system, for example, have a link back to their index page. In the second situation, pages with links to popular pages are also expected to have high number of backlinks since many users visit them during Web navigation. In both cases, results are regarded biased. Finally, it was found that pages with baclinks within the same domain (Web site) received higher rankings than pages which have outward backlinks from pages found in different domains. As a result, rankings produced by Google are generally regarded of low relevance by the users. The prelemiary experimantal evaluation reported in section 4 tends to support this claim.

To overcome these pitfalls, we propose a heuristic based solution to re-rank the results returned by a text-based search engine. In this solution, we attempt to differentiate the significance of various back links by assigning a different weight factor to these various back links depending on the location of the link in the directory tree. In this way, it is hoped that the element of bias in the current ranking process would diminish. The three heuristics we use are shown below. More discussion on the nature of these heuristics is given in section 3.3.

h1: a page with outward backlink (from different domain) should receive a score of 1

h2: a page with downward backlink should receive a score between 0.75

h3: a page with upward backlink should receive a score of 0.5

Design and implementation: Our prototype system, coined as DiffRank, consists of two parts, a ranking system consisting of a spider and a computational kernel and a local database system for saving and retrieving the ranking results. The ranking system is implemented using Visual Basic and is built on top of the existing text-based search engine AltaVista. The main purpose of this ranking system is to search for URLs and compute the score for ranking pages. It saves the URLs as well as their rank scores into a backend Access database.

Figure 2 shows the user interface. The interface provides two windows to list the root set URLs and the base set URLs. Users can watch the growth of the root set and base set while the spider program is running in the background. When a user inputs query strings into the keyword field of the interface, the system first sends them to a text-based search engine. All the URLs returned by the selected search engine make up the root set. For each page in the root set the spider parses the page to search for any existing links. The root set is expanded into the base set by adding newfound URLs referenced by pages of the root set. The constructed base set is also saved in the database. Links in the base set have tags indicating pages of the root set from which they were extracted. This helps in calculating the hub value for pages in the root set. If a page contains a repeated link it will be counted only once. Figure 2 shows the content of both the root and base sets when search is performed for the query “computer science”.

Development phases: There are two main phases in the development of this ranking system. The first is the search and growth phase. Here, the ranking system first constructs a collection of Web pages about a query

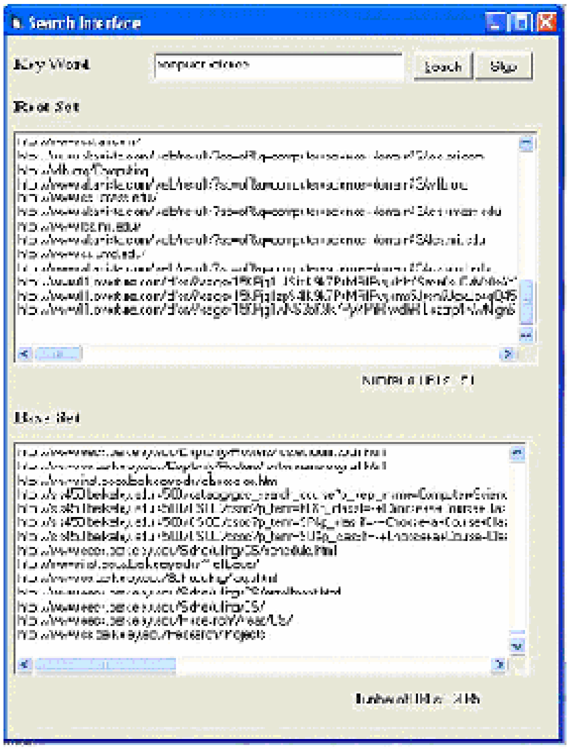


Fig. 2: Expansion of the root set into base set

string. Since the search results may contain millions of pages, the number of Web pages in the collection must be limited to a reasonable quantity so that the system can reach a compromise between obtaining a collection of pages highly relevant and saving computational effort. For constructing such a collection of pages, the ranking system makes use of the results given by a text-based search engine. The search engine will return a set of documents which are determined by its own scoring function as a root set. It then extends the root set, by adding any additional document that is pointed to by a document already in the root set. This is shown in Fig. 2. The new collection is then renamed the base set. In this way, the link structure analysis can be restricted to this base set, which is expected to be relatively small, rich in relevant pages and contains most of the strongest authorities.

The second is the weight and propagation phase, in which the results returned by the first stage are evaluated. Here, the ranking system calculates the rank score of each page based on the link structure between any node pairs in the base set and extracts good authorities and hubs from the overall collection of pages. It is worth noting that both these phases are considered typical phases found in all link-based ranking systems such as Google. The difference lies only in the way weights are assigned to the links based on their position in the base set directory.

