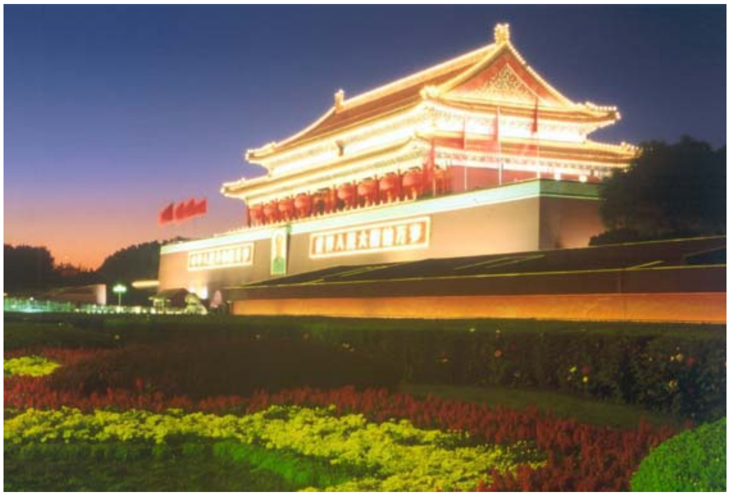
WSS 2004

The Second International Workshop on Web-based Support Systems

In Conjunction with IEEE/WIC/ACM WI/IAT'04



Beijing, China, September 20, 2004

Edited by

J. T. Yao
University of Regina

V. V. Raghavan
University of Louisiana at
Lafayette

G.Y. Wang
Chongqing University of Posts
and Telecommunications

ISBN: 0-9734039-6-9

WSS 2004

The Second International Workshop on Web-based Support Systems

In Conjunction with IEEE/WIC/ACM WI/IAT'04

Beijing, China, September 20, 2004

Edited by

J. T. Yao
University of Regina

V. V. Raghavan
University of Louisiana at Lafayette

G.Y. Wang
Chongqing University of Posts and Telecommunications

Table of Contents

Keynote

Regular Papers

Preface

Following the success of the first workshop held in Halifax, Canada on October 13, 2003, the Second International Workshop on Web-based Support Systems is held in Beijing, China on September 20, 2004. It aims to provide a forum for the discussion and exchange of ideas and information by researchers, students, and professionals on the issues and challenges brought on by the Web technology for various support systems. One of the goals is to find out how applications and adaptations of existing methodologies on the Web platform benefit the decision-making, research, learning, as well as other activities.

Web-based support systems (WSS) is a newly identified research area. The research of Web-based support systems can be viewed as a natural evolution of the existing research. The first step is the extension of decision support systems, computer aided design, etc. to computerized support systems. With the emergence of Web technology and Web Intelligence (WI), the needs to study Web-based support systems are obvious. The current and previous proceedings have demonstrated that WSS is and will attract more research interests.

We are quite pleased with the quality and diversity the accepted papers. Each paper has been reviewed by three program committee members. The current 22 papers included in the proceedings were selected from more than 30 submissions.

A Web site devoted to the research of WSS has been set up at http://www2.cs.uregina.ca/~wss/. There are articles on the Bibliography page of the Web site. If you want your publications to be listed on the page or identify yourself as a researcher in the area of WSS, please send information to wss@cs.uregina.ca. The current proceedings are also online at http://www2.cs.uregina.ca/~wss/wss04/wss04.pdf. The previous proceedings are at http://www2.cs.uregina.ca/~wss/wss03/wss03.pdf.

We would like to express our gratitude to program committee members for assisting with the reviewing process and helping us put together a very solid program in a short time. Many people helped the workshop in one way or another. We appreciate very much the support and encouragement from WI/IAT'04 Conference Chair Dr. Jiming Liu, Program Chair Dr. Ning Zhong and Workshop Chair Dr. Pawan Lingras. Thanks to Dr. Yiyu Yao for kindly agreeing to give a keynote speech. Thanks to Wei-Ning Liu and Songlun Zhao of University of Regina for maintaining WSS Web site and submission account and managing PDF documents. Most importantly, we would like to thank all authors for their submissions and participation in the workshop.

Enjoy the workshop, and have fun in Beijing.

WSS'04 Chairs
JingTao Yao, Vijay V. Raghavan and Guoyin Wang

Program Chairs

Dr. JingTao Yao (University of Regina, Canada)

Dr. Vijay V. Raghavan (University of Louisiana at Lafayette, USA)

Dr. Guoyin Wang (Chongqing University of Posts and Telecommunications, China)

Program Committee Members

C. Butz (University of Regina, Canada)
B. V. Dasarathy (Editor-in-chief, Information Fusion Journal, Consultant, USA)
E. Diaz (University of Louisiana, USA)
I. Duntsch (Brock University, Canada)
J.C. Han (California State University, USA)
T. Hu (Drexel University, USA)
Y.G. Hu (China Agricultural University, China)
D. Jutla (Saint Mary's University, Canada)
Kinshuk (Massey University, New Zealand)
V. Kreinovich (University of Texas at El Paso, USA)
J. Z. Li (University of Calgary)
Y. F. Li (Queensland University of Technology, Australia)
T.Y. Lin (San Jose State University, USA)
J. M. Liu (Hong Kong Baptist University, China)
Q. Liu (Nanchang University, China)
P. Lingras (Saint Mary's University, Canada)
J. Lu (University of Technology Sydney, Australia)
J. F. Peters (University of Manitoba, Canada)
V. V. Raghavan (University of Louisiana, USA)
G. Ruhe (University of Calgary, Canada)
H. Sever (Baskent University, Turkey)
S. Tsumoto (Shimane Medical University, Japan)
G. Y. Wang (Chongqing University of Posts and Telecommunications, China)
Z. H. Wu (University of Louisiana, USA)
Y. Xie (University of Louisiana, USA)
J. T. Yao (University of Regina, Canada)
Y. Y. Yao (University of Regina, Canada)
Y.-Q. Zhang (Georgia State University, USA)
N. Zhong (Maebashi Institute of Technology, Japan)

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 1-6, 2004.

Web-based Research Support Systems

Y.Y. Yao

Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2
E-mail: yyao@cs.uregina.ca
http://www.cs.uregina/∼yyao

Abstract

Web-based research support systems (WRSS) are a specific type of Web-based support systems (WSS) that support research activities for scientists. They are motivated by the challenges and opportunities of the Web, as well as the needs of scientists. The recent advancements of computer and Web technologies make the implementation of WSS feasible. A general framework of WSS is presented by considering its basic issues. Within the framework, the principles and issues of WRSS are examined. From the conceptual point of view, WRSS may be considered as a new subfield of study. It focuses on a more systematic and coherent treatment of existing isolated studies of research support. From the implementation point of view, WRSS are based on assembling, integration, and adaptation of existing computer technology and information systems for the purpose of research support. The results of WRSS may lead to new and viable research tools.

1. Introduction

The advance and development of the Web has lead to many innovations in the applications of Web technology. One has to reconsider the existing methods and re-design or modify the existing systems to meet the challenges, as well as take the advantages of the Web. The Web is used as a universal interface and the underlying infrastructure for Intelligent Web Information Systems (IWIS) [24]. Web Intelligence (WI) emerged naturally as a new field of study to cover such recent research that explores the information, structures, and semantics of the Web for the design and implementation of Web empowered systems [10, 24, 25, 26].

Many types of Web-based Support System (WSS) have been considered recently by many researchers [19, 21]. An example of such systems is Web-based Decision Support Systems (WDSS) [13]. Based on the observations of existing studies, Yao and Yao argue that it is the time to treat Web-based support systems as a new and separate sub-area of Web intelligence [21]. The first workshop of WSS was held successfully in 2003 (http:/www.cs.uregina/∼wss). The papers published in WSS 2003 proceedings cover a variety of Web-based support systems, including decision support [9, 11], research support [14, 17, 18], retrieval support [3], teaching and learning support [4, 16], data mining support [18], and many more.

Web-based Research Support Systems (WRSS) can be viewed as a specific type of Web-based support systems. By examining the basic issues of WRSS, we attempt to demonstrate the usefulness of the concept of WSS. In addition, the study of WRSS can bring more insights into other fields of studies. For example, intelligent agents and data mining can be applied to support research, and research methods in turn can be used to guide studies of data mining [23].

2. Overview of Web-based Support Systems

The motivations and basic issues of WSS, the needs for WSS, the scope of WSS, the characteristics of WSS, and a general framework of WSS have been discussed in detail by Yao and Yao in their WSS 2003 paper [21]. Instead of repeating the same discussion, the original paper is reprinted here for easy reference. In this section, we only briefly address a few additional issues.

There are two important features of WSS. They can be understood as extensions of existing research in two dimensions, as shown in Table 1. In the application dimension, represented by the rows in the table, WSS cover support systems in many different domains. They can be viewed as natural extensions of decision support systems [15]. In the technology dimension, represented by columns in the table, WSS use the Web as a new platform for the delivery of support. Along the application dimension, the lessons and experiences from DSS can be easily applied to other domains. Along the technology dimension, the new advances in tech-

	Technology		
Application domain	Computer technology	Web technology	...
Decision making	DSS	WDSS	...
Business application	BSS	WBSS	...
Information retrieval	IRSS	WIRSS	...
Scientific research	RSS	WRSS	...
Teaching	TSS	WTSS	...
Medical application	MSS	WMSS	...
Knowledge management	KMSS	WKMSS	...
Data mining	DMSS	WDMSS	...
...	...	...	...

$$\mathcal{A}\ \text{(a particular domain)}\ +\ \text{support systems}\ = \mathcal{A}\ \text{support systems}$$
$$\text{Web}\ + \mathcal{A} +\ \text{support systems}\ = \text{Web-based}\ \mathcal{A}\ \text{support systems}$$

Table 1. Two diemensional view of WSS

nology can lead to further innovations in support systems.

