PROCEEDINGS OF THE 8th ANNUAL CONFERENCE ON WORLD WIDE WEB APPLICATIONS

6-8 September 2006
Bloemfontein
South Africa

Editor:
P.A. van Brakel

Publisher:
Cape Peninsula University of Technology
PO Box 652
Cape Town
8000

Proceedings published at
http://www.zaw3.co.za

ISBN-10: 0-620-37309-1

ISBN-13: 978-0-620-37309-8

TO WHOM IT MAY CONCERN

Herewith to testify that each paper accepted for the 8th Annual Conference on World Wide Web Applications, 6-8 September 2007, Bloemfontein have been peer-reviewed by two independent peer-reviewers.

Prof P.A. van Brakel
Conference Chair
8th Annual Conference on WWW Applications
c/o e-Innovation Academy
Cape Peninsula University of Technology
Cape Town
+27 21 469-1015 (landline)
+27 82 966-0789 (mobile)

Working paper: Using LSA matrix comparison to improve the relevancy of search engine answers

S. Nkukwana
e-Innovation Academy
Cape Peninsula University of Technology
Cape Town
South Africa
nkukwanas@cput.ac.za

M. Weideman
e-Innovation Academy
Cape Peninsula University of Technology
Cape Town
South Africa
meliusw@yahoo.com

Abstract

The principal objective of this research project is to analyse and apply the use of Latent Semantic Analysis (LSA) as a support mechanism for Internet searching. The research aim is to improve the standard of search engine results where accommodation in South Africa is the search key, using the Ananzi search engine. This paper contains a detailed literature survey and a proposed methodology to achieve this aim. LSA is a theory and a method for extracting and representing the contextual meaning of words by statistical computations applied to a large text section. It analyses word-word, word-passage, and passage-passage relationships. This makes it feasible to compare words by paragraphs, paragraphs by paragraphs, and paragraphs by documents for the relevancy of data. Most of the existing search engines base their information retrieval purely on the keyword search mechanism. This implies that results are retrieved based on the matching of these keywords, ignoring the meaning and the sense they make towards documents to be retrieved. The strength of the proposed design is the ability to use the keyword technique, concentrate on the meaning of words and the sense they make in the webpage document. In order to test, analyse, and apply this technique and its ability, an implementation of a search tool based on LSA technology will be developed.

Keywords: LSA, latent semantic analysis, SMME's, search engines, accommodation, comparison, Ananzi.

1. Background to the research problem

The Ananzi website consists of two sections; the Search Engine and South African Directory parts. In order for a business to add its site into the Ananzi search engine, it has to first submit it to the SA Directory - from where it could be indexed should it meet the Ananzi acceptance criteria (Ananzi, 2005).

This research focuses on adding greater value to the Ananzi search engine where the user interest is searching for South African accommodation.

The Ananzi search engine home page contains banners and adverts which provides weblinks to registered webpages. Some of these weblinks could be indexed by Ananzi for search purposes. The impact of these banners may seem improper and inefficient specifically when user's interest is finding a specific business webpage, or a specific product to buy. By using the normal search box, results returned may still seem irrelevant and cumbersome. User's interest when searching for accommodation may include "bed and breakfast along the coast in Cape Town for two nights". This input contains a number of keywords that may link to various documents, which may not be necessary linked to "Cape Town" or "the coast", or specifically for "two nights". This therefore may produce irrelevant long listing of answers, which adds to the user's frustration.

In this research project, the meaning of input keywords and their relation to the retrieved page will be the determinant of the information retrieved. This implies that all documents containing text which seem to be related to "bread and breakfast in Cape Town" without having to occur or appear physically will be ranked.

In latent semantic analysis (LSA), the meaning of a word is represented as a vector in a high-dimensional semantic space. Different meanings of a word or different senses of a word are not distinguished. Instead, word senses are appropriately modified as the word is used in different contexts (Kintsch, 2001: 173-202).

2. Problem statement

Webpages seem to be poorly structured documents, and their logical inter-relationships are represented by hyperlinks. Due to the enormous size of the web, search engines play a more and more important role as a primary tool for locating and retrieving information. Most search engines compete against each other ito the number of indexed pages and their ranking, quality of returned pages, and response time.

