Toward an Interactive Internet Search Engine

By

Imad M. Akl

A Thesis

Submitted in Partial Fulfillment of the
Requirements for the Degree of Master of
Science in Computer Science

Department of Computer Science
Faculty of Natural and Applied Sciences
Notre Dame University – Louaize
Zouk Mosbeh, Lebanon

June 2002

Toward an Interactive Internet Search Engine

By

Imad Akl

Approved:

Fouad Chedid

Fouad Chedid: Associate Professor of Computer Science.
Advisor.

Omar Rifi: Assistant Professor of Computer Science.
Member of committee.

Khladoun el Khaldi: Assistant Professor of Computer Science.
Member of Committee.

Bassem Ghalayini: Assistant Professor of Mathematics.
Member of committee.

Date of thesis Defense: June 18th, 2002

ACKNOWLEDGEMENTS

Many thanks to Dr. Fouad Chedid whose support, assistance and guidance have made this thesis more than a duty to be achieved, in fact a new horizon in research methods.

I am forever grateful to my parents, whose foresight and values paved the way for a privileged education and to my brothers who gently offer counsel and unconditional support at each turn of the road.

ABSTRACT

This thesis uses the HITS algorithm as a basis to propose an interactive internet search engine. With the Web becoming a major source of information for many users, it became a necessity to be able to search the Web efficiently. The main problem resides with broad topic queries. These are queries for which a typical text-based search engine like Altavista would return thousands of pages. A remedy for these situations was proposed by Kleinberg in his HITS algorithm in which he uses the hyperlink structure of the Web as a major source of information about the contents of the Web.

In this thesis, we experiment with the HITS algorithm using the keyword "cancer" as a broad topic. We report on the performance of HITS for different parameter values.

The main contribution of this thesis is an attempt to create an interactive search engine that allows the user to specify the rank of each page in the root set. Our results show a significant improvement over the results reported by HITS.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

CHAPTER I

THE PROBLEM

1.1 Introduction

The Web can be regarded as a huge and complex mass of interconnected, or hyperlinked information sources. The past few years have witnessed a dramatic and unplanned increase in the amount of information accessible from the Web. People seeking information have to browse specialized search engines in order to get Web pages relevant to their topic. Unfortunately, most of these search engines are text-based, which lead sometimes to poor result accuracy. One weak point in text-based search engines is that the content of the page may not be self-descriptive like pages containing only pictures. Adding to that, we can face information overload with broad topics. A topic like cancer, for example, may have millions of results, but in fact relevant ones are usually hundreds – relevancy is a subjective thing. Also clever programmers can fool this software by inserting keywords that are not relevant to the topic.

To face this problem, a way has to be found to filter the results in order to get the most relevant pages.

1.2 Link-based Search Engine

Link-based search engines are proposed as a way to deal with the weaknesses described in the previous section. A collection of hyperlinked pages can be viewed as a directed graph where the nodes correspond to the pages and the edges are the links between them. For example, there is an edge from node p to node q when page p links to page q. Disregarding the navigational links and those created for advertising reasons, links have a lot to say about the content of the linked pages. Hyperlinks encode a considerable amount of latent human judgment, [3] therefore the creator of page p, by including a link to page q, has in some measure conferred authority on q. Kleinberg calls a page having a lot of outlinks as a hub page and one that has a lot of in links as an authority page. [3] Kleinberg introduced the HITS (hyperlink-induced topic search) algorithm to return the most authoritative and hub pages for any sufficiently broad topic. HITS is centered on the mutually reinforcing relationship between hub and authoritative pages: “A good hub points to many good authorities; a good authority is pointed to by many good hubs”.

The rest of this thesis is organized as follows: chapter II describes the HITS algorithm, reviews related mathematical background and describe three other algorithms: ARC clever, Page rank citation and Google. Chapter III describes our implementation of the HITS algorithm. Chapter IV reports on a number of experiments related to parameters that affect the performance of HITS. Chapter v, introduces the idea of interactive search engines. Chapter VI is a conclusion.

CHAPTER II

LITERATURE REVIEW

2.1 The HITS Algorithm

Studying and analyzing the structure of the Web can lead to amazing results. The structure of the Web, related to a specific topic can in a way or another be a very rich and informative source of data. All these positive points are condition to good and right understanding of the link structure. The HITS algorithm relies a lot on the use of links to analyze the structure of the web and to get the most "authoritative" pages on such topics.

While searching in broad topics, a text based search engine returns sometimes millions of pages. The difficulty itself exists in the abundance problem:[3] "The number of pages that could reasonably be returned as relevant is far too large for a human user to digest". The best solution is to filter results or to narrow the user demand from the first step.[2]

Some other weaknesses and complications are in order: Text based search engines also assume that any relevant page to the topic will mention the keywords right from the start. Therefore, words appearing near the top of a web page, such as in the headline or in the first few paragraphs are given more weight than those appearing further in the text. Another major factor in determining the relevancy of a page is the frequency of keywords. A search engine will analyze how often keywords in relation to other words appear within the same page. Pages with higher frequency are assumed to be more relevant than other pages. All text-based search engines use this location - frequency method, but each has its own added specifications. For this reason, we find that search engines differ in their results for the same topic. Add to this the fact that no search engine has the exact same collection of pages to search through.

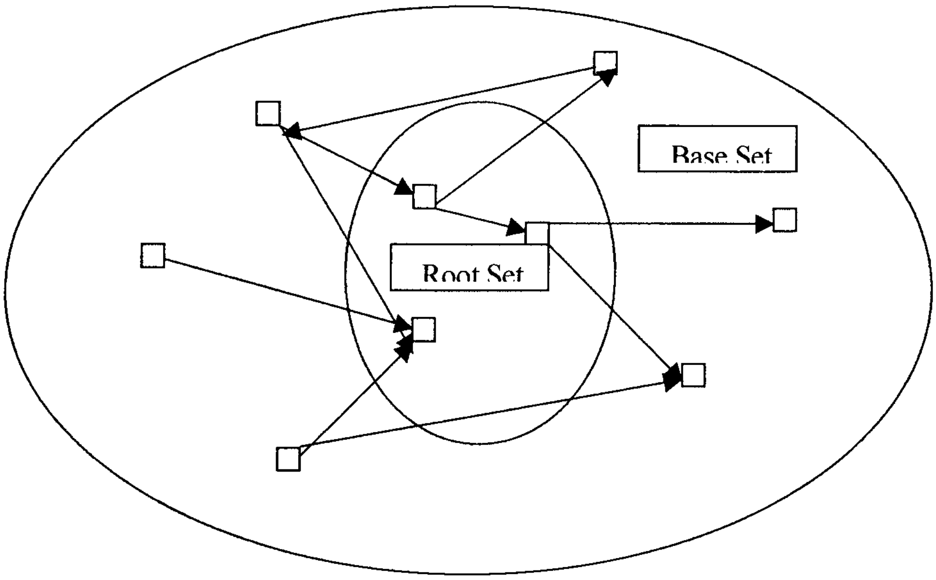


Fig 2.1 Interaction between pages in the root and base set

In analyzing the link structure, we found that hyperlinks encode a considerable amount of latent human judgment, and we claim that this type of judgment is precisely what is needed to formulate a notion of authority. Another issue is the distinction between the criteria of relevance and popularity. For example, www.yahoo.com seems to be very authoritative due to its high number of inlinks.

In our work, we follow Kleinberg's link-based model for the conferral of authority, and show how it leads to a method that consistently identifies relevant, authoritative WWW pages for broad search topics.

The HITS algorithm mentioned focuses on a collection of pages with the following properties:

- Should be relatively small to afford computational cost by applying non-trivial algorithms.
- Should be rich in relevant pages
- Should contain most of the strongest authorities

2.1.1 Constructing the Root and Base sets

Start with any broad topic and use a text-based search engine to collect the highest ranked pages. The highest ranked pages will form the root set of the algorithm. In our example, we will limit the root set to 20 pages(however in kleinberg it was set to 200 pages).

Such a root set satisfies the first two conditions set in the previous section. Next, we can increase the number nodes by expanding the root set along the links that enter and leave it.

Before proceeding with the calculation of hubs and authorities, the *HITS* algorithm defines two simple heuristics to clean up the links:

1. Delete all intrinsic links (navigational links)
2. When a large number of pages from a single domain point to a single page, this corresponds to a mass endorsement, advertisement, or some other type of "collusion" among the referring pages, for ex: "this site designed by ... "

   The HITS algorithm allows up to m pages from a single domain to point to any given page p.

2.1.2 Computing Hubs and Authorities

The HITS algorithm is based on the relationship between hubs and authorities, so with each page p is associated a hub-weight h(p), and an authority weight a(p). The sets of authority and hubs weights are placed in two distinct vectors **a** and **h**. The components of both vectors are initialized to 1. The algorithm runs K iterations during which it replaces a(p) by the sum of hub weights of pages pointing to p, and replaces h(p) by the sum of the authority weights pointed to by p. This process is based on the idea that "a good hub page points to many good authorities and a good authority is pointed to by many good hubs." With the increased number of iterations, results will begin to converge. This convergence depends on whether the topic is "wired" or not. The more "wired" is the topic, the faster is the convergence.

The pseudo-code of the HITS algorithm as it appears in Kleinberg's paper is:[3]

```
Iterate(G,k)
Begin
        G: a collection of n linked pages
        K: a natural number
        Let z denote the vector (1,1,1 ...1) E $R^n$
        a:=z //initializing the authority weights
        h:=z //initializing the hub weights
For I:=1 to k do
        For I:=1 to n do
                A[i] := $\sum$ h[j], where the link (j,i) exists and j $\neq$ i
        For I:=1 to n do
                H[I]:= $\sum$ a[j], where the link (i,j) exists and j $\neq$ I
        Normalize a.
        Normalize h.
Return (a,h)
End.
```

2.2 Mathematical Background

The results returned by the HITS algorithm are related to the eigenvectors of some matrices described later in this section.[6]

Let A be a nxn matrix. The number λ is an eigenvalue of A if there exists a nonzero vector v such that: $Av = \lambda v$, where v is called an eigenvector of A corresponding to λ. The eigenspace of A corresponding to λ is the set of all vectors satisfying $Av=\lambda v$.

