van Leeuwen, J.P., Jessurun, A.J. and Jansen, G. 2005. "Searching with Experience - A Search Engine for Product Information that Learns from its Users." In B. Martens and A. Brown(eds.): Learning from the past - proceedings of CAAD Futures 2005. Vienna: Vienna University of Technology. 203-214.

Searching with Experience

A Search Engine for Product Information that Learns from its Users

VAN LEEUWEN Jos, JESSURUN Joran, JANSEN Glenco
*Department of Architecture, Building and Planning,
Eindhoven University of Technology*

Keywords: search engine, page ranking, product information, construction supply chain, machine learning

Abstract: This paper describes the motivation and development of a new algorithm for ranking web pages. This development aims to enable the implementation of a search engine that can provide highly personalised results to queries. It was initiated by a request from the Dutch CAD industry, but has generic potential for the development of web search engines. The paper describes the algorithm, its position in relation with existing algorithms, and its potential drawbacks and advantages.

1 INTRODUCTION AND MOTIVATION

Product information is one of the main resources in the design stage of every construction project. The capability of a designer to find and utilise product information in their design activities may well be the main factor of success for both the designer and the construction projects that eventually result from his design activities. A survey by Josephson and Hammarlund (1999) has shown that design mistakes are a major cause (15-30%) of defect costs during production of buildings. After construction, during maintenance, the effect of design mistakes on defect costs is even more dramatic, 40-55%. The same study shows that over 60% of the defect costs in construction that are caused by design mistakes can be traced to a lack of knowledge or information. Presumably, in many cases this refers to *internally produced* information that is project specific and generated by various design participants during the course of a design process. However, it is safe to assume that the availability of *external* sources of information to a design team is a very essential asset and one of the factors that allow designers to excel.

Experienced designers and a high-quality library of product documentation and construction methods are perhaps the most valuable resources of any architect's office. Yet, human knowledge is as volatile as life itself unless it is represented in some form of media. The medium we are increasingly using is the Internet. Therefore, finding and accessing relevant information on Internet will become more and more a critical success factor, also in the practice of architectural design. The value of the architect's office's documentation library mainly lies in the proven relevance of the documents, which has a strong relation with the personal experiences in the office. Also, the library forms a very focused collection and is, ideally, uncluttered with irrelevant data.

This paper describes the results of a short research project that was initiated from a request from industry. A Dutch software developer of CAD software for the construction industry, called De Twee Snoeken, provides its customers (a considerable segment of Dutch architectural practice) with a collection of product information from the Dutch supply chain. While this information is currently distributed at regular intervals using CD-Rom as a medium, the CAD vendor is aware of the limitations of this approach. They seek to improve this form of distributing product information, without loosing the opportunity to integrate the information with their CAD software. The integration of specific product information with the generic CAD functionality is in fact one of the core features of the CAD software package and regarded a valuable asset by its architect customers.

Distributing product information in this specific format on CD-ROMs clearly has advantages for the possibilities of integration, but also involves a number of important and rather trivial drawbacks. The most important disadvantages are that information on such fixed media is quickly outdated, generally static, and providing merely a snap-shot overview of the relevant suppliers. Today, utilising suppliers' information through Internet is a much more obvious approach. To retain the features of dedicated information that can be integrated with CAD functionality, Internet-based information sources must meet certain requirements, but this effort is compensated by the advantages of having online resources.

Assuming that web-based documents will indeed form the main information resources for the next generation CAD integrated product documentation tool, the vendor is now challenged with new requirements for this software, hence their request for the present research project. One of the main challenges is to develop an adequate method to support users in finding the desired information. Where the limited set of information on a CD-Rom can be accessed through fixed and well-composed indexes, the ever-expanding sources on Internet require a more sophisticated approach. The utilisation of algorithms that are used by search engines seems obvious. However, a survey we performed among users in many different functions in the construction industry has pointed out that they rely heavily on their personal experience when selecting product information for construction projects. Yet user-experience is something that search engines generally do not deal with.

