A Question Answering service for information retrieval in Cooper

Bas Giesbers[1], Antonio Taddeo[2], Wim van der Vegt[1], Jan van Bruggen[1], Rob Koper[1],

[1]Open University of the Netherlands
{Bas.Giesbers, Jan.vanBruggen, Wim.vanderVegt, Rob.Koper}@ou.nl

[2]Advanced Learning and Research Institute, Italy
antonio.taddeo@alari.ch

Abstract. In Cooper, part of the student support will be provided by a Question Answering application in the form of a webservice. Question Answering allows a user to use the content of project document as input to find related documents as well as related experts. Latent Semantic Analysis as an underlying technique is briefly discussed followed by a description of our Latent Semantic Analysis engine and the software architecture that was developed. Issues for further development are also mentioned. The final section contains a specific case study of an environment in which an implementation is planned.

Keywords: Latent Semantic Analysis; Question Answering; Information Retrieval; Singular Value Decomposition

1 Introduction

The Cooper platform aims to support teams of students who work on complex problems in a Virtual Company educational scenario [1, 2] and will consist of several elements that support students' performance including tools for information retrieval. Question Answering (QA) is one of these elements. This paper will (1) discuss what we mean by Question Answering and why we want to use it; (2) show how Latent Semantic Analysis (LSA) is used as a basis for QA and present a modular built engine for complete LSA analysis; (3) demonstrate the preliminary version of the engine and (4) describe our plan for the near future to integrate the LSA engine at the Advanced Learning and Research Institute.

1.1 Why Question Answering?

In a Virtual Company educational scenario students run projects that consist of several phases [2]. Students will be active during the project start phase, the project work phase and the final phase in which results are delivered. Each phase includes several 'standard' activities that will be performed by all students in all projects. For example, during the project start phase students have to gather background information about related (previous) projects and the people who have been involved in those projects. This entails that students must have access to all documents that are related to previous projects. Next, students must also be able to search these documents and retrieve those most relevant to their current project.

Besides the support of finding related projects, we also want to support students in finding the right people to provide information about related projects as well as field experts.

Following [3], the information need of our users can be individual, may change through time and is context-dependent. Search results of the typical search engine using lexical methods provide query results independent of user and context and are incomplete and imprecise [4]. Instead, we want students to find matches based on semantic similarity. Also, the methods behind typical search engines are aimed at vast amounts of material with a very diverse nature. Our users' needs are dependent on a much smaller and more precise context such as the document collection maintained by Cooper partners like ALaRI, Open University and L3S. A useful method to use for our purpose is Latent Semantic Analysis (LSA) of which an extensive overview including applications is available [5, 6].

1.2 Latent Semantic Analysis

LSA is a technique with which documents can be compared to each other by representing them as text vectors. The basis of LSA is the representation of a corpus of documents into a term by document matrix, holding the frequency for every occurring term per document. This allows a document to be represented as a vector of term frequencies.

When analysing a corpus we turn the material into a 'bag of words' without syntactic information. However, we can then calculate similarities between the documents by calculating the distance or angle between the vectors. Furthermore, LSA takes this a step further in that it projects vectors in a multidimensional space that is abstracted from the original data. This can be explained by looking at each of the three steps that construct LSA.

During the first step, the original term x document matrix is used for singular value decomposition (SVD). This results in a diagonal matrix containing singular values and two orthogonal rotation matrices. The number of singular values > 0 are the number of dimensions in the data.

During the second step the material is reduced. This means that the smallest singular values and corresponding rows and columns in the rotation matrices are identified and excluded from further analysis.

During the third step the reduced material is used as a basis to reproduce the original material.

In other terms, LSA explains the content of a text as the weighted sum of underlying constructs. This makes LSA similar to more commonly used methods as Principal Component analysis and Factor Analysis.

When doing a query (comparing a document to the vector model created earlier), before it can lead to any meaningful results from the corpus, the query document is subjected to the same process as described above. A schematic overview of the comparison of a query with a corpus is given in Figure 1.

