2008

Recognizing anchoring text patterns on the web

Shruti K. Bhandari
Lehigh University

Bhandari, Shruti K.

Recognizing Anchor Text Patterns on the Web

January 2008

Recognizing Anchor Text Patterns on the Web

by

Shruti K. Bhandari

A Thesis

Presented to the Graduate and Research Committee

Of Lehigh University

In Candidacy for the Degree of

Master of Science

in

Computer Science

Lehigh University

January, 2008

This thesis is accepted and approved in partial fulfillment of the requirements for the Master of Science.

7 Dec. 2007

Date

Thesis Advisor

Chairperson of Department

Acknowledgments

I would like to sincerely thank Prof. Brian D. Davison for mentoring me throughout this study. He has provided me the much needed guidance and the necessary infrastructure to carry out the experiments. He has also been very receptive to my ideas and very patient, even at times when I could not give in my best. I would also like to thank my senior colleagues XiaoGuang Qi and Lan Nie for their advice and guidance, whenever required.

Table of Contents

Abstract

Anchor text, the hyperlinked text, on a page gives a visitor concise information about the page it links to. Studies and experiments have been carried out in the past to extract the features of anchor text for better search engine results. Researchers have tried to find the similarity between the anchor texts, user queries submitted to the search engines and the titles of web pages.

In this Master's thesis, our task is to extract the anchor text patterns found on the web and to study them to better understand the hyperlink structure formed within the web pages. We investigate various kinds of distributions such as those of the regular and spam anchor texts, and regular and spam pages in the UK dataset (.uk Web pages). We have answered questions such as given an anchor text, what are the different pages it points to and vice versa, that is, given a page, what are the different anchor texts pointing to it.

We compare two methods to find the most significant page for a given anchor text (which can be used as a query) by ranking the pages it points to according to i) the number of times it points to that page and ii) weighting the number of times it points to that page by the number of other anchor texts pointing to that page. We have also investigated into a possible method of classifying a given anchor text or a page as regular or spam.

To ease the viewing process of the different results for a subset of our experiments, we have also built a user interface. This user interface provides a number of search options such as the search of the pages pointed to by a given anchor text, search of the anchor texts pointing to a given page, and more.

Chapter 1

Introduction

Anchor text, the hyperlinked text, on a page gives a visitor concise information about the page it links to. Efforts have been made by researchers to make use of the inherent features of anchor text to enhance the quality of the search engine results. In this chapter, we explain the concept of anchor text and the associated hyperlink structure of the World Wide Web. We further present our goals and the questions we are trying to answer using the conclusions from our work. The chapter concludes with a brief summary of the chapters to follow.

1.1 Anchor Text and Hyperlink Structure

Suppose that you are browsing though a page on the Internet. While reading through the text of the page, you come across a part of the text which is clickable. When you click on this text, you are redirected to another page. The new page usually contains information which is indicated by the text which is clicked to reach this page. This clickable text is called anchor text. The reference from one section of a document to another section or from one document to another document is called a hyperlink. A hyperlink can be in the form of text, image or media. The hyperlinked text on a web page which when clicked leads to a new page is called anchor text. The new page which is opened is called the target page. The anchor text of a page is associated with the URL of the target page.

Hyperlinks link a unit of information to another unit of information over the Internet either through direct links or through a sequence of links. Hyperlinks are therefore essential to the existence of the World Wide Web.

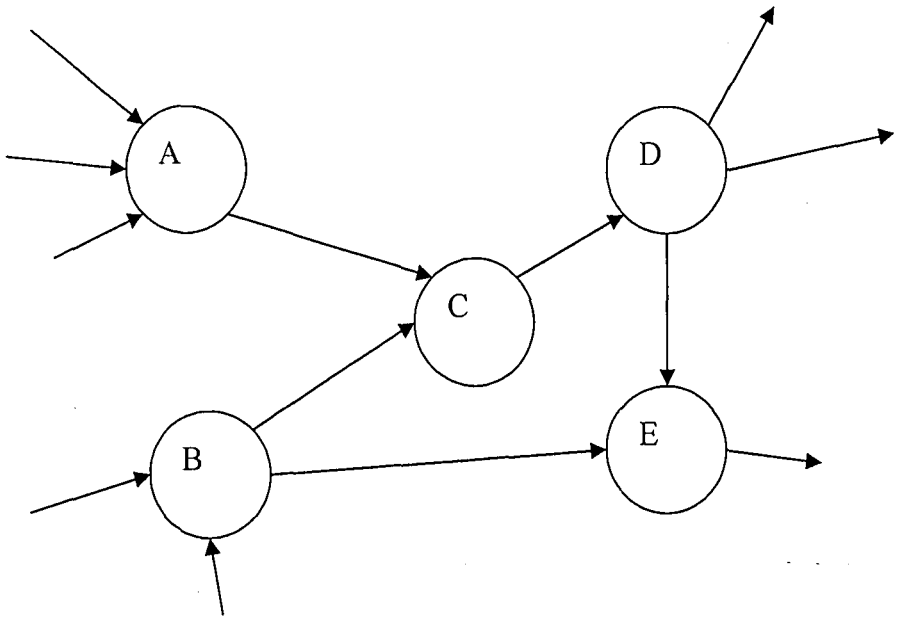


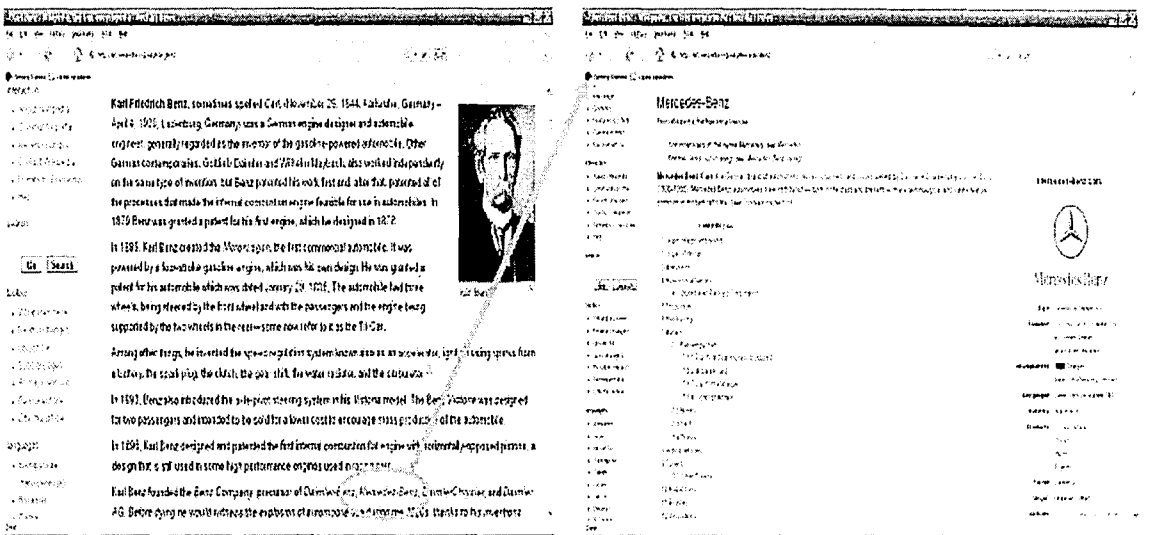
Figure 1.1

In Figure 1.1, we can see a sample of the hyperlink structure. Each node in the figure represents a Web page and each directed link represents a hyperlink between the pages. For example, the directed arrow from B to C represents a hyperlink from page B to page C. Page B contains anchor text which, when clicked, directs a user to page C.