One approach to improve the availability of information design and construction processes, is to provide information middlemen services dedicated to this end, such as proposed by (Finne 2003). Research has been carried out on understanding the search behaviour of professionals in the AEC industry (Shaaban et al. 2003), which has led to statistically clustered behaviour patterns. In the research project presented in this paper, we focus on the individual user's search interests.

2 A NEW SEARCH ALGORITHM?

The procedure of searching the web, as it is performed behind the scenes, can be subdivided into a number of processes:

1. Gathering pages from the web, to build up a searchable, indexed, and possibly categorised database;
2. Acquiring the search query from the client;
3. Interpreting the client's query and reforming it into a database query or set of database queries;
4. Adding a sort algorithm to the database query, to obtain the top number of pages that are most relevant within the context of the client's query;
5. Presenting the results of the query to the client.

The value of any web search engine can be determined by four main characteristics:

a. The quality of the database of web pages that the engine uses, in relation with the area of interest of the user. This may vary per user and per usage. Personalised approaches of search algorithms, such as those deployed by, e.g. Google Personalized, focus on the usage of categorisation of the web space and do not take detailed user data into account. An additional aspect to this is the refresh rate of the database; how often is the data updated, how soon will changes on the web be propagated into the search database?

b. The capabilities of the engine to enhance the search terms using, for example, linguistic or semantic rules. This capability can expand a search on, e.g., the term 'modelling' to include also the terms 'model' and 'models.' The quality of this enhancement, regarded in the context of the user's query, will determine how 'intelligent' the engine will be perceived.

c. The capability to sort search results in a way that puts pages that are most relevant to the user on top of the list. A commonly used algorithm for sorting search results is the PageRank algorithm, which is discussed below.

d. And of course, the speed of producing the search results.

The current project focuses on improving aspects a and c above, and has the intention to address aspect b as well in the near future. Its objectives are to use a domain specific database of resources on the web and to present users with search results that are sorted in a way that takes their personal preferences in account.

2.1 How does PageRank™ work?

An obvious way to value the 'importance' or 'relevance' of a page on the web is to count the number of hyperlinks (also called citations or backlinks) to that page found on other pages on the web. We can regard this approach as a ballot system that gives the authors of web pages the right to vote (Rogers 2002). Every hyperlink to a page that is found on the web is regarded as a vote in favour of that page. There is no such thing as a vote against a page.

The sort mechanism that is most popular in the development of search engines is called PageRank™ (Page et al. 1998). It was developed at Stanford University by Page and Brin who are the founders of Google (Brin and Page 1998). This algorithm uses the backlink count to rank pages, but takes into account the importance of the referring page. Also, the 'vote' of the referring page is normalised by the number of votes that are made from the referring page (the higher the number of hyperlinks on a referring page, the less important is its vote). A simplified view on the PageRank algorithm can be denoted as follows:

$$PR(p) = (1-d) + d \sum_{r \in L(p)} \frac{PR(r)}{C(r)} \tag{1}$$

Where *PR(p)* is the PageRank of page *p*; *r* is a page from the set of pages *L(p)* that contain a hyperlink to page *p*; *PR(r)* is the PageRank of that page *r*; *C(r)* is the number of hyperlinks on page *r*; and *d* is a dimming factor.

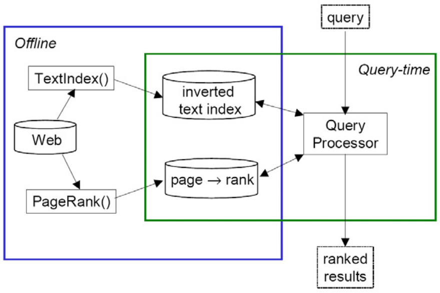


Figure 1 Search engine schema using PageRank (Haveliwala 2003)

Figure 1 shows a simplified schema of how a search engine using PageRank works. From this schema it becomes clear that the PageRank can be calculated prior to the time of querying the engine. It also shows that the PageRank is not related to the text index that is used for the query. In other words: the PageRank is not influenced by what the user is actually searching, it is entirely determined by the hyperlinks found on the web. The user's query is used to make a selection of web pages through the text index; these are then ordered according to their PageRank. Additional mechanisms, such as counting the number of times a word occurs on a page and giving increased importance to words that appear first in the query, are applied after the initial ranking.