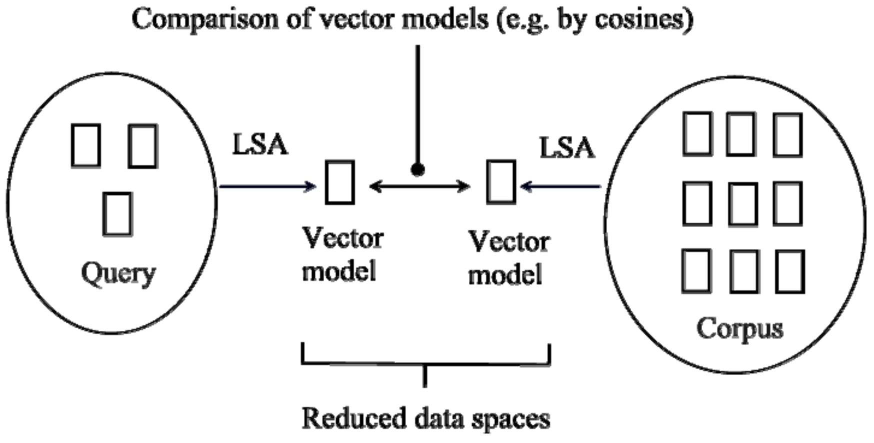


Fig. 1. Schematic overview of the LSA process, taken from [4] p. 7.

1.3 Assumptions and requirements

Because a semantic space is a mathematical representation of an amount of text, the corpus underlying it must be large enough so the machine can 'learn the language' in order to produce meaningful query results. Because the corpora an institution like ALaRI can offer is not big enough, it must be enlarged by using a general corpus that serves the process of 'learning the language'. Documents contained in this corpus will of course not be presented to the user as query results. Several large corpora that can serve this purpose are available, for examples see [14]. A lot of these were constructed by laborious work and are therefore only available freely for research purposes. Of course, the language of the corpus must be the same as the language of the query; otherwise results will have no meaning.

In Cooper, the basic idea is to use a document as a query with which related people, projects and documents can be found. This means that a connection to the underlying Cooper database containing that information is required. It is our intention to provide an LSA engine as a webservice that can be 'plugged' into the platform. The query process is designed as follows:

First, a query result will lead to a number of semantically related documents. They can be listed depending on the strength of their relationship with the query document. Depending on the amount of returned results it is useful to list only the documents with a relation above a certain threshold, for example 0.6. However, sometimes the highest correlation may only be 0.5 or lower. Depending on experiments it should be decided where a meaningful cut-off point should be set.

Second, the authors of each document are found in the database underlying the Cooper platform. These authors are listed as experts on a certain area.

Third, other data such as availability of a person can be checked and reported along with the name. The user' personal profile of the Cooper platform provides several entries for people to add their location, availability and so forth.

Finally, the student who performed the query now knows what documents are related, from which projects they originated and who were involved in writing them and where and when they may be reached.

Depending on the rules within the organization, several restrictions can be added to the system. These must be visual for the student. For example, external parties may not be contacted after a project ends so in the query results they should be listed as such or not listed at all.

These assumptions and requirements have implications on a technical level. Most of these are known but not all have been evaluated in practice. The fourth paragraph introduces a first implementation at the Advanced Learning and Research Institute (ALaRI).

2 Toolbox

Most of the currently available tools for LSA, such as GTP [12] and R [13] are not equipped to perform the required steps for a complete latent semantic analysis (i.e. pre-processing of the documents, turning them into a Term x Document matrix and perform the SVD) in a web based environment. Because the Cooper platform is web based, we develop an LSA engine that is built in a modular fashion and is suitable for our web based experiments.

Available tools (especially GTP) are very sensitive to the format of the source documents. GTP works best with plain ASCII [12] and may not perform the SVD correctly (or may not perform it at all) if the source documents don't fulfil its requirements. During the development of our LSA engine we eliminated this problem by allowing it to directly access source formats like *.pdf, *.doc and *.ppt without initial conversion to plain ASCII. We intend to eventually support Unicode with our LSA engine. That would allow the processing of materials in different languages and character sets. Our LSA engine supports importing source documents from (hard) disk, file transfer protocol (ftp) and network news transfer protocol (nntp). Once the location of the source documents is known pre-processing can take place.

Pre-processing the source materials should result in a Term x Document matrix which can be use to perform the SVD analysis. However, this requires a fixed order for some steps in the pre-processing phase. Scripting the process allows for certain steps to be optional, thus offering the user a choice to incorporate these steps or not. Also, a series of analyses can be scripted.

To perform the SVD, an application like SVDLibC or GTP is used. The use of a single and separate application for complex calculations makes the LSA engine less sensitive to errors. Because of the modular setup other analysis tools like Support Vector Machines (SVM) or Semi-discrete Decomposition (SDD) are easily substituted in the process. The LSA engine saves the result of each step in the process.