Note that the eigenvalues and the eigenvectors can be of complex numbers as well as real numbers.

The equation $Av=\lambda v$ can be rewritten by moving its right side to the left and factoring out the vector v as: $(A-\lambda I)v = 0$, where I is the nxn identity matrix.

By multiplying this equation from both sides by the inverse of $(A-\lambda I)$, we get:

$((A-\lambda I)^{-1})(A-\lambda I)v = ((A-\lambda I)^{-1})0 \Leftrightarrow Iv = 0 \Leftrightarrow v = 0$

Which contradicts the definition that v must be a non-zero vector. Thus (A-λI) must not be invertible, or in other words the determinant of (A-λI) must equal 0.

We notice three facts about eigenvalues.

- The dimension of the eigenspace corresponding to an eigenvalue is less than or equal to the geometric multiplicity of that eigenvalue.
- The techniques for computing eigenvalues and eigenvectors used above are practical for 2x2 or 3x3 matrices. Eigenvalues and eigenvectors of larger matrices are often found using other techniques, such as iterative methods.
- Matrices A^TA and AA^T have the same eigenvalues.

2.3 ARC -CLEVER

ARC, Automatic resource compilation is a variation of the HITS algorithm enhanced with textual analysis implemented in the context of the clever project by a group of researchers of IBM Alamdin Research Center.[7,5,10]

The structure of ARC is identical to HITS, the main difference is in the weighing process of the hubs and authorities. HITS works solely on the link topology while ARC introduces the idea of analyzing the textual content in the vicinity of the link, called the anchor window, which is of B bytes wide to both sides of the anchor tag (href=" ...").

If we let x denote the number of matches between terms in T in the anchor window of the link (p,q), then we set the entry in the adjacency matrix for (p,q) = 1+ x.

2.4 The Page Rank Citation Ranking Algorithm

This algorithm was developed at Standford University under the WebBase project, which eventually resulted in the famous Google search engine. It builds up on the HITS algorithm and can be considered as yet another variation of HITS. [7]

The main idea is that simple citation counting does not relate well to the human common sense of importance. Attempts to formulate importance in this case resulted in a simple heuristic: A page has high rank if the sum of the ranks of its inlinks is high.

The algorithm outline is as follows:

Given a URL u, let Fu be the set of pages that u points to, and Bu the set of pages that point to u, also Nu=|Fu| the cardinality of Fu and c<1 a normalization factor. Then a simple formulation of the rank of u would be c multiplied by the sum of the parent ranks each divided by its N, i.e. dividing a page rank evenly into its forward links.

2.5 The Google Search Algorithm

Nowadays, Google is known as the best and most successful search engine. In fact, it is a combination of textual and link structures in a more accurate and sophisticated way compared to ARC. Google uses the PageRank function, which relies on the "democratic nature of the web" . A link from page A to page B is considered a vote for page B, by page A.[8]

Google do not restrict its ranking to the number of votes, it also analyses the page that issues the vote. If, for example, we have a vote from an important page, this vote weights heavily in google. Important sites receive a higher PageRank, which is remembered by google. Google combination of PageRank with sophisticated text-matching goes beyond the location / frequency method of traditional text-based search engines and examines all aspects of the page's content in order to determine if it is a good match for the query.

As the HITS algorithm relies on the concept of "Authority", the PageRank concept relies a lot on the notion of " Importance".

2.6 Conclusion

This chapter presented a brief overview of text-based search engines and link-based search engines. In the next chapter, we will describe our implementation of HITS algorithm.

CHAPTER III

Implementation of HITS

In order to implement the HITS algorithm, we have to work on the following: issues, crawling, storing and computing.

3.1 The Structure of the HITS Algorithm

In crawling, we collect Web pages that constitute the root set and later the base set. After we store the pages in tables for future processing using specific code. We choose to work on a wide topic such as "cancer". Using AltaVista as a text-based search engine, it returns 4,639,759 web pages.

First, we created three access tables (pages, links, weights) of the following form:

Field name	**Type**	**Size**	**Primary key**
Url Id	Number	Integer	Primary
Url	Text	50	None
Source	Memo	Unlimited	None
Rankings	Number	Long integer	None
Epsilon	Number	Double	None

Table 3.1 Pages table

Field Name	**Type**	**Size**	**Primary key**
From	Number	Long integer	Primary
To	Number	Long integer	None

Table 3.2 Links Table

Field Name	**Type**	**Decimal places**	**Primary Key**
Urlid	Number	Auto	Primary
Aweight	Double	15	None
Hweight	Double	15	None

Table 3.3 Weights table

The relationship between the three tables is as follows:

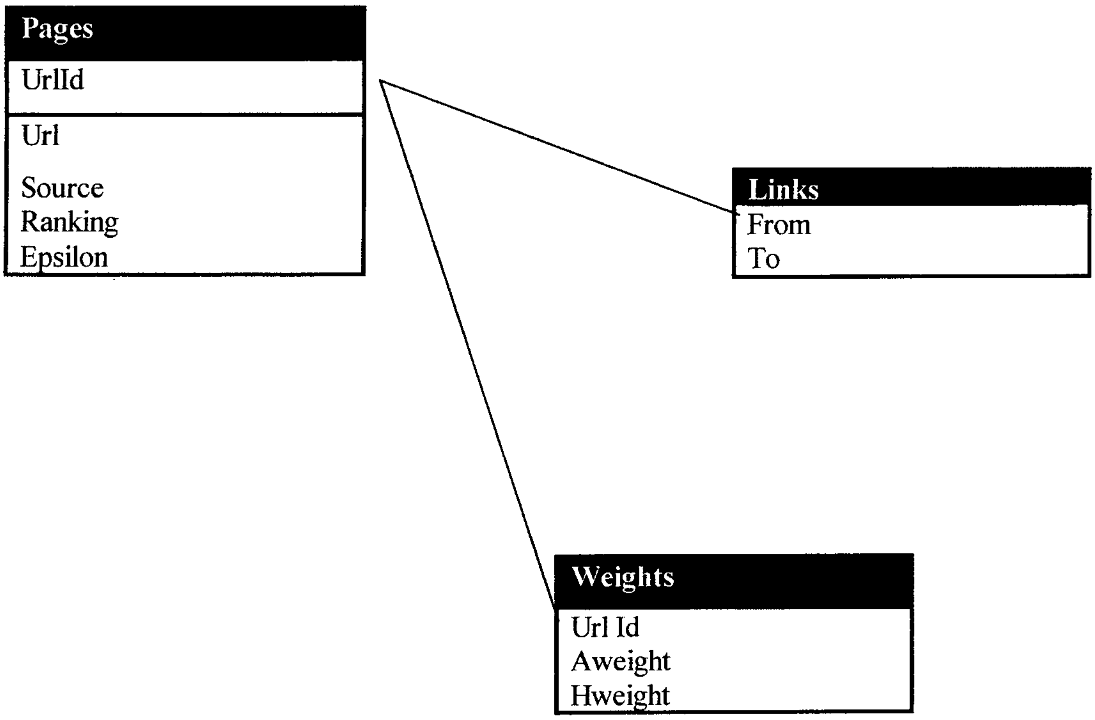


Fig 3.1 Relationship between the tables

3.2 Details About the Base Set

The HITS algorithm starts with a root set of 200 pages, mainly The ones returned by the text-based engine AltaVista. However, due to our inability to gain access to a database that indexes the web, we had to start with much smaller root set of size 20, and the data will be collected by hand. To expand this process we are going to add all the inlinks and the outlinks for the first 20 ranked pages. We mean by inlinks the pages pointing to that page whereas outlinks the pages pointed to by this page. To find the outlinks, we browse each page separately, then we save all the links to external pages; note that we should delete all navigational and commercial links. For the inlinks we work with the google search engine using the protocol “link”. We save these data in order to build our database later. Our page table is now expanded from 20 pages (the root set) to a larger one, the base set.

3.3 HITS Main Functions

Second, we have to fill the link table, in order to know which source page is linking to which target page. We implemented the following procedures: For more result accuracy we need to filter the url of the pages collected previously. Let us say we have at first: http://dmoz.org/health/. After running the function “cleanstr”, it becomes: dmoz.org/health. A small code takes the new URL of the first page and search the source text of all pages in the base set one by one. Whenever there are matches the “ URL value of the first page is inserted in the “From” and the URLID of the page searching in it in the “To”.