Much research still has to be done on the way humans interact with websites, what their preferences are and how they perceive the use of webpage elements (colour, location, images, etc). Tamborello *et al.* (2005) have done research on how the visual salience of items interacts with "information scent". The effect of highlighted phrases and bold headlines on the user's perception of the usefulness of an information source, amongst others, was inspected.

Ananzi uses a spider algorithm (search engine indexing technology) which determines the website ranking based on the Meta-tags and (to a lesser degree) the body text on the page. Overall, sites using frames are also more difficult to index, meaning that its ranking might actually be lower than a site using tables (Cozahost, 2004). If a site uses frames, the chances are that a specific site may never be retrieved as part of the results even though it may be the number one preferred site.

Secondly, the Ananzi spider indexing algorithm indexes every page that may contain one or more of input keywords phrases, based on their number of occurrence on that specific page. Metatags are hidden code/text used by search engines for identifying a website's title and content when indexing. Thirdly, Ananzi's ranking algorithm may enforce most webpage owners attempting to register with Ananzi to include Metatags for search engine optimization. This implies that websites that do not use Metatags will not gain visibility to the search engines.

3. Literature survey

3.1 Search engine indexing

3.1.1 Web spiders

The search engine spider crawls websites in order to retrieve information. It is used by search engines to retrieve website information, and include it in its index. Plaza (2002) identifies the four component parts of a search engine as being; a spider, a parser or indexer, a query engine, and a web interface. The explanation of Thomas and Shearer (2000) on web robots states the web robots' task as being webpage visits. These authors state that these programs are called spiders, crawlers, or harvesters.

Search engines and other web services primarily rely on web spiders to collect large amount of data analysis. The design of a high performance web spider is a challenging task due to the large scale of the web. There are two important aspects in designing efficient web spiders, i.e. crawling strategy and crawling performance. Crawling strategy deals with the way hyperlinks are followed (eg depth-first or breadth-first), while crawling performance deals with the way spider performance is optimised. Characteristics of a web spider include scalability, robustness, flexibility and reconfiguration (Garodia, 2005).

The complexity of spider design signifies the degree of development of a search engine. This may imply the degree at which spiders “deep crawls” webpages, or the way that image mapping is performed, or the way metatags are used whilst testing metatag abuse. The inconsistency of result listings obtained when searching for identical keywords is determined by the choice of a spider used.

The following is a list of commonly used search engines:

- AltaVista,
- Excite (which also owns Web Crawler and Magellan),
- Google,
- Lycos,
- Inktomi (which influences the search results of HotBot, AOL Search, and MSN Search).

Search engines such as Excite, Inktomi and Lycos neither crawls nor indexes frame pages while Alta Vista, FAST and Northern Light do. This implies that if an image map is one of the main pages for links to other areas of a website, problems will arise regarding indexing pages with the spiders of Excite, FAST, Google, Inktomi and Lycos.

3.1.2 Indices

An index is the second search engine component, which contains a copy of website information harvested by web spiders. Plaza (2002) relates the concept of indexing as one of indexing books. This author explains its operation by the example of how books are normally indexed. Book index pages provide page references for a particular word, while the search engine’s index contains words along with references to the objects containing those words. When a user makes a query, this index is then used to find out

which pages contain the query terms or keywords entered by the user. The fetched object is analyzed for new links, and the new URLs found are then fed to the spider.

The artificial intelligence used by a search engine is a deciding factor of the objects which will best suit the user's query. LSA will be playing a greater role at this stage for filtering these indexed objects using singular value decomposition (SVD) so as to achieve best search answers.

3.1.3 The query engine

The query engine can boost pages with query terms occurring in the page title, or pages with many referring links. Usually users do not directly interact with the query engine, they make queries via a web interface. The web interface takes the query from the user and sends it to the query engine in a format understood by the engine. The query engine returns the results in a suitable format, for example, XML. The web interface then analyses the results and presents them to the user. To keep the index up to date, the spider must repeatedly check the objects for modifications.

3.1.4 Ranking of websites

According to Alimohammadi (2003), many schemes have been developed to organize digital information and compensate for search engine weaknesses. Creating metatags and using them as means of controlling the process of web indexing is one of those schemes.