1.2 Anchor Text Patterns on the Web

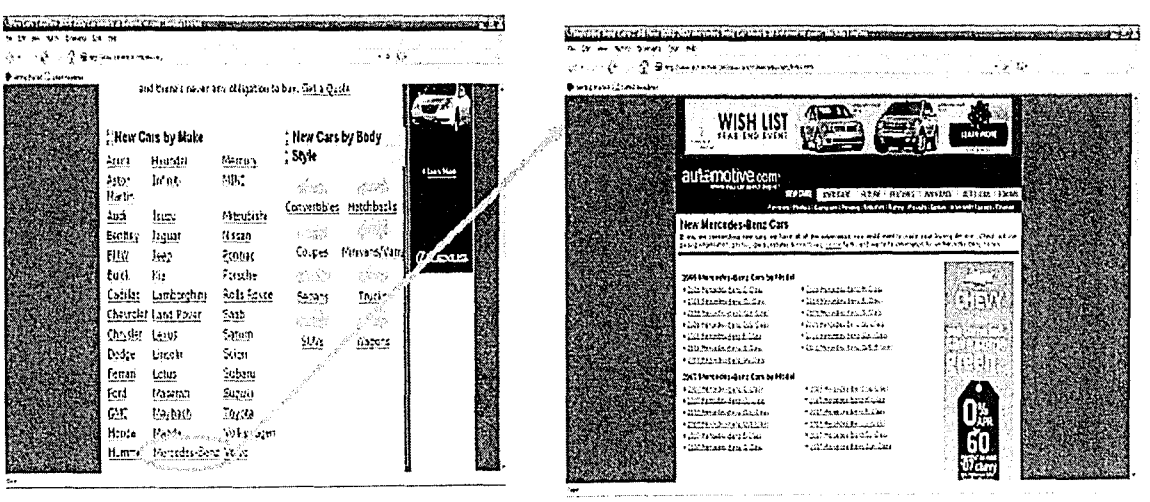
Anchor text can be indicative of the information available on the target page. This inherent property of the anchor text can be used as one of the features for the classification of the target page. It can also be used to rank the pages returned as a result

of searching a query on the Web. A considerable amount of research has been done to find out similar uses of incorporating anchor text features in the field of search engines. In this paper, we perform data mining on a subset of .uk web pages by extracting and studying anchor text patterns in these pages.



Page A Page B

Figure 1.2 Example of anchor text



Page C Page D

Figure 1.3 Example of anchor text

Consider the scenarios in Figure 1.2 and Figure 1.3. As we can see, the anchor text 'Mercedes Benz' on page A points to page B. The same anchor text is present on page C but it points to page D and not to page B. Thus, the same anchor text, present in different pages, can point to different target pages. Similarly, one page can be pointed to by different anchor texts, present in different pages. This interlinking of anchor texts and target pages can be represented by a bipartite graph as shown in Figure 1.4.

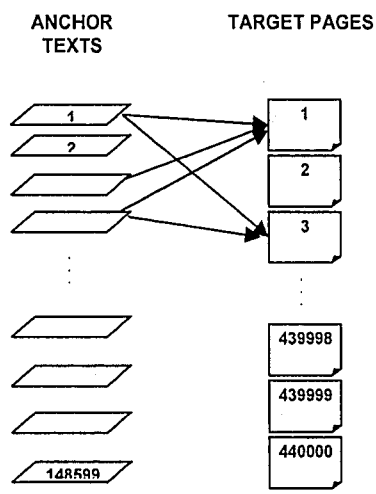


Figure 1.4 Bipartite Graph representing interlinking of anchor texts and target pages

In this paper, our goal is to find out how the anchor texts and the target pages are arranged on the Web. We have answered the following questions in this paper. Which is the most popular page in the given dataset or which page is pointed to by the maximum number of anchor texts? Given an anchor text, what is the distribution of the pages that it points to? Similarly, given a page, what is the distribution of the anchor texts that point to this page? Do these distribution curves follow the Zipf power law? Are the distribution

curves for the regular anchor texts different than that of the spam anchor texts? Do the distribution curves for the regular and spam pages differ from each other? Is there a possible way to recognize spam text and spam pages using the features inherent in the distribution curves? We use two different methods to rank the set of pages pointed to and anchor texts pointing to a given anchor text and a given page respectively. We have also compared the set of pages returned for a given anchor text using our method with the set of pages which are relevant according to the BM25F scores when the same anchor text is used as a query for relevance search.

We present the existing work involving the usage of anchor texts in Chapter 2. We present our approach in Chapter 3. Chapter 4 enlists and explains the various experiments carried out as a part of this work. We made use of an open-source Java library called Lucene [8] to perform the indexing and search operations. The functionalities of this library and some of the required APIs of this library are explained in Chapter 5. We present the results obtained, after performing the various experiments, in Chapter 6. We conclude with our findings and a description of possible future work in Chapter 7.

Chapter 2

Prior Work

A considerable amount of research work has been done to harness the usefulness of the anchor text in the field of Web search and related areas as shown by Amity [19] and others, but there exists practically no or very little published work to cite the patterns formed by the anchor texts and the target pages on the Web. In this chapter, we present a brief overview of the work which has shown the significance of anchor text and has motivated us to mine the anchor text patterns with the hope to discover useful knowledge.

2.1 Improving Search Engine Results

One of the earliest and the most famous works incorporating the features of anchor texts and hyperlinks is published in the paper by Brin and Page [9] wherein they present their search engine Google. They have described a prototype of a large scale search engine which makes significant use of the hyperlinked structure in the Web. They made use of link structure and anchor text to improve the relevance judgments and quality filtering of the results returned in response to a user query.

2.2 Similarity of Anchor Text to User Queries and Titles of Web pages

Eiron and McCurley [1] conducted experiments to investigate several aspects of anchor text. They extracted features from the anchor text and based on the results of their

experiments showed that the anchor text pointing to a target page bears similarity to the title of that page and also to the user queries aimed to retrieve this target page. They demonstrated how anchor text, alone, can be used to satisfy a large number of queries.

2.3 Link Bombing

Sibel et al. [2] have mentioned that there is some correlation between the content of the keywords used in the attacking pages and the actual attacked page. All the attacking links had the same keywords in the anchor text pointing to the attacked page. In general, the other links pointing to this attacked page did not show such a correlation.

2.4 Web Page Classification

Glover et al. [3] and Johannes [4] concluded, after performing experiments, that the anchor text in the pages pointing to a target page, when available, often has a greater discriminative and descriptive power than the text in the target page itself. Thus, it is often easier to classify a web page using a combination of anchor text information provided on pages that point to it and actual information on the page instead of using the information provided on the page itself. The same proposition was put forth by Shen et al. [5] that the anchor text can be used to create implicit links between pages to help in web classification.

2.5 Identifying User Goals

Based on the observation by Eiron et al. [1] that the Web queries and anchor text are highly similar, Lee et al. [6] proposed that the anchor link distribution can be used to identify user goals automatically. Given a query, all the pages which have anchor texts similar to the query can be found and then the destination pages pointed to by these anchor texts can be obtained. By ranking these target pages, the user goals behind the query can be studied.

2.6 Anchor Text Mining for Translation of Web Queries

Lu et al. [10] have proposed an effective approach to find translation equivalents of query terms and to construct multilingual lexicons through the mining of Web anchor texts and link structures. They have exploited anchor texts as bilingual corpora to alleviate the difficulties of cross language Web search and have proved that this is particularly effective for extracting multilingual translation equivalents of query terms containing proper names or a new terminology. To deal with the translation when there is a lack of sufficient anchor texts in a particular language pair, they have extended the above approach by adding a phase consisting of indirect translation via an intermediate language.

2.7 Query Refinement

Lee et al. [6] proposed that terms appearing in anchor texts can be added to the original user queries to make the search more specialized. Kraft and Zien [7] also state that the similarity between the anchor text and the queries can be exploited to automatically complete, refine or show related queries and help users find relevant search results. They suggested a naïve algorithm which would look into the user query, find all similar anchor texts and present them to the user as query refinements or related queries.

2.8 Information about Protected Pages and Pages which cannot be indexed

Some sites restrict their pages from being crawled. The information about such pages, which have not been crawled, cannot be indexed. Using anchor text, though, information about such protected pages and other pages such as images, programs and databases can be obtained to help in the search process as the anchor text pointing to a target page is indicative of the content in that page. This idea was put forth by Brin and Page [9].

Chapter 3

Extracting Anchor Text Patterns

Observing the tremendous use of anchor text and the linked structure of the Web in various ways, we propose to get an insight of the patterns formed by these anchor texts and the corresponding target pages. Existing work exploits the link structure for different purposes but the researchers have not mined the Web to study the statistical information about the distribution of the anchor texts and the target pages. In this chapter we present our work and a detailed explanation of the various kinds of information that we have mined from the structure formed within the Web pages.