The two-dimensional view of WSS provides an easy classification. Schematically, suppose $\mathcal{A}$ is a specific domain, a computerized support system for domain $\mathcal{A}$ can be termed as an $\mathcal{A}$ support system. The use of the Web results in Web-based $\mathcal{A}$ support systems. Based on such a scheme, we repeated the same searches reported by Yao and Yao [21]. The results are summarized in Table 2. By comparing the number of hits in 2003 and 2004, respectively, one can observe that there is a growing interest in Web-based support systems. By examining some of the returned lists from Google, one can also see that WSS workshops have a solid contribution to such a growth.

3. Web-based Research Support Systems

Many computerized systems, although not designed specifically for research support, have in fact been used by scientists in different stages of research. Web-based research support systems aim at pooling together all these isolated efforts and un-integrated systems with a common goal of research support.

Research activities can be broadly classified into two levels, the institutional level and the individual level [14]. The institutional level deals with the management of research and research projects in an institution. The individual level is the the actual research process of a scientist. We restrict the discussion to the individual level support [23].

3.1 Scientific Research in the Web Age

The impact of computer technology on research can be felt by every scientist. Computer software and information systems have been implemented to support scientists in many activities, such as communication, literature search, data analysis and manuscript preparation. As new technologies evolve and existing technologies expand, a scientist needs to adjust accordingly and make full use of them when carrying out research.

The advantages of the Web are often emphasized without mentioning the related difficulties. Ideally, we need to consider the problems coming with the Web, to be consistent with metaphor that the same coin has two sides. The growth of the Web, as well as information, tools, software, and services on the Web, makes scientific research easier from the point of view of easy access of information and tools. On the other hand, the limited human processing capacity becomes even more pronounced with the explosion of information, tools, and services. The opportunities and challenges offered by the Web for a scientist are summarized as follows:

- **Information on the Web.** With the fast growth of the Web and easy availability of information on the Web, we have arrived at a new information age. The Web provides a new medium for gathering, storing, processing, presenting, sharing, and using information. There is a tremendous amount of online materials, such as articles, journals, newspapers, databases, digital libraries, and so on.

  The easy accessibility and huge amount of information on the Web result in many difficulties for scientists, such as information overload, misinformation, fees, poorly designed navigation, retrieval, and browsing tools [6]. How to make effective use of information on the Web becomes a serious problem. How to digest the materials on the Web and evaluate their quality are related difficult problems.

- **Web-based tools.**

  The advance of science and technology normally leads

	# of Hits	
Search phrase	Aug. 2003	Aug. 2004
Decision support system	212,000	241,000
Decision support systems	332,000	402,000
Web-based decision support system	891	745
Web-based decision support systems	583	629
Web-based decision support	3,460	5490
Business support system	4,180	5,090
Business support systems	11,400	12,600
Web-based business support system	3	4
Web-based business support systems	27	30
Web-based business support	87	147
Negotiation support system	1,270	965
Negotiation support systems	1,680	1,710
Web-based negotiation support system	96	273
Web-based negotiation support systems	294	100
Web-based negotiation support	408	383
Information retrieval support system	39	31
Information retrieval support systems	98	184
Web-based information retrieval support system	0	2
Web-based information retrieval support systems	33	80
Web-based information retrieval support	33	82
Research support system	750	743
Research support systems	48	475
Web-based research support system	2	15
Web-based research support systems	25	44
Web-based research support	33	69
Teaching support system	231	237
Teaching support systems	118	89
Web-based teaching support system	1	9
Web-based teaching support systems	2	8
Web-based teaching support	108	160
Medical support system	1,180	914
Medical support systems	1,010	809
Web-based medical support system	0	2
Web-based medical support systems	0	6
Web-based medical support	33	49
Knowledge management support system	433	286
Knowledge management support systems	90	78
Web-based knowledge management support system	340	184
Web-based knowledge management support systems	1	2
Web-based knowledge management support	414	224
Data mining support system	7	26
Data mining support systems	2	10
Web-based data mining support system	0	2
Web-based data mining support systems	0	2
Web-based data mining support	0	2

Table 2. Summary of the Google search results on WSS

to new and improved tools and equipment for scientists. The development of the Web is no exception. There are also many products, tools and services on the Web, such as news groups, downloadable software, document delivery systems, and so on. A wide spectrum and a huge number of available software systems, such as those used for data analysis, simulation, graphical representation, and document preparation are available. Those tools enable scientists to increase their research quality and productivity.

With the increased number of tools, software, and services, it is a challenge to select the right tools and techniques. It may also take more time to learn to use a new tool or software. Scientists are faced with the problem of keeping in pace with the development of new tools, which may take up their valuable research time.

The quantity of information on the Web does not imply an increased quality and productivity in research. Similarly, the existence of new tools and software does not automatically lead to new scientific discoveries. Scientists can only make new advances by making effective use of information and tools. It is therefore necessary to support scientists to meet such challenges. Web-based research support systems are built for such a purpose. They will support scientists by automatizing many routine activities, effectively managing available information and tools, transforming information into useful knowledge, and so on.

3.2 Research Process and Methods

It is generally agreed that there are some basic principles and techniques that are commonly used in most types of scientific investigations [2]. The study of research methods adopts the view that scientists follow a fairly systematic process in scientific investigations [1, 2, 5, 8, 12]. A model of the research process can be briefly described [5, 7, 12]:

- **Idea-generating phase**. The objective is to identify a topic of interest to study. It may also be referred to as the preparation [2] or the exploration phase. Curiosity, interest, enthusiasm, intuition, imagination, tolerance of uncertainty, diversity, and communication with colleagues are some of the critical ingredients in idea generation [2, 8]. Literature search and reading also play important roles in this phase [2, 8].

- **Problem-definition phase**. The objective is to precisely and clearly define and formulate vague and general ideas generated in the previous phase. Problem definition involves careful conceptualization and abstraction. The success in problem definition increases the probability of a successful research project. With respect to a precisely defined problem, it is relatively easy to find related and solved problems, as well as potential solutions.

- **Procedure-design/planning phase**. The objective is to make a workable research plan by considering all issues involved, such as expected findings and results, available tools and methodologies, experiments, system implementation, time and resource constraints, and so on. This phase deals with planning and organizing research at the strategic level [2].

- **Observation/experimentation phase**. The objective is to observe real world phenomenon, collect data, and carry out experiments. Depending on the nature of the research disciplines, various tools and equipment, as well as different methods, can be used.

- **Data-analysis phase**. The objective is to make sense out of the data collected. One extracts potentially useful information, abstraction, findings, and knowledge from data. Statistical software packages can be used.

- **Results-interpretation phase**. The objective is to build rational models and theories that explain the results from the data-analysis phase. It is necessary to investigate how the results help answer the research question, and how this answer contributes to the knowledge of the field. The connections to other concepts and existing studies may also be established.

- **Communication phase**. The objective is to present the research results to the research community. Communication can be done in either a formal or an informal manner. Books and scientific journals are the traditional communication media. Web publication is a new means of communication. Oral presentation at a conference, or discussion with colleagues, is an interactive means of communication.

It is possible to combine several phases into one, or to divide one phase into more detailed steps. The division between phases is not a clear cut. Moreover, the research process does not follow a rigid sequencing of the phases. Iteration of different phrases may be necessary [5].

3.3 Functionalities of WRSS and Related Computer Technologies

To support a large spectrum of research activities, WRSS must be flexible and have many functionalities. These functionalities and the required computer technologies are summarized in this section, based on a recent paper by Yao [23].

Profile management. The profile management deals with a scientist's profiles. There may exist different classes of profiles, such as research interest, personal libraries, address books, Web bookmarks, and many more. A main component of profile management is the knowledge base, which serves as the basis of WRSS. Profile management module collects, organizes, and stores all relevant information for a scientist.

Resource management. Many types of resources exist for supporting research. Examples are human resources, tool resources, and information/knowledge resources. The main functions of human resource management are the maintenance of expert reservoir, and the matching and retrieval of a useful group of experts. The reservoir of experts may be a virtual one, which consists of links to other systems, databases, or scientists' home pages. The tool resource and information/knowledge resource management modules maintain different types of objects, but have similar functions. The information resources are combined from many sources, such as libraries, digital libraries, and the Web. Database, knowledge base, information retrieval, and agent technologies can be used. Web search engines can be used for retrieval.

Data/knowledge management. Typically, research involves the collection and processing of a large amount of data. WRSS must have a module to record the useful data, information and knowledge during the entire research process. The module must contain some data/knowledge operations and retrieval facilities. Database and information retrieval systems can be used.