This allows further processing of intermediate results like word counts for tag clouds. A uniform format is used to automatically transform the output of the SVD. This allows for further processing of the final results using tooling different from ours.

The LSA engine is usable in multiple scripting environments such as PHP, ASP or VB/Jscript. Further adaptation to the users needs in scripting environments are possible, which further increases flexibility. At the moment, a layer for several PHP versions is available. Other layers like ActiveX for ASP and VB/Jscript can be created when desired. Of course, it is also possible to build the engine into a 'classic' stand-alone application.

We want to further develop the LSA engine to support (1) the import of websites and sitemaps; (2) cutting documents; (3) Unicode; (4) scaling a query vector and (5) Support Vector Machines.

2.1 Architecture

Figure 2. shows a schematic overview of the architecture.

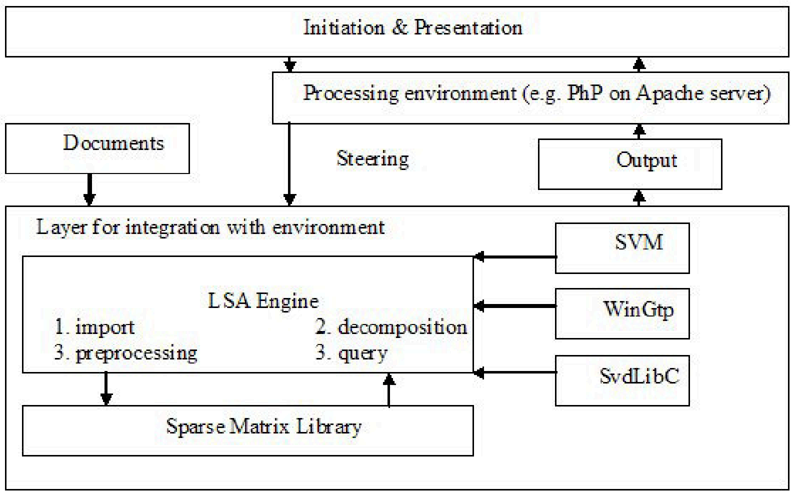


Fig. 2. Architecture of the LSA engine

At the bottom layer of our architecture there is a sparse matrix library which was designed separately. Here the mathematical processing is performed, while separate applications that take care of the SVD are called.

On top of this layer, the actual LSA engine is built that performs the following tasks:

- Pre-process source documents;
- Construct the Term x Document matrix;
- Call a separate SVD application;
- Execute a query.

The final layer around the LSA engine processes the call of the engine in an environment like PHP. This layer facilitates the use of the engine in interactive (web) applications.

3 Future assessment: implementation at ALaRi

In this section we present a specific case study: the implementation of LSA techniques at the Masters of Engineering in Embedded Systems Design at the Advanced Learning and Research Institute (ALaRI), University of Lugano, Switzerland [8]. ALaRI is a higher education institution and a school of excellence with geographically dispersed students and/or tutors. It offers an innovative teaching and research program for specialisation in embedded systems.

The case study offers a rather unique learning and collaboration scenario due to the roles played by industrial and academic partners of the AlaRI institute. In particular, the two sides contribute to the training of the master students in different ways, but both share the problem of integrating remote and face-to-face interaction with the students and with other stakeholders. A unique characteristic of ALaRI is that it represents a remote faculty with face-to-face learning in which teachers from different universities around the world come to ALaRI for a lecture period of one/two weeks. Moreover, during their Master program, students participate in multidisciplinary projects focusing on real and relevant problems which are usually identified by external institutions. The different actors revolving around a Master project also need to interact remotely to participate in the completion of the research work.

A first implementation of a remote collaboration platform for the ALaRI institute has been developed in [9]. However, an important requirement not addressed in [9] is to fully develop a Knowledge Base (KB) for the institute and its partners enabling the sharing of information among the different actors.

3.1 Question Answering and the ALaRI knowledge base

In this section we present the use of the Question Answering service in a first implementation of the ALaRI KB developed in the context of the Cooper project. Such a KB can integrate several services such as the Question Answering service for information retrieval. The web application was designed using a high-level modelling language called WebML [10] and then deployed using an automatic code generator, Webratio [11].