3.1 Introduction

We hypothesize our work on the fact that the actual anchor text distribution on the Web can help in enhancing the existing solutions, for various search engine related issues, that make use of anchor texts or can give a new direction to scientists pursuing research related to anchor text. To gather information about the distribution of anchor texts and the target pages, our approach is to index the pages in our dataset, a subset of the .uk Web pages, and then collect different kinds of statistics by plotting various distribution curves.

3.2 Mining the Hyperlink Structure

We started our work by parsing a subset of the .uk Web pages and extracting the information required for further analysis. This information collected from each of the Web pages was then indexed using a Java library, available for indexing and searching purposes, called Lucene (further explained in Chapter 5). This indexed data was then used to search and group data and find anchor text patterns and distributions. The generated index consisted of the following data.

ANCHOR

- A word from an anchor text of a Web page.

Each anchor text from every parsed page is split into individual words and each word, thus obtained, is indexed in the field 'anchor'. Stop words have been removed for the purposes of this study. Stemming was not used.

THREF

- The URL of the page pointed to by the anchor text.

Whenever an anchor text is encountered in a page being parsed, the target URL is extracted and stored in the field 'thref'. The target URL is canonicalized before it is indexed meaning that relative URLs have been expanded and encoding used in URLs have been made consistent before indexing.

SHREF

- The URL of the page containing the anchor text.

Whenever an anchor text is encountered in a page, the URL of the source page, which contains this anchor text, extracted at the start of parsing the page, is stored in the field 'shref'.

POS

- The relative position of the word in the anchor text.

When an anchor text is encountered, the relative position of the word, being indexed in the field 'anchor', is calculated relative to the start of the anchor text and stored in the field 'pos'.

DOCID

- The document ID of the page being indexed.

Every document (or a page) in the crawled set of pages is numbered as it is indexed. A count is used to maintain this ID which is incremented after every document is parsed. This document ID is indexed in the field 'docid'.

LCNT

- The count of the link being indexed.

Every target URL pointed to by an anchor text on a page is associated with a count. This count is incremented every time a new anchor text and the corresponding target link is found and is initialized for every new page. This link count is indexed in the field 'lcnt'.

ATEXT

- The complete anchor text.

Along with every word in an anchor text separately stored in the field 'anchor', the complete anchor text is also stored in the field 'atext'. The text to be stored in both 'anchor' and 'atext' is first converted into lowercase and then indexed.

For every anchor text in a given page, the above fields are populated and indexed as a Lucene document. This document can be used in the search phase for querying purposes. The indexing was carried out by dividing the entire dataset into subsets. These datasets were indexed in parallel. Using the merge facility, provided by Lucene, we then merged these indexes to obtain the final complete index. Based on the information stored in this index, we provide different search functionalities. These are explained below.

Subset Search grouped by Target

Given a word or a group of words as a query, this search option retrieves all the URLs pointed to by the anchor texts containing these words in the same sequence as in the search query but with zero or more words before and after it. The unique target URLs are grouped and the results are ranked based on the number of times a target URL is pointed to by the query words as the anchor text. The page pointed to the anchor text, containing the query string, the maximum number of times receives the top rank. This function helps in finding the distribution of pages based on the anchor texts pointing to them.

Subset Search grouped by Anchor Text

Given a word or a group of words as a query, this search option retrieves all the anchor texts containing these words in the same sequence as in the search query but with zero or more words before and after it. The unique anchor texts are grouped and the results are ranked based on the number of times an anchor text containing the query words is found in the complete index. The unique anchor text appearing most frequently and containing the query words receives the top rank. This function helps in finding the distribution of anchor texts with similar words.

Exact Search grouped by Target

Given a word or a group of words as a query, this search option retrieves all the URLs pointed to by the anchor texts containing exactly these words in the same sequence. The unique target URLs are grouped and the results are ranked based on the number of times a target URL is pointed to by the query as an anchor text. This function helps in finding the distribution of pages based on the anchor texts pointing to them.

In the above three search methods, the query is first converted into lowercase and then submitted for search purposes.

Search by Target

Given a URL as a query, this search option retrieves all the anchor texts that point to the given target URL. The unique anchor texts are grouped and the results are ranked based on the number of times an anchor text points to the query URL. This helps in finding the anchor text distribution based on the target URL that they point to.

These search options are used for performing various experiments as explained in Chapter 4.

3.3 Dataset Used

The dataset used for the experiments consists of the subset of pages from the .uk domain. A crawl of millions of Web pages within the .uk domain (UK2006 [17]) was already

available but the parsing and indexing of the required information from these pages was done as a part of this experimental work. Each page of this dataset was parsed and given a document ID to identify each document (or Web page) within the dataset. An example of the page contained in this dataset is *http://www.esds.ac.uk/*. The crawl of the pages was stored in .gz file format. The pages in this .gz file are stored in a contiguous manner in the WARC [11] format. Thus, for every page in the dataset, the information such as its title, URL, etc. could be extracted from the WARC file header. This data is then followed by the actual page content in the HTML form. This data was parsed to remove the HTML tags and extract the plain text of the Web page. The anchor texts and the corresponding URLs of the target pages pointed to by these anchor texts were also extracted using the HTML tag information. We have been able to successfully parse and index about 3.5 million pages out of about 3.8 million pages in the UK2006 [17] dataset.

Chapter 4

Experiments

In this chapter, we describe, in detail, the various experiments carried out as part of this thesis. We also present the description of the user interface which we designed as a part of this work to facilitate the manual exploration of the results obtained by the various search functionalities.

4.1 Implementation of Search Features

As described in Chapter 3, Section 3.2, we have studied the distributions of the anchor texts and the target pages using four different kinds of search functionalities. In this section, we explain, with an example, the method of 'subset search grouped by target'.

Consider the scenario shown in Figure 4.1. For the given anchor text 'mercedes benz', we can see that pages A, B, C and D are pointed to from various pages using anchor texts containing the given query anchor text as a subset. Thus, when we search for the target pages pointed to by the anchor text 'mercedes benz' using the 'subset search grouped by target' method, the result set will consist of pages A, B, C and D.

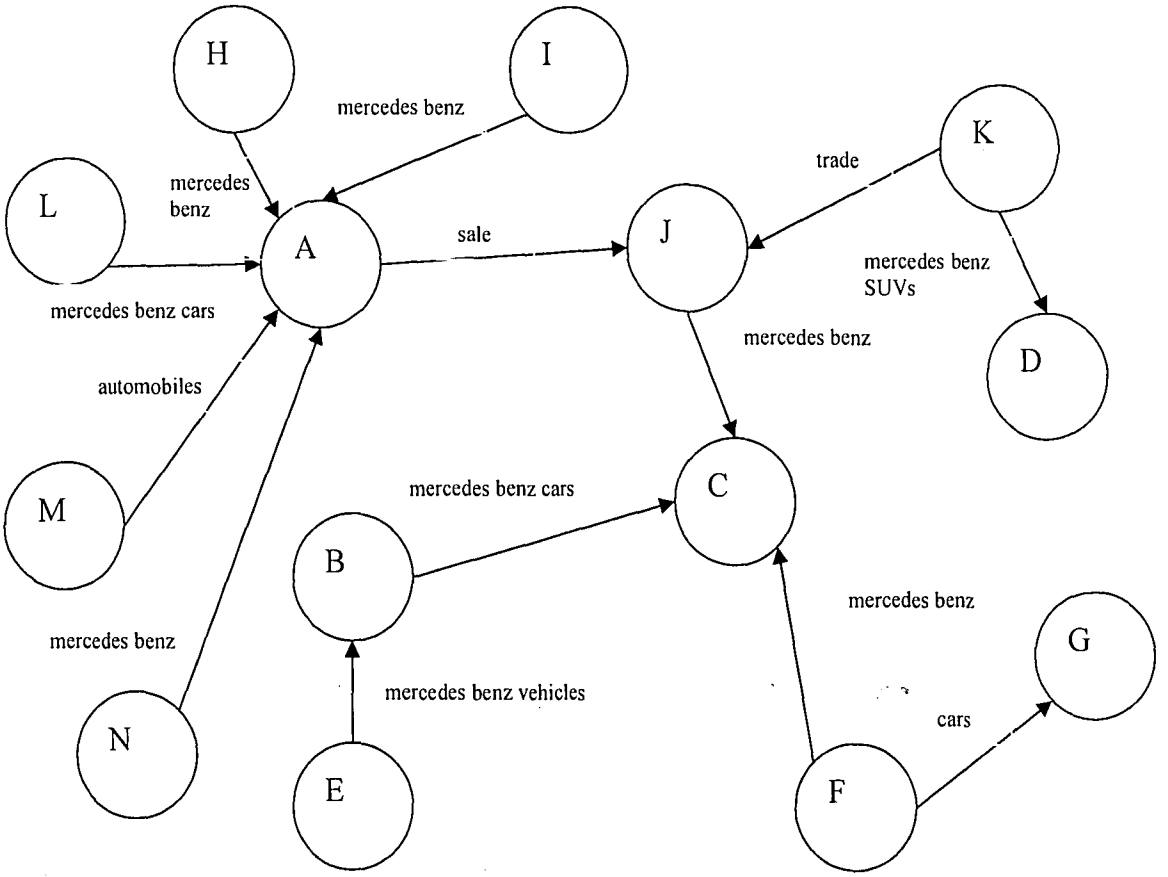


Figure 4.1

Similarly, the other search functions are implemented to return the required result sets.

