

Using the Generic Document Profile to Cluster Similar Texts.

Jeremy Ellman¹

School of Computing & Information Systems
University of Sunderland, UK.
Email : Jeremy.Ellman@mari.co.uk

Abstract

An Exemplar Text is the ideal model result for Web searches: It characterises what we would like our queries to find. Yet, this rarely happens, since the World Wide Web (WWW) suffers from several deficiencies as an Information Retrieval system. Firstly, the Web does not contain a document collection since the material it contains has not been gathered for one subject or purpose. It is also incompletely indexed because of its size. Consequently retrieving text from a search engine is time consuming and laborious.

Hesperus is a system designed to address this problem. Using an electronic encyclopaedia as a source of subject defining texts, queries are made to MetaCrawler. This searches several search engines in parallel returning the best 10-20 links. Hesperus retrieves these web pages, and computes their conceptual similarity to the Exemplar Text using a method based on thesaurally determined lexical chains.

Initial results show that users prefer the Hesperus page order to MetaCrawler's statistical ordering. The technique consequently shows promise as a way to improve the effectiveness of Web searching.

KEYWORDS: Web searching, conceptual similarity, thesaurally defined lexical chains.

1. Introduction

The Internet World Wide Web[1] is an incredibly useful resource that has several deficiencies as an Information Retrieval system. Unlike most collections, it contains a highly heterogeneous document set. This means that query terms may be interpreted with greater ambiguity with respect to those used in a single subject IR collection. Krovetz and Croft[2] for example have noted that the nature of the collection may act to disambiguate query terms, whilst Sanderson[3] has shown the word sense disambiguation needs to be highly accurate to positively influence retrieval performance. Nevertheless, simple queries to the common search engines consequently generate many possible links that seem unrelated to the search terms used.

There is no consistent editorial content control over Internet Web pages. The web consequently contains material of very variable quality. Since downloading pages is frequently time consuming, poor material may hide more informative pages considered less relevant by a search engine. This phenomenon has been called "Information Burial"[4]

There is also a problem that everyone meets when working in a new area: Using the correct terminology is essential for accurate searching. This so-called "Vocabulary Problem" was noted long ago[5] although in the context of Human Computer Interaction.

One approach to these problems is to use what we call "Exemplar Texts". An exemplar is "a person or thing to be imitated; the ideal model, an example" (source: Microsoft Bookshelf). Thus an Exemplar text represents the kind of output that would exemplify a successful search. Since it also contains the key terminology that users new to an area may not recognize as distinctive, it would also alleviate the vocabulary problem.

Exemplar texts could be found by personal recommendation, or through "Recommender Systems"[6,7]. As in relevance feedback[8], we anticipate improved retrieval performance, by reference to a document known to be more appropriate for a query. Unlike relevance feedback though, we wish to use exemplars as "known" examples of good quality replies, without interactive user involvement. This could be considered a type of automatic query expansion.

Query expansion has been studied in Information Retrieval for almost thirty years. Qiu & Frei [9] for example report a system that improves retrieval performance by about 20-30% by using an automatically constructed similarity thesaurus based on domain knowledge. Magennis[10] has however argued that automatic techniques are inferior to interactive methods. The context of the Internet makes this undesirable though since retrieving texts is subject to noticeable delay.

For our approach to work, we need to be able to match Exemplar texts against candidates retrieved from the

¹ Postal Address: MARI Group Ltd Wansbeck Business Park Ashington NE 63 8QZ

Web. The most intuitive option is to stem the words in the texts, and generate term vectors. These could then be compared using standard algorithms such as cosine similarity (eg [7]). The problem here though is that there are simply too many terms: Texts would need to be very similar in order to match. Furthermore, we would still not have tackled the vocabulary problem. Our exemplars and candidates would still need to contain the exact same terms for relatedness to be measured.

The ‘Generic Document Profile’ is designed to address these concerns. It is a weighted vector of semantic categories derived from Roget’s thesaurus. This contains approximately one thousand subject headings. Although this number is large, it is possible to rank attribute-value vectors of this size using a nearest-neighbour algorithm in a classic case based reasoning approach. This is appropriate since the Exemplar texts represent previously successful problem solutions.

This approach is similar to that of Rada & Bicknell[11]. However, their thesaurally based ranking system was specific to the MeSH medical collection. The approach described here is intended to be general.