To run this code and manipulate its different functions, we used Access and it is as follows:

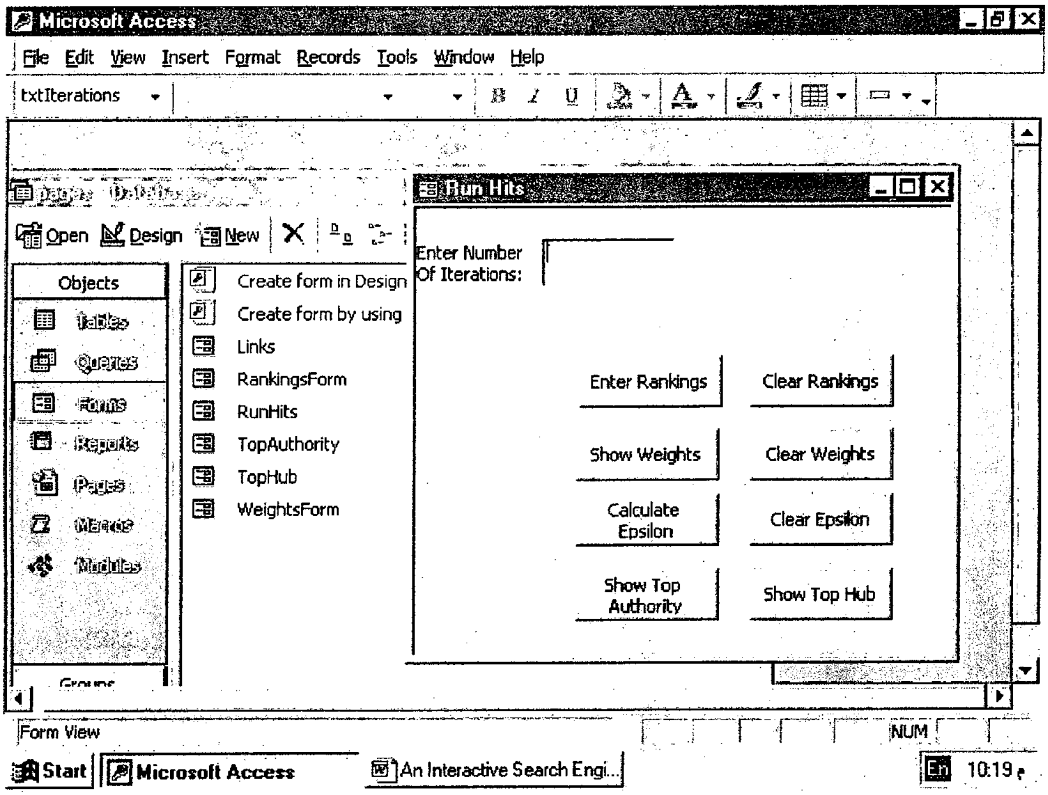


Fig 3.2 Interface of *HITS* algorithm

The code of HITS algorithm consists of the following functions

- Iterate_operation
- I-operation
- O_operation
- Normalize
- Update_authority_weight
- Update_hub_weight
- Clear_hweight
- Clear_aweight

The table weight contains the number of rows, which is equal to the number of pages in the base set, adding to that all rows are initialized to 1. The table weight has the following form.

URLID	Aweight	Hweight
5	1	1
19	1	1

Table 3.4 Initial Weights table

I. The function Iterate_operation is the main function. It consists of:

- Open page "pages"
- Loop on all fields

  Call I-operation
- Loop on all fields

  Call O-operation
- Normalize

This function iterates once. Using a query we can iterate as much as we want.

II. To get the authority weights, we apply the following rules.

A.**h** = **a**

where A stands for Adjacency Matrix, **h** for Hub Vector and **a** Authority vector

See appendix B for details.

III. After we normalize the values to avoid large number.

Using a mathematical formula new value = old value/ square root (v1 x v1 + V2 x v2)

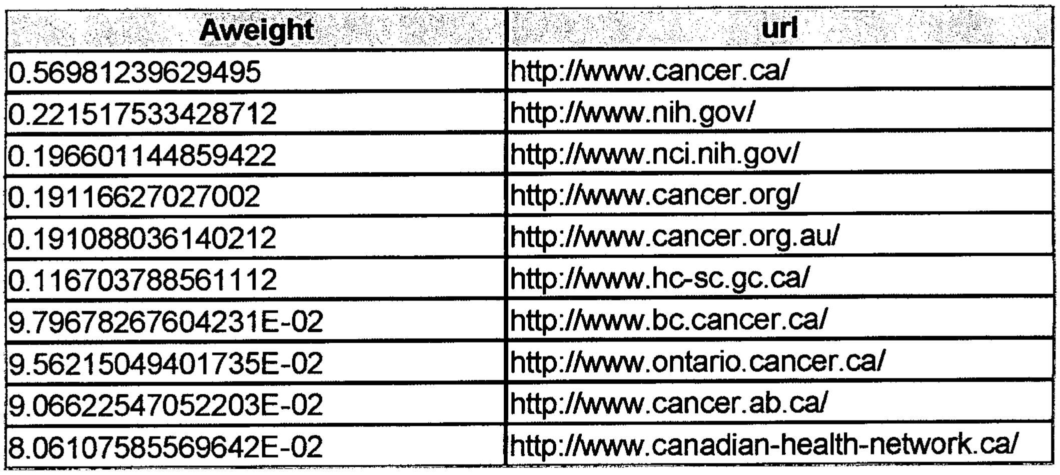
3.4 Running HITS

For the "cancer" topic, AltaVista returns 4,639,759 pages. This huge number of relevant pages reflects the breadth of the topic. The first 20 pages of this set formed the root set we began with. Here are the pages in the order they appeared in, at the time we ran our test in January 2002

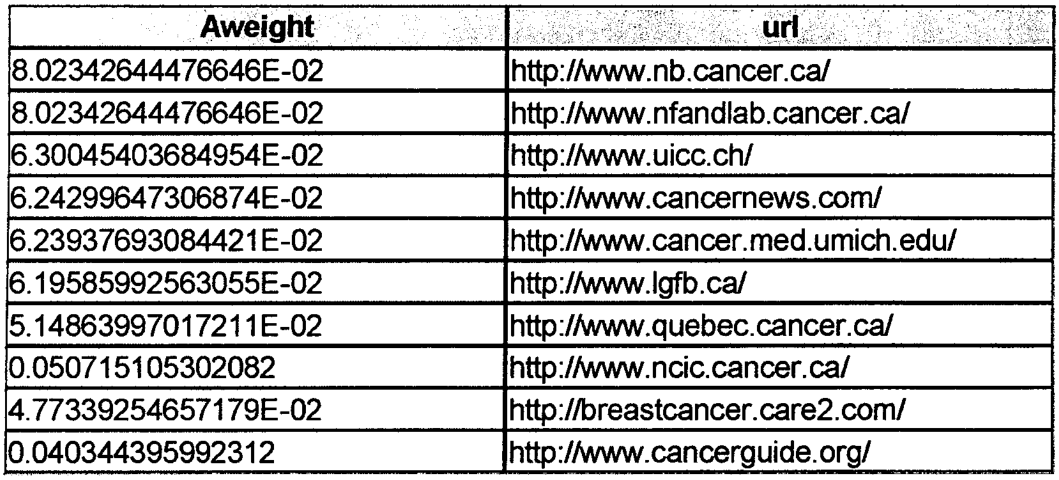
Order	URL of the pages
1	http://www.cancer.ca/
2	http://www.cancercharities.com/
3	http://breastcancer.care2.com/
4	http://www.lovetest.com/astromate/cancer.html
5	http://www3.cancer.gov/cancercenters/
6	http://www.omega23.com/books/med/chemo2.html
7	http://www.ignio.com/e/daily/tod/cancer.html
8	http://www.prostatecancerclimb.org/
9	http://www.cancer.se/
10	http://www.prostatecancer-tests.com/
11	http://www.endcancernow.com/
12	http://www.cancerdirectory.com/
13	http://www.cancer.ie/
14	http://www.cancer.ab.ca/links/index.htm
15	http://www.crc.org.uk/cancer/Aboutcan_common3.html
16	http://www.nabco.org/
17	http://www.cancer.org.au/
18	http://a-ten.com/books/cancer/index.html
19	http://www.cancer.gov/dictionary/
20	http://www.acc-vkb.be/index.cfm

Table 3.5 Root Set for the Cancer Query

This root set includes many relevant pages. After adding the needed inlinks and outlinks, the cardinality of the base set becomes 150. Running the HITS algorithm on this set has produced the following authorities:



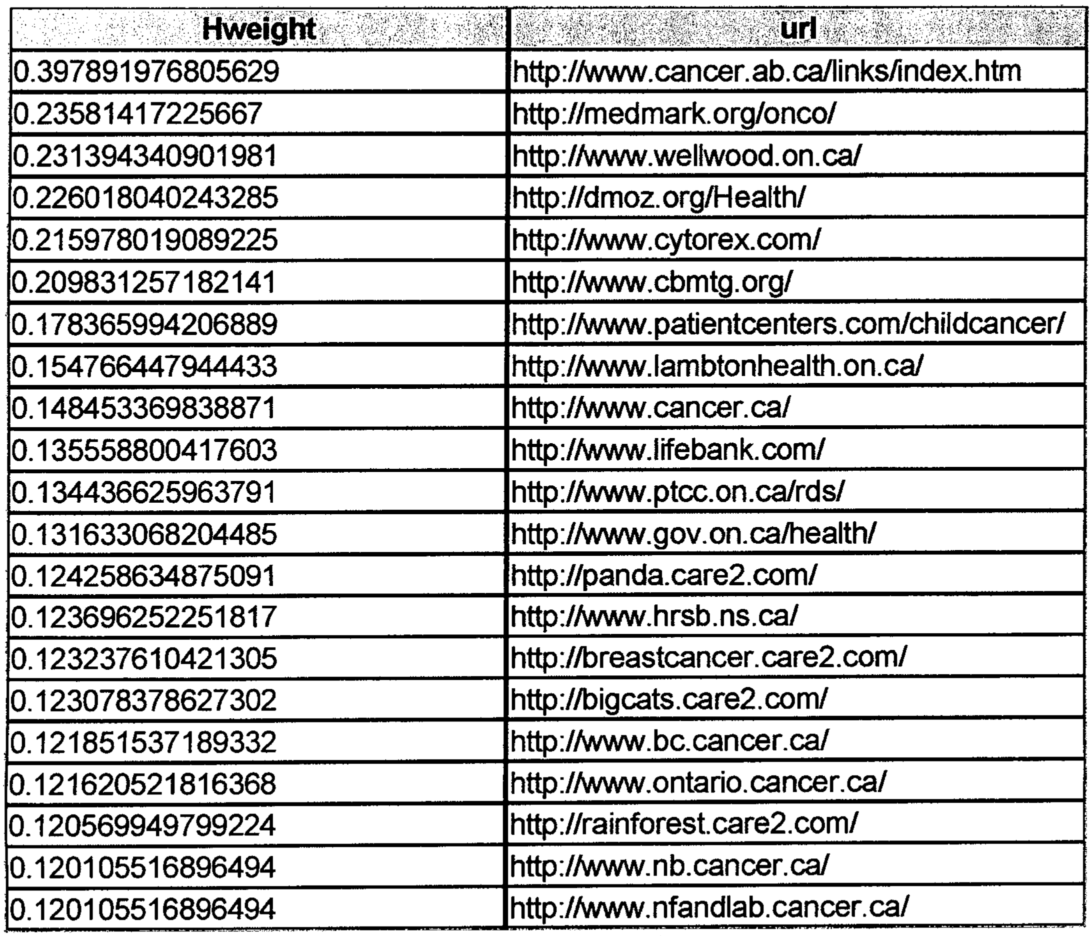
Aweight	url
0.56981239629495	http://www.cancer.ca/
0.221517533428712	http://www.nih.gov/
0.196601144859422	http://www.nci.nih.gov/
0.19116627027002	http://www.cancer.org/
0.191088036140212	http://www.cancer.org.au/
0.116703788561112	http://www.hc-sc.gc.ca/
9.79678267604231E-02	http://www.bc.cancer.ca/
9.56215049401735E-02	http://www.ontario.cancer.ca/
9.06622547052203E-02	http://www.cancer.ab.ca/
8.06107585569642E-02	http://www.canadian-health-network.ca/