2.2 Personalisation of PageRank

The PageRank algorithm as utilised by Google leads to a single ranking where all web pages are positioned relative to each other. *Personalisation* of the algorithm to rank query results has been further subject of research, also at Stanford University (Haveliwala et al. 2003). The approach discussed in (Haveliwala 2003) is also based on the PageRank algorithm, but focuses on calculation of multiple rankings for various subsets of the collection of pages on the web. This leads to the so-called topic-sensitive page ranking which is presumably used in the Personalized Web Search beta from Google that is based on the topics from the Open Directory Project; see http://labs.google.com/personalized and http://dmoz.org. Before the user can start this personalised web search, he is asked to create a profile, which consists of a selection from the topics in the ODP that the user is interested in (see Figure 2).

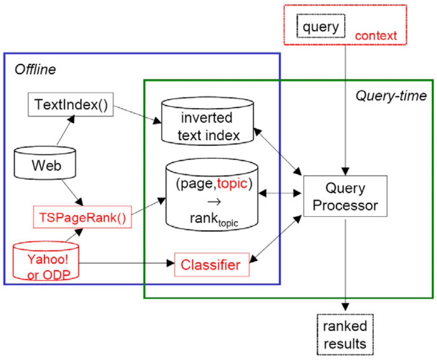


Figure 2 Search engine schema using Topic-Sensitive PageRank (Haveliwala 2003)

Although this approach leads to a more biased ranking of pages, we would not call it a personalised ranking for two reasons. Firstly, the categories in the ODP are not personal categories. Secondly, the ranking is still determined by the citations on the web. Three important issues, which have a strong relevance to the problem as identified at the beginning of this paper, remain unaddressed in this approach:

1. Personalised and dynamic recognition of the user's interests.
2. Relevancy of pages that is determined directly by the user's interest rather than by cross-citations on the web.
3. In subsets of the web, particularly subsets covering a commercial domain, pages are likely to represent competing companies that will not cross-link. Relying on citations to determine the importance of pages may produce non-representative results. Techniques to 'improve' a page's ranking are known and not rarely applied.

2.3 The HITS Algorithm

HITS, or Hyperlink-Induced Topic Search (Kleinberg 1998), is a ranking mechanism that is in a way similar to the PageRank approach. It is based on the distinction between pages with good coverage of the topic, called authorities, and directory-like pages with many hyperlinks to useful pages on the topic, called hubs. The goal of HITS is to identify good authorities and hubs for the topic of the user's query, which makes HITS a query-based algorithm. A good authority has many backlinks from good hubs and vice versa. This algorithm also determines the importance of a page on the basis of its backlinks and also requires iterative calculation.

2.4 Recommender Systems

The issue of recognising the user's interest is in a way addressed by the application of recommender systems (Sarwar et al. 2000, Amento et al. 2003). These systems provide recommendations of items (e.g. books or web pages) based on collective interests of dynamically constructed groups of users. In these systems a list of item recommendations is presented to the user. This list is composed by comparing the history of the current user's interest in items with the history of the interests other users. It is assumed that when a particular history is close to the current user's history that other items in that history are also interesting to the user. Research on recommender systems focuses on algorithms that compare the history of item interests of users and on algorithms for creating a sorted list of recommendations from this. The main objective of these systems is to *enlarge* the scope of the user's interest. Contrastingly, the project described in this paper aims to *narrow* the search space for the user's queries.