4.2 User Interface

To ease the viewing of the results obtained by the various search methods, we designed and developed a Perl-CGI user interface (UI). The search results obtained by querying the indexed data are directed to the socket established and this data is read by the CGI interface program and displayed as an HTML page.

The UI displays the total number of results available for the given query. It displays 50 actual results at a time. One can view the next or the previous 50 results by using the

navigational links provided. For a given anchor text as a query, when the URLs of the target pages are displayed, 2 links are provided for each result. The first link directs the user to use that target link as a query for a new search using 'search by target' method. The second link leads the user to the actual page associated with the target URL. When the anchor texts are returned as results for the search types 'search by target' and 'search grouped by anchor text', two links are provided for each result. The first link, when clicked, uses the anchor text as a query and generates a new search for the type 'subset search grouped by target'. The second link, when clicked, uses the anchor text as a query and generates a new search for the type 'exact search grouped by target'.

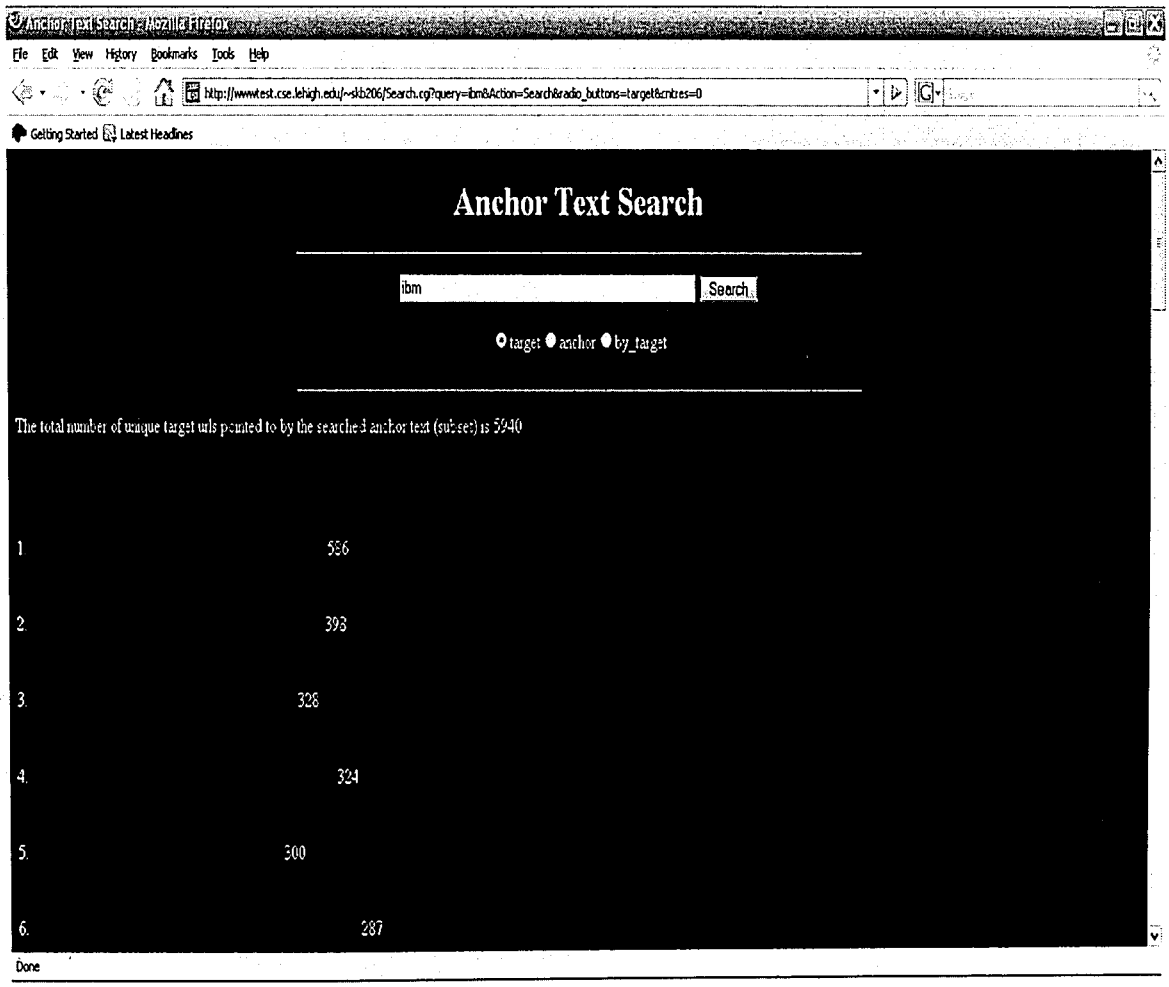


Figure 4.2 '*ibm*' used as the query anchor text to retrieve the target pages

Figure 4.3 Search of anchor texts corresponding to a target URL

4.3 Ranking the Result Sets

The results obtained by the various search options are ranked according to the scores generated by two methods.

For subset search grouped by target, using the first method of ranking, given an anchor text, all the unique target URLs that it points to are grouped and each of these unique target URLs (pages) is ranked based on the number of times it is pointed to by the given anchor text. Thus, the frequency of occurrence of the result determines its rank. Higher the frequency, better the rank.

Consider the scenario in Figure 4.1. Page A is pointed by anchor texts containing 'mercedes benz' from pages I, H, L and N. Page B is pointed to by the same anchor text from page E. Similarly, page C is pointed to by pages J, B and F and page D is pointed to by page K. Thus, according to the first ranking method, when 'mercedes benz' if given as a query, the scores for the result pages A, B, C and D are 4, 1, 3 and 1 respectively. Thus, page A receives the top rank and page C stands second in the ranking.

The second ranking method considers the significance of the anchor text for a given target URL in addition to the frequency of occurrence of that URL. The following is the formula used for calculating the score for the second method.

$$\text{Weighted score} = \frac{\text{original score of the target URL calculated by rank 1}}{\text{Number of unique anchor texts pointing to the target URL}}$$

Page A is pointed to by 2 different unique anchor texts, page B by 1, page C by 1 and page D by 1.

Thus, according to the second ranking method, the weighted new scores for page A is 4/2 i.e. 2, for page B is 1, for page C is 3 and for page D is 1. If we now rank the result pages based on the scores, page C is the top result followed by page A. The effect is that we have revised the first ranking method to give importance to a page which is more

significant for the given query anchor text than a page which is pointed to heavily by the given query anchor texts but also by other anchor texts. The modified ranking method is similar to the *tf-idf* weighting method [18] which is used to evaluate how important a word is to a document in a collection. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.

Similar score calculations are applied to the results obtained for search by target and exact search grouped by target. For the purposes of further study of the two ranking methods, we performed exact search by target on a set of 45 different anchor texts and ranked the results using the scores calculated by both the ranking methods. We then compared the ranking methods using the Kendall Tau distance.

Kendall Tau distance

The Kendall Tau distance [12] is a metric that counts the number of pair wise disagreements between two lists. The larger the distance, the more dissimilar the two lists are. Kendall Tau distance is also called bubble-sort distance since it is equivalent to the number of swaps that the bubble sort algorithm would make to place one list in the same order as the other list.