The profile, resource, and data managements form a solid basis of WRSS. Consider now the following specific supporting functionalities:

- **Exploring support**. In the early stage of research, a scientist may have a vague idea and may not be aware of the works of fellow researchers. Exploration plays an important role. There are many means of exploration, such as browsing databases, libraries, and the Web. A scientist's profile may be useful in focusing the exploration areas. If the Web is used for browsing, the historical data can be tracked. The collected data can be analyzed using machine learning and data mining tools to provide a scientist useful information and hints. The profile can also be updated. Currently, Web browsers are a useful exploration tool. Their functions need to be expanded for providing research support.

- **Retrieval support**. Once a scientist forms relatively solid ideas, it is necessary to search the literature to find relevant information. Retrieval support assists retrieval related activities, such as browsing, searching, organization, and utilization of information [20, 22, 23].

- **Reading support**. Reading critically and extensively is important, especially in the preparation stage [2, 8]. The advances in digital libraries and electronic publications make the reading support possible. Software packages exist so that a reader can add book marks, make notes, link different parts of an article, and make logical connections of different articles. A reading support system needs to assist a reader in actively finding relevant materials, as well as constructing cognitive maps based on the materials read. Reading support systems can be combined with exploring and retrieval support systems. Machine learning and text mining methods can be used to assist a reader by learning from the reading history. Agent technology can be used to actively look for useful information and periodically inform scientists with new information. On-line dictionaries may also be useful in reading support.

- **Analyzing support**. Successful analyzing support depends on tool management. It is necessary to help a scientist to find the right tool for a particular problem in analyzing data. In addition, the system should also assist a scientist in using a tool. An explanation feature may be needed, which answers the question why a particular tool is used. If the functions of tools are described as plain text, information retrieval systems can be used to find the right tool. Computer graphics and visualization may be useful in analyzing support.

- **Writing support**. There are many writing support software tools, such as word-processor and typesetting software. Many packages come with additional functions, such as spelling-checking, grammar-checking, and various other agents. A writing support system should also contain some functions mentioned in the retrieval support systems. For example, a writing support system can find relevant articles based on the text written by a scientist and suggest possible references.

A research support system consists of many sub-systems to support different activities. They share common data and knowledge bases. As one can not have a clear classification of research activities, it is difficult to have a clear classification of different types of support sub-systems.

4 Conclusion

The emerging interdisciplinary study of Web-based support systems is motivated by the challenges and opportunities of the Web. It focuses on the theories, technologies and tools for the design and implementation of Web-based systems that support various human activities.

As a specific type of WSS, Web-based research support systems assist scientists to improve their research quality and productivity. The feasibility of such systems is based on the assumption that there exists a relatively systematic approach in scientific research. Furthermore, a general research process can be established, consisting of several steps or phases, such as idea generation, exploration, problem definition, procedure design and planning, observation and experimentation, data analysis, results interpretation, and communication. A number of activities are involved in each of these phases. A WRSS supports each of the activities, such as exploring, retrieving, reading, analyzing, and writing.

The study of Web-based support systems, a useful subclass of intelligent Web information systems, will result in many applications of Web intelligence.

References

[1] Adams, G.R. and Schvaneveldt, J.D. *Understanding Research Methods*, Longman, New York, 1985.

[2] Beveridge, W.I.B. *The Art of Scientific Investigation*, Vintage Books, New York, 1957.

[3] Curra, K. and Higgins, L. A Web-based intelligent case-based reasoning legal aid retrieval information system, in [19], 63-67, 2003.

[4] Fan, L. and Yao, Y.Y. Web-based learning support systems, in [19], 43-48, 2003.

[5] Graziano, A.M and Raulin, M.L. *Research Methods: A Process of Inquiry*, 4th edition, Allyn and Bacon, Boston, 2000.

[6] Hoggan, D.B. Challenges, strategies, and tools for research scientists: using Web-based information resources, *Electronic Journal of Academic and Special Librarianship*, **3**, 2002.

[7] Hult, C.A. *Researching and Writing Across the Curriculum*, Wadsworth Publishing Company, Belmont, California, 1990.

[8] Ladd, G.W. *Imagination in Research: An Economist's View*, Iowa State University Press, Ames, Iowa, 1987.

[9] Li, J. and Ruhe, G. Web-based decision support for software release planning, in [19], 13-20, 2003.

[10] Liu, J. Web Intelligence (WI): what makes Wisdom Web? *Proceeding of IJCAI 2003*, 1596-1601, 2003.

[11] Lu, J., Zhang, G. and Shi, C. Framework and implementation of a Web-based multi-objective decision support system: WMODSS, in [19], 7-11, 2003.

[12] Martella, R.C., Nelson, R. and Marchard-Martella, N.E. *Research Methods: Learning to Become a Critical Research Consumer*, Allyn and Bacon, Boston, 1999.

[13] Power, D.J. and Kaparthi, S. Building Web-based decision support systems, *Studies in Informatics and Control*, **11**, 291-302, 2002.

[14] Tang, H., Wu, Y., Yao, J.T., Wang, G.Y. and Yao, Y.Y. CUPTRSS: a Web-based research support system, in [19], 21-28, 2003.

[15] Turban, E. and Aronson, J.E. *Decision Support Systems and Intelligent System*, Prentice Hall, New Jersey, 2001.

[16] Wetprasit, R. Developing an intelligent Web-based Thai tutor: some issues in the temporal expert, in [19], 49-53, 2003.

[17] Xiang, X., Huang, Y. and Madey, G. A Web-based collaboratory for supporting environmental science research, in [19], 29-36, 2003.

[18] Xu, J., Huang, Y. and Madey, G. A research support systems framework for Web data mining, in [19], 37-41, 2003.

[19] Yao, J.T. and Lingras, P. (Eds.), *Proceedings of 2003 WI/IAT Workshop on Applications, Products and Services of Web-based Support System (WSS 2003)*, Saint Mary's University, Canada, 2003.

[20] Yao, J.T. and Yao, Y.Y. Web-based information retrieval support systems: building research tools for scientists in the new information age, *Proceedings of the IEEE/WIC International Conference on Web Intelligence*, 570-573, 2003.

[21] Yao, J.T. and Yao, Y.Y. Web-based support systems, in [19], 1-6, 2003.

[22] Yao, Y.Y. Information retrieval support systems, *Proceedings of FUZZ-IEEE'02*, 773-778, 2002

[23] Yao, Y.Y. A framework for Web-based research support systems, *Proceedings of COMPSAC'2003*, 601-606, 2003.

[24] Yao, Y.Y., Zhong, N., Liu, J. and Ohsuga, S. Web Intelligence (WI): research challenges and trends in the new information age, *Web Intelligence: Research and Development, LNAI 2198*, Springer, Berlin, 1-17, 2001.

[25] Zhong, N. Toward Web intelligence, *Advances in Web Intelligence, LNAI 2663*, Springer, Berlin, 1-14, 2003.

[26] Zhong, N., Liu, J. and Yao, Y.Y. (Eds.), *Web Intelligence*, Springer, Berlin, 2003.

Web-based Support Systems

J.T. Yao Y. Y. Yao

Department of Computer Science, University of Regina
Regina, S4S 0A2, Canada
E-mail: {jtyao, yyao}@cs.uregina.ca

Abstract

Web-based support systems (WSS) concern multidisciplinary investigations which combine computer technologies and domain specific studies. Domain specific studies focus on the investigation of activities in a particular domain. Computer technologies are used to build systems that support these activities. Fundamental issues of WSS are examined, a framework of WSS is presented, and research on WSS is discussed. It is expected that WSS will be accepted as a new research area.

1. Introduction

The advances in computer technologies have affected everyone in the use of computerized support in various activities. Traditional decision support systems focus on computerized support for making decision with respect to managerial problems [11]. There is an emerging and fast growing interest in computerized support systems in many other domains such as information retrieval support systems [12, 14], research support systems [14], teaching and learning support systems, computerized medical support systems [9], knowledge management support systems[1, 5], and many more. The recent development of the Web generates further momentum to the design and implementation of support systems.

This paper investigates the emerging field of computerized support systems in general and Web-based support systems (WSS) in specific. WSS are viewed as a multidisciplinary research involving the integration of domain specific studies and other disciplines such as computer science, information systems, and the Web technology, to only name a few. There is a sufficient evidence showing a strong trend for studies of computerized support systems in addition to decision support systems. Investigations of WSS in a wide context may result in many new research topics and more effective systems.

In the rest of the paper, we focus on the following specific objectives:

- to provide a precise characterization of computerized support systems, and to identify and examine the needs, rationalities, as well as trends of such systems (Section 2.1);
- to understand, study and analyze the feasibility and advantages of transferring support systems to the Web platform (Section 2.2);
- to identify the scope of WSS (Section 2.3);
- to establish a general framework for Web-based support systems (Section 3);
- to address some basic research issues related to WSS (Section 4).

2. Issues of Web-based Support Systems

2.1. Computerized Support systems

It is a dream of every computer scientist to develop a fully automated computer system which has the same or even a higher level of intelligence as human beings. However, the technologies we mastered can only design and develop systems that have some abilities to assist, support, and aid us for various activities. In fact, one of the popular definitions of artificial intelligence (AI) is "*the study of how to make computers do things at which, at the moment, people are better*" [7]. AI is one of the important and popular research topics in computer science. The research proves that it is almost impossible to replace human intelligence with computer systems, at least within the foreseeable future. With this restriction, we have to lower the expectation of our dreams. Decision support systems (DSS), computer aided software engineering (CASE), and computer aided design (CAD) systems are some examples of such systems to fulfill more practical goals.