Metatags are non-displaying or hidden HTML tags that may provide site owners and authors with a degree of control over how a web page is indexed (Henshaw and Valauskas, 2001: 86-101). Metatags were designed to give the webmaster the power to have the website ranked higher in certain circumstances. However, due to spammer abuse, these tags have lost their value to most search engine crawlers (Weideman & Strumpfer, 2004). Few of the major search engines spiders read and index metatags. Ananzi is one of the few search engines which allocates weight to the use of metatags.

Some web masters build their websites in a way keywords appearing on webpages are repeated excessively for search engines to be able to allocate a higher ranking. This does imply a greater disadvantage when using search engines which does not entirely base their indexing and searching on keywords. The navigation structure of a website also plays an important role in both the user experience and the crawler evaluation of a given webpage. Users often prefer graphical interfaces, using aids such as breadcrumbs and drop-down menus, above traditional text-based menus. Blacmon *et al.* (2005) executed research on methods to identify and repair webpage navigation problems.

3.1.5 Conclusion

Web spiders are the fundamental tool when information is to be retrieved from webpages, and indexed to databases. The performance of search engines is influence largely by the type of the spider software used. Competing search engines are also an influence when deciding on the implementation or improvement of the spider software. There are many ways in which these spiders can be improved to help achieve better performance. This research aims to implement LSA as an effective tool in bettering the manner in which indexing and information retrieval is performed.

3.2 LSA

Quesada *et al* (2001: 117-131) states that:

> "LSA is a machine-learning model that induces representations of meaning of words by analyzing the relation between words and passages in large bodies of representative text".

Ishwinder *et al* (2005) did a study towards the design of a predictive tool, which would simulate human visual search behaviour, in an attempt to assist interface designers. One of three semantic systems (PMI-IR) was chosen to best perform this function.

LSA is used in industry to develop technological applications, and the theory of knowledge representation is used to model well known experimental effects in text comprehension and priming. LSA allows one to define the meaning of words as a vector in a high-dimensional semantic space. A matrix is constructed whose columns are words and whose rows are documents (Kintsch, 2001: 173-202). Guandong *et al. (2005)* have researched a web usage mining method, which utilizes web user usage and page linkage information to capture user access patterns based on a Probabilistic Latent Semantic Analysis (PLSA) model. Results have indicated that this model can be used to generate user profiles, which could reflect common access interest.

According to Deerwester *et al* (1990: 391-407), the mathematical component used by LSA uses the singular value decomposition (SVD) technique to decompose the existing text-document matrix. This is done to three other matrices of a very special form, the resulting matrices containing "singular vectors" and "singular values". These special matrices show a breakdown of the original relationships into linearly independent components or factors (Deerwester *et al*,).

SVD involves a significant part of LSA processing. LSA applies the SVD tool to reduce the dimensions of a matrix. SVD compresses a large amount of co-occurrence information into a much smaller space. This compression step is somewhat similar to the common feature of neural network systems where a large number of inputs are connected to a fairly small number of hidden layer nodes. If there are too many nodes, a network will "memorize" the training set, miss the generalities in the data, and consequently perform poorly on a test set (Wiemer-Hastings, 1999: 932-937). The concept of dimensional reduction in relation to matrix factorisation is an efficient technique used by in various computational tasks (Osinski, 2004).

3.2.1 SVD with LSA

LSA starts with a very large text corpus (e.g., an encyclopedia, a series of textbooks in an area, or a large set of smaller documents, or a webpage). Next, all words that occur in the corpus are found. Those that occur with extremely high frequency (e.g., "the", "of", "a") are removed. For all the remaining words (potentially tens of thousands), the frequency with which they co-occur in a given context (usually defined as a paragraph) is then counted (Schunn, 1999). The matrix is normally produced from the occurrence of examined words, in a number of corpuses, with a size N by N. The size of this matrix can be enormous depending on the regularity at which these keywords occur in the given number of paragraphs in a document.

According to Maletic and Valluri (1999: 251-254), LSA relies on a SVD of a matrix M x N where M is the number of rows. N represents the number of columns derived from these corpuses of text which relates to the knowledge in the particular field of interest. The description by these authors of SVD states that:

> "… it is a form of factor analysis and acts as a method for reducing the dimensionality of a feature space without serious loss of specificity".

This illustrates that SVD reduces the number of dimensions without great loss of descriptiveness of words in a corpus matrix N x N. These two authors also indicate that the use of SVD is reflected in a number of applications, including; statistical principal component analysis, text retrieval, pattern recognition, dimensionality reduction and natural language understanding.