The Kendall Tau distance between two lists τ_1 and τ_2 is

$$K(\tau_1,\tau_2) = \left|(i,j) : i < j, (\tau_1(i) < \tau_1(j) \wedge \tau_2(i) > \tau_2(j)) \vee (\tau_1(i) > \tau_1(j) \wedge \tau_2(i) < \tau_2(j))\right|.$$

$K(\tau_1,\tau_2)$ will be equal to 0 if the two lists are identical and $n(n-1)/2$ (where n is the list size) if one list is the reverse of the other. In our work, Kendall Tau distance is normalized by dividing by $n(n-1)/2$. So a value of 1 indicates maximum disagreement. The normalized Kendall Tau distance therefore lies in the interval [0,1].

Kendall Tau distance may also be defined as the total number of discordant pairs as

$$K(\tau_1, \tau_2) = \sum_{\{i,j\} \in P} \bar{K}_{i,j}(\tau_1, \tau_2)$$

where

P is the set of unordered pairs of distinct elements in τ_1 and τ_2

$\bar{K}_{i,j}(\tau_1, \tau_2) = 0$ if i and j are in the same order in τ_1 and τ_2

$\bar{K}_{i,j}(\tau_1, \tau_2) = 1$ if i and j are in the same order in τ_1 and τ_2

We first compared the 2 rankings by considering all the results returned by exact search grouped by target, using 45 different anchor texts as queries. We further compared the 2 rank lists for the top k results generated by the 2 ranking methods. We varied k to be 5, 10, 15 and 20. The Kendall Tau distance calculation described before cannot be used for comparing the top k results, when k is less than the total number of results, as Kendall Tau distance formula requires that the 2 ranking lists to be compared should contain the same set of elements. Hence, we used the generalized version of Kendall Tau distance calculation as suggested by Fagin et al. [13] in their work for comparing top k lists. They have presented 4 cases possible while comparing the top k lists and have assigned a penalty Kendall Tau score to each pair wise comparison. In the case, where the results *a* and *b* in a pair to be compared lie in the top k results of one ranking list and none of them

appear in the second ranking list, the penalty we have decided is 0.5. This is a neutral approach to guessing the order of the results *a* and *b* (not in the top k results) in the second ranking list.

Note: In case of a tie between the scores of two pages, forming the pair in Kendall Tau distance measure, in the first ranking set and also in the second ranking set, then the pair is concordant even if the order between the two pages in the two ranking methods differs.

4.4 Distribution Plots

We plot different kinds of distribution patterns for the anchor texts and the target pages using the 4 kinds of search features implemented.

While parsing the pages and indexing, we maintained a list of URLs of all the Web pages that were indexed. Using this list, we searched for the number of links to each page in the dataset to find the most popular page. Also, for each page, we found the number of unique anchor texts pointing to it. Using this data, we plotted an overall distribution of anchor texts pointing to all the Web pages in the dataset.

We arbitrarily created four different lists of about 40-50 regular texts, spam texts (anchor texts believed likely to be pointing to spam pages), URLs or regular pages and URLs of spam pages. For each anchor text in the regular and the spam texts, we found the distribution of the unique target pages each points to, using the 'subset search grouped by target' and 'exact search grouped by target' methods. Similarly, for each URL

corresponding to the regular and spam pages, we found the distribution of the unique anchor texts pointing to each, using 'search by target' method.

The distribution curves were plotted using the scores calculated for the returned set of results by the first method of frequency ranking explained in Section 4.3. The distributions, thus obtained, were studied in detail to find out regularities and irregularities. We studied the peculiarities in the distribution plots of the spam texts and spam pages. We also carried out an experiment to find out if these distributions can be put to use in detecting spam from regular pages and texts. We also studied the distribution of the anchor texts and target pages on a special set of Web pages which belong to Wikipedia.

Chapter 5

Lucene

Lucene [8] is a high performance, scalable Information Retrieval (IR) library. It lets you add indexing and searching capabilities to an application. It is a mature, free, open-source project implemented in Java. Lucene can index and search any data that can be converted into a textual format. In the real world, however, documents in plain-text format are diminishing and in their place, we increasingly find information presented in rich media documents. For example, PDF and Microsoft documents, World Wide Web containing data in HTML.

Although Lucene does not support indexing of documents that are not plain text, the HTML data from the web pages was extracted and fed in text form to Lucene for indexing purposes for this study. A utility called HTML Tidy [14] was used to pull out the Document Object Model elements and then further processing was done to extract the text from these elements.

The Lucene index and search APIs and features used in this study are explained in the following sections.

5.1 Index

IndexWriter

IndexWriter is the central component of the indexing process. This class creates a new index and adds documents to an existing index. It gives write access to the index but does not let us read or search it.

Directory

The Directory class represents the location of a Lucene index. In our work, we used a path to an actual file system directory to obtain an instance of Directory, which we passed to IndexWriter's constructor. We have stored our Lucene index on a disk. To do so, we have used FSDirectory, a Directory subclass which is used by IndexWriter to create our index and maintains a list of real files in the directory specified in the file system.

Analyzer

The text is passed though an Analyzer before it is indexed. The Analyzer is in charge of extracting tokens out of the text to be indexed and eliminating the rest. Lucene comes with several implementations of the abstract class, Analyzer. Some deal with skipping stop words (such as a, an, and the); some convert the uppercase letters to lowercase and so on. In our work, we have made use of StandardAnalyzer which tokenizes based on a sophisticated grammar that recognizes email addresses, acronyms, Chinese-Japanese-Korean characters, alphanumerics, and more; lowercases; and removes stop words.

Document

A Document represents a collection of fields, a virtual document that can be retrieved at a later time. Fields of a document represent meta-data associated with that document. We created a new Document for each word for every anchor text encountered while parsing the Web pages.

Field

Each Document in the index contains one or more names fields, embodied in a class called Field. Each field corresponds to meta-data that can be queried against during search. Lucene offers four different types of fields. Each field has a name and a value associated with it. As mentioned in Chapter 3, we have stored the data in the index in the fields named *anchor, thref, shref, docid, pos, lcnt* and *atext.* Of these, the field *anchor* is stored as *Text* type of field. This type of field is analyzed and indexed. The remaining fields are stored as *Keyword* type of field. This type is not analyzed, by Analyzer, but is indexed and stored in the index verbatim. This is used to search the original value as it is during the search phase.

5.2 Search

IndexSearcher

IndexSearch has a similar meaning in searching as IndexWriter has in indexing. It is the central link to the index which contains several search methods. It is a class that opens an index in a read-only mode.

Term

A Term is the basic unit for searching the Lucene index. Similar to a Field object, it is associated with a name and a value. We can search for a value stored in a Field by specifying the name of the field and the value being searched for in the Term object.

Query

Query is an abstract class provided by Lucene for several search methods. It has several subclasses such as TermQuery, PrefixQuery and so on. The most elementary way to search an index is for a specific term by using TermQuery. This can be used to search for a Field which is indexed using *Keyword* type to get the exact match of results with the queried terms. We have used PrefixQuery to search for pages which are present in a website, whose URL is given as a query, as this type of query matches documents containing terms beginning with a specified string.

Hits

The Hits class is a container of pointers to ranked search results – documents that match a given query. Thus, when a search is made for a given query, the returned results can be accessed through a Hits object.

Chapter 6

Results

In this chapter, we present the results generated by the various experiments as explained in Chapter 4. We also present our studies of the various distribution plots.

6.1 Analysis of Regular and Spam-Finding Anchor Texts

While plotting the distribution plots, the number of results formed data point values for the X axis and the frequency of occurrence of each of the results in the dataset with respect to the query formed the data point value for the Y axis. These data points to be plotted on the graph were normalized for both the axes and then the Log values for each of these data points were plotted using the GNU Plot function [15].