Note that this representation is designed as a secondary ordering metric over that produced by the web search engines. It depends upon non stopwords being present in the thesaurus. Thus, it is not appropriate for specific queries that contain proper names, which tend to perform well in web searches anyway.

We have built an Internet web agent to investigate the utility of the Generic Document Profile. This system (known as ‘Hesperus’) posts queries to MetaCrawler [12], and compares the output to the Exemplar texts chosen. We have carried out initial experiments of the systems, and compared the output to rankings obtained from human judgements - with interesting results.

The paper proceeds as follows: Firstly, we describe the architecture of Hesperus. Then we describe how text similarity may be assessed using the generic document profile. Next we describe experiments to evaluate the approach, present some conclusions and planned future work.

2. Hesperus: A System to Meta-Search the Web for Similar Texts

Hesperus is a research system designed to enhance the results of current search engines. It is made up of two conceptual modules (see fig 1 below), a User Interface (UI), and Page Processor. The UI accepts the query that identifies the exemplar text, and posts this to MetaCrawler [12]. This queries AltaVista, Excite, Yahoo! etc, and collates the replies into a common list of the thirty or so most highly ranked.

Hesperus retrieves these pages using a multi-threaded Java core. This greatly increases task speed, since most of the time spent in retrieving web pages is used in establishing connections, waiting for slower sites etc. By multi-threading, pages from the faster sites can still be retrieved whilst several threads are blocked waiting for pages from the slower ones.

When the pages have been retrieved, they are passed on to a ‘Page Processor’ suite. This determines the generic document profile for each page and places the result in a Case Library. This process is described in more detail below.

The profiles can then be clustered for similarity to the exemplar texts using the nearest neighbour algorithm as is common in Case Based Reasoning [13]. The architecture of the system is shown graphically below in Figure 1.

3. Text Similarity Assessment

In Case Based Reasoning (CBR) a query and examples are usually represented as attribute value pairs. Thus to apply CBR to document comparison, both the text acting as a query, and the documents to be compared against need to be represented equivalently. If this representation is based on simple terms (i.e. words), the problem becomes hugely complex, since there are about 100,000 words in the English language. This would also be a fragile approach, since semantically equivalent words would not count as equal. However, If we represent a document as Roget categories, the problem becomes tractable, since there are only 1024 main thesaurus categories². We call this representation the ‘*Generic Document Profile*’, since it is not word specific, and is derived from the whole text. Now all that remains is to describe how a text may be analysed so that values in each particular category can be determined. This is based upon the ‘lexical chains’ the text contains.

² There is considerable work on lexical chaining using WordNet 1.5 (eg Stairmand & Black 96). However, with 91,595 semantic categories this would not be computationally tractable either.