Link analysis: The ranking system will rearrange the URLs in the root set by using the hyperlink information for all the Web pages in the base set. The process of analyzing hyperlinks is done twice. The first occurs while the spider performs the task of building the base set. Later, the spider will crawl all the newly found Web pages in the base set in order to perform the hyperlink analysis used for computing rank scores. A Web page can link to many other pages, which may in turn reference the Web page. When the spider crawls each Web page in the root set, it not only executes the task of extracting URLs in the Web page it is visiting but also registers the newfound URLs as outward links for that Web page. After the base set has been constructed, the spider walks through all the newfound Web pages, extracting the URLs of each visiting Web page again and comparing them with all existing URLs. If the outward link points to a URL which is in the base set, the URL in the base set is registered as its outward link and the Web page itself is registered as an inward link of the URL in the base set. In this case, the spider only cares about URLs that already exist in the base set. After walking through all the URLs in the base set, the hyperlinks among Web pages are recorded and saved for use in the ranking process.

Calculation of rank scores: Two things are considered when calculating the rank of a page (1) hub and authority values and (2) transverse and intrinsic links. Regarding hub and authority values, each page p in the base set has two values:
The authority value which is the number of pages pointing to p, X(p). It is normalized as:

$$\sum (X(p)^2) = 1$$

p € Bs

The hub value which is the number of pages that points to p, Y(p). It is normalized as:

$$\sum (Y(p)^2) = 1$$

p € Bs

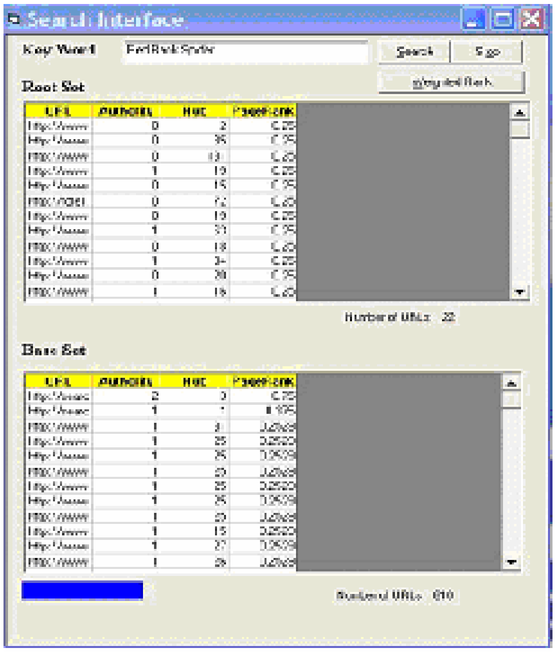
A good hub page is a page that points to good authority pages and a good authority page is the one pointed to by good hub pages. An alternative way to express the relationship between hub and authority is: if a page p points to many pages with high values of X, then p should get a high value of Y. Also if page p is pointed to by many pages with high Y values, then it should get high X value. According to this, the value of X and Y can be computed as follows: X(p) for a page p = sum of Y(q) overall pages q that point to p. In other words, $X(p) = \sum Y(q)$ such that q ➔ p and same with Y where: $Y(p) = \sum X(q)$ such that p ➔ q[9]. For each page p in our pool, DiffRank finds out X(p) which is the number of back links pointing to p and Y(p) which is the number of outward links to p. These values are then used to calculate the rank value for a page.

The other thing considered when calculating the rank of a page is the transverse and intrinsic links. The key that distinguishes these two types of links is their domain name where the domain name is the first level of the URL string associated with a page. The link is transverse if it is between pages with different domain name and it is intrinsic if it is between pages with the same domain name. The intrinsic links include different types of links that can exist between pages of the same domain such as: upward, downward and crosswise links whereas the transverse links are outward links. The intrinsic links exist usually for navigation purposes[10]. They contain less information than outward links. This is because the outward links convey information on the authority of the page it points to.

Google assigns equal weights to all the links regardless of their direction or type. The heuristic based approach suggested in this paper is reflected in the method of assigning weights to the hyperlinks when counting transverse and intrinsic links of a web page. Transverse links receive higher weights. For the intrinsic links, downward links receive higher weights than upwards links, since the child page is usually a content rich page on a certain topic where the main page mostly contains links to its child pages[10]. The important issue here is the selection of the weights. According to our heuristics, downward links get a weight of 0.75 while the upward links get 0.50. This may not be the most efficient weight assignment scheme. A more acceptable solution would be to rank intrinsic links through several cycles with successive increase in their weights and then take an average weight. For example, we can initially assign a weight of 0.75 to all downwards links found in the base set and come up with the rank scores for the pages. Next, we need to repeat the same process with the weight of 0.85 and calculate the page ranks. Finally, we repeat the process for the weight of 0.95 and then take the average of these three ranks for each page. The same applies to upward links. This method should provide us with more representative insights into the effectiveness of this differential weight assignment process. We hope to have this scheme incorporated in the next version of DiffRank.