The KB is composed of a data model and a navigational model. The data model of the KB shown in Figure 3 is a simplified representation of the EntityRelationship diagram, in which only the main entities are drawn, with their minimum set of attributes. Besides the classical relationships between Author, Document and Personal Identity Profile (meaning the user), there are two specific relations: the Linked_by and the Ownership. The first takes into account the number of users which have "linked" a given document into their document folder. Such a relationship represents a sort of "ranking" of a given document. The higher the number of "links" to a document, the higher it will be valued. The Ownership relationship is added to store the information regarding the owner of a document who has some access rights on it (e.g. the owner can modify and/or delete his uploaded documents).

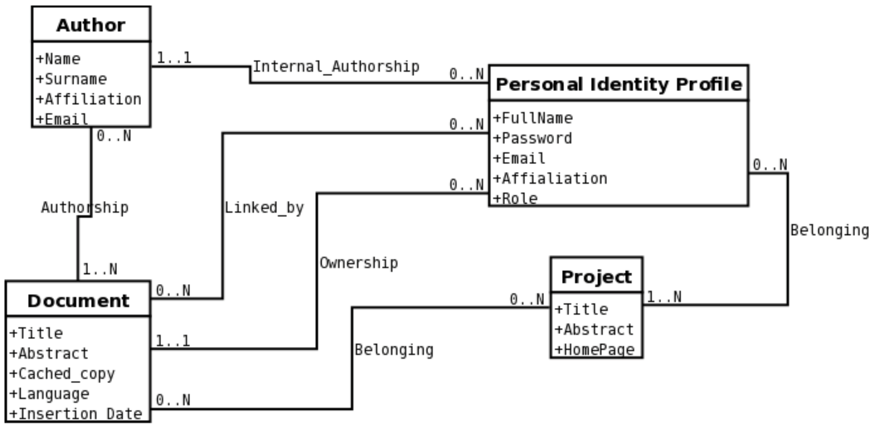


Fig. 3. Knowledge Base Data Model

Furthermore, a document may belong to a project. In that case a record is added to the relationship among Project and Document. The data model is then used for developing the navigational model. The navigational model describes the composition of site views, areas and pages with the links between them. Moreover, it also includes the page's contents and structure. For more details about the navigational model see [10] and [11].
The KB system has been developed according to the following requirements: The users allowed to access the KB are enabled to upload documents, search the repository, organize all interesting documents into private folders and build and share a project bibliography.

The KB provides a multidirectional navigation capability among authors, documents and folders. For instance, the user might start browsing the KB, find one interesting document, get details on it; see its owner, or its authors profiles; move on to the related projects; get reports about those projects, see which professor has supervised them, and so on. Details of a specific document can be used as a basis to decide if that document is worth while to be used in a LSA query. If so, all semantically related documents and additional detail information are retrieved by the Question Answering service.

Several examples of KB pages with their contents are shown below. Figure 4 shows a sample structure of a possible Document detail page. Notice that a specific unit (in the top left corner) is in charge to publish, at runtime (once the page will be requested), the recommended documents related to the selected one. An example of how the Document detail page structure is rendered in a web page is shown in Figure 5. The Document Detail web page includes basic document metadata (title, abstract, etc.) plus links to: (1) cached copy of the document; (2) authors, (3) related projects; (4) related virtual folders of the logged user and (5) the profile of the user who uploaded the document. Available actions (in the bottom area) are (1) adding document to own folders (including bibliography) and (2) link document to a project's inbox.

Finally, in the highlighted box are the recommended documents provided by the LSA engine, ordered by correlation value (highest on top). The Project's Inbox acts as a container for suggested documents, in which a user recommends documents manually. Any user that found an interesting document can suggest the given document by "sending it" to a Project Inbox for further evaluation.