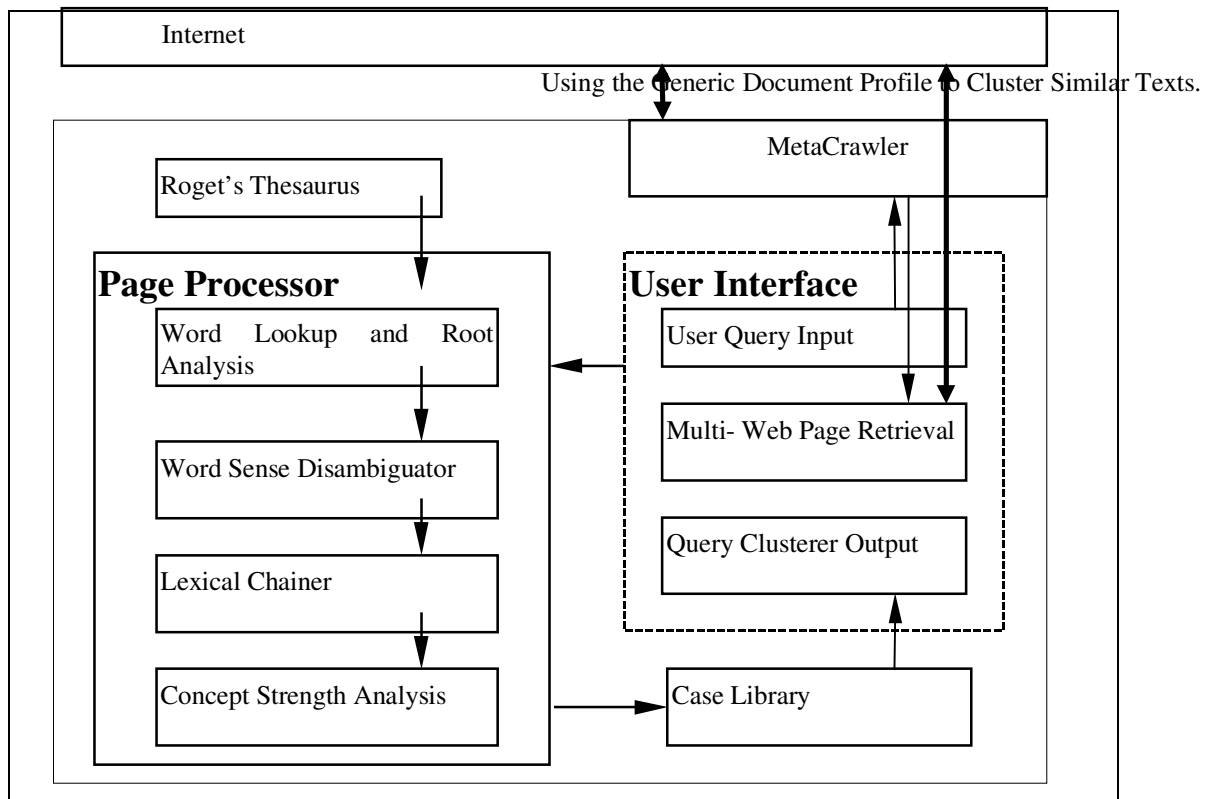


Fig1. Hesperus System Architecture

What is a lexical chain?

A lexical chain is a set of words in a text that are related by both proximity, and by relations between the words that may be determined using a Thesaurus, such as Roget's or WordNet. Since Morris and Hirst[14] introduced the idea, lexical chains have been applied to several different areas of language processing. These include word sense disambiguation and text segmentation [15], malapropism detection [16,17], detection of HyperText links in newspaper articles[18], and lexical cohesion and Information Retrieval[19,20]

We look for several relations between word pairs to decide if they are members of the same lexical chain. This is a subset of the relations suggested by (and described fully in) [14]. The most important of these is term (or term stem) repetition, known as the ID relation. Next we check for membership of the same thesaural category (CAT relation). Since Roget's thesaurus contains groups of categories we also check if words are members of appropriate neighbouring thesaural categories may be linked (GROUP). Roget categories often refer to other categories. Words may therefore be related where one word's entry refers to an entry that contains the other word (ONE).

Creating Chains from Text

The algorithm used to create chains was derived from that given in StOnge[16]. The aim is to chain every non stopword, and store the resulting chains in a value ordered stack. This offers the strongest chains for further growth to promote context. The algorithm is summarised below.

Creating the Generic Document Profile

The Generic Document Profile is created from the lexical chains. The strength of every link in every chain is taken, and summed into the appropriate profile category. This gives the required attribute value representation.

Now we have described Hesperus, we have to determine how well it works. This was done experimentally.

The Lexical Chaining Algorithm

1. get a new nonstop word from the input or terminate
2. Disambiguate Word based on local context
3. For all chains in the queue
4. for all the links in the chain
5. try to link the input word to a word in that chain using ID
6. IF link is made add word to the front of the chain
7. IF a link is made using one (of several) word senses
8. then DELETE other word senses
9. calculate new chain weight, and re-sort the queue
10. goto STEP 1
11. IF input word greater than the MAXDIST (ie 200 words) break
12. IF no ID link Repeat step 2-10 using CAT check
13. IF no CAT link Repeat step 2-10 using ONE check
14. IF no ONE link Repeat step 2-10 using GROUP check
15. IF word can not be linked form a new chain

16. The Experiments

The purpose of the experiments was to evaluate Hesperus' Similarity matching in a realistic usage scenario. This involves comparing its rating judgements to those of human assessors and the raw MetaCrawler ranking. Of course, it is perfectly possible that an IR program would produce quite different document rankings to MetaCrawler if given the retrieved pages as a 'mini-collection'. To check for this, the page set retrieved was indexed using SWISH³. The raw query was then posed against this index as a control

Experimental Design

There were three stages in the experiment:

1. Selected Queries were posted to MetaCrawler
2. web page set retrieved were ranked in order by people
3. Pages were re-indexed and query posted against this page set.