3 A USER-BASED METHOD FOR PAGE RANKING

The prototype search engine that was developed in this project implements a ranking method for web pages that is an alternative to the PageRank algorithm. Where the PageRank algorithm is based on cross-citations *on the web*, the proposed algorithm is based on a preference profile that is continuously updated for each individual user. This preference profile is built up by acquiring citations from the user. However, the preference profile does not consist of the collected citations, but rather of the indexed contents from the cited pages. For each user, the engine's database contains a collection of terms (words) that were found on pages that the user has appreciated in past visits. Each term is valued by an integer score that distinguishes its relative relevance in the collection of terms.

As other search engines, our system consists of three main components:

A. A crawler component that builds textual indexes of web pages;

B. A component that deals with the interpretation of the user's query;

C. A component that sorts the search results for the particular user.

The key feature of the system is that it learns from the experiences of the user, as he or she indicates which search results appear to be useful. The system learns in two ways. Firstly, the system learns to recognise the contextual meaning of terms that the user prefers to search with. This way, the system is able to add context to otherwise ambiguous terminology entered by the user when searching. Secondly, the system builds up a memory of the context of preferred search results. It then uses this memory to sort the results by examining the context of a resulting page, and calculating a ranking of the pages from this particular context of the user's preference. In this paper, we will focus on the sorting algorithm, component C from the list above.

3.1 User-based Ranking algorithm

$$R(p,u) = \sum_{i \in S(p)} \frac{T(i,u)}{C(i)} \qquad (2)$$

R(p,u) Ranking of page *p* for user *u*

T(i,u) Score of term *i*, for user *u* (the interest profile)

C(i) Number of URL's that contain the term *i*

S(p) Set of all terms on page *p*

This algorithm calculates the ranking of a page by summing up the weighted scores of the terms (words) on the page, as continuously determined by the user. While browsing the web or browsing the results of a search query, the user is asked to indicate his preference for a page, using client-side functionality (e.g., an 'I-like-this' button in the browser). This preference is communicated to the server, which increases the user-scores for all the terms on that page. The score for each term, which is now page-independent, is used in future ranking of all pages that contain the term. To allow distinction between exceptional and very common terms, the score of each term is divided by the number of pages that contain the term. This will boost the user's appreciation for exceptional terms, such as those specific for the user's professional domain, and will reduce the relevance of very common terms, such as those found in common language.

3.2 Implementation Considerations

Implementation of the user-based ranking algorithm can follow two approaches: query-time calculation and off-line pre-calculation. Off-line pre-calculation implies that the page scores are stored for each user and are then readily available during query-time. This will result in fast queries, but requires extra storage space per user, the size of which depends on the broadness of the user's interest. When the user's interest profile changes, i.e. when the user marks a newly found page as interesting, the off-line user-based ranking needs to recalculated. In terms of scalability, this means that the time needed for changes in the user's interest profile to be reflected in new queries may become a concern for very large datasets. In practice, we should view this possible delay in relation to the time interval between subsequent queries by the same user, which can be assumed to be at least one or two minutes.

Query-time calculation of the user-based ranking of pages does not store the rankings but calculates them each time the user performs a query. This approach does not require extra storage space per user but will slow down the querying process for very large datasets. On the other hand, changes to the user's interest profile are immediately reflected in the ranking results.