As a field of study, computerized support systems is an interdisciplinary research area. A particular support system with specific domain knowledge provides support to a specific field. The most popular and successful example is the decision support systems (DSS). DSS was defined as *"computer-based information systems that combine models and data in an attempt to solve nonstructured problems with extensive user involvement through a friendly user interface"* [11]. It can be viewed as a hybrid product of two domains of studies. DSS are derived from management science and computer science. The same principle applies to other types of support systems. For instance, a medical support system or a medical expert system is the product of the marriage between medical science and computer science. Research support systems are the combination of research methodology and computer science [14]. In general, a specific support system aims to support activities and operations of the specific domain.

Various support systems have been studied for a long time. Schematically, suppose $\mathcal{A}$ is a specific domain, a support system for domain $\mathcal{A}$ can be termed as an $\mathcal{A}$ support system. Following this, we used one of the most popular search engines Google [3] for our background studies. Table 1 shows the search results we obtained in August 2003. The first column 'Search Phrase' is the phrase we used for exact phrase search. The second column '# of Hits' is the number of links returned by Google with the search phrase. It can be seen that people have done numerous research on various support systems. Decision support system(s), business support system(s), negotiation support system(s) and medical support system(s) are amongst the highest returned hits. An interesting observation from Table 1 is that the majority of support systems with high hit rates are business and management oriented. Technical oriented support systems had not been paid attention by researchers. Therefore, we should investigate more on technical oriented support systems such support as for data mining, research, and learning. Further more, there are also emerging needs for moving support systems to the Web platform.

2.2. Support systems in the Web age

The Web provides a new medium for storing, presenting, gathering, sharing, processing and using information. The impacts of the Web can be felt in almost all aspects of life. We aim to study the issues and challenges brought on by the Web technology for various support systems. One of the goals is to find out how applications and adaptations of existing methodologies on the Web platform benefit our decision-makings and various activities. A list of benefits of the Web technology is given bellow.

1. The Web provides a distributed infrastructure for information processing.
2. The Web is used as a channel to discuss one of the most popular support systems, DSS [4].
3. The Web can deliver timely, secure information and tools with user friendly interface such as Internet Explorer and Netscape.
4. The Web has no time or geographic restrictions. Users can access the system at any time, any place.
5. Users can control and retrieve results remotely and instantly.

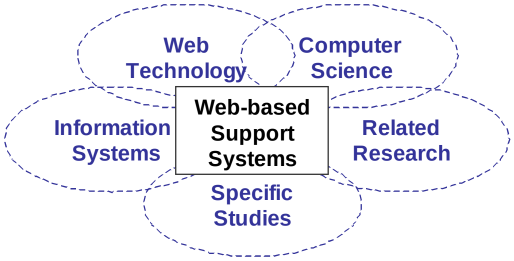


Figure 1. WSS: A multidisciplinary research

Although the advantages of applying the Web technology to support systems are obvious, the concept of Web-based support systems has not been paid enough attention by researchers. It is clear to see from the search results obtained in Table 1 that the number of hits for each type of Web-based support systems is dramatically lower than its computerized support system counterpart. For instance, the hits of the search of "Medical support system" and "Medical support systems" both reached 1,000. However, there was none when we change the phrase to "Web-based medical support system" or "Web-based medical support systems". The majority of returns from "Web-based medical support" were not related to computerized systems. Although the hits were 33, Google returned only 18 links with similar sites omitted according to its criteria. In fact, 13 out of 18 links pointed to a single research paper entitled "Intranet Health Clinic: Web-based medical support services employing XML" [8]. Web-based decision support systems [6] is one of the pioneer research areas of WSS. The returns of "Web-based decision support system(s)" were also higher than others.

2.3. Scope of Web-based support systems

WSS is a multidisciplinary research area as depicted in Figure 1. It involves many research domains. We classify the scope of WSS in four categories: WSS for specific domains, Web-based applications, techniques related to WSS,

Search Phrase	# of Hits
Decision support system	212,000
Decision support systems	332,000
Web-based decision support system	891
Web-based decision support systems	583
Web-based decision support	3,460
Business support system	4,180
Business support systems	11,400
Web-based business support system	3
Web-based business support systems	27
Web-based business support	87
Negotiation support system	1,270
Negotiation support systems	1,680
Web-based negotiation support system	96
Web-based negotiation support systems	294
Web-based negotiation support	408
Information retrieval support system	39
Information retrieval support systems	98
Web-based information retrieval support system	0
Web-based information retrieval support systems	33
Web-based information retrieval support	33
Research support system	750
Research support systems	48
Web-based research support system	2
Web-based research support systems	25
Web-based research support	33
Teaching support system	231
Teaching support systems	118
Web-based teaching support system	1
Web-based teaching support systems	2
Web-based teaching support	108
Medical support system	1,180
Medical support systems	1,010
Web-based medical support system	0
Web-based medical support systems	0
Web-based medical support	33
Knowledge management support system	433
Knowledge management support systems	90
Web-based knowledge management support system	340
Web-based knowledge management support systems	1
Web-based knowledge management support	414
Data mining support system	7
Data mining support systems	2
Web-based data mining support system	0
Web-based data mining support systems	0
Web-based data mining support	0

Table 1. Search results with Google

and design and development of WSS. Some suggested topics are listed below:

- Web-based support systems for specific domains:
  - Web-based decision support systems
  - Enterprise-wide decision support systems
  - Web-based group decision support systems
  - Web-based executive support systems
  - Web-based business support systems
  - Web-based negotiation support systems
  - Web-based medical support systems
  - Web-based research support systems
  - Web-based information retrieval support systems
  - Web-based education support systems
  - Web-based learning support systems
  - Web-based teaching support systems
- Web-based applications
  - Web-based knowledge management systems
  - Web-based groupware systems
  - Web-based financial and economic systems
  - Internet banking systems
  - Web-based multimedia systems
- Techniques related to WSS:
  - XML and data management on the Web
  - Web information management
  - Web information retrieval
  - Web data mining and farming
  - Web search engines
- Design and development of WSS:
  - Web-based systems development
  - CASE tools and software for developing Web-based applications
  - Systems analysis and design methods for Web-based applications
  - User-interface design issues for Web-based applications
  - Visualizations of Web-based systems
  - Security issues related to Web-based applications

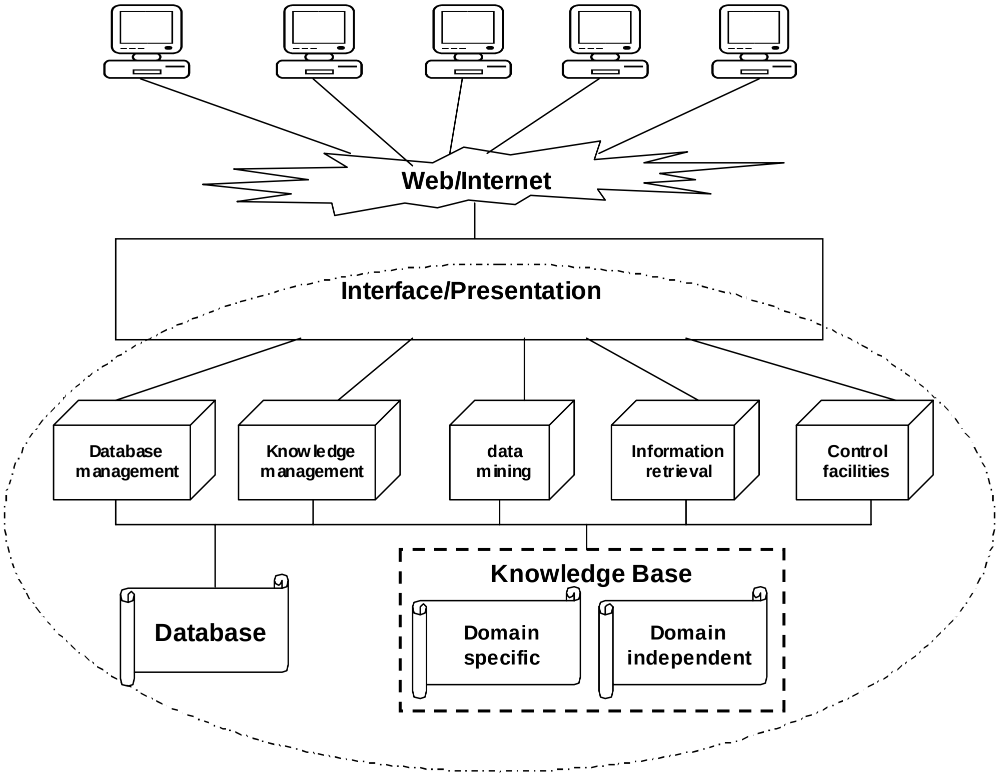


Figure 2. An Architecture of Web-based Support Systems

3. A Framework of Web-based Support Systems

Interface, functionality, and databases are some of the components which need to be considered when we design a system. We can view the architecture of WSS as a (thin) clint/server structure [2] as shown in Figure 2. The users, including decision makers and information seekers, are clients on the top layer. They access the system with browsers via the Web and the Internet. The interface that is designed on the server side will be presented on the client's side by browsers. The lower layers and components encapsulated by the oval dotted line are, in fact, very similar to conventional computerized support systems. In other words, a Web-based support system is a support system with the Web and Internet as the interface.