Two principal factors need to be in selecting queries: The identification of material for the Exemplar texts, and the length of queries to the Web search engines.

There are similarly several criteria for the selection of exemplar texts for Hesperus:

1. Are general --rather than specialist (since Proper nouns can not be chained)
2. Do not reflect experimenter bias
3. Are available in Machine Readable form (for experimentation and replication)

Articles from Microsoft Encarta '96 (a multimedia encyclopaedia CD) fulfil all these criteria. Additionally, Encarta has a subject index that is a mixture of general topic names, and proper names. The latter are inappropriate for Hesperus, since proper nouns are not usually included in Roget's thesaurus, but are well served by the more usual (ie tf*idf) approaches to Information Retrieval used by the common search engines.

It is well known [22] that Internet queries average about 1.7 words long. This poses a significant problem since we need to choose queries that are random, and for which there is a known exemplar text. Queries were consequently selected by selecting index entries from Encarta at random. Since proper names are not appropriate for this technique, when one was found it was skipped. The two queries used for our initial pilot tests were:-

- (a) Modern Science.
- (b) Indigenous Peoples.

Queries were posted to Hesperus, which then retrieved all the links identified by MetaCrawler. MetaCrawler typically, indicates at most thirty links, of which a percentage can not be retrieved since they have moved or the server is down etc. Of the pages retrieved a number only contain graphics, or have little or no text. These are consequently eliminated from the sample set.

The pages retrieved were then rated by human assessors. To simplify the experiment the first two pages retrieved of the best ten remaining pages were given to fifteen University undergraduates who were instructed to order the best five pages in order of usefulness for the general query posted to Hesperus. As a control measure, the Exemplar text from Encarta was included in the test set.

³. SWISH is a freely available basic IR program that understands HTML format, and increases the rank of terms found in HTML heading tags. URL: <http://sunsite.berkeley.edu/SWISH-E/>

Finally, the mini page set was queried using the original query posted to MetaCrawler.

Results: Query 1

The results of the first query “Modern Science” are shown in fig 2 below.

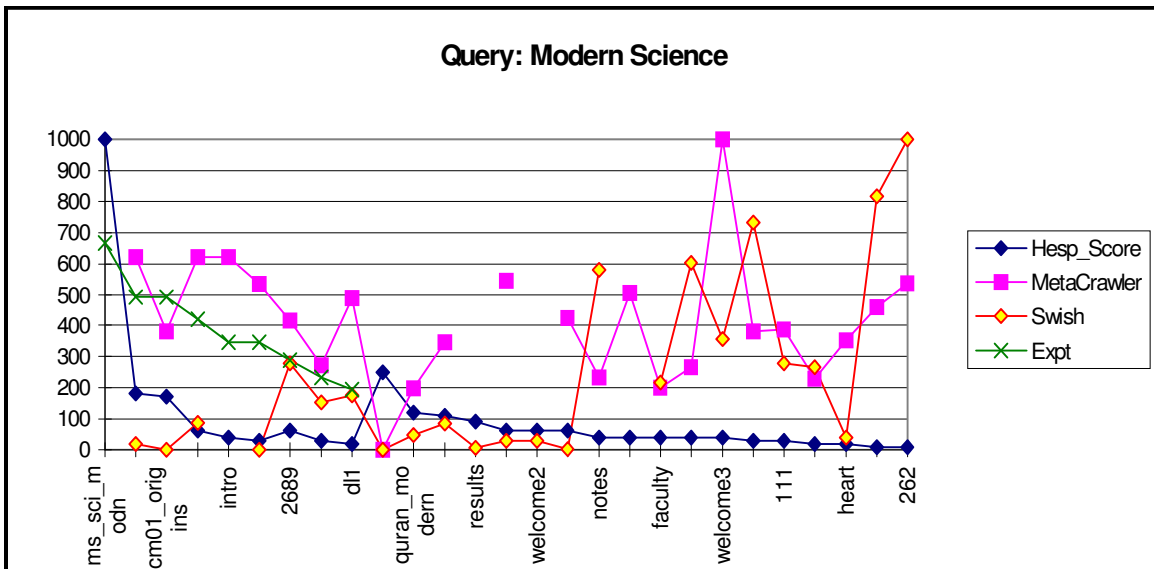


Fig 2: Experiment 1

The human experimental data is represented by the mean from the subject pool. Although there was poor inter-subject agreement, note that the “ms-science” extract was rated most useful.