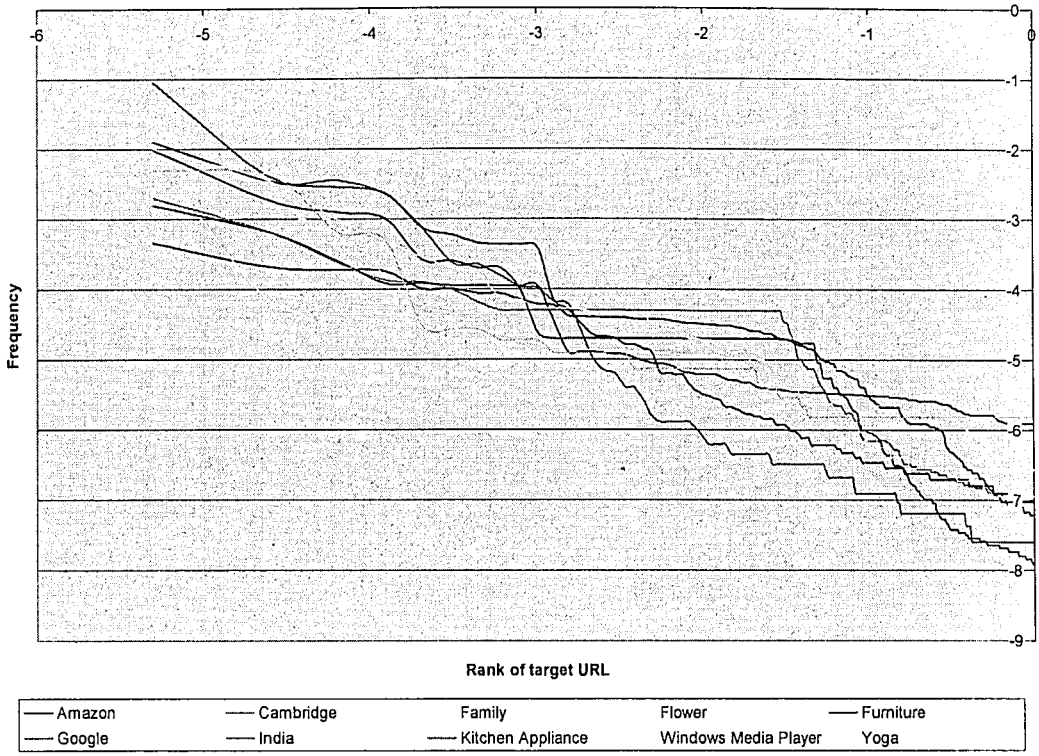


Figure 6.1 Distributions of pages pointed to by regular anchor texts

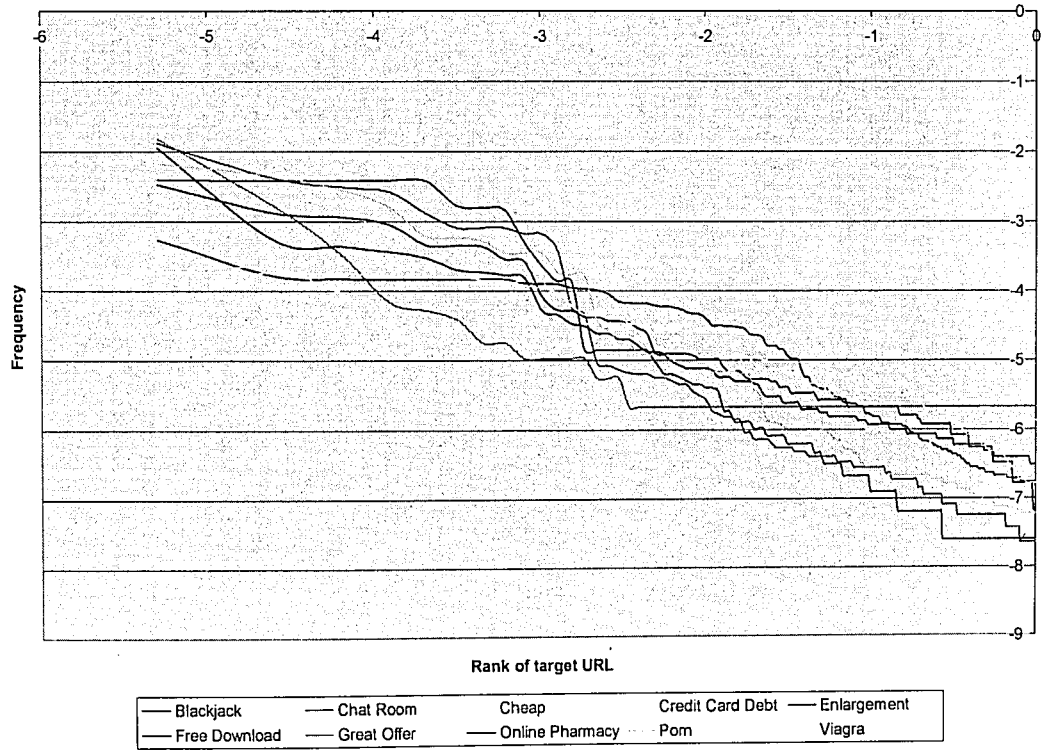


Figure 6.2 Distributions of pages pointed to by spam anchor texts

Figures 6.1 and 6.2 correspond to examples of distributions of pages for regular and spam texts respectively. A detailed study of rank-frequency distributions of pages pointed to by regular and spam anchor texts showed that on an average, these two types of graphs have the same shape. Also, by definition, they have the same direction. We found two average distributions, one each, of the distribution plots of pages pointed to by 30 regular and 30 spam texts, considering only the top 200 results. We compared these two averaged distributions with each of the 60 distribution plots using the Pearson's correlation coefficient. The same experiment was carried out considering the top 500 results. We observed that on an average, the coefficient value is more than 0.95 which indicates that the distribution of pages for both regular and spam anchor texts are highly correlated.

The distribution plots are log-log graphs. By studying all the distribution plots, we can observe that most of these follow the Zipf power law. This is true for the middle region of most of the graphs. The start of the graphs has a peak and some variations whereas the graph trails at the end and remains constant suggesting many unique unimportant pages for the given anchor text.

We studied the pages with equal frequencies corresponding to data points forming horizontal lines in the graphs. For most of the anchor texts, the pages - pointed to by them - showing such a pattern belong to the same Web site. The anchor texts are focused to point to the same Web site but are distributed among the pages of the site.

6.2 Analysis of Regular and Spam Pages

We studied the distribution plots for the regular and spam pages. The number of unique anchor texts for most of the regular pages was very low. Hence, we could not further study these graphs as we did not have sufficient data points to make any findings. We then plotted the data for the distribution of unique anchor texts for an identified set of spam pages. The study of these plots revealed some interesting facts.

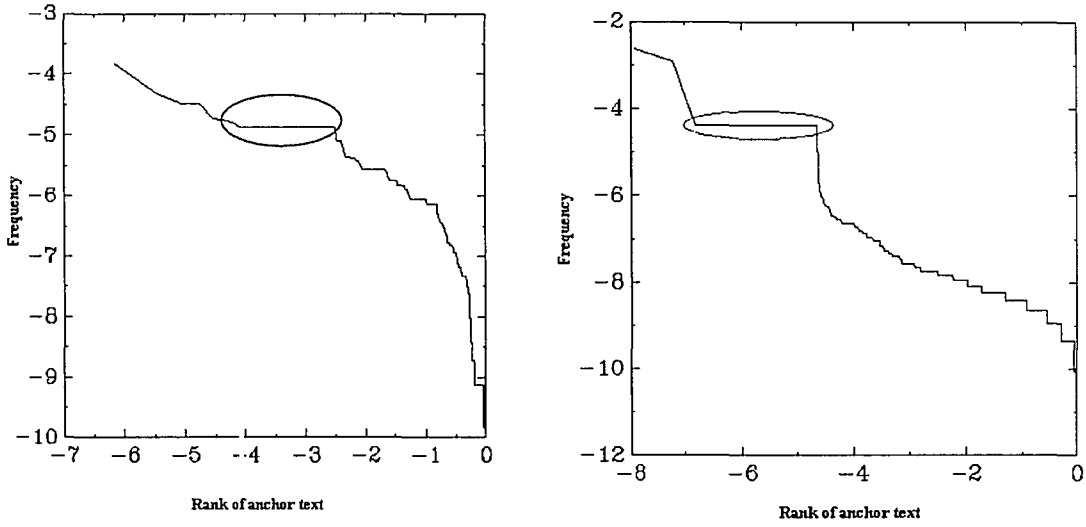


Figure 6.3 Distributions of anchor texts for spam pages 'http://24hourhealth.co.uk' and 'http://a2zcheats.co.uk'

We can see in Figure 9 that the plots for spam pages contain horizontal regions (marked in red) which show that some anchor texts point in equal numbers to these pages. We studied these anchor texts and found out that these anchor texts have something in common with each other. They share some common features.