Schunn (1999) conceptually relates this reduction in size to factor analysis or multidimensional scaling. This author states:

> "… it produces a more compact representation of the important statistical regularities in the larger matrix. In the reduced matrix, each word can be thought of as a vector or point in an M-dimensional space. Since the reduced matrix is derived from frequency co occurrence information, it represents words that occur in similar contexts with similar representations".

By this, the process of using SVD for dimensional reduction forces synonyms to have very similar representations in the corpuses. Even though they rarely both occur in the same context, they co-occur with the same words.

3.2.2 Matrix composition

A large body of text is represented as an occurrence matrix (N × N) in which rows represent individual word types. Columns represent meaning carrying passages such as sentences or paragraphs. Each cell then contains the frequency with which a word occurs in a passage (Maletic and Valluri, 1999: 251-254).

One component matrix describes the original row entities as vectors of derived orthogonal factor values; another describes the original column entities in the same way. The third is a diagonal matrix containing scaling values such that when the three components are matrix multiplied, the original matrix is reconstructed.

3.2.3 An LSA example with SVD

Following is an example of LSA/SVD that illustrates the analysis and demonstrates some of what it accomplishes. This example uses as text passages the titles of nine technical memoranda, five about human computer interaction (HCI), and four about mathematical graph theory, topics that are conceptually rather disjoint. Thus the original matrix has nine columns, and 12 rows have been allocated to it, each corresponding to a content word used in at least two of the titles. The titles, with the extracted terms italicized, and the corresponding word-by-document matrix, is shown in Figure 1.

The linear decomposition is shown in Figure 2; except for rounding errors, its multiplication perfectly reconstructs the original as illustrated.

Lastly, a reconstruction based on just two dimensions that approximates the original matrix is shown in Figure 3. This uses vector elements only from the first two shaded columns of the three matrices shown in the previous figure (which is equivalent to setting all but the highest two values in S to zero).
Each value in this new representation has been computed as a linear combination of values on the two retained dimensions, which in turn were computed as linear combinations of the original cell values. Note, therefore, that if we were to change the entry in any one cell of the original, the values in the reconstruction will have reduced dimensions (Landauer et al, 1998).

The matrix X formed to represent this text is shown in Figure 1. It has nine columns, one for each title. Twelve rows were allocated, each corresponding to a content word that occurs in at least two contexts. Cell entries are the number of times that a word (rows) appeared in a title (columns) for words that appeared in at least two titles.

Figure 1: A word by context matrix, X, formed from the titles of five articles about human-computer interaction and four about graph theory

{X} =	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

c1: Human machine interface for ABC computer applications
c2: A survey of user opinion of computer system response time
c3: The EPS user interface management system
c4: System and human system engineering testing of EPS
c5: Relation of user perceived response time to error measurement
m1: The generation of random, binary, ordered trees
m2: The intersection graph of paths in trees
m3: Graph minors IV: Widths of trees and well-quasi-ordering
m4: Graph minors: A survey

Figure 2 shows the dimension reduction step which has collapsed the component matrices. This was done in such a way that words that occurred in some contexts now appear with greater or lesser estimated frequency, and some that did not appear originally now do appear, at least fractionally.

Figure 2: Complete SVD of matrix in Figure 1

{X} = {T} {S} {D}

{T} =

0.22	-0.11	0.29	-0.41	-0.11	-0.34	0.52	-0.06	-0.41
0.20	-0.07	0.14	-0.55	0.28	0.50	-0.07	-0.01	-0.11
0.24	0.04	-0.16	-0.59	-0.11	-0.25	-0.30	0.06	0.49
0.40	0.06	-0.34	0.10	0.33	0.38	0.00	0.00	0.01
0.64	-0.17	0.36	0.33	-0.16	-0.21	-0.17	0.03	0.27
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.30	-0.14	0.33	0.19	0.11	0.27	0.03	-0.02	-0.17
0.21	0.27	-0.18	-0.03	-0.54	0.08	-0.47	-0.04	-0.58
0.01	0.49	0.23	0.03	0.59	-0.39	-0.29	0.25	-0.23
0.04	0.62	0.22	0.00	-0.07	0.11	0.16	-0.68	0.23
0.03	0.45	0.14	-0.01	-0.30	0.28	0.34	0.68	0.18