Care should be taken in interpreting the numeric data used in the graph. This has been rebased to fit on to the 0-1000 scale used by MetaCrawler, but the numbers were derived on completely different bases and should not be directly compared. However, it is appropriate to compare order, or relative rank information.

Spearman’s rank correlation coefficient between the human judgements and Hesperus was calculated: $r_s=0.9041677$. This is highly significant ($p << 0.01$).

There are no significant pairwise correlation between the human judgements, MetaCrawler, or Swish.

Results: Query 2

The results of the second query “Indigenous Peoples” are shown in fig 3 below.

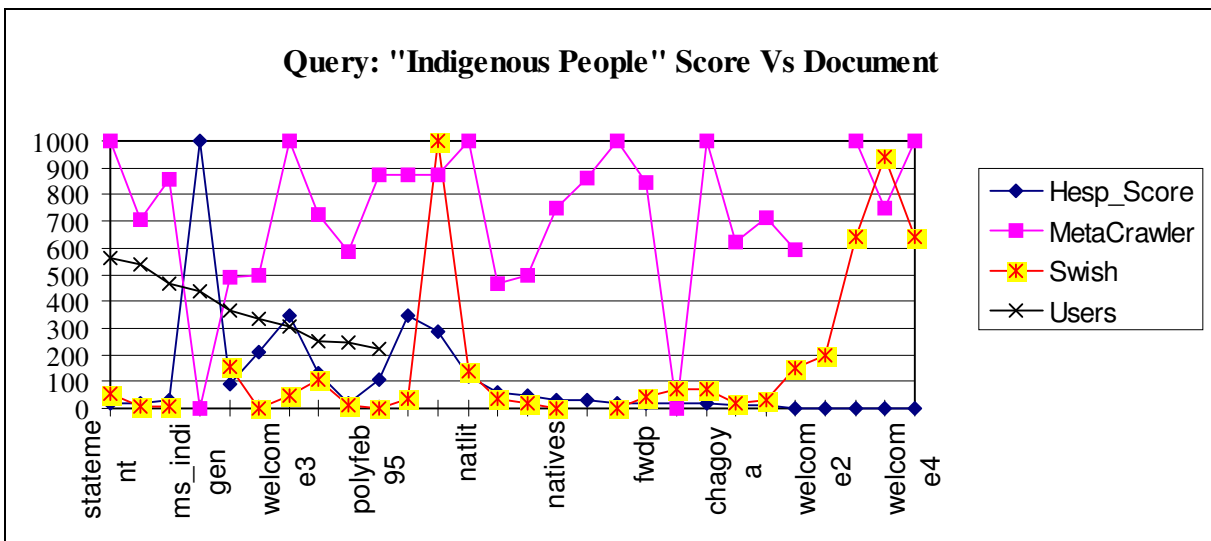


Fig 3: Experiment 2

Unlike Experiment 1, Experiment 2 yielded no significant rank correlation between Hesperus, human judgement, 1st Annual CLUK Research Colloquium 1998

Swish, or MetaCrawler. Furthermore, the Exemplar Text (identified as “ms_indi”) was only ranked as fourth most useful by the users.

Had the exemplar text actually been the page rated most highly by user Hesperus would again have produced a significant rank correlation: $r_s = .63333$. ($p < 0.05$).

We can speculate on the reasons for this. Firstly, the two queries are quite different in content. ‘Modern Scientists’ do not usually refer to themselves in this way, whereas ‘Indigenous People’ is the generic term used by a variety of peoples to refer to themselves. Thus the document preferred by the users ‘Statement of Rights of Indigenous Peoples’ may have been a better exemplar than Encarta.

Another observation that may be seen in both sets of experimental data is that both MetaCrawler and Swish rate documents increasingly higher as Hesperus ratings tend to zero. These pages often contain no more than (say) ‘Indigenous Peoples Home Page, Click here if you’re using Netscape’. This would seem to be an artefact of the tf*idf algorithm, which may rate documents higher if term frequency is measured against document length. This is a heuristic assessment, *which* may indicate useful links to follow. Since Hesperus only analyzes actual content, it can not make similar judgements.