The prominent horizontal region for 'http://24hourhealth.co.uk' corresponds to the following anchor texts having the same frequency - family medicine, nuclear medicine, emergency medicine, sleep medicine, vascular medicine, preventive medicine, sports medicine, internal medicine, comparative medicine, mountain medicine, occupational medicine, narrative medicine and tropical medicine. Similarly, the horizontal region for 'http://a2zcheats.co.uk' corresponds to the following anchor texts having the same frequency – f, g, d, e, b, c, n, o, l, m, j, k, h, i, w, v, u, r, q, p, y, x, a, z. Thus, we can see that the anchor texts with the same frequency follow some common pattern. Some graphs also have anchor texts with common words. We studied about 30 other similar graphs and saw that they follow the same trend.

6.3 Comparison of Ranking Methods

As explained in Chapter 4, we ranked the search results using two different scoring methods. We compared the 2 ranking lists, thus generated, using Kendall Tau distance metric. The comparison was done for all the results, top 5, 10, 15 and 20 results. This study was carried out for 45 anchor texts. The results obtained are presented in Tables 1 and 2.

Anchor Texts	Kendall Tau Distance
amazon	0.18348
amsterdam	0.37451
award	0.31516
bank loan	0.25882
bank	0.32552
bmw	0.34051
cambridge university	0.44594
celebration	0.46125
cell phone	0.28626
china	0.33111
clip art	0.32890
family	0.41015
flower	0.32169
furniture	0.40389
gemini	0.34719
germany	0.38773
google	0.08891
hair style	0.07422
hello	0.37940
ibm	0.32578
india	0.35194
internet explorer	0.32484
kitchen appliance	0.31226
literature	0.37131
maternity	0.21351
music video	0.22635
olympic	0.35354
online dating	0.13971
palace	0.43569
queen	0.34083
rain	0.25858
result	0.35966
roger	0.43492
rose	0.35090
salary	0.26447
search engine	0.39176
student	0.33041
tennis	0.38770
tower	0.45270
united kingdom	0.40698
united states	0.45784
windows media player	0.31109
yoga	0.36514
Average	**0.33099**

Table 6.1 Kendall Tau Distance for comparison of complete result sets obtained from the two ranking methods

Anchor Texts	KT Dist. (k=5)	KT Dist. (k=10)	KT Dist. (k=15)	KT Dist. (k=20)
amazon	0.00000	0.09091	0.07500	0.10672
amsterdam	0.13333	0.27273	0.22794	0.19368
award	0.00000	0.00000	0.00833	0.07905
bank loan	0.20000	0.10909	0.13971	0.15942
bank	0.28571	0.32967	0.36190	0.32615
bmw	0.33333	0.21818	0.19853	0.19203
cambridge university	0.42857	0.55147	0.61231	0.60963
celebration	0.23810	0.31868	0.44928	0.47782
cell phone	0.00000	0.00000	0.04167	0.11594
china	0.61905	0.42424	0.30719	0.24275
clip art	0.64286	0.46154	0.45022	0.43350
family	0.42857	0.27273	0.22059	0.18421
flower	0.00000	0.00000	0.05147	0.09486
furniture	0.33333	0.42857	0.45059	0.47312
gemini	0.33333	0.27273	0.24561	0.31746
germany	0.33333	0.37363	0.28333	0.20346
google	0.72222	0.45455	0.33333	0.24275
hair style	0.20000	0.16364	0.12500	0.09524
hello	0.00000	0.23077	0.31429	0.38161
ibm	0.69444	0.52747	0.41176	0.32246
india	0.46429	0.33333	0.31373	0.26087
internet explorer	0.52381	0.23636	0.15000	0.19000
kitchen appliance	0.28571	0.32051	0.26471	0.23333
literature	0.33333	0.23636	0.29474	0.35185
maternity	0.00000	0.09091	0.05833	0.08225
music video	0.20000	0.09091	0.17544	0.30788
olympic	0.66667	0.40659	0.39048	0.42118
online dating	0.26667	0.10909	0.13235	0.19000
palace	0.53571	0.62092	0.57609	0.56439
queen	0.20000	0.11111	0.04762	0.02632
rain	0.06667	0.10909	0.14379	0.19000
result	0.50000	0.52500	0.49012	0.48387
roger	0.33333	0.32967	0.45290	0.50379
rose	0.20000	0.31868	0.39394	0.38360
salary	0.53571	0.50476	0.42632	0.35333
search engine	0.20000	0.09091	0.12500	0.12857
student	0.23810	0.18182	0.16176	0.13439
tennis	0.64286	0.57143	0.38562	0.34333
tower	0.60714	0.57353	0.62108	0.63333
united kingdom	0.60714	0.42308	0.35263	0.32806
united states	0.26667	0.15152	0.15441	0.18841
windows media player	0.77778	0.69118	0.62338	0.57258
yoga	0.38095	0.29487	0.24183	0.20553
Average	**0.32797**	**0.28494**	**0.27298**	**0.27397**

Table 6.2 Generalized Kendall Tau distance for comparison of top k results

From Tables 1 and 2, we observe that, on an average, there is 33% disagreement between the two ranking methods. We can see that the weighting of the frequency score of a page by the number of number of anchor texts pointing to it does not significantly change the ranking of the page. This indicates that the pages are mostly associated with one important anchor text containing the query (which contributes to the main score to the page) and few other unimportant anchor texts. Also, we can see that there is higher degree of agreement between the two ranking methods for a higher value of k. This shows that the two ranking methods differ mostly in the top results.

6.4 Comparison of our ranking methods with Relevance Ranking method using Precision@10

For a set of 25 queries, we have a list of .uk Web pages which are evaluated to be relevant or irrelevant for each of these queries. For these 25 queries, we obtained the complete set of results by using them as anchor texts in our search method. We then found the top 10 results from this set which also lie in the evaluated set of pages. (These were not necessarily the top 10 ranked pages in our result set but could be ranked lower.) Using the evaluation present for these 10 pages, we calculated the number of pages which are relevant. Similarly, we calculated the number of relevant results from the top 10 evaluated results obtained using the BM25F [16] ranking method. The results are presented in Table 6.3. The performance achieved was the same for both our ranking methods. So, we present only one set of results in Table 6.3.

Query	Prec@10 using Our Ranking Method	Prec@10 using BM25F ranking
weather	0.70	0.70
toys	0.70	0.70
playstation	0.50	0.40
auctions	0.70	0.30
disney	0.30	0.20
software	0.80	0.80
wine	0.50	0.60
wedding	0.50	0.30
chat room	0.70	0.50
microsoft	0.50	0.40
digital camera	0.80	0.70
electronics	0.50	0.60
music	0.10	0.50
photography	0.70	0.60
mp3	0.50	0.10
furniture	0	0.20
credit cards	0.20	0.20
golf	0.60	0.40
used cars	0.30	0.40
travel	0.20	0.20
shoes	0.30	0.40
flowers	0.30	0.60
shopping	0.30	0.50
magazines	0.50	0.80
hotels	0.80	0.90
Average	0.48	0.48

Table 6.3 Prec@10 Comparison of 'our ranking method using anchor text' and the ranking using BM25F scores

We can see that ranking the pages using anchor texts containing the query string gives comparable results as that of the pages returned by the BM25F ranking method. We should note that we get a comparable performance using our method in spite of not using the results which are ranked top 10 but are not evaluated. These unevaluated results are relevant in most of the cases as shown in the next section.

6.5 Manual Relevance Check

The top 10 results of the same 25 queries used in the previous section were evaluated manually for relevance. We observed that almost 90% of these results are relevant. In Table 6.4, we present the top 10 results for 3 of the queries.