There are two components on the data layer. Database is a basic component in any modern system. WSS is not an exception. Another major component is the knowledge base. It stores all rules, principles and guidelines used in supporting activities. We intend to divide the knowledge base into two parts: domain specific knowledge base and domain independent knowledge base. The former is the knowledge specific to the domain. The latter involves general knowledge for all support systems.

Knowledge management, data management, information retrieval, data mining and other control facilities form the management layer. They serve as the middleware of the three-tier client/server architecture. They are the intermediaries between interface and data layers. Reasoning, inference and agent technologies will play important roles on this layer. The split of data and user results in a secure and standardized system. To take advantage of the Web technology, these processes are distributed over the Internet to form a virtual server. In fact, databases and knowledge bases on the lower tier are also distributed.

Web-based support systems can be classified into three levels. The first level is support for personal activities. An example of such support is research support for individuals [14]. Personal research activities such as search, retrieval, reading and writing are supported. The second level is the organizational support, such as research support on an institute level [10]. The top level is the network level. The collaborations between organizations or decision making by a group of people like in group decision support systems fall in this level. The group decision support room may be a virtual room on the Web.

4. Research on Web-based Support Systems

The research on Web-based support systems can be classified into a few categories. The first class is the study of a specific support system and related technology as indicated in Section 2.3. There are four types of existing research, namely, WSS for specific domains, Web-based applications, techniques related to WSS and design, and development of WSS, that can be classified as WSS research.

On a more general level of research on WSS, we may include the study of WSS operations and support facilities. The study of WSS operations aims to understand the needs of supporting domains such as business logic and management concerns. The study of support facilities focuses on potential support functionalities that computer science and Web technology can provide. There are two types of operations, i.e, domain independent operations and domain specific operations. Domain independent support facilities and domain specific support facilities are two types of support facilities.

The study of operations will help us to gain a deeper understanding of WSS. Domain independent operations may include operational controls such as report generating and graphical multimedia presentation, managerial control such as negotiation and evaluation, strategic planning such as technology adoption and quality assurance. These domain specific operations may include class schedules for teaching support and images processing for medical support.

With the understanding of operations, various support facilities can be studied. They may include techniques such as data mining, information retrieval, optimization, simulation heuristics, and inference. The support facilities could also be classified into levels. For instance, a Web-based research support may provide two levels of support: managing support for management staff and activities support for individual researchers [10, 14].

5. Conclusion

The research of Web-based support systems is a natural evolution of the existing research. The first step is the extension of decision support systems to computerized support systems. With the emergence of Web technology and Web intelligence, the need to study Web-based support systems are obvious. We identify the domain and scope of Web-based support systems. A framework with the viewing angle from a client/server facility is presented. We also discuss the issues of research on WSS. It is expected that WSS, as a new identified research area, will attract more research.

References

[1] M. Ginsburg, A. Kambil, Annotate: A Web-based Knowledge Management Support System for Document Collections, *Procedeeings of HICSS-32*, 1999.

[2] J. Goldman, P. Rawles, J. Mariga, *Client/server information systems: a business-oriented approach*, John Wiley & Sons, 1999.

[3] Google: http://www.google.com

[4] ISWorld DSS research page:
http://www.isworld.org/dss/index.htm.

[5] R. Otondo, J. Simon, A Model for the Study of Knowledge Management Support Systems, *Proceedings of the 6th Americas Conference for Information Systems*, Long Beach, 2000.

[6] D.J. Power, S. Kaparthi, Building Web-based decision support systems, *Studies in Informatics and Control*, **11**, 291-302, 2002.

[7] E. Rich, K. Knight, *Artificial Intelligence*, McGraw-Hill, 1991.

[8] G. Stalidis, A. Prentza, I.N. Vlachos, G. Anogianakis, S. Maglavera, D. Koutsouris, Intranet Health Clinic: Web-based medical support services employing XML, *Proceedings of the Medical Informatics Europe*, pp1112-1116, 2000.

[9] G. Stalidis, A. Prentza, I.N. Vlachos, S. Maglavera, D. Koutsouris, Medical support system for continuation of care based on XML Web technology, *International Journal of Medical Informatics*, **64**, 385-400, 2001.

[10] H. Tang, Y. Wu, J.T. Yao, G.Y. Wang, Y. Y. Yao, CUP-TRSS: a Web-based Research Support System, *Proceedings WSS'03*, 2003.

[11] E. Turban, J.E. Aronson, *Decision Support Systems and Intelligent System*, Prentice Hall, New Jersey, 2001.

[12] J.T. Yao, Y.Y. Yao, Web-based information retrieval support systems: building research tools for scientists in the new information age, *Proceedings of the IEEE/WIC International Conference on Web Intelligence*, 2003.

[13] Y.Y. Yao, Information retrieval support systems, *Proceedings of FUZZ-IEEE'02*, 773-778, 2002

[14] Y.Y. Yao, A framework for Web-based research support systems, proceedings of COMPSAC'2003, Dallas, USA, Nov 2003 (to appear).

A Learning Algorithm for Multiple Rule Trees

Jiujiang An, Guoyin Wang, Yu Wu
Institute of Computer Science and Technology
Chongqing University of Posts and Telecommunications
Chongqing, 400065, P. R. China
anjiujiang@tom.com wanggy@cqupt.edu.cn wuyu@cqupt.edu.cn

Abstract

It is one of the key problems for web based decision support systems to generate knowledge from huge database containing inconsistent information. In this paper, a learning algorithm for multiple rule trees (MRT) is developed, which is based on ID3 algorithm and rough set theory. MRT algorithm can quickly generate decision rules from inconsistent decision information tables. Both space and time complexities of MRT algorithm are just polynomial, while those of Skowron's default decision rule generation algorithm are exponential. With the increasing of the number of records and core attributes of an information table, Skowron's default algorithm needs more memory and time for generating rules than MRT algorithm. In some cases, Skowron's default decision rule generation algorithm could not generate rules due to the lack of memory. It's proved by our simulation experiment results that MRT algorithm is effective and valid.

1. Introduction

Rough set theory has been applied successfully in such fields as machine learning, data mining and etc., since Prof. Z. Pawlak developed it in 1982 [1]. Generating rules from a decision table is one of the major research topics of rough set theory. Reduct is an important contribution of rough set theory for data mining, and its results are in accordance with rules. In uncertain or inconsistent cases, the default decision rule generation algorithm developed by Prof. Skowron (DDRG) can generate all rules, certainty factors of which are greater than a predefined threshold α_c [2]. At the same time, conflicts between these rules are solved by use of some block rules. It is proved that rules generated by this algorithm have high flexibility in processing unseen data. Based on DDRG algorithm, Prof. Wang developed a self-learning model under uncertain condition [3,4]. This model can automatically get the threshold α_c to generate rules without any prior domain knowledge. Simulation experiment results demonstrate that this model could generate a rather smaller number of rules than DDRG algorithm, and has high correct recognition rate for unseen data.

During the rule generation process of DDRG algorithm an inconsistent decision table will be projected onto many subtables by dropping one of its core attribute. Then DDRG generates rules with discernibility matrix from each subtable repeatedly. Thus, the cost for generating rules of this algorithm is determined by the number of records and core attributes of each subtable. It is illustrated in section 4 that its space and time complexities are exponential. When the number of records and core attributes of an inconsistent decision table are small, DDRG algorithm can quickly generate rules from it. However, with the increasing of the number of records and core attributes, it needs much more memory and time for generating rules, and even fails to generate rules due to the lack of memory.

ID3 is a classical machine-learning algorithm for generating rules from decision tables [5]. A typical tree generation process of ID3 algorithm is as follows. Firstly, a condition attribute with the minimal information entropy is chosen from a decision table. Then this attribute is used to divide the decision table into different classes. The above two steps are repeated until each record of the decision table only belongs to one of these classes. Unfortunately, ID3 algorithm can build a decision tree for a consistent decision table only. It cannot process an inconsistent decision table.

To solve the above problems, a learning algorithm for multiple rule trees (MRT) based on ID3 algorithm and rough set theory is proposed in this paper. MRT algorithm can generate multiple rule trees from an inconsistent decision table, and the certainty factor of each rule is greater than or equal to a predefined threshold. In the rule tree generation process, MRT algorithm computes and stores only a part of the original decision table to create an internal node of rule tree. With the predefined threshold, MRT algorithm judges whether an internal node of rule tree has a branch of child node or not. Thus, the depth and width of rule tree is limited. Its space and time complexities are polynomial. Compared with DDRG algorithm, MRT algorithm generates decision rules from an inconsistent decision table at less memory and time cost. In addition, MRT algorithm generates multiple rule trees to overcome a drawback of ID3 algorithm that rules resulted from it centralize on a few condition attributes. MRT algorithm could be further used to deal with the problem of generating knowledge from huge databases containing inconsistent information for web based intelligent decision support systems.

In Section 2, some basic concepts of rough set theory and rule tree are discussed. MRT algorithm is presented in detail in Section 3. Its space and time complexities are analyzed and compared with DDRG algorithm in Section 4. In Section 5, some simulation experiments are done to test our results. In Section 6, we draw a conclusion for this paper.

2. Basic concepts

For the convenience of later discussion, some related concepts about rough set theory and rule tree are introduced here.