{S} =

3.34								
	2.54							
		2.35						
			1.64					
				1.50				
					1.31			
						0.85		
							0.56	
								0.36

{D} =

0.20	0.61	0.46	0.54	0.28	0.00	0.01	0.02	0.08
-0.06	0.17	-0.13	-0.23	0.11	0.19	0.44	0.62	0.53
0.11	-0.50	0.21	0.57	-0.51	0.10	0.19	0.25	0.08
-0.95	-0.03	0.04	0.27	0.15	0.02	0.02	0.01	-0.03
0.05	-0.21	0.38	-0.21	0.33	0.39	0.35	0.15	-0.60
-0.08	-0.26	0.72	-0.37	0.03	-0.30	-0.21	0.00	0.36
0.18	-0.43	-0.24	0.26	0.67	-0.34	-0.15	0.25	0.04
-0.01	0.05	0.01	-0.02	-0.06	0.45	-0.76	0.45	-0.07
-0.06	0.24	0.02	-0.08	-0.26	-0.62	0.02	0.52	-0.45

Figure 3: Two dimensional reconstruction of original matrix of Figure 1

{X} = (Final)

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04

computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

This reconstruction is based on shaded columns and rows from SVD as shown in Figure 2. Comparing shaded and boxed rows and cells of Figures 1 and 3 illustrates how LSA induces similarity relations by changing estimated entries up or down to accommodate mutual constraints in the data.

r (human.user) = .94
r (human.minors) = -.83

Considering the two shaded cells for *survey* and *trees* in column m4, the word *tree* did not appear in this graph theory title. But because m4 did contain *graph* and *minors*, the zero entry for *tree* has been replaced with 0.66, which can be viewed as an estimate of how many times it would occur in each of an infinite sample of titles containing *graph* and *minors*. By contrast, the value 1.00 for *survey*, which appeared once in m4, has been replaced by 0.42, reflecting the fact that it is unexpected in this context and should be counted as unimportant in characterizing the passage. In constructing the reduced dimensional representation, SVD, with only values along two orthogonal dimensions to go on, has to estimate what words actually appear in each context by using only the information it has extracted. This text segment is best described as having so much of abstract concept one and so much of abstract concept two, and this word has so much of concept one and so much of concept two, and combining those two pieces of information (by vector arithmetic), a guess is that word X actually appeared 0.6 times in context Y.

A comparison is now done of the shaded rows for the words human and user in the original and in the two-dimensionally reconstructed matrices (Figures 1 and 3). It shows that while they were totally uncorrelated in the original - the two words never appeared in the same context - they are quite strongly correlated (r = .9) in the reconstructed approximation. Thus, SVD has done just what was required. When the contexts contain appropriate "concepts", SVD has filled them in with partial values for words that might well have been used but were not.

The italicized single-cell entries in the two figures show this phenomenon in a slightly different way. The word "tree" did not appear in graph theory title m4. But because m4 did contain graph and minor the zero entry for tree has been replaced with 0.66, which can be

viewed as an estimate of the proportion of times it would occur in each of an infinite sample of contexts containing graph and minor. By contrast, the value 1.00 for survey, which appeared once in m4, has been replaced by 0.42. This reflects the fact that it is unexpected in this context and should be counted as unimportant in characterizing the context itself. Notice that if we were to change the entry in any one cell of the original, the values in the reconstruction with reduced dimensions might be changed everywhere (Landauer *et al*, 1998).

4. Proposed methodology

4.1 Create a webpage database

- A number of existing South African accommodation webpages will be downloaded.

- A database will be created to store the collection of downloaded webpages. This database will be a folder on a disk containing URLs of all downloaded accommodation webpages in the Western Cape.

4.2 Create HTML webpage

- A JavaScript webpage with the following capabilities will be created:

Text input box with two pushbuttons:

- LSA
- Indexing

The initial phase of this process is to compare the difference between indexing and the use of LSA and produce a graph.

4.3 LSA matrix construction

- Perform LSA on each webpage and store matrices on database
- Write a word tokenizing tool
  - remove unwanted words
- Perform LSA on input phrase and store matrix to database.

4.4 Coloration between Input and webpages

- Perform statistical coloration between input matrix and webpage matrices
- Get coloration index

5. Expected Outcomes

The outcome of this research could prove the advantage in the use of LSA for information retrieval. The possible design outcomes may be as follows:

- an integrated Ananzi search tool used for accommodation search in South Africa,

- an independent search engine website designated for accommodation search in South Africa which will be invoked by Ananzi when accommodation is searched for.