Weather	Hotels	Photography
http://www.timesonline.co.uk/weather	http://www.placestoholiday.com	http://photography.abcaz.co.uk
http://www.thisisyork.co.uk/york/weather/	http://www.hotels-stay.co.uk/	http://photography.ebay.co.uk/
http://www.tiscali.co.uk/weather/	http://www.cheapaccommodation.com	http://www.ncl.ac.uk/photos/
http://www.met-office.gov.uk/weather/europe/uk/uk.html	http://www.hotels-england.co.uk/sitemap4.htm	http://www.waggydog.com/posters/photography/
http://weather.linkspider.co.uk/	http://www.findhotel.co.uk/help.htm	http://audiovisual.kelkoo.co.uk/b/a/c_100325623_photography.html
http://stage.manchesteronline.co.uk/weather/	http://www.airport-hotels-discount-travel.co.uk	http://www.minigallery.co.uk/search/index.asp?sMedia=photography
http://www.sundaymail.co.uk/weather	http://www.travel-europe-direct.co.uk/index.cfm	http://helpzone.leedsmet.ac.uk/photography.htm
http://weather.yahoo.com/regional/UKXX.html	http://www.hotels-discount-travel.co.uk	http://www.bristolcameras.co.uk/digital-photography.htm
http://weather.travel-guide.com/weather/	http://www.westcountrylinks.co.uk/content.htm	http://photography.findandcompare.com
http://www.bbc.co.uk/weather	http://www.1000-hotels.co.uk/france/france-directory.html	http://www.hush-hush.co.uk/directory/Photography/

Table 6.4 Top 10 results for 3 queries used for anchor text search

We can observe that almost all of the top 10 results for these 3 queries, when used as an anchor text, are relevant. The same was observed for the other 22 queries. Looking at these results, we believe that our ranking method may give higher precision (perhaps better than BM25F) if the complete set of pages in the result set for a given query is evaluated.

Chapter 7

Conclusions

This final chapter presents our findings and conclusions based on the experiments carried out and the results thus obtained. We also put forth the possible applications of this work. We intend to pursue further research into this field which could not be done as a part of this thesis. The ideas for the same are presented in the last section of this chapter.

7.1 Findings

Based on the results obtained from the experiments, we have been successful in answering the questions which we posed at the start of our work. We have indexed the data and have been able to calculate the rank-frequency distribution of the pages pointed to by a given anchor text and of the anchor texts pointing to a given page. We have observed that these distributions follow the Zipf power law for most of the plot and then have a trailing end with low frequency results. Thus, on the whole, we see that these graphs appear to have heavy tailed distributions.

We also found out that the rank-frequency distributions of pages pointed to by regular and spam anchor texts are highly correlated indicating a significant similarity between the ways in which the regular and spam pages are being organized using the hyperlink structure. It was observed that the anchor texts pointing to a given spam page with the

same frequency display common patterns and common words. Also, given an anchor text, the target pages with the same frequency mostly belong to the same Web site. The experiment carried out by averaging the distributions of pages pointed to by regular and spam texts and then comparing the average with each of the individual distributions, to train a classifier for classifying pages as spam, was unsuccessful due to the high correlation between the distributions.

We can also conclude from the results obtained in Sections 6.4 and 6.5 that our ranking method using anchor text gives comparable performance as that achieved by ranking the pages using the BM25F method. We believe to get better performance if we have an evaluation of all the result pages returned for a query.

7.2 Applications

Based on the prior work presented in Chapter 2, the following are the proposed applications of this study and the results obtained from it.

- The anchor text distribution can be used to identify user goals as suggested by Lee et al. [6]. Given a query, the distribution of the anchor texts, which are similar to the query, can be obtained. Let us consider the top k (variable parameter) ranked anchor texts from this distribution. We can obtain the distribution of the target pages for each of these anchor texts. The information obtained from the top ranked pages can then be used to help in automatic identification of user goals.

- The distribution can also help in finding better methods to deal with the problem of link bombing. For example, a page linked by a high frequency anchor text and other equally distributed frequency anchor texts can suggest that the page is possibly under link bombing attack according to the work carried out by Sibel et al. [2].
- The distribution curves also suggest ways to refine the user queries. By obtaining the distribution of all the anchor texts containing the query words, the top ranked anchor texts can be suggested to users as variations for the given query.

Also, the results of the study, once performed on a larger and varied dataset, can throw a different light on the current use of anchor text in Web search.

7.3 Future Work

The ideas proposed in Section 7.2 may be considered for experimentation in the future. Apart from these, the study can be extended to index a larger set of Web pages from the same .uk domain. Also, datasets from other domains can be parsed and indexed, and the distributions for this data can be studied and compared with the results we have obtained in the current study. This will help us better analyze the commonalities or differences in the anchor text distribution patterns. Also, while indexing, the stop words can be preserved and the results for the same can be studied. For the purpose of our current work, the results for a given search of an anchor text were obtained by searching anchor texts having the words in the query in the same sequence. In future, one can consider two

anchor texts to be similar even if the sequence of words in them differs. Also, the 'pos' field in the current index can be put to use by considering the similarity between anchor texts and the query on the basis of some threshold distance rather than an exact match of the words. A variation of the weighted score ranking of the target pages can be tried wherein the fraction of all incoming links can be used to weight the frequency of results instead of using the fraction of all unique anchor texts as explained in Section 4.3.

We would like to study and compare the distributions of the words in regular text and anchor text in the given dataset to find out if there is a difference in the ways in which people use the wordings of the content of the pages and the anchor texts used to link other pages. Also, we would like to be able to study the distribution patterns in more depths to find out ways to find pages with similar distribution. We can then find out if we can use this information for purposes of clustering or classification of Web pages.

Bibliography

[1] N. Eiron and K. S. McCurley. Analysis of anchor text for web search. *In Proceedings of the 26th SIGIR,* 2003.

[2] S. Adah, T. Liu and M. Magdon-Ismail. An Analysis of Optimal Link Bombs.

[3] E. J. Glover, K. Tsioutsiouliklis, S. Lawrence, D. M. Pennock and G. W. Flake. Using web structure for classifying and describing web pages. *In Proceedings of the 11th International Conference on World Wide Web,* pages 562–569, Honolulu, Hawaii, USA, 2002.

[4] Johannes Furnkranz. Exploiting Structural Information for Text Classification on the WWW. *In Proceedings of IDA,* 1999.

[5] D. Shen, Z. Chen, Q. Yang, H. Zeng, B. Zhang, Y. Lu and W. Ma. Web-page Classification through Summarization. *In Proceedings of SIGIR,* 2004.

[6] U. Lee, Z. Liu and J. Cho. Automatic Identification of User Goals in Web Search. *In Proceedings of WWW,* 2005.

[7] R. Kraft and J. Zien. Mining Anchor Text for Query Refinement. *In Proceedings of WWW,* 2004.

[8] O. Gospodnetic and E. Hatcher. *Lucene IN ACTION*, Manning Publications 2005.

[9] Sergey Brin and Lawrence Page. The Anatomy of a large-scale hypertextual web search engine. *In Proceedings of WWW,* 1998.

[10] W. H. Lu, L. F. Chien and H. J. Lee. Anchor Text Mining for Translation of Web Queries: A Transitive Translation Approach. *ACM Transactions on Information Systems,* 2004.

[11] *http://www.iwaw.net/05/kunze.pdf*

[12] *http://en.wikipedia.org/wiki/Kendall_tau_distance*

[13] R. Fagin, S. R. Kumar and D. SivaKumar. Comparing top k lists. In *Proceedings of 14th ACM-SIAM Symposium On Discrete Algorithms (SODA),* 2003

[14] *http://tidy.sourceforge.net/*

[15] *http://www.gnuplot.info/*

[16] S. Robertson, H. Zaragoza and M. Taylor. Simple BM25 Extension to multiple Weighted Fields. *CIKM'04,* November 8-13, 2004, Washington, DC, USA

[17] Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Massimo Santini and Sebastiano Vigna. A Reference Collection for Web Spam. *In ACM SIGIR Forum,* Vol.40, No.2, pages 11-24, Dec 2006.

[18] *http://en.wikipedia.org/wiki/Tfidf*

[19] Einat Amitay. Hypertext: The Importance of being Different. MSc Dissertation, Centre for Cognitive Science, The University of Edinburgh, Sep 1997 and Technical Report No. HCRC/RP-94.

Vita

Name: Shruti Krishna Bhandari

Place of birth: Sirsi, Karnataka, India

Date of birth: June 5, 1983

Father's name: Krishna V. Bhandari

Mother's name: Shailaja K. Bhandari

Bachelor of Engineering: Computer Engineering, Maharashtra Institute of Technology, Pune, India, 2004

END OF TITLE