Aweight	url
8.02342644476646E-02	http://www.nb.cancer.ca/
8.02342644476646E-02	http://www.nfandlab.cancer.ca/
6.30045403684954E-02	http://www.uicc.ch/
6.24299647306874E-02	http://www.cancernews.com/
6.23937693084421E-02	http://www.cancer.med.umich.edu/
6.19585992563055E-02	http://www.lgfb.ca/
5.14863997017211E-02	http://www.quebec.cancer.ca/
0.050715105302082	http://www.ncic.cancer.ca/
4.77339254657179E-02	http://breastcancer.care2.com/
0.040344395992312	http://www.cancerguide.org/

Fig 3.3 Authorities for the Cancer Query

The top Hubs are as follows:



Hweight	url
0.397891976805629	http://www.cancer.ab.ca/links/index.htm
0.23581417225667	http://medmark.org/onco/
0.231394340901981	http://www.wellwood.on.ca/
0.226018040243285	http://dmoz.org/Health/
0.215978019089225	http://www.cytorex.com/
0.209831257182141	http://www.cbmtg.org/
0.178365994206889	http://www.patientcenters.com/childcancer/
0.154766447944433	http://www.lambtonhealth.on.ca/
0.148453369838871	http://www.cancer.ca/
0.135558800417603	http://www.lifebank.com/
0.134436625963791	http://www.ptcc.on.ca/rds/
0.131633068204485	http://www.gov.on.ca/health/
0.124258634875091	http://panda.care2.com/
0.123696252251817	http://www.hrsb.ns.ca/
0.123237610421305	http://breastcancer.care2.com/
0.123078378627302	http://bigcats.care2.com/
0.121851537189332	http://www.bc.cancer.ca/
0.121620521816368	http://www.ontario.cancer.ca/
0.120569949799224	http://rainforest.care2.com/
0.120105516896494	http://www.nb.cancer.ca/
0.120105516896494	http://www.nfandlab.cancer.ca/

Fig 3.4 Hubs for the Cancer Query

Notice that, if we compare Table 3.5 to Figure 3.3, we will find that the first page returned by AltaVista is the same returned by HITS. However, the second page in the root set appears in the 19th position in the authorities list.

3.5 Unrecognized Hubs / the Unrecognized Contributors

Returning to the Kleinberg's point of view about hubs and authorities, he mentioned that hub pages are pages that have links to multiple relevant authoritative pages. It is theses hub pages that "pull together" authorities on a common topic, and allow us to throw out unrelated pages of large in-degree. The purpose of searching for such web pages is to observe the rate of the pages that are excellent contributors to a certain subject, yet very few other pages know about them. However the presence of www.cancer.ab.ca in the root set of the "cancer" query played a major role in pulling together the best cancer sites.

We will see in the next tables that most of the top hubs have almost no inlinks neither from the root set nor from the base set, and most of those that are pointed to by other pages have a good authority weight under HITS.

Rank	**URL**	**Base Set in links**	**Global In links**
1	http://www.cancer.ab.ca/links/index.htm	0	1
2	http://medmark.org/onco/	0	0
3	http://www.wellwood.on.ca/	0	0
4	http://dmoz.org/Health/	0	0
5	http://www.cytorex.com/	0	0
6	http://www.cbmtg.org/	0	1
7	http://www.patientcenters.com/childcancer/	0	0
8	http://www.lambtonhealth.on.ca/	1	1
9	http://www.cancer.ca/	2	39
10	http://www.lifebank.com/	0	0

Table 3.6 In links count for the top Hubs of the Cancer query

For instance, the best hub is ranked as the best authority, and this explains why it is recognized contrary to other hub pages. A Hub page that is an authority can be the center of a different structure on the web. However, the presence of such structures is less evident than the presence of communities of hubs and authorities.

3.6 Characteristics of Authoritative Pages

Rank	Page Url	Outlinks
1	www.cancer.ca	14
2	www.nih.gov	3
3	www.nic.nih.gov	0
4	Www.cancer.org	1

Table 3.7 Outlinks count for some authorities of the cancer query

It is clearly seen that authoritative pages on a given topic link to other pages in the base set but not in large quantity. We can deduce that www.cancer.ca is a reference in this topic.

3.7 Emerging Communities in Cyberspace

The use of the term community does not mean that these structures have been constructed in a planned fashion, but they may be a consequence of the way in which page creators on the web link to each other. A community can be either explicit (gathered by human ontological effort) or implicit (emerging due to the link structure, even without individual users knowing about it).[4]

A community is constructed by taking the top ten authorities and top ten hubs for a specific query.

The set of densely linked hubs and authorities returned by the HITS algorithm for a given query is considered to be a principal community for that query. It was shown that this community relates to the principal eigenvectors of AA^T and A^TA. For this reason, it is called the principal community. As for the non-principal communities, we combine the highest components of the non-principal eigenvectors of AA^T and A^TA.

Top Hubs	Top Authorities
www.cancer.ab.ca	www.cancer.ca
Medmark.org/onco/	www.nih.gov
www.wellwood.on.ca	www.nci.nih.gov
Dmaoz.org/health	www.cancer.org

www.cytorex.com	www.cancer.org.au
www.cbmtg.org	www.hc-sc.gc.ca
www.patientcenters.com/childcancer/	www.bc.cancer.ca
www.lambonhealth.on.ca	www.ontario.cancer.ca
www.cancer.ca	www.cancer.ab.ca
www.lifebank.com	www.canadian-health-network.ca

Table 3.7 Community for the "Cancer " query

3.8 Conclusion

The implementation of the HITS algorithm gives a clear overview about the web map and how different pages interact. Adding to that it shows a clear and more accurate search results.

CHAPTER IV

Experimenting With HITS Parameters

In this chapter, we are going to make different experiments concerning the parameters that influence the performance of HITS. Some of these parameters include the number of iterations, the size of the root set , etc.

4.1 Changing the Size of the Root Set

Table 4.1 shows the variation of the root set which will result in direct variation in the size of the base set. Still working with the "Cancer" topic, we begin with a root set of five pages, then we increase the set by adding five pages at a time.

R	T
5	55
10	73
15	104
20	150

Table 4.1 size of base set relevant to the size of the root set

The next table shows and compares the top authorities returned by the *HITS* algorithm for the different base sets obtained from the previous table.

	R=5	R=10	R=15
1	www.breastcancer.care2.com	www.nih.gov	www.cancer.ca
2	www.quebec.cancer.ca	www.cancer.ca	www.nih.gov
3	www.cancer.ca	www.cancer.ab.ca	www.ontario.cancer.ca
4	www.cancer.org	www.uicc.ca	www.cancer.ab.ca
5	www.cancer.med.umich.edu	www.quebec.cancer.ca	www.nb.cancer.ca
6	www.endcancernow.com	www.breastcancer.care2.com	www.uicc.ch
7	www.cancer.ie	www.ontario.cancer.ca	www.lgfb.ca
8	www.nih.gov	www.cancer.se	www.quebec.cancer.ca
9	www.ontario.cancer.ca	www.endcancernow.com	www.breastcancer.care2.com

10	www.uicc.ch	www.cancercharities.com	www.cancernews.com

Table 4.2 Authorities for "Cancer" query with different root set

For broad topics such as "cancer" text-based search engines can return sometimes relevant answers, but the problem remains in the abundance problem. In comparison with the three listed results of the top authorities, it is upon human judgment to differentiate between the most relevant. However the table shows that the larger the root set is chosen to be, the better the results reached. Something very important to notice ,if we were selective in our choice while filling our root set, we would have even ensured better results.

4.2 Changing the Base Set

We can also expand the base set in a way that all pages linking in and out will be added. In our example the root set is about 20 pages, the base set is about 150 and the expanded base set will become approximately 700 pages.

Working with a larger base set will increase dramatically the computation cost. On the other side, the results will not be better than the previous ones.

4.3 Varying the Number of Iterations

Staring with a root set of twenty pages, the convergence of the major authorities for the "Cancer" query is attained after running three iterations of the HITS algorithm. The next table shows the changes in the rankings of the major authorities when the value of k increases. Most of theses authorities are grouped in the top 25 when k=3. In other words, running the HITS algorithm until convergence of the hub and authority vectors succeeds in filtering the most authoritative sources.