Def. 1 A decision table is defined as $S=<U, R, V, f>$, where U is a finite set of objects and $R=C\cup D$ is a finite set of attributes. C is the condition attribute set and D is the decision attribute set, $V=\cup V_a$ is a union of the domain of each attribute of R. Each attribute has an information function $f: U\times R\rightarrow V$.

Def. 2 Given a decision table $S=<U, R, V, f>$, C and D are its condition attribute set and decision attribute set separately. Condition class E_i is defined as $E_i\in U/IND(C)$, where, $i=1,\ldots,m$, and $m=|U/IND(C)|$. Decision class X_j is defined as $X_j\in U/IND(D)$, where, $j=1,\ldots,n$, and $n=|U/IND(D)|$.

Def. 3 Given a decision table $S=<U, R, V, f>$, C is its condition attribute set. A condition class E_i is called consistent if and only if its all objects have the same decision value. Otherwise, it is called inconsistent.

Def. 4 A decision table S is called certain if and only if its condition classes are all consistent. Otherwise, it is an uncertain one.

Def. 5 Given a decision table $S=<U, R, V, f>$, where $R=C\cup D$, C is its condition attribute set and $D=\{d\}$ is its decision attribute set. The certainty factor of a decision rule $A\rightarrow B$, $CF(A\rightarrow B)$, is defined as $CF(A\rightarrow B)=|X\cap Y|/|X|$, where X is the object set with condition attribute values satisfying formula A, while Y is the object set with decision attribute satisfying formula B.

Def. 6 Given a decision table $S=<U, R, V, f>$, $P\subseteq R$ is an attribute set, $U/IND(P)=\{X_i, 1\leq i\leq n\}$, we define the entropy of P as $H(P)=-\sum_{i=1}^{n}p(X_i)\log(p(X_i))$, where, $p(X_i)=|X_i|/|U|$.

Def. 7 The conditional entropy of an attribute set $Q\subseteq R$ $(U/IND(Q)=\{Y_1,Y_2,\ldots,Y_m\})$ with reference to another attribute set $P\subseteq R$ $(U/IND(P)=\{X_1,X_2,\ldots,X_n\})$ is $H(Q|P)=-\sum_{i=1}^{n}p(X_i)\sum_{j=1}^{m}p(Y_j|X_i)\log(p(Y_j|X_i))$, where, $p(Y_j|X_i)=|Y_j\cap X_i|/|X_i|$, $1\leq i\leq n$, $1\leq j\leq m$.

Def. 8 A rule tree [6] is defined as follows:

(1) A rule tree is composed of one root node, some leaf nodes and some internal nodes.

(2) The root node represents the whole rule set.

(3) Each path from the root node to a leaf node represents a rule.

(4) Each internal node represents an attribute testing. A branch of rule tree represents each attribute class, and a new child node is created from each branch.

Def. 9 A collection consisting of some rule trees is called a rule tree set.

3. MRT algorithm

3.1 MRT algorithm description

The main idea of MRT algorithm is as follows.

Firstly, a condition attribute with the minimal condition entropy with reference to the decision classes is chosen from the original decision table.

Secondly, the chosen condition attribute is used to divide the original decision table into different classes. Each class is further processed in the following way. If its certainty factor with some decision class is 1, a new child node of current node is created and inserted into the rule tree. If its certainty factor is bigger than or equal to the predefined threshold, a new child node of current node is created and inserted into the internal node queue.

Thirdly, the above two steps are repeated until the internal node queue becomes NULL. The whole rule tree is completely built and added into the rule tree set.

At last, the corresponding condition attribute of the root node of the previous rule tree is deleted from the original decision table, the above three steps are done until the original decision table is empty. Consequently, the rule tree set is generated.

For the convenience of illustrating MRT algorithm, the class *CTreeNode* is firstly defined, which means a node of rule tree, contains a decision table and other memberships.

```
Class CtreeNode
{  Table; //A decision table
   NodeID; //Node's ID
```

ParentID; //Parent node's ID
IsLeafNode; //TRUE for leaf node, while FALSE for internal node
};

The other variables needed in describing MRT algorithm are illustrated in Table 1.

Table 1 Variables and their meanings

Variable	Meaning
RootNode	The root node of a rule tree
CurrentNode	Current node
ChildNode	The child node of the current node
InternalNodeQueue	An internal node queue with some internal nodes
Tree	A rule tree
ListTree	A rule tree set with some rule trees
NodeNumber	The sequence number of node in a rule tree.

Then, based on the above definitions, MRT algorithm is given in Algorithm 1.

Algorithm 1: MRT algorithm

Input: Original decision table and a predefined threshold

Output: Rule tree set

Step 1: Initialize *InternalNodeQueue*=NULL, *Tree*=NULL, *ListTree*=NULL, *NodeNumber*=0.

Step 2: If the original decision table is empty, then go to Step 7, else continue.

Step 3: Initialize *RootNode* and insert it into *InternalNodeQueue*, where,

RootNode.Table is set to be the original decision table.

RootNode.ParentID=-1;

RootNode.NodeID=0;

RootNode.IsLeafNode=FALSE.

Step 4: Repeat step 4.1 and 4.2 until *InternalNodeQueue* is NULL.

Step 4.1: Pop the first node from *InternalNodeQueue* and set it to be *CurrentNode*.

Step 4.2: Choose a condition attribute (C_i) with the minimal condition entropy with reference to the decision classes from *CurrentNode.Table*.

Step 4.3: According to the chosen condition attribute (C_i), divide the decision table of the current node into classes $\{E_k|E_k \in CurrentNode.Table|IND(C_i)\}$, where, $k=1, 2, \ldots, m$, and $m=|CurrentNode.Table \mid IND(C_i)|$.

for k=1 to m

{

If($\exists_{X_j \in CurrentNode.Table|IND(D)}$ (CF(|$E_k \cap X_j$|/|E_k|)==1)), then create a new child node *ChildNode* and insert it into *Tree*, where,

ChildNode.Table is set to be NULL for leaf node.

ChildNode.NodeID=++*NodeNumber*;

ChildNode.ParentID=*CurrentNode.NodeID*;

ChildNode. IsLeafNode=TURE.// Leaf node.

If(the predefined threshold $\leq Max_{X_j \in CurrentNode.Table|IND(D)}$ {CF(|$E_k \cap X_j$|/|E_k|)}<1), create a new child node *ChildNode* and insert it into *InternalNodeQueue*, where,

ChildNode.Table is set to be a decision table containing those records of E_k in the current node' table except the column of the condition attribute $C_{i.}$

ChildNode.NodeID=++*NodeNumber*;

ChildNode.ParentID=*CurrentNode.NodeID*;

ChildNode. IsLeafNode=FALSE.//Internal node.

}

Step 4.4: Release the decision table of the current node, and insert *CurrentNode* into *Tree*.

Step 5: Insert *Tree* into *ListTree*, and *Tree*=NULL, *NodeNumber*=0.

Step 6: Delete the corresponding condition attribute of the root node of the previous rule tree from the original decision table, and go to Step 2.

Step 7: Output *ListTree* and stop.

3.2 An example to illustrate MRT algorithm

Table 2 is an inconsistent decision table used in several papers [2,3,4]. It contains three condition attributes, A, B, C, a decision attribute D and 100 records. To illustrate the MRT algorithm more clearly, Table 2 is used as an example to explain the rule generation process of the MRT algorithm here.

Table 2 An inconsistent decision table

U	A	B	C	D
E_1	1	2	3	1(50x)
E_2	1	2	1	2(5x)
E_3	2	2	3	2(30x)
E_4	2	3	3	2(10x)
E_5	3	5	1	3(4x)
E_6	3	5	1	4(1x)

Suppose the predefined threshold is 0.6. The rule tree generation process of MRT algorithm for Table 2 is given as follows.

(1) From Table 2, choose condition attribute *A*, whose condition entropy with reference to the decision classes is minimal. According to *A*, Table 2 is divided into three condition classes $U|IND(A)=\{\{E_1, E_2\}, \{E_3, E_4\}, \{E_5, E_6\}\}$. According to the decision attribute *D*, Table 2 is divided into four decision classes $U|IND(D)=\{\{E_1\},\{E_2, E_3, E_4\}, \{E_5\}, \{E_6\}\}$.

(2) Each condition class is processed in different ways according to the step 4.3 of MRT algorithm.

- Because $CF(\{E_1, E_2\}\rightarrow\{E_1\}) = |E_1 \cap(E_1\cup E_2)|/| E_1\cup E_2)|=0.91$, create a new child node (*A*=1) and insert it into the internal node queue.
- Because $CF(\{E_3, E_4\}\rightarrow\{E_2, E_3, E_4\}) =|(E_3\cup E_4)\cap(E_2\cup E_3\cup E_4)|/|(E_3\cup E_4)|=1$, create a new child node (*A*=2, *D*=2) and insert it into the rule tree. This new child node is a leaf node.
- Because $CF(\{E_5, E_6\}\rightarrow\{E_5\}) =|(E5\cup E6) \cap E5|/|(E5\cup E6)|=0.8$, create a new child node (*A*=3) and insert it into the internal node queue.

The first rule tree gotten at this step is shown in Figure 1.