After finding out the hub and authority values from the root and base sets (Fig. 3), our DiffRank ranking system uses heuristics to re-rank the results by assigning different weights to back links. Following is a brief account of the process. At the beginning links in the root set are assigned an initial weight of 0.50, since they were received directly from AltaVista and thus their parents are not known[11]. However, links in the base set were derived from root set, hence their parents are known and there is no need to use the initial value of 0.50.

For each URL in the base set, the DiffRank system compares its domain with that of its parent page, i.e. page in the root set that points to this page. If the child page is from a different domain, it will be assigned a weight of 1. As an example, if the child URL is http://www.amonline.net.au/factsheets/redback.htm and the parent URL is http://www.powerup.com.au/~glen/spider.htm, then the two domains are different, hence the child URL will receive a weight of 1. When the child and parent pages are from the same domain, the following three cases are considered: (1) the child URL is a downward back link and hence is assigned a weight of 0.75, (2) the parent URL is an upward back link and hence the child URL is assigned a weight of 0.50 and (3) both the parent and the child links are at the same level of hierarchy and hence the child URL gets a weight of 0.50.



Key Word: PageRank Spider

Root Set

URL	Authority	Hub	PageRank
[illegible]	0	2	0.25
[illegible]	0	35	0.75
[illegible]	0	13	0.20
[illegible]	1	19	0.25
[illegible]	0	15	0.75
[illegible]	0	72	0.20
[illegible]	0	19	0.25
[illegible]	1	31	0.75
[illegible]	0	18	0.20
[illegible]	1	34	0.25
[illegible]	0	38	0.75
[illegible]	1	18	0.20

Number of URLs: 22

Base Set

URL	Authority	Hub	PageRank
[illegible]	2	3	0.75
[illegible]	1	1	[illegible]
[illegible]	1	3	[illegible]
[illegible]	1	25	0.2520
[illegible]	1	25	[illegible]
[illegible]	1	20	[illegible]
[illegible]	1	25	0.2520
[illegible]	1	25	[illegible]
[illegible]	1	20	[illegible]
[illegible]	1	15	0.2520
[illegible]	1	27	[illegible]
[illegible]	1	26	[illegible]

Number of URLs: 810

Fig. 3: Hub, authority and initial rank values

Experimental evaluation: To assess the usefulness of our heuristic based ranking over that of Google, we are currently in the process of conducting several experiments. These experiments address the hypothesis that heuristic analysis of hyper links based on their location in the Web tree subdirectory tend to produce rankings which are more relevant to user query. In particular, these experiments attempt to evaluate the quality of rankings produced by Google and DiffRank. However to accomplish that, we need a test bed against which the two rankings are compared. We decided to use the explicit rankings provided by the users to serve as the benchmark. In other words, the explicit user ranking of the pages is used to represent a centeroid of the relevancy space. Rankings produced by the Google are compared against this cenetroid to measure the distance between the two rankings for the same set of the URLs. The same is done for the rankings produced by the DiffRank system which produces differential weights of the links based on their direction and domain. Next, we need to compare these two computed distances to get some insights into their relevancy to the user query. Shorter distance from the centeroid would imply less difference in the rankings of the URLs and hence high relevancy or better quality, whereas larger distances would indicate more difference and hence lower relevance or poor ranking quality. To achieve the above aims, we are in the process of running two main experiments. In the first one (reported below), rankings generated by Google for the top 20 URLs across five different queries are compared with the explicit user rankings of the same URLs. In the second experiment, we plan to compare the rankings produced by DiffRank with rankings generated by Google and the ones assigned by the users. We hope to complete the second experiment very soon.

MATERIALS AND METHODS

A total of five different queries and 10 users are used in this experiment, as shown by Table 6. Each user is asked to judge the relevancy of the first 20 top links ranked by Google against each of the five queries. It has been widely reported that users normally tend to look at only top 20 links returned by a search engine in response to a query, hence it was decided to focus on the first 20 URLs[1,3,4,12]. Each user looks at the page and assigns a score from 1 to 20, representing a ranking for that page in relevance to the query. This implies that each user ranks a total of 100 pages for 5 queries, hence resulting in a total of 1000 rankings produced by all 10 users. For each page, a total of 10 different scores ranging from 1-20 are given by the users. The average of these 10 scores is used to serve an overall explicit ranking for that page. Next, both the explicit rankings and that of Google for the 20 links are compared, differences to determine the degree of variations are computed and a simple statistical 2-tailed test is used to find out if the differences in rankings are significant or not.