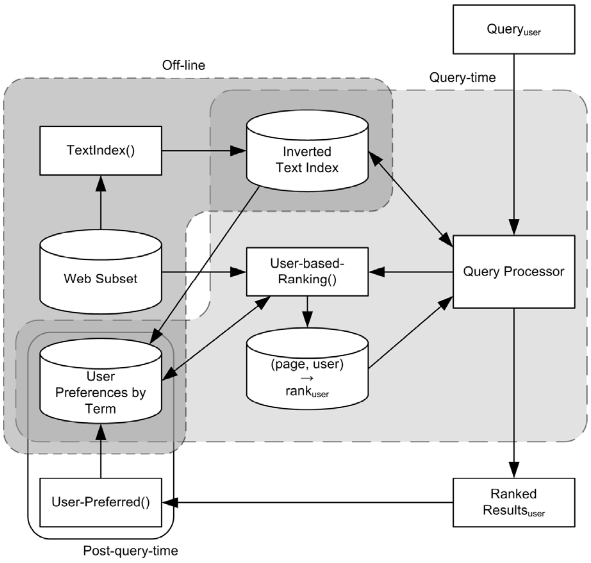


Figure 3 Search engine schema using user-based ranking of pages

Regarding the performance of the algorithm, it should be noted that the ranking per page has a linear relation with the number of terms on the page and with the number of pages in the dataset. There is no interdependency between the rankings of various pages; the algorithm is therefore not iterative, unlike the PageRank algorithm.

Figure 3 shows the schema of the prototype search engine that was implemented. This schema follows the query-time implementation of the algorithm. The prototype was used for a relatively small dataset that is domain specific and does not reflect the entire web. For this objective, the query-time computation of the user-based ranking appeared to be acceptable. Further development and experiments have to provide evidence of the true scalability of the two variant approaches mentioned.

3.3 Potential Advantages

Apart from having achieved the advantage that formed the objective of this project, i.e. user-based ranking, the algorithm has some potential advantages that now can be easily implemented. Using the database of preferred terms that is built up from the user's indicated appreciations, the ranked results from a query can be regarded as an implied list of favourites for the particular user on the topic of the query. Pages that have been visited before and were appreciated will have a high ranking. Compared to traditional bookmarks, this implied list of favourites has the advantage that it does not require any maintenance. The server will keep the URL's in its database up to date, which ensures that 'outdated bookmarks' are removed and replaced by new pages that automatically appear in the ranking. Also, it is not necessary for the user to organise his bookmarks, as a simple query with terms well-known to the user will quickly lead to the required results. No more aging bookmarks on a local disk; the user can access these implied favourites from anywhere on the web.

While the user can build up a personal profile of preferred terms, the ranking method we propose also allows for shared profiles. A mechanism that allows users to share their profile of term-scores, or a part thereof, would not be difficult to implement. With this approach it is possible to foresee a number of scenarios, including the forming of communities with similar interests and the exchange of search profiles that are known to lead to search results that excel in a particular domain (e.g., one could buy or hire the profile of a professional to be able to find high-quality web pages in the area of that professional's expertise).

3.4 Enhancements

Sorting the search results by using a ranking method is only one way of achieving a high quality search engine. Counting the number of times a term appears on a page is another commonly applied method to find relevant pages. Interpretation of the user's query is an important way to further enhance the search. Google applies text-matching techniques to add words to the query that are linguistically related to the words the user entered. In our project we intent to enhance the user's query by:

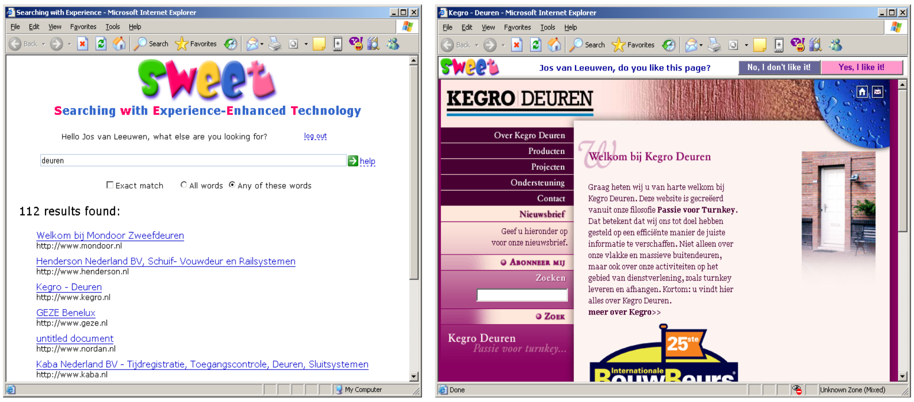
- Semantic matching using a domain specific thesaurus such as the LexiCon (Woestenenk 2000). This enables us to do cross-lingual searches.
- Semantic matching using unit transformations (e.g., converting metres to millimetres, etc.).
- Including relationships between terms in the user preferences. This is an important enhancement as it is expected to improve the user's preference profile considerably. For example, if the user searches for 'aluminium door,' the combination of these two terms is of special interest to the user, much more than the two words separately. Our system will record a memory of the user's interest for aluminium doors by adding the combination of the terms in the user's profile. This allows the system to