1. Conclusion and future work

We have implemented a WWW meta searching agent known as ‘Hesperus’ that clusters web pages based on their similarity to Exemplar texts. The agent identifies the lexical chains in a text using Roget’s thesaurus as a knowledge source. This is used to create an attribute value vector of thesaural categories that we have called the Generic Document Profile. Using this profile, we can compute similarity between a web page retrieved and an Exemplar.

Initial experiments using Hesperus have given mixed results. In the case of one query, Hesperus’ clustering was significantly correlated with that of human judges. However in the case of a second query, no correlation could be found with the judges, with MetaCrawler, or by locally re-indexing web pages retrieved using Swish. Perhaps, Encarta is not the best source of Exemplar texts.

Our current work is focused on improving the accuracy of the Generic Document Profile. This is subject to the word sense ambiguity problem and is being addressed by improving the disambiguation techniques used. Further user trials are also planned to understand the anomalous results from the second user query.

We are also considering the problem of unknown words. Our work has shown that 60-80% of chain links identified were due to term repetition. If a term not in thesaurus is repeated however, Hesperus can not now recognize that this is a distinctive aspect of the document. If these could be incorporated into the chaining approach, this would improve Hesperus’ general applicability.

2. References

- [1] Berners-Lee, T., Cailliau, R., Luotonen, A, Nielsen H.F, and Secret A, ‘The World Wide Web’, CACM Vol 37, 8 August 1994
- [2] Krovetz R and Croft W B, ‘Lexical Ambiguity and Information Retrieval’, ACM Transactions on Information Systems, Vol. 10(2), pp. 115-141, 1992.
- [3] Sanderson M. 1994 "Word Sense Disambiguation and Information Retrieval" in Proc. ACM SIGIR 1994 pp 142-151
- [4] Ellman J. and Tait J. “*INTERNET Challenges for Information Retrieval*”, proc BCS IRSG Conference March 1996
- [5] Furnas, G.W., Landauer, T.K., Gomez, L.M., Dumais, S. T., ‘The vocabulary problem in human-system communication.’ Communications of the Association for Computing Machinery, 30 (11), Nov 1987, pp. 964-971.
- [6] Resnick P & Varian H. “Recommender Systems” Communications of the Association for Computing Machinery, 40 (3), Mar 1997, pp. 56-58
- [7] Davies J., Week R., Revett M., & McGrath A “Using Clustering in a WWW Information Agent” proc BCS IRSG Conference March 1996
- [8] Salton and Buckley ‘Improving Retrieval Performance by Relevance Feedback’ JASIS 41(4) pp288-297 1990
- [9] Qiu Y. Frei HP “Concept Based Query Expansion” Proc ACM SIGIR 1993 pp160-169

- [10] Magennis M "Expert rule based query expansion" Proc BCS IRSG 1995
- [11] Rada R. & Bicknell E. "Ranking Documents with a Thesaurus" JASIS 40(5) pp304-310 1989
- [12] Selberg Erik and Etzioni Oren 1995 "Multi-Service Search and Comparison Using the MetaCrawler" Proc WWW4
- [13] Aamodt A and Plaza E 1994 "Case Based Reasoning: Foundational Issues, Methodological Variations and System Approaches" AI Communication Vol 7, 1 March 1994
- [14] Morris J. and Hirst G. 1991 "Lexical Cohesion computed by thesaural relations as an indicator of the structure of text" Computational Linguistics Vol 17 (1) pp 21-48
- [15] Okumura M. and Honda T "Word Sense Disambiguation and text segmentation based on lexical cohesion" Proc COLING 1994 vol 2 pp 755-761
- [16] Hirst and St Onge 1995 "Lexical Chains as representations of context for the detection and correction of malapropisms" to appear in [21]
- [17] St-Onge David 1995 "Detecting and Correcting Malapropisms with Lexical Chains" MSc Thesis University of Toronto. (available via WWW)
- [18] Green S 1996 "Using Lexical Chains to build Hypertext links in Newspaper Articles" in proc AAAI Symposium
- [19] Stairmand M and Black W J 1996 "Conceptual and Contextual Indexing using WordNet-derived Lexical Chains" in proc BCS IRSG
- [20] Stairmand M 1996b "A Computational Analysis of Lexical Cohesion with Applications in Information Retrieval" Unpublished PhD Thesis. UMIST Computational Linguistics Laboratory
- [21] Felbaum 1998 "Wordnet" The MIT Press Cambridge Mass
- [22] Croft B 1996 D-LIB Magazine January 1996.