Rank	K=1	K=2	K=3
1	www.cancer.ca	www.cancer.ca	www.cancer.ca
2	www.nih.gov	www.nih.gov	www.nih.gov
3	www.cancer.org	www.nci.nih.gov	www.nci.nih.gov
4	www.nci.nih.gov	www.cancer.org	www.cancer.org
5	www.hc-sc.gc.ca	www.hc-sc.gc.ca	www.hc-sc.gc.ca
6	www.dhhs.gov	www.bc.cancer.ca	www.bc.cancer.ca
7	www.bc.cancer.ca	www.ontario.cancer.ca	www.ontario.cancer.ca
8	www.ontario.cancer.ca	www.cancer.ab.ca	www.cancer.ab.ca

9	www.easyscopes.com	www.canadian-health-network.ca	www.canadian-health-network.ca
10	Bigcats.care2.com	www.nb.cancer.ca	www.nb.cancer.ca
11	Breastcancer.care2.com	www.nfandlab.cancer.ca	www.nfandlab.cancer.ca
12	Rainforest.care2.com	www.cancernews.com	www.cancernews.com
13	www.canadian-health-network.ca	www.uicc.ch	www.cancer.med.umich
14	www.cancer.ab.ca	www.cancer.med.umich	www.uicc.ch
15	www.cancernews.com	www.lgfb.ca	www.lgfb.ca

Table 4.3 Rankings of the authorities when k increases

Many factors, like the size of the root set and the value of epsilon (the error factor that determines convergence), affect the number of iterations in the *HITS* algorithm. In the following chart, we present the effects of these factors on the value of k. The x-axis represents epsilon; The y-axis represents the number of iterations k.

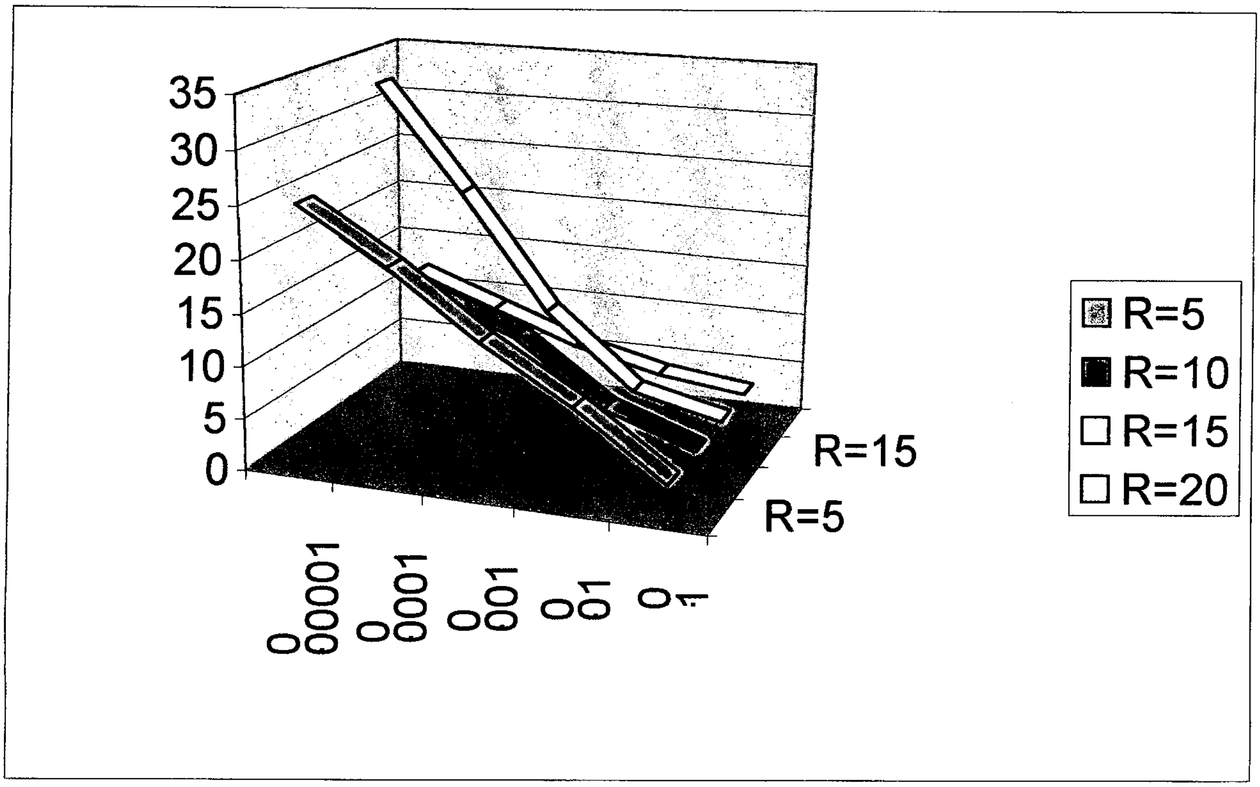


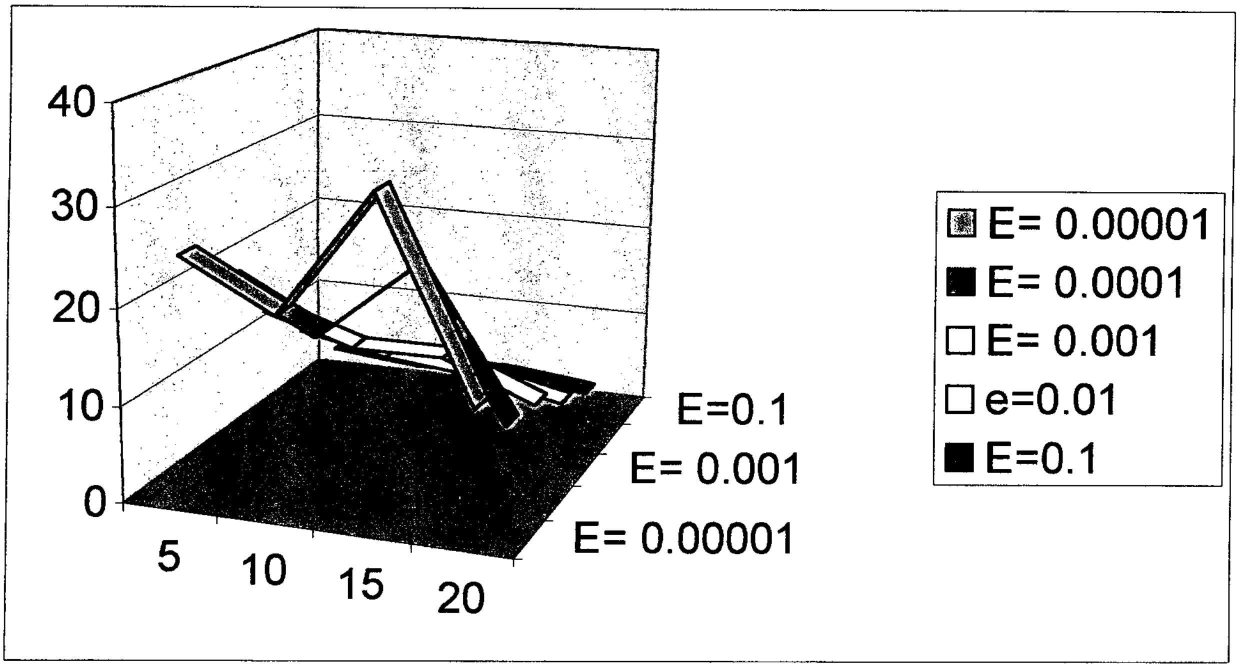
Fig 4.1 K versus Epsilon for the Cancer query

From the table shown above we can deduce the following:

- The number of iterations is not necessarily proportional to the size of the base set. For epsilon =0.00001, the largest root set (20 pages) needs the lowest number of iterations (12 iterations).
- Note also that all root sets reach their convergence for k =3 when epsilon is high (0.1).

The next chart shows the relationship between the number of iterations and the size of the root set as we change the value of Epsilon. The x=axis represents the size of the root set R and the y-axis represents the number of iterations k.

Fig 4.2 R versus K for the Cancer query



This chart shows clearly that whatever the value of epsilon is, most of the main authorities and hubs converge when R=20. This fact is due to the existence of a major hub page, which helped pulling together most of the authorities, and consequently formed a much more wired base set.

4.4 Conclusion

Different experiments on the parameters of the HITS are experimented with, in this chapter, which showed that the size of the root set play an important role in the success of HITS. The next chapter introduces a modified version of the HITS algorithm called an Interactive Search Engine.

CHAPTER V

Interactive Search Engines

5.1 The weaknesses in Present Search Engines

All search engines, text-based and link-based, do not get the user involved in the search results. In fact, they are almost batch processes. Depending on the topic, the user may get thousands of pages in return. Also, we note the large difference in accuracy between several search engines like AltaVista and Google. As a matter of fact, the results have to be narrowed. The problem becomes apparent with broad topics such as cancer, which results in real dissatisfaction on the user part (having as a result 100,000 pages).

5.2 A New Approach to Solve the Problem

To solve theses problems and to have accurate, sharp and precise answers, we are going to implement an interactive method in designing search engines.

In fact this method relies on the user's evaluation and ranking of pages while working on a specific topic. It is well known that users work and explore ordinarily the first twenty or thirty pages returned by any available search engine. Our method suggests the following: the user is asked at the end of each search result to rank the pages explored according to his evaluation and need. In other words, users will be able to confer a relative authority to a specified page.

5.3 Implementation of the New Algorithm

First, we begin by editing the table " pages" , we added the "ranking" and "new weight" fields. (shown in Table 3.1). These fields are set to null values.

We created an interface using Access software that allows the user to edit the " ranking " field. In our example, we are working on the root set, mainly the first twenty pages. The user enters values from 1 to 20 depending on his evaluation.

The new weight field is updated automatically by a specific code using the following formula:

- NewWeight calculation:

  NW = 1 / number of pages.