For each of the 5 queries, shown in Table 6, we computed the percentage of the matches among the rankings produced by the users and Google, percentage of the mismatches, standard deviation in these two rankings, the mean of the difference between these rankings and finally the statistical significance of the difference between the two rankings. It is worth noting that all the users involved in this experiment were 4th and 5th year undergraduate Computer Science students who participated in the study on a voluntarily basis.

RESULTS

Tables 1 to 5 show how the users ranked the top 20 URLs returned by Google for the five queries shown in Table 6. In each table, the first column shows the top 20 URLs returned by Google, while the second one shows user rankings of the same URLs and the last column shows the difference in these two rankings for each and every URL. For example, in Table 1 we see that the URL ranked second by Google was regarded of low relevance by the users and hence received a ranking of 10 by them. The difference shows how far the two rankings for a URL are apart. The difference value is an absolute value. Larger difference values tend to indicate high degree of variations in two rankings, while small values tend to suggest high similarity level among any two URLs. Since we consider the explicit ranking as the benchmark, higher difference values would suggest that the users did not consider the quality of ranking produced by Google to be high and vice versa. Zero difference values indicate identical ranking.

Table 6 summarizes the overall analysis of all the five queries. For the first query, Q1, there is only one (5%) direct matching in the rankings, namely for the first URL returned by Google. The remaining 19 links (95%) received different rankings from the users. Standard Deviation (SD) of the two sets (Google vs. Explicit) is 7.182, hence indicating relatively large variations among the rankings of the corresponding URLs in the two set. The Mean of the Difference (MD) values among the two sets is 6.25. It is expected that high SD values would be associated with high MD values. From Table 1, this relationship is clear. High SD values suggest larger variations in the rankings for the corresponding URLs of the two sets. This would result in large difference values and hence large overall MD values. Hence, larger SD and MD values would imply that users did not regard the rankings returned by Google to have high relevance to the query, while smaller SD and MD values could imply that users considered rankings provided by Google to be of high relevance.

Among all the queries, as shown in Table 6, query 4 had the lowest SD and MD values, hence suggesting low variations in the two rankings of corresponding URLs. Table 4 supports this claim and shows that there are 5 (25%) direct matching among URLs in the two sets. Although the remaining 15 (75%) URLs were ranked differently by the users, the degree of variation in the two rankings is small, ranging from 1 to 5, compared to 1 to 11 for query 1 as shown in Table 1. This implies that users generally saw the rankings produced by Google to be of high quality and more relevant to the query. Query 5 produced some interesting user rankings. Similar to query 1, this query had only 1 (5%) direct matching and 19 (95%) mismatches. However, it has far more smaller SD and MD values. This is attributed to having smaller degree of variations between the rankings of Google and explicit. This outcome invites us to have more focus on SD and MD values rather than on the percentages of matches and mismatches. A query can have 100% mismatch in two rankings but still produce lower SD and MD values if variations between two rankings are small.

Finally, despite being clear in showing the variations between the two sets of rankings, the results reported here did not indicate any statistical

Google	Expl	Diff
1	1	0
2	10	8
3	2	1
4	9	5
5	3	2
6	17	11
7	15	8
8	19	11
9	20	11
10	18	8
11	5	6
12	6	6
13	4	9
14	8	6
15	7	8
16	12	4
17	13	5
18	16	2
19	14	5
20	11	9

Table 1: Query 1

Google	Expl	Diff
1	1	0
2	2	0
3	3	0
4	7	3
5	19	14
6	4	2
7	5	2
8	6	2
9	8	1
10	9	1
11	18	7
12	20	8
13	10	3
14	11	3
15	12	3
16	13	3
17	14	3
18	17	1
19	15	4
20	16	4

Table 2: Query 2

Google	Expl	Diff
1	1	0
2	3	1
3	6	3
4	4	0
5	5	0
6	2	4
7	14	7
8	15	7
9	20	11
10	19	9
11	7	6
12	8	4
13	11	2
14	10	4
15	9	6
16	12	4
17	18	1
18	13	5
19	16	3
20	17	3

Table 3: Query 3

Google	Expl	Diff
1	1	0
2	2	0
3	3	0
4	4	0
5	5	0
6	9	3
7	10	3
8	12	4
9	6	3
10	7	3
11	8	3
12	16	4
13	12	1
14	13	1
15	14	1
16	20	4
17	19	2
18	17	1
19	18	1
20	15	5