> have a bias for pages describing aluminium doors the next time the user searches for the term ‘door.’

Although the algorithm we propose in this paper focuses entirely on the ranking of web pages on the basis of user preferences, we do not deny the importance of using citations on the web to determine relevance. Another development that would benefit the quality of search results involves a combination of user-based ranking with citation-based ranking. This can be done in a sequential manner, using the user-based ranking to pre-select or post-sort the results from the PageRank algorithm.

4 PROTOTYPE SYSTEM

A fully functional prototype system has been developed to evaluate and experiment with the user-based ranking algorithm. The prototype, given the working title of SwEET (Searching with Experience-Enhanced Technology), works with a database of URL’s that we received from the CAD vendor that approached us with the search problem. The crawler component of the system creates a text index of the pages and is implemented in Java. The query component is implemented in ASP.NET. The first experiments have demonstrated that the implementation is feasible and that the query-time is acceptable for the given set of pages in the particular domain of the Dutch construction supply chain. As mentioned in section 3.2, the user-based ranking algorithm is implemented in this prototype following the query-time approach. This is done partly in a collection of SQL procedures stored at the DBMS, partly in SQL procedures that are dynamically prepared in ASP.NET code. This allows for optimisation on the side of the SQL stored procedures and flexibility in the code behind the engine’s web pages.



Searching with Experience - Microsoft Internet Explorer

Searching with Experience-Enhanced Technology

Hello Jos van Leeuwen, what else are you looking for? log out

deuren

□ Exact match ○ All words ● Any of these words

112 results found:

Welkom bij Mondoor Zweefdeuren
http://www.mondoor.nl

Henderson Nederland BV, Schuif- Vouwdeur en Railsystemen
http://www.henderson.nl

Kegro - Deuren
http://www.kegro.nl

GEZE Benelux
http://www.geze.nl

untitled document
http://www.nordan.nl

Kaba Nederland BV - Tijdregistratie, Toegangscontrole, Deuren, Sluitsystemen
http://www.kaba.nl

Kegro - Deuren - Microsoft Internet Explorer

Jos van Leeuwen, do you like this page?

Geef u hieronder op voor onze nieuwsbrief.

Welkom bij Kegro Deuren

Graag heten wij u van harte welkom bij Kegro Deuren. Deze website is gecreëerd vanuit onze filosofie **Passie voor Turnkey.** Dat betekent dat wij ons tot doel hebben gesteld op een efficiënte manier de juiste informatie te verschaffen. Niet alleen over onze vlakke en massieve buitendeuren, maar ook over onze activiteiten op het gebied van dienstverlening, zoals turnkey leveren en afhangen. Kortom: u vindt hier alles over Kegro Deuren.
meer over Kegro>>

Figure 4 SwEET prototype search engine with user-based ranking of pages

5 CONCLUSIONS AND FUTURE WORK

PageRank is used to calculate a single ranking of all pages on the web. Topic-Sensitive PageRank is used to calculate multiple rankings for subsets of the web using the categorised topics in the Open Directory Project at http://dmoz.org. Both algorithms are based on determination of a page's importance by the citations of the page found on the web.