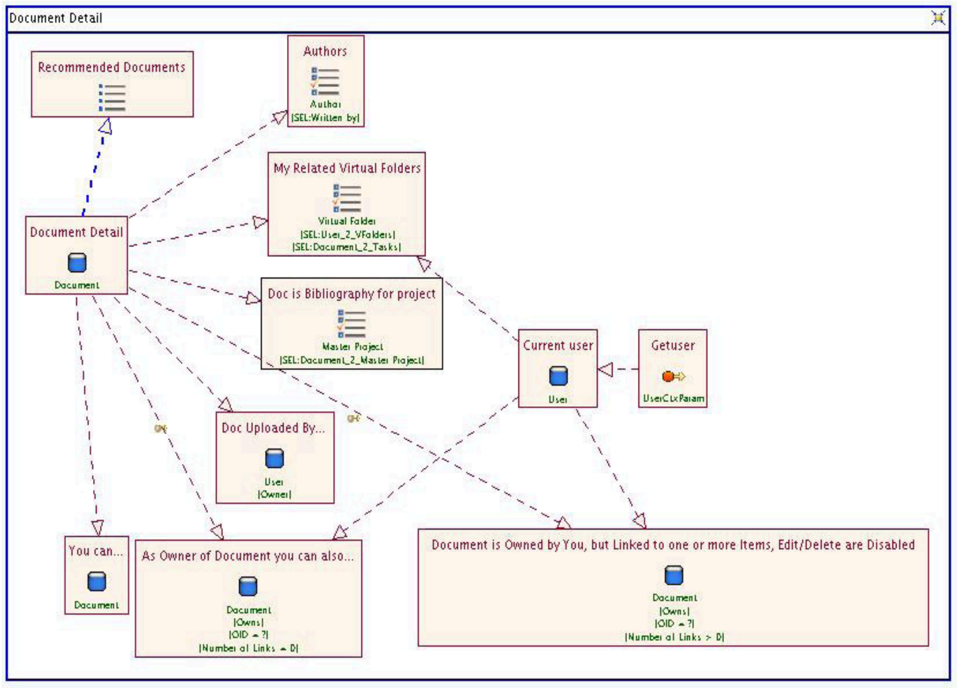
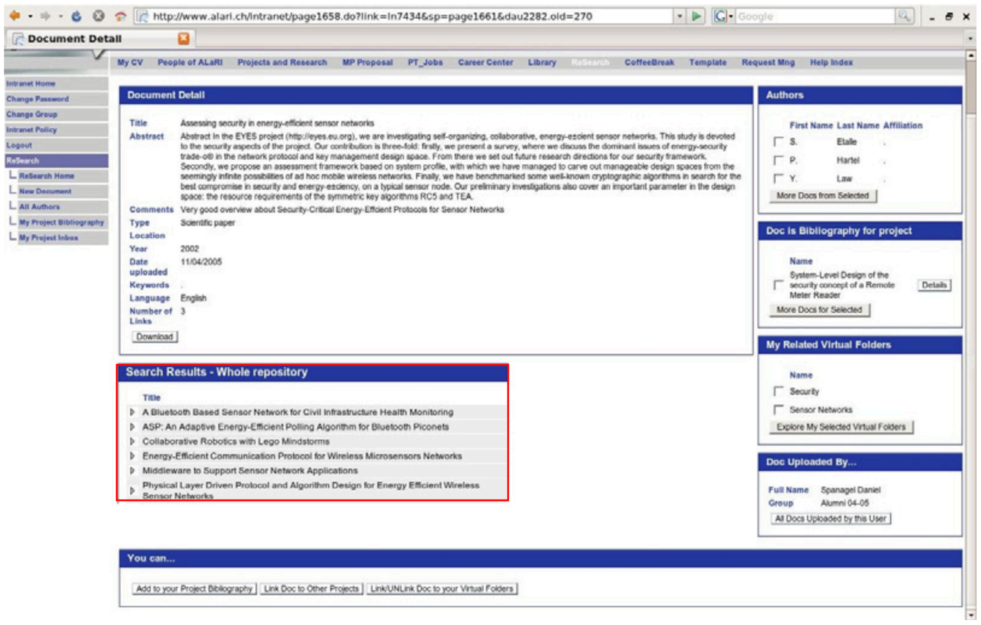


Fig. 4. Document Detail page Model



Document Detail

Title	Assessing security in energy-efficient sensor networks
Abstract	Abstract In the EYES project (http://eyes.eu.org), we are investigating self-organizing, collaborative, energy-e±cient sensor networks. This study is devoted to the security aspects of the project. Our contribution is three-fold: firstly, we present a survey, where we discuss the dominant issues of energy-security trade-o® in the network protocol and key management design space. From there we set out future research directions for our security framework. Secondly, we propose an assessment framework based on system profile, with which we have managed to carve out manageable design spaces from the seemingly infinite possibilities of ad hoc mobile wireless networks. Finally, we have benchmarked some well-known cryptographic algorithms in search for the best compromise in security and energy-e±ciency, on a typical sensor node. Our preliminary investigations also cover an important parameter in the design space: the resource requirements of the symmetric key algorithms RC5 and TEA.
Comments	Very good overview about Security-Critical Energy-Efficient Protocols for Sensor Networks
Type	Scientific paper
Location	
Year	2002
Date uploaded	11/04/2005
Keywords	.
Language	English
Number of Links	3

Download

Authors

First Name	Last Name	Affiliation
S.	Etalle	.
P.	Hartel	.
Y.	Law	.