  NW = 1 / 20 = 0.05. (we are working on 20 pages only)

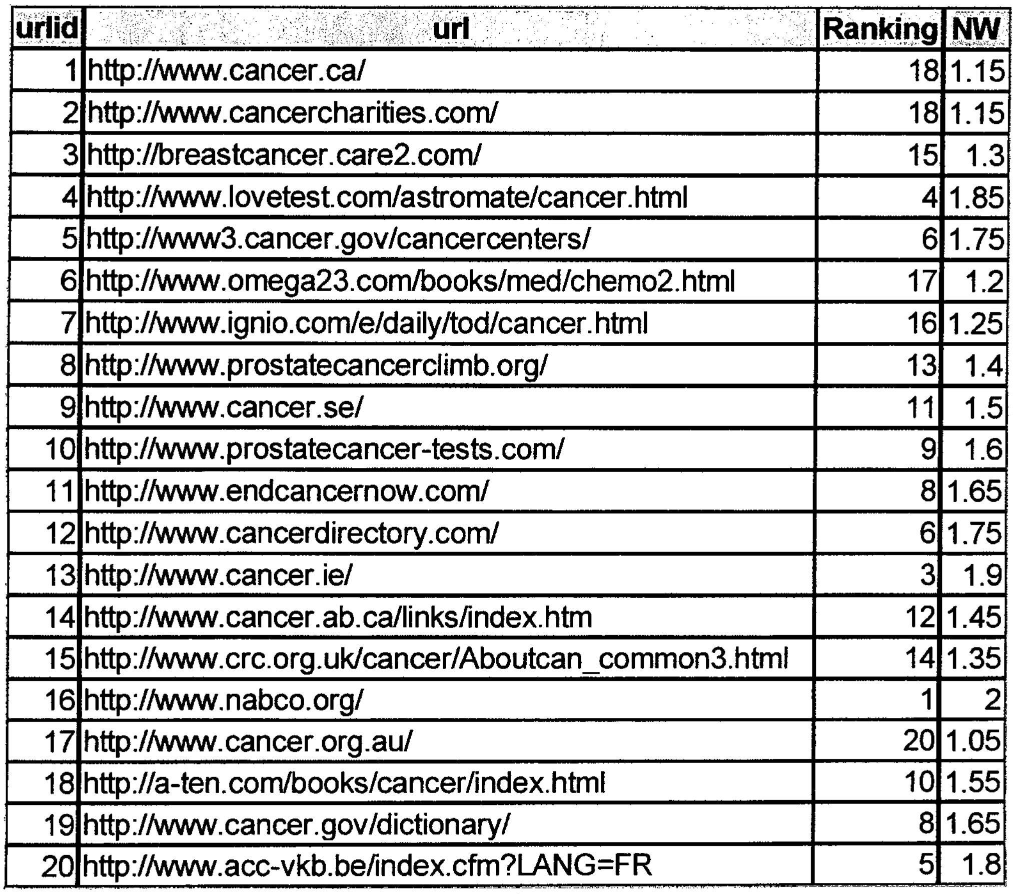
  E is between 1 at maximum and 0.05 at minimum.

- Weight calculation:

  Weight = (((Number of pages + 1) –(ranking)) * E) + 1)

  Ex:

  A page ranked by user as fifth one will have additional weight = (((20 + 1) - (5) * 0.05) + 1) = 0.8



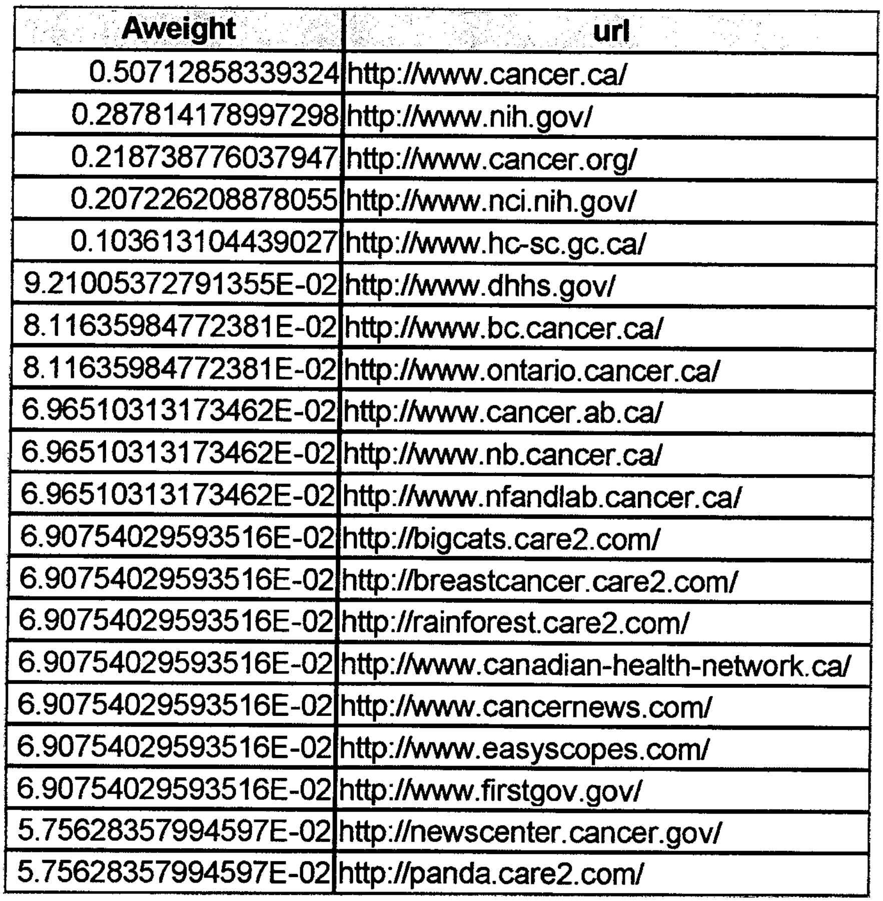
urlid	url	Ranking	NW
1	http://www.cancer.ca/	18	1.15
2	http://www.cancercharities.com/	18	1.15
3	http://breastcancer.care2.com/	15	1.3
4	http://www.lovetest.com/astromate/cancer.html	4	1.85
5	http://www3.cancer.gov/cancercenters/	6	1.75
6	http://www.omega23.com/books/med/chemo2.html	17	1.2
7	http://www.ignio.com/e/daily/tod/cancer.html	16	1.25
8	http://www.prostatecancerclimb.org/	13	1.4
9	http://www.cancer.se/	11	1.5
10	http://www.prostatecancer-tests.com/	9	1.6
11	http://www.endcancernow.com/	8	1.65
12	http://www.cancerdirectory.com/	6	1.75
13	http://www.cancer.ie/	3	1.9
14	http://www.cancer.ab.ca/links/index.htm	12	1.45
15	http://www.crc.org.uk/cancer/Aboutcan_common3.html	14	1.35
16	http://www.nabco.org/	1	2
17	http://www.cancer.org.au/	20	1.05
18	http://a-ten.com/books/cancer/index.html	10	1.55
19	http://www.cancer.gov/dictionary/	8	1.65
20	http://www.acc-vkb.be/index.cfm?LANG=FR	5	1.8

Fig 6.1 the epsilon values

The NewWeight values showed in the previous table are the values received from the formula plus one, because the value of the hub weights and authority weights are set initially to one.

In this way the user is given the authority to edit the weights of pages inside the *HITS* algorithm.

After this step we run *HITS*, but with new values assigned for Hweight and Aweight.



Aweight	url
0.507128583339324	http://www.cancer.ca/
0.287814178997298	http://www.nih.gov/
0.218738776037947	http://www.cancer.org/
0.207226208878055	http://www.nci.nih.gov/
0.103613104439027	http://www.hc-sc.gc.ca/
9.21005372791355E-02	http://www.dhhs.gov/
8.11635984772381E-02	http://www.bc.cancer.ca/
8.11635984772381E-02	http://www.ontario.cancer.ca/
6.96510313173462E-02	http://www.cancer.ab.ca/
6.96510313173462E-02	http://www.nb.cancer.ca/
6.96510313173462E-02	http://www.nfandlab.cancer.ca/
6.90754029593516E-02	http://bigcats.care2.com/
6.90754029593516E-02	http://breastcancer.care2.com/
6.90754029593516E-02	http://rainforest.care2.com/
6.90754029593516E-02	http://www.canadian-health-network.ca/
6.90754029593516E-02	http://www.cancernews.com/
6.90754029593516E-02	http://www.easyscopes.com/
6.90754029593516E-02	http://www.firstgov.gov/
5.75628357994597E-02	http://newscenter.cancer.gov/
5.75628357994597E-02	http://panda.care2.com/

Fig 6.2 new top authorities

In comparison with Fig 3.3 we can observe a slight difference in the rankings of top authorities. However www.cancer.ca remains in the highest rank due to the large number of inlinks pointing to that page.

5.4 A Comparison Between HITS and our Modified HITS-Based Interactive Search Engine

In table 6.1, the differences between the results returned by HITS algorithm for 20 iterations and the ones returned by modified HITS_based interactive search engine for 20 iterations are compared.

HITS	**Modified** ***HITS***
www.cancer.ca	www.cancer.ca
www.nih.gov	www.nih.gov
www.nci.gov.nih.gov	www.cancer.org
www.cancer.org	www.nci.gov.nih.gov
www.hc-sc.gc.ca	www.hc-sc.gc.ca
www.bc.cancer.ca	www.dhhs.gov
www.ontarion.cancer.ca	www.bc.cancer.ca
www.cancer.ab.ca	www.ontarion.cancer.ca
www.canadian-health-network.ca	www.firstgov.gov
www.nb.cancer.ca	www.cancer.ab.ca
www.nfandlab.cancer.ca	www.nb.cancer.ca
www.uicc.ch	www.nfandlab.cancer.ca
www.cancernews.com	www.canadian-health-network.ca
www.cancer.med.umich.edu	www.cancernews.com
www.lgfb.ca	www.easyscopes.com
www.quebec.cancer.ca	Bigcats.care2.com
www.ncic.cancer.ca	Breastcancer.care2.com
Breastcancer.care2.com	www.rainforest.care2.com
www.cancerguide.org	www.newscenter.cancer.gov

Table 6.1 Comparison between HITS and Modified HITS

When the user enters his/her rankings, the conferred authority will affect the root set only. The first two pages returned by the HITS algorithm remains in their places for several reasons: these pages have very high inlinks compared to other pages within the base set. In our evaluation, the weight given by a single user to a page is equal to the weight conferred by a single page.