Our user-based ranking algorithm calculates a ranking of web pages based on user preferences, by taking into account a detailed user profile that represents the user's appreciation of terms found on web pages. We have built a prototype search engine that implements the user-based ranking algorithm. This prototype implementation has successfully demonstrated the technical feasibility of implementing the algorithm and the viability of a functional search engine. Initial tests with the prototype have shown positive results, yet extensive testing and up-scaling are necessary and planned. Regarding the original request from the Dutch CAD vendor, we can conclude that it is technically possible to achieve the objectives of building search tools for domain specific online information sources, tools that take advantage of both personal and corporate experience in architectural design offices.

Future work is planned into several directions. The first is improvement of the interpretation of the user's query, utilising the LexiCon, semantic conversions, and linguistic resources. Second, we aim to include relationships between terms in the ranking algorithm to further bias the ranking for the user's particular interests. A third direction for future work is to search collaboration with existing search engines in order to make use of a larger dataset of web pages.

A parallel development will regard the usage of the system in communities, such as research communities with a particular set of interests. The common profile of such a community can form the basis for knowledge sharing in an informal manner, comparable to such initiatives as http://www.stumbleupon.com. This application of the system in communities brings us back to the introduction of this paper, which identified the need for sharing people's online information resources in the context of an architect's office.

ACKNOWLEDGEMENT

The authors give thanks to De Twee Snoeken (http://www.tweesnoeken.nl/) who have provided the construction context for this project and have helped in the initial tests.

REFERENCES

Amento, B., Terveen, L., Hill, W., Hix, D., and Schulman, R. 2003. Experiments in Social Data Mining: The TopicShop System, *ACM Transactions on Computer-Human Interaction*, 10, 1 (March 2003) , 54-85.

Brin, S. and Page, L. 1998. *The Anatomy of a Large-Scale Hypertextual Web Search Engine*. Computer Science Department, Stanford University, Stanford, CA http://www-db.stanford.edu/~backrub/google.html

Finne, C. 2003. How the internet is changing the role of construction information middlemen, *ITcon* Vol. 8, Special Issue eWork and eBusiness, pg. 397-412. http://www.itcon.org/cgi-bin/papers/Show?2003_28

Haveliwala, T.H. 2003. Topic-Sensitive PageRank: A context-sensitive ranking algorithm for web search. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 4, July/August 2003.

Haveliwala, T., Kamvar, S., and Jeh, G. 2003 *An Analytical Comparison of Approaches to personalizing PageRank*. Technical Report, Stanford University. http://newdbpubs.stanford.edu/pub/2003-35/

Josephson, P.-E. and Hammarlund, Y. 1999. The causes and costs of defects in construction – a study of seven building projects. *Automation in Construction*, vol.8, pp.681-687.

Kleinberg, J. 1998. Authoritative Sources in a Hyperlinked Environment. *Proc. 9th Ann. ACM-SIAM Symp. Discrete Algorithms*, ACM Press, New York, 1998, pp.668-677.

Page, L., Brin, S., Motwani, R., and Winograd, T. 1998. *The PageRank citation ranking: Bringing order to the web*. Stanford Digital Libraries Working Paper. http://newdbpubs.stanford.edu/pub/1999-66/

Rogers, I. 2002. *The Google Pagerank Algorithm and How It Works*. IPR Computing Ltd. http://www.iprcom.com/papers/pagerank/

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. 2000. Analysis of Recommendation Algorithms for E-Commerce. *EC'00*, October 17-20, 2000, Minneapolis, Minnesota, p. 158-167.

Shaaban, S., McKechnie, J., and Lockley, S. 2003. Modelling information seeking behaviour of AEC professionals on online technical information re-sources, *ITcon* Vol. 8, Special Issue eWork and eBusiness, pg. 265-281. http://www.itcon.org/cgi-bin/papers/Show?2003_20

Woestenenk K. 2000. Implementing the LexiCon for Practical Use. In: Gudnason G, editor. *Construction Information Technology*, Reykjavik, Iceland.

All URL's in the list of references and in the paper were valid at the moment of submission of this paper (2 February 2005).