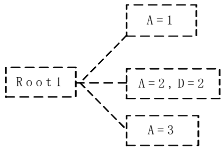


Fig. 1 First rule tree

(3) Pop the first node from the internal node queue and set it to be the current node. Its decision table is shown in Table 3.

Table 3 A part of Table 2

U'	B	C	D
E_1'	2	3	1(50x)
E_2'	2	1	2(5x)

In Table 3, the condition attribute *C* is the one with minimal condition entropy with reference to the decision classes. It is used to divide Table 3 into two condition classes $U'|IND(C)=\{\{E_1'\},\{E_2'\}\}$. According to the decision attribute *D*, Table 3 is divided into two decision classes $U'|IND(D)=\{\{E_1'\}, \{E_2'\}\}$. Each condition class is processed in different ways according to the step 4.3 of MRT algorithm.

- Because $CF(\{ E_1'\}\rightarrow\{ E_1'\}) = | E_1'\cap E_1'|/| E_1'|=1$, create a new child node (*C*=3, *D*=1) and insert it into the rule tree.
- Because $CF(\{E_2'\}\rightarrow\{E_2'\}) = |E_2'\cap E_2'|/| E_2'|= 1$, create a new child node (*C*=1, *D*=2) and insert it into the rule tree.

The first rule tree gotten at this step is shown in Figure 2.

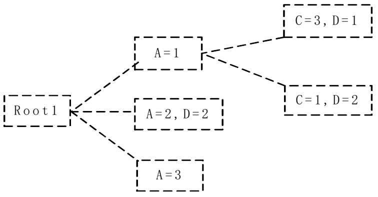


Fig. 2 First rule tree

(4) Repeat step (3) until the internal node queue becomes NULL. The first rule tree is completely built and shown in Figure 3.

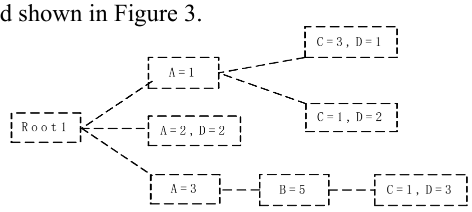


Fig. 3 First rule tree

(5) Delete the corresponding condition attribute (*A*) of

the root node of the first rule tree from Table 2, and do step (1), (2), (3) and (4) until Table 2 is NULL. The rule tree set gotten at this step is shown in Figure 4.

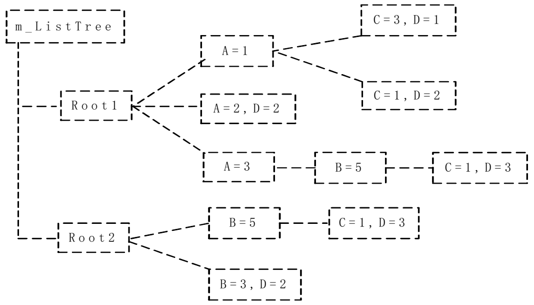


Fig. 4 Rule tree set

(6) Generate the following rules from Fig. 4. Note that the parameter behind " | " is the certainty factor of each rule.

R_1:	$A_1C_3 \rightarrow D_1$ |0.91	R_4:	$A_3B_5C_1 \rightarrow D_3$ |0.80
$R_{2:}$	$A_1C_1 \rightarrow D_2$ |0.91	$R_{5:}$	$B_5C_1 \rightarrow D_3$ |0.80
$R_{3:}$	$A_2 \rightarrow D_2$|1	$R_{6:}$	$B_3 \rightarrow D_2$ |1

4. Space and time complexities of MRT and DDRG algorithm

At first, suppose the number of record of a decision table is n, and the number of condition attribute is m. In a rule tree, the maximal number of leaf node is n, the maximal depth of the path from the root node to a leaf node is m. Thus, the maximal number of node is mn.

4.1 Time complexity of MRT algorithm

Suppose the maximal number of condition class is b, therefore, the time for computing the condition entropy of one condition class with reference to the decision classes is O(bmn), and the time complexity for generating a rule tree is O($bmn*mn$). MRT algorithm generates no more than m rule trees. So, the time complexity of MRT algorithm at worst case is O(bm^3n^2).

4.2 Time complexity of DDRG algorithm

If a decision table has only one core attribute, the time complexity for generating rules with DDRG algorithm is O(mn^2). If the original decision table has m attribute cores, the maximal number of subtables projected from the original decision table is 2^m-1. So, its time complexity at worst case is O(2^m mn^2).

4.3 Space complexity of MRT algorithm

Suppose the maximal number of record in a condition class is q, therefore, the memory for storing the internal node is O(qm), and the memory for a rule tree is O($qm*mn$). MRT algorithm generates no more than m rule trees. So, the space complexity of MRT algorithm at worst case is O(qm^3n).

4.4 Space complexity of DDRG algorithm

If the original decision table has m core attributes, the maximal number of subtables projected from the original decision table is 2^m-1, and the memory for storing each subtable is O(mn). So, the space complexity of DDRG algorithm at worst case is O(2^m mn^2).

Thus, we can find that the space and time complexities of MRT algorithm are better than those of DDRG algorithm.

5. Simulation experiments

Our simulation experiments are done on a PC with 2.4GHZ CPU, 512MB memory. DDRG algorithm in RIDAS system is used for the comparison with MRT algorithm of this paper [8].

The simulation experiments have been conducted on 9 data sets from the UCI Machine Learning Repository [9]. At first, 50 percent of each data set is used to generate rules with these two algorithms separately. Then, the other 50 percent of each data set is used to test the recognition rate of rules by these two algorithms. In addition, the objects' combination based simple computation of attribute core algorithm is used to calculate attribute cores of each data set [10].

Simulation experiment results are shown in Table 4. The time cost for generating rules and recognition rate of these rules are compared in it.

Simulation experiment results show that when the number of records and core attributes of an inconsistent decision table are small, such as Experiments 1 and 2, the performance of MRT and DDRG algorithms are almost the same. In Experiments 3 to 9, we can find that MRT algorithm is more effective and valid than DDRG algorithm. Especially, in Experiments 6 to 9, DDRG algorithm fails to generate rules due to the lack of memory when the number of records and core attributes are high. It just proves that the cost for generating rules of DDRG algorithm is greatly affected by the number of records and core attributes of an inconsistent decision table.

6. Conclusion

MRT algorithm can generate multiple rule trees from inconsistent decision tables. On one hand, the certainty factor of each rule is higher than or equal to a predefined threshold. On the other hand, in the rule tree generation process of MRT algorithm, for creating an internal node of rule tree, only a part of an inconsistent decision table needs to be kept in memory and computed. It reduces the depth and width of rule tree with the predefined threshold. Therefore, MRT algorithm can quickly generates rules from an inconsistent decision table. Compared with DDRG algorithm, MRT algorithm needs less memory and time for generating rules. By our simulation experiments, MRT algorithm is proved to be more effective and valid than DDRG algorithm. As future research task, we will study various measures for selecting condition attribute to improve this algorithm, and further use MRT algorithm to solve the problem of generating knowledge from huge databases containing inconsistent information for web based intelligent decision support systems.

Table 4 Simulation experiment results

	Data set	Number of Record	Number of Attribute	Number of Core attribute	MRT algorithm		DDRG algorithm	
					Time (S)	Recognition Rate (%)	Time (S)	Recognitio n Rate (%)
1	HAYES_BOTH	102	5	4	0.016	98.07	0.625	98.07
2	IRIS	150	5	3	0.016	98.68	<0.001	98.68
3	LIVER_DISDORE	1260	7	5	0.422	99.68	1431.812	100
4	AUSTRALIAN CREDIT APPROVA	345	15	5	0.375	100	57.437	100
5	WINE	148	14	3	0.250	100	0.359	100
6	POST_OPERATIVE	90	9	8	0.051	100	*	*
7	ZOO	71	16	8	0.045	100	*	*
8	BACT_T	6000	155	20	1009.813	44.13	*	*
9	LETTER_RECOGNI TION	20000	17	15	84.469	92.62	*	*

Note: "*" means that DDRG algorithm cannot generate rules from data set due to the lack of memory.

Acknowledgements

This paper is supported by National Natural Science Foundation of P.R. China (No.60373111), PD Program of P.R. China, Application Science Foundation of Chongqing, and Science & Technology Research Program of the Municipal Education Committee of Chongqing of China.

References

[1] Pawlak Z. Rough set. International Journal of Computer and Information Sciences, 1982,11(5):341-356.

[2] Mollestad T. Skowron A. A rough set framework for data mining of propositional default rules. In: Ras Z. W. Michalewicz M. eds. Foundations of Intelligent Systems·9th International Symposium, ISMIS'96, Berlin: Springer-Verlag, 1996. 448-457.

[3] G.Y. Wang, Y. Wu, F. Liu, Generating Rules and Reasoning under Inconsistencies, 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation, Japan, 2536-2541.

[4] Wang GY, He X. A Self-learning Model under Uncertain Condition. Journal of Software, 2003, 14(6):1096-1102.

[5] Quinlan JR. Induction of decision trees. Machine Learning, 1986, 1(1), 81-106.

[6] Shi ZZ. Knowledge Discovery. Tsinghua University Press, 2003 (in Chinese).

[7] Wang GY. Rough set theory and knowledge acquisition. Xi'an: Xi'an Jiaotong University Press, 2001 (in Chinese).