5.5 Conclusion

This chapter describes a new approach toward interactive search engine. In fact, this method relies mostly on user's corporation to ameliorate the Internet

CHAPTER VI

Conclusion

The WWW has grown into a hypertext environment with enormous complexity and the process underlying the growth has been driven in a chaotic fashion by the individual actions of numerous participants. Our experience with HITS suggests, however, that in many respects the end product is not chaotic as one might think. The aggregate behavior of user populations on the WWW can be studied through a mathematically clean technique for analyzing the Web's link topology, and one can use this technique to identify themes about hyperlinked communities that appear to span a wide range of interests and disciplines.

In this thesis, the HITS algorithm is described in details as well as providing a practical implementation for testing purpose. The code used can serve for future research in the field, since it is clearly written and explained.

Variations of parameter, root and base set allow a sharp understanding of the HITS algorithm.

The idea we proposed in this thesis; that is, about interactive search engines, is definitely worth further investigations.

REFERENCES

[1] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins, "Trawling the web for emerging crbyer-communities", Proc. 8th WWW Conf., 1999

[2] J Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, A tomkins, "The web as a graph: measurements, models, and methods".

[3] J. Kleinberg, "Authoritative sources in a hyperlinked Environment", Proc. ACM-SIAM Symposium on Discrete Algorithms,1997.

[4] David Gibson, Jon Kleinberg, Parbhakar Raghavan, "Inferring Web Communities from link Topology".

[5] S. Chakrabarti, B. dom, P. Raghavan, S.Rajaghopalan, D. Gibson, j.Kleinberg, "Automatic resource compilation by analyzing hyperlink structure and associated text"

[6] Gilbert Strang, "Linear Algebra and its applications"

[7] Brian Amento, Loren terveen, Will Hill, "Does Authority mean quality? Predicting Expert Quality Ratings of Web Documents"

[8] Goggle search engine, www.goggle.com

[9] Krishna Bharat, Andrei Broder, Monika Henzinger, Ouneet Kumar, Suresh Venkatasubramanian, " The connectivity Server: fast access to linkage information on the Web".

[10] Kemal Efe, Vijay Raghavan, C. Henry Chu, Adrienne L.Broadwater, Levent Bolleli, Seyda Ertekin," The shape of the web and its implications for searching the Web"

Appendix A

Option Compare Database

This is the main function in the link code. We run this function first of all to fill the table link. The following will take the URL of each page and check it in the source code of the following code. If it is found, a new record is opened in the link table and the urlid of the source and target page are entered.

```
Sub initialize_links() ' intializing a function
  Dim nLoc, strTmp  ' declaration of 2 variables
  Dim rst, rst1 As ADODB.Recordset  ' these 3 lines are used whenever we want to work
  Set rst = New Recordset ' with tables
  Set rst.ActiveConnection = CurrentProject.Connection
  rst.Source = "select * FROM pages" 'in rst we have now all the content of table pages
  rst.Open 'first time the pointer is on the first record
  Do While Not rst.EOF ' begin of the loop
    strTmp = cleanstr(rst("url")) ' call function cleanstr and assigning the url
    Set rst1 = New Recordset ' working with the page "to"
    Set rst1.ActiveConnection = CurrentProject.Connection
    rst1.Source = "SELECT urlid FROM pages WHERE page LIKE '%" & strTmp & "%'"
    rst1.Open   ' taking the urlid of the selected URl
    Do While Not rst1.EOF   ' begining the loop to search each source page
        Insert_link rst1("urlid"), rst("urlid")  ' insert in the table link
         rst1.MoveNext  ' move to next record
    Loop
    Set rst1 = Nothing
    rst.MoveNext
   Loop
  Set rst = Nothing
 End Sub
```

```
'This function clean the URl from http:// ,www and the / at the end
```

```
Function cleanstr(strURL)

Dim nLoc, strTmp
  nLoc = InStr(strURL, "http://")  'compare url to http://
  If nLoc = 1 Then ' if true
    strTmp = Right(strURL, Len(strURL) - 7)  ' delete the first 7 letter
```

```
 End If
 nLoc = InStr(strTmp, "www.") 'compare url to www
 If nLoc = 1 Then ' if true
    strTmp = Right(strTmp, Len(strTmp) - 4) ' delete the first 4 letter
 End If
 If Right(strTmp, 1) = "/" Then 'if the last character is /
    strTmp = Left(strTmp, Len(strTmp) - 1) 'delete
 End If
 cleanstr = strTmp

End Function
```

This function is called from the main one, its role is only to insert the value in the table link in the from and to field.

```
Public Sub Insert_link(FromID, ToID)
   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset ' opening the table
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "INSERT INTO LINKS ([From],[to])VALUES(" & FromID & "," & ToID &
")"
   rst.Open ' insert values in fromID and ToID in the tabel
   Set rst = Nothing
End Sub
```

This function deletes rows from the table link delete each row where the from = to

```
Public Sub delete_record()
   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "DELETE * FROM LINKS WHERE From =to"

     rst.Open
   Set rst = Nothing
End Sub
```

Appendix B

The source code of *HITS* algorithm

```
Option Compare Database
```

This is the main function in this module the following function open the table pages and take the values of the urlid field it calls I_operation for each urlid it calls O_operation for each urlid. Then it normalize the two vector in order to have values between 0 and 0.1

```
Sub ITERATE_OPERATION()
Dim WORK_TABLE
Set WORK_TABLE = New ADODB.Recordset
Set WORK_TABLE.ActiveConnection = CurrentProject.Connection
WORK_TABLE.Source = "SELECT URLID FROM PAGES"
WORK_TABLE.Open
  Do While Not WORK_TABLE.EOF
    I_OPERATION WORK_TABLE("URLID")
    WORK_TABLE.MoveNext
  Loop
WORK_TABLE.MoveFirst
  Do While Not WORK_TABLE.EOF
    O_OPERATION WORK_TABLE("URLID")
    WORK_TABLE.MoveNext
  Loop
Set WORK_TABLE = Nothing
Normalize
End Sub
```

'The following function take a single page number and calculate its authority weight

```
Sub I_OPERATION(Page_ID)
  Dim TOTAL_HUB_WEIGHT
  Dim SQLSTR As String
```

The work of the following SQL code is the basic of the HITS algorithm we are working now on 2 tables the LINK table and the WEIGHTs table.
'We have to search for all pages linking to that page_id (to fing its authority)
'We make an inner join on these tables using the "from" from Links and "urlid" from WEIGHTS
'we add all the value in the hub weight column, this is the authority weight for page_id

```
    SQLSTR = "SELECT SUM(weights.Hweight) as [TOTAL_WEIGHT] "
    SQLSTR = SQLSTR & "FROM Links INNER JOIN weights ON Links.[From] = "
    SQLSTR = SQLSTR & "weights.urlid WHERE Links.[To]=" & Page_ID
 'SQLSTR = "SELECT SUM (WEIGHT.HWEIGHT) AS [TOTAL_WEIGHT]"
 'SQLSTR = SQLSTR & "FROM LINKS INNER JOIN WEIGHTS ON LINKS.[FROM] =
 WEIGHTS.URLID"
 'SQLSTR = SQLSTR & "WHERE LINKS.[TO]=" & Page_ID

 Dim rst As ADODB.Recordset
 Set rst = New ADODB.Recordset
 Set rst.ActiveConnection = CurrentProject.Connection
 rst.Source = SQLSTR
 rst.Open
```

we update now the value in the authority column

```
 If Not IsNull(rst("TOTAL_WEIGHT")) Then
    TOTAL_HUB_WEIGHT = rst("TOTAL_WEIGHT")
    Update_Authority_Weight Page_ID, TOTAL_HUB_WEIGHT
 End If
 Set rst = Nothing
```