More Docs from Selected

Doc is Bibliography for project

Name	
System-Level Design of the security concept of a Remote Meter Reader	Details

More Docs for Selected

My Related Virtual Folders

Name
Security
Sensor Networks

Explore My Selected Virtual Folders

Doc Uploaded By...

Full Name	Spanagel Daniel
Group	Alumni 04-05

All Docs Uploaded by this User

Search Results - Whole repository

Title
A Bluetooth Based Sensor Network for Civil Infrastructure Health Monitoring
ASP: An Adaptive Energy-Efficient Polling Algorithm for Bluetooth Piconets
Collaborative Robotics with Lego Mindstorms
Energy-Efficient Communication Protocol for Wireless Microsensors Networks
Middleware to Support Sensor Network Applications
Physical Layer Driven Protocol and Algorithm Design for Energy Efficient Wireless Sensor Networks

You can...

Add to your Project Bibliography | Link Doc to Other Projects | Link/UNLink Doc to your Virtual Folders

Fig. 5. Document Detail page in Knowledge Base system

4 Conclusion

This article first introduced Question Answering and its use by means of Latent Semantic Analysis in the Cooper platform. LSA was briefly explained as well as the development of a toolbox and the architecture behind its implementation in the platform. Then a more elaborate description was given of an environment in which the implementation of Question Answering techniques is planned.

Much work still needs to be done in order to develop a fully operational LSA webservice that can be plugged into the Cooper platform. This will include further development of the LSA engine and the webservice layer, on the refinement of pre-processing and the use of corpora and queries as well as work on the Cooper environment itself and the implementation of the webservice into that environment. The paths to be taken are clear and chances are good we can reach the aims we have set.

References

[1] Cooper is an EU funded collaborative research project from the sixth framework programme of the Information Society Technologies IST (contract No. FP6 IST - 027073) http://www.cooper-project.org

[2] Spoelstra, H., Matera, M., Rusman, E., van Bruggen, J., Koper, R. (2006). Bridging the gap between instructional design and double loop learning. Current Developments in Technology-Assisted Education. http://www.formatex.org/micte2006/pdf/1396-1400.pdf

[3] Almeida, R. B., & Almeida, V. A. F. (2004). A community-aware search engine. WWW2004, New York, May 17-22, 413-421. New York, USA: ACM.

[4] Macedo, A.A., Pimentel, M.G.C., Cammacho-Guerrero, J.A. (2002). An infrastructure for open latent semantic linking. In: Proc ACM HT'02, pp 107–116

[5] Iofciu, T., Zhou, X., Giesbers, B., Rusman, E., van Bruggen, J., Ceri, S. (2006). State of the Art Report in Knowledge Sharing, Recommendation and Latent Semantic Analysis.

[6] Landauer, T.K., McNamara, D.S., Dennis, S., Kintsch, W. (eds.): Handbook of Latent Semantic Analysis (2007). Mahwah, New Jersey: Lawrence Erlbaum Associates

[7] L3S Research Center, http://www.l3s.de/web/sv3a.do

[8] ALaRI, web site, http://www.alari.ch.

[9] L. Negri and U. Bondi, "The ALaRI Intranet: a Remote Collaboration Platform for a Worldwide Learning and Research Network", proc. EDMEDIA 04, Lugano, Switzerland, 2004(1), 50425047, AACE Press, 2004.

[10] Ceri, S., Fraternali, P. & Bongio, A. (2000). "Web Modeling Language (WebML): a Modeling Language for Designing Web Sites". WWW9 Conference, Amsterdam, Holland.

[11] Webratio, web site, http://www.webratio.com.

[12] Giles, J. T., Wo, L., & Berry, M. W. (2001). GTP (General Text Parser) Software for Text Mining. In Statistical Data Mining and Knowledge Discovery (chap. 27). CRC Press. Retrieved from http://www.cs.utk.edu/~berry/papers02/GTPchap.pdf

[13] R-Project web site, http://www.r-project.org/

[14] LSA spaces website at Colorado University, http://lsa.colorado.edu/spaces.html