Table 4: Query 4

Google	Expl	Diff
1	2	1
2	4	2
3	5	2
4	1	3
5	3	2
6	6	0
7	8	1
8	7	1
9	15	6
10	11	1
11	13	2
12	9	3
13	10	3
14	18	4
15	12	3
16	20	4
17	14	3
18	19	1
19	16	3
20	17	3

Table 5: Query 5

Table 6: Summary of overall analysis over five queries and ten users

Query	Matches	Mismatch	Std Dev	Diff Mean	t-test
Q1: Banks in Oman	1 (5%)	19 (95%)	7.182	6.25	.456
Q2: Oman News Agency	3 (15%)	17 (85%)	4.646	3.20	.187
Q3: Oman Chamber of Commerce	3 (15%)	17 (85%)	4.995	4.00	.382
Q4: Educational Technology Portals	5 (25%)	15 (75%)	2.585	1.95	.436
Q5: Textual Version of World News	1 (5%)	19 (95%)	2.828	2.40	.401

significance in the variations among any two sets of the five queries. With p-value = .187, as shown by Table 6, query 2 is the only query which came close to having significant variations among corresponding URLs of the two sets. This implies that despite having differences in the two rankings for all the five queries, rankings assigned manually by the users are not that far from the ones generated by Google. Therefore, the overall level of the ranking quality of Google does seem to come close to what the users consider to be of high relevance. It still remains to be seen how the results of rankings produced by DiffRank stand when compared against that of Google.

CONCLUSION

In this study we have proposed an alternative solution to improve the ranking capability of Google. The solution uses a heuristic approach which is based on the application of differential weights to incoming (back) links into a page depending on the location of the link in the Web directory space. The proposed approach was implemented using a modified PageRank algorithm called DiffRank. The prototype system incorporating DiffRank algorithm has been successfully developed. To get some insights into the superiority of proposed method, we designed two main experiments. The first one, reported in this paper, attempts to assess the quality of ranking produced by Google across five different queries. This is achieved by asking ten different users to explicitly re-rank the top 20 URLs returned by Google for each query. The user ranking is regarded as a benchmark against which rankings of Google are compared. The second experiment, still under investigation, compares the performance of DiffRank against that of Google. In addition, it also evaluates the quality of ranking produced by DiffRank against the user-based benchmark mentioned above. The results of the experiment reported here have shown that across all five queries, results of the top 20 rankings of Google rarely match the rankings specified in the benchmark. However, differences in variations are not significantly large, hence implying that the overall quality of Google's rankings is in fact considered by the users to be of reasonably high relevance.

REFERENCES

1. http://www.lib.berkeley.edu/TeachingLib/ Guides/Internet/ SearchEngines.html (Retrieved: June 2005)
2. Monika, R.H., 2001. Hyperlink analysis for the Web. IEEE Internet Computing, 2001 URL: http://maya.cs.depaul.edu/~classes/ds575/papers/hyperlink.pdf
3. http://www.ecsl.cs.sunysb.edu/ ~chiueh/ cse646 /cn4/ cn4.html (Retrieved: June 2005)
4. Han, J. and K. C.-C. Chang, 2002. Data mining for web intelligence. IEEE Computer, IEEE Computer Society, Washington, 2002. URL:http://www-faculty.cs.uiuc.edu/~kcchang/Papers/dmweb-ieeecomputer02.pdf.

5. http://www.belllabs.com/ user /minos/ Papers/ widm99.pdf (Retrieved: June 2005)
6. http://pr.efactory.de/e-pagerank-algorithm.shtml (Retrieved: June 2005)
7. Sergey, B. and L. Page, (year?). The anatomy of a large-scale hypertextual web search engine. Stanford University Computer Science Department, Stanford, CA 94305, USA. URL: http://www-db.stanford.edu/~backrub/google.html
8. Ashraf, K. and Y. Liu, 2004. Experiments with pagerank computation. Indiana University, Department of Computer Science. URL: http://www.cs.indiana.edu/~akhalil/Papers/pageRank.pdf
9. Ozsoyoglu, G. and A. Al-Hamdani, 2003. Web information resource discovery: Past, present and future. invited paper, ISCIS, 2003.
10. Ramadhan, H. and K. Shihab, 2000. Improving the engineering of web search engines. Intl. Conf. Internet Computing, USA, pp: 29-35.

11 http://www.webworkshop.net/pagerank.html (Retrieved: June 2005)

12. Brian, S. and L. Page, 1998. The anatomy of a large-scale hyper textual Web search engine. Computer Networks, 30: 107-117.