End Sub

The work of this function is similar to the I-operation the calculations are made on the authority vector

```
Sub O_OPERATION(Page_ID)
   Dim TOTAL_AUTHORITY_WEIGHT
   Dim SQLSTR As String

   SQLSTR = "SELECT SUM(weights.Aweight) as [TOTAL_WEIGHT] "
   SQLSTR = SQLSTR & "FROM Links INNER JOIN weights ON Links.[To] =
weights.urlid "
   SQLSTR = SQLSTR & "WHERE Links.[From] =" & Page_ID

'SQLSTR = "SELECT SUM (WEIGHTS.AWEIGHT) AS [TOTAL_WEIGHT]"
'SQLSTR = SQLSTR & "FROM LINKS INNER JOIN WEIGHTS ON LINKS.[TO] =
WEIGHTS.URLID"
'SQLSTR = SQLSTR & "WHERE LINKS.[FROM]=" & Page_ID

Dim rst As ADODB.Recordset
Set rst = New ADODB.Recordset
Set rst.ActiveConnection = CurrentProject.Connection
```

```
rst.Source = SQLSTR
rst.Open
'we update now the value in the authority column


If Not IsNull(rst("TOTAL_WEIGHT")) Then
   TOTAL_AUTHORITY_WEIGHT = rst("TOTAL_WEIGHT")
   Update_Hub_Weight Page_ID, TOTAL_AUTHORITY_WEIGHT
End If
Set rst = Nothing

End Sub


Public Sub Update_Authority_Weight(Page_ID, New_Weight)
   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "UPDATE weights SET Aweight = Aweight + " & New_Weight & "
WHERE urlid=" & Page_ID
   rst.Open
   Set rst = Nothing
End Sub


Sub Update_Hub_Weight(Page_ID, New_Weight)
   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "UPDATE weights SET Hweight = Hweight + " & New_Weight & "
WHERE urlid=" & Page_ID
   rst.Open

   Set rst = Nothing
End Sub



Sub Normalize()
   Dim Total_Aut_W, Total_Hub_W
   Dim rst As ADODB.Recordset

   Total_Aut_W = 0
   Total_Hub_W = 0

   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "SELECT Aweight, Hweight FROM weights"
   rst.Open
```

```
    Do While Not rst.EOF
       Total_Aut_W = Total_Aut_W + rst("Aweight") ^ 2
       Total_Hub_W = Total_Hub_W + rst("Hweight") ^ 2
       rst.MoveNext
    Loop

    Total_Aut_W = Sqr(Total_Aut_W)
    Total_Hub_W = Sqr(Total_Hub_W)
    rst.Close

    rst.Source = "UPDATE weights SET weights.Aweight = [weights].[Aweight]/" &
 Total_Aut_W
    rst.Open
    Set rst = Nothing

    Set rst = New ADODB.Recordset
    Set rst.ActiveConnection = CurrentProject.Connection
    rst.Source = "UPDATE weights SET weights.Hweight = [weights].[Hweight]/" &
 Total_Hub_W
    rst.Open

    Set rst = Nothing
 End Sub



 Public Sub clear_aweights()
    Dim rst As ADODB.Recordset
    Set rst = New ADODB.Recordset
    Set rst.ActiveConnection = CurrentProject.Connection
    rst.Source = "UPDATE weights SET Aweight = 1"
    rst.Open
    Set rst = Nothing

 End Sub


 Public Sub clear_hweights()
    Dim rst As ADODB.Recordset
    Set rst = New ADODB.Recordset
    Set rst.ActiveConnection = CurrentProject.Connection
    rst.Source = "UPDATE weights SET hweight = 1"
    rst.Open
    Set rst = Nothing

 End Sub


 Public Sub clear_rankings()
```

```
    Dim rst As ADODB.Recordset
    Set rst = New ADODB.Recordset
    Set rst.ActiveConnection = CurrentProject.Connection
    rst.Source = "UPDATE pages SET Ranking = NULL"
    rst.Open
    Set rst = Nothing

End Sub


Public Sub EnterEpsilon()

   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "UPDATE pages SET epsilon = ((21 - ranking) * 0.05) + 1 where ranking is
not null"
   rst.Open
   Set rst = Nothing

End Sub


Public Sub clear_epsilon()
   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "UPDATE pages SET Epsilon = NULL"
   rst.Open
   Set rst = Nothing

End Sub



Public Sub UpdateEpsilonWeight()

   Dim rst As ADODB.Recordset
   Set rst = New ADODB.Recordset
   Set rst.ActiveConnection = CurrentProject.Connection
   rst.Source = "UPDATE pages INNER JOIN weights ON pages.urlid = weights.urlid SET
weights.aweight = pages.Epsilon, weights.hweight = pages.Epsilon where pages.ranking is
not null"
   rst.Open
   Set rst = Nothing
End Sub
```

Appendix C

The Base set of "Cancer" query

Urlid	Url
1	http://www.cancer.ca/
2	http://www.cancercharities.com/
3	http://breastcancer.care2.com/
4	http://www.lovetest.com/astromate/cancer.html
5	http://www3.cancer.gov/cancercenters/
6	http://www.omega23.com/books/med/chemo2.html
7	http://www.ignio.com/e/daily/tod/cancer.html
8	http://www.prostatecancerclimb.org/
9	http://www.cancer.se/
10	http://www.prostatecancer-tests.com/
11	http://www.endcancernow.com/
12	http://www.cancerdirectory.com/
13	http://www.cancer.ie/
14	http://www.cancer.ab.ca/links/index.htm
15	http://www.crc.org.uk/cancer/Aboutcan_common3.html
16	http://www.nabco.org/
17	http://www.cancer.org.au/
18	http://a-ten.com/books/cancer/index.html
19	http://www.cancer.gov/dictionary/
20	http://www.acc-vkb.be/index.cfm?LANG=FR
21	http://www.abdiagnostics.com/
22	http://www.bbbonline.org/r2.cfm?ID=372000185
23	http://www.cancer.org/
24	http://www.tricky.com/liz/
25	http://www.cancer.ca/
26	http://www.cancerboard.ab.ca/
27	http://www.cancer.au.org./
28	http://www.cpcn.org/
29	http://medweb.bham.ac.uk/cancerhelp/
30	http://www.cancernews.com/quickload.htm
31	http://www.cancernews.com/quickload.htm
32	http://www.cancerindex.org/guide2.htm
33	http://www.cancer.med.umich.edu/
34	http://www.ncic.cancer.ca/
35	http://www.hc-sc.gc.ca/hpb/lcdc/bc/index.html
36	http://www.uicc.ch/
37	http://www.lgfb.ca/
38	http://www.nsabp.pitt.edu/

Urlid	Url
39	http://imsdd.meb.uni-bonn.de/cancernet
40	http://cancer.med.upenn.edu/
41	http://communities.msn.ca/ProstaidCalgary/homepage
42	http://www.terryfoxrun.org/
43	http://www.bc.cancer.ca/
44	http://www.nb.cancer.ca/
45	http://www.nfandlab.cancer.ca/
46	http://www.ontario.cancer.ca/
47	http://www.breast.cancer.ca/
48	http://www.tobaccotruth.com/
49	http://www.5to10aday.com/
50	http://www.actcancer.org/
51	http://www.accv.org.au/cancer1/
52	http://www.cancerguide.org/
53	http://www.cancernews.com/
54	http://www.graylab.ac.uk/cancerweb.html
55	http://www.cansearch.org/
56	http://www.aihw.gov.au/
57	http://www.icr.ac.uk/
58	http://www.uicc.ch/
59	http://www.mayoclinic.com/
60	http://www.ncci.org.au/
61	http://www.actcancer.org/
62	http://newscenter.cancer.gov/
63	http://www.oncolink.com/
64	http://www.petermac.org/
65	http://medmark.org/onco/
66	http://cornhill.ludwig.edu.au/cara2/
67	http://www.ludwig.edu.au/cdct/
68	http://www.ctc.usyd.edu.au/
69	http://www.iarc.fr/
70	http://www.ncbi.nlm.nih.gov/PubMed/
71	http://telescan.nki.nl/
72	http://www.cancersa.org.au/
73	http://nt.citysearch.com.au/E/V/NORTH/0033/04/70/
74	http://www.cancerwa.asn.au/
75	http://www.cancerwa.asn.au/
76	http://www.qldcancer.com.au/
77	http://www.cancer.org/
78	http://newscenter.cancer.gov/
79	http://www.nih.gov/
80	http://www.dhhs.gov/
81	http://www.firstgov.gov/
82	http://www.netnanny.com/
83	http://www.webtrendslive.com/
84	http://www.ssmg.be/

Urlid	Url
85	http://www.rmg.ssmg.be/
86	http://www.hon.ch/
87	http://panda.care2.com/
88	http://rainforest.care2.com/
89	http://bigcats.care2.com/
90	http://www.breastcancerfund.org/care.htm
91	http://www.guru*HITS*.com/
92	http://www.easyscopes.com/
93	http://www.0800-horoscope.com/
94	http://www.drive-your-man-wild.com/
95	http://www.one-and-only.com/
96	http://www.nci.nih.gov/
97	http://www.futuresedge.org/
98	http://motorheaven.bizland.com/
99	http://futuresedge.org/
100	http://www.research.oxydex.com/
101	http://www.a-ten.com/
102	http://www.amazon.com/exec/obidos/subst/
103	http://www.gwpharm.com/cann_index.html
104	http://www.lindesmith.org/medicalmarijuana/
105	http://www.ukcia.org/medical/default.html
106	http://www.psykedelbok.se/cannabis_medicin.html
107	http://www.dieminger.com/gorgojo/
108	http://www.cancer.ab.ca/
109	http://www.bc.cancer.ca/
110	http://www.mb.cancer.ca/
111	http://www.nb.cancer.ca/
112	http://www.nfandlab.cancer.ca/
113	http://www.ontario.cancer.ca/
114	http://www.quebec.cancer.ca/
115	http://www.pei.cancer.ca/
116	http://www.sk.cancer.ca/
117	http://www.communication.gc.ca/
118	http://www.iapacificlife.com/
119	http://www.hrsb.ns.ca/
120	http://www.cbmtg.org/
121	http://www.wellwood.on.ca/
122	http://www.logicorp.ca/
123	http://host.perth.igs.net/
124	http://www.canadian-health-network.ca/
125	http://www.muggah.org/
126	http://www.mdscollaborate.com/
127	http://www.lifebank.com/
128	http://www.csih.org/
129	http://www.psynergie.com/
130	http://www.patientcenters.com/childcancer/

Urlid	Url
131	http://cbrpe.uwaterloo.ca/
132	http://www.hc-sc.gc.ca/
133	http://www.gov.on.ca/health/
134	http://www.canoe.ca/HealthReference/
135	http://www.manpros.org/
136	http://www.qe2-hsc.ns.ca/care
137	http://www.cancerresources.com/
138	http://www.tsrcc.on.ca/supcare.htm
139	http://dmoz.org/Health/
140	http://www.cytorex.com/
141	http://medbio.utoronto.ca/student
142	http://www.lambtonhealth.on.ca/
143	http://www.geocities.com/~mlshams/acronym/
144	http://www.ptcc.on.ca/rds/
145	http://www.for.gov.bc.ca/protect/
146	http://www.mednets.com/hemeoncoass.htm
147	http://www.cardiffjitsu.com/donate4free.htm
148	http://members.tripod.com/~thepops/intro-b.html
149	http://www.geocities.com/griefpoetry/
150	http://www.everyclickcounts.org/