Both these options could prove the efficiency of LSA when used as a search tool. Time constrains may make it only possible to achieve the first option.

6. References

Alimohammadi, D. 2003. Meta-tag: a means to control the process of Web indexing. *Online Information Review*, 27(4):238-242.

Ananzi, 2005. How does Ananzi work? [Online]. Available WWW: http://search1.ananzi.co.za/faq/works.html (Accessed 27 April 2006).

Blackmon, M.B., Kitajima, M, and Polson, P.G. 2005. Tool for accurately predicting website navigation problems, non-problems, problem severity, and effectiveness of repairs. In: *Proceedings of the SIGCHI conference on Human factors in computing systems,* Portland, Oregon, USA, 31-40.

Cozahost, 2004. How search engines and their spiders work. [Online]. Available WWW: http://www.cozahost.com/info/sespiders.asp (Accessed 27 April 2006).

Deerwester, S., Dumais, T., Landauer, K., Furnas, W. and Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the Society for Information Science,* 41(6):391-407

Garodia , R. 2005. Web Spiders. [Online]. Available WWW: http://www.allconferences.com/conferences/20050423181743/ (Accessed on 20 May 2006).

Guandong, X., Yanchun, Z., Jiangang, M. and Xiaofang, Z. 2005. Discovering user access pattern based on probabilistic latent factor model. In: *Proceedings of the sixteenth Australasian conference on Database technologies - Volume 39,* Newcastle, Australia, 27 - 35.

Henshaw, R., and Valauskas, E.J. 2001. Metadata as a catalyst: experiments with metadata and search engines. *Libri*, 51(2):86-101.

Ishwinder, K, and Hornof, A.J. 2005. A comparison of LSA, wordNet and PMI-IR for predicting user click behaviour. In: *Proceedings of the SIGCHI conference on Human factors in computing systems,* Portland, Oregon, USA, 51-60.

Kintsch, W. 2001. Predication. *Cognitive Science,* 25:173-202.

Landauer, T. K., Foltz, P. W., and Laham, D. 1998. Introduction to Latent Semantic Analysis. *Discourse Processes*, 25:259-284.

Maletic, J., and Valluri, N. 1999. Automatic Software Clustering via Latent Semantic Analysis. In: *Proceedings of the 14th IEEE International Conference on Automated Software Engineering (ASE'99)*, October 12-15, 251-254.

Osinski, S. 2004. Dimensionality Reduction Techniques for Search Results Clustering. [Online]. Available WWW: http://www.cs.put.poznan.pl/dweiss/carrot-bin/osinski04-dimensionality.pdf (Accessed 10 October 2005).

Plaza, S. 2002. Multimedia Search Engines White Paper. [Online]. Available WWW: http://www.medialab.sonera.fi/workspace/MultimediaSearchEngines.pdf (Accessed 10 May 2006).

Quesada, J.F, Kintsch, W. and Gomez, E. 2001. A computational theory of complex problem solving using the vector space model (part I): Latent Semantic Analysis, through the path of thousands of ants. In*: Proceedings of the 2001 Cognitive research with Microworlds meeting*, 117-131. J.J. Cañas (Ed.)

Schunn, C. D. 1999. The presence and absence of category knowledge in LSA. In: *Proceedings of the 21st Annual Conference of the Cognitive Science Society*. Mahwah, NJ: Erlbaum.

Tamborello, F.P., Byrne, M.D. 2005. Information search: the intersection of visual and semantic space**.** In: *Proceedings of the SIGCHI conference on Human factors in computing systems,* Portland, Oregon, USA, 1821-1824.

Thomas, A., and Shearer, J. 2000. *Internet searching and indexing.* New York: The Harworth Information Press.

Weideman, M., and Strumpfer, C. 2004. An empirical evaluation of one of the relationships between the user, search engines, metadata and websites in three-letter .com websites. *Information Technology and Libraries,* 23(2):58-65.

Wiemer-Hastings, P. 1999. How latent is Latent Semantic Analysis? In: *Proceedings of the 16th International Joint Conference on Artificial Intelligence,* August 1999. San Francisco: Morgan Kaufmann: 932-937