[8] Wang GY, Zheng Z, Zhang Y. RIDAS-A Rough Set Based Intelligent Data Analysis System, Proceedings of the First International Conference on Machine Learning and Cybernetics, pp.646~649, 2002.

[9] http://www.ics.uci.edu/~melean/MLRepository.htm

[10] Zheng Z, Wang GY, Wu Y. Objects' Combination Based Simple Computation of Attribute Core.Proceedings of the 2002 IEEE International Symposium on Intelligent Control, Vancouver, pp.514~519,2002.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 20-27, 2004.

Semi-Structured Complex List Extraction

Anders Arpteg
Department of Technology
University of Kalmar
SE-39192 Kalmar, Sweden
anders.arpteg@hik.se

Abstract

The semi-structured information available in HTML and similar documents provide valuable information that can be used for information extraction applications. This information together with other technical information about how to retrieve pages can be used to automatically extract pieces and various types of lists. The goal is to put as much intelligently as possible in the system so that as little knowledge and work as possible is required by the users, i.e. a user-driven extraction system. The advantage of a user-driven system is that the service provided by the system is available not only for experts, but for also ordinary users and thereby making the service available for a wide audience.

A problem with some lists in documents are that the structure is different for the elements in the lists, and thus it becomes more difficult to take advantage of the semi-structural information. The agent-oriented system described in this paper allows a user without expert skills to train an extraction system to extract singleton, lists, and also complex lists. The complex list type shall be able to handle these complex lists with varied structure.

The experiments conducted show that a user can train the system to extract information pieces from different sites with very little knowledge and small amount of work. However, there are still additional work needed to be able to handle more advanced extraction tasks.

1. Introduction

The information extraction (IE) concept has been given a number of definitions such as the task of semantic matching between user-defined templates and documents written in natural language text, a process that takes unseen text as input and produces fixed-format, unambiguous data as output, and extract relevant text fragments and piece them together into a coherent framework [1, 2, 3]. The preferred definition for this paper is that information extraction is the process to find relevant subsets of textual information for a given task or question and organize them into a clearly defined data structure. This is different from the area of text understanding that attempts to capture the semantics of whole documents.

Examples of applications of IE are shopping agents that locate information about products or services at different retailers and compares them to find the best retailer, event agents that collect information about events that occurs at different locations and times, and news agents that collect news articles from different sources and presents articles relevant for a specific user. Information stored in free natural text or with semi-structured format would be too difficult to handle directly without IE for these applications.

The area of information retrieval (IR) has attracted a lot of attention due to the increased popularity of the World Wide Web. Services such as Google are known by most Internet users and are an essential part of the Web today. The main difference between IR and IE is that IR returns a set of documents rather than a set of answers or phrases related to the query. Thus, the information is not translated to a defined data structure in IR. The advantage of IR is that it is possible to cover a large number of domains, whereas IE typically requires domain-dependent knowledge and is therefore limited in the number of covered domains. These two areas can be combined and complement each other to provide useful services.

The concept structured text as used in this paper refers to textual information stored in a clearly defined data model, for example in a relational database. The advantage of clearly defined structured information is that the information can be automatically analyzed and processed more effectively. Semi-structured text may not have the clear data model representation as structured text, but have more structured information than natural language text, e.g. HTML documents with presentational information combined with the content. These semi-structured documents are often less grammatically correct than natural language texts with choppy sentence fragments [4]. The natural language processing (NLP) methods designed for unstructured natural language text does usually not work as well for semi-structured information. It has also been shown that the extraction task can be performed with very high accuracy using only the semi-structured information, without use of any NLP technique [4].

The term wrapper has been given different definitions depending on the context. In the database community, it represents a software component that converts data from one data model to another. In the Web context, it represents a software component that convert information in a Web page to a structured format, e.g. into a database. The latter corresponds to the preferred definition in this paper. The term wrapper represents an IE software component that takes semi-structured textual input and generates structured text as output. Automatic wrapper generation and wrapper induction are terms that refer to the automatic construction of wrapper, for example using machine-learning techniques.

The performance of semi-structured IE system (wrappers) is often measured differently than with information retrieval systems. The precision and recall measure is typically very high and therefore not a useful measure of the system. Only systems with 100% precision and recall are of interest for sources with significant amount of semi-structured information [5]. These systems are evaluated by their expressiveness and efficiency that measures the coverage of the wrapper (percentage of sources that have 100% precision and recall) and how easily the wrapper can be adapted to new domains.

The type of information extraction system that is described in this paper works in a different way and has different types of applications. Instead of trying to identify pieces in the natural language text using linguistic techniques, the focus is to keep track of pieces in the semi-structured information. A typical application is to keep track of items in some kind of list in documents, i.e. list extraction. For example, a shopping agent needs to keep track of products available from different retailers. Information about these products can be available in some kind of list on the retailer's homepage, e.g. a table with a row for each product. Other examples are calendar of events, and ads in action sites. It is this type of semi-structured extraction tasks that is in focus for this paper.

A problem with information extraction system, as opposed to information retrieval systems, is that it is difficult to handle a large amount of domains. Search engines based on information retrieval techniques can create word-index and rank document for a huge number of domains, but it is very difficult to create a general information extraction system. Therefore, instead of creating a general information extraction system, a system is created that can easily be adapted to new domains. This is called a user-driven information extraction system [6], where non-experts shall be able to train the system to handle new domains. The user shall not need programming skills nor be required to spend a lot of time to train the system. It must be easy to adapt to new domains if a large amount of domains shall be available.

2. The ISSIE System

The ISSIE (Intelligent Semi-Structured Information Extraction) system is a user-driven information extraction system that use semi-structured information to extract content from various types of lists.

There are several possible ways to design a system for the semi-structured information extraction task described above. Since it should be user-driven, it should not require the user to have expert skills in either the knowledge domain or to have expert programming skills. It is therefore not appropriate to require the user to design an ontology for the domain or to ask for complicated regular expression rules. The basic approach taken by the ISSIE system is to monitor details of the surfing behavior of the user and try to repeat the extraction process. The user is asked to start from a given page and then to navigate to the pages that contains the extraction pieces of interest, using a traditional web client. The user shall also give a few examples of which pieces that he/she is interested in. When this "training phase" is complete, the system shall go into the "examination phase" where it tries to repeat the navigation and locate the given pieces by itself. If the examination is successful, the system can go into the third phase, the "extraction phase". In this final phase, the system can automatically extract data from the information sources.

The examples that are provided by the user can currently be of three different types: singleton, list, or complex list. The singleton examples means that a single node in the parse tree shall be extracted (see section 2.1.1 for more information about the parse tree transformation). It could for example be a title in a page, or some similar piece of information that do not have any siblings to extract. The list type shall be used when a simple list in a page shall be extracted. For example, the list could be all cells in a column of a table of products at a retailer's homepage. The user only needs to provide two examples of cells in the list, and the system will find the remaining siblings by itself. See section 2.1.2 for additional information about the sibling algorithm.

Some lists have more complicated structure, where the siblings do not share the same path in the parse tree. It is this parse tree path that is used to find the siblings, so if this path differs between the siblings, the simple list will not work. An example of such a complex list is the list of news in Google (see experiment in section 4.2). To handle such lists, the user has to provide more than two examples.

At least one example for each sibling that have different parse tree path.

The navigation performed by the user is currently monitored by the system using a proxy server. All requests and responses sent between the web client and web server are seen by the proxy and information about them is stored in a database. If the system is able to repeat the extraction process and identify the wanted extraction pieces, then the training is complete and the system is ready to extract the pieces by itself. Remember that the type of extraction tasks that is in focus here is that pieces, e.g. information about products in a retailer's homepage, shall be identified and extracted. As that information changes over time, the extraction system shall be able to extract the new or changed pieces.

The advantage of using a proxy server is that technical details of the retrieval of web pages are captured by the system. The user can use web clients as usual and the system can still retrieve important technical information. In this way, it is possible for the system to handle web sites that depend on features such as cookies, form posts, and browser dependencies. The goal is to add as much intelligence as possible to the system, to handle technical details automatically, and require as little work and expertise as possible from the user. An alternative approach is to let the system try to navigate by itself without the information received from the proxy server. This has been attempted using reinforcement learning techniques, but it is difficult to be able to extract from advanced sites using this technique [7].

The architecture of the system is shown in figure 1. The two main parts of the system are the agent-system that handles the analysis and automated extraction tasks, and the user interface that allows the user to manage and train the system to extract information. The rest of this section will give a brief overview of how these parts work and how the extraction process works.

2.1. The Agent Sub-System

The part of the system that is responsible for handling the automated extraction process is developed using the JADE platform [8]. The motivation for using an agent-oriented approach for the design and execution of the system, is mainly for software engineering reasons. The agent-oriented way to decompose, abstract, and organize relationships can be more intuitive and efficient [9]. The system consists of Surfer agents that are able to download and handle web pages on the Internet, Analyzer agents that analyze documents to find the relevant pieces of information, and a Butler agent that communicates with the user and other systems.

The communication between the agents is made using an ontology developed with Protégé-2000 [10]. The ontology designed with Protégé can be automatically used in JADE agent communication by using the Bean-generator plug-in for Protégé [11]. Also, the same ontology can be used for reasoning with the Jess logical engine in the agents. The ontology can be imported into Jess using the JessTab plug-in [12]. The integration of Protégé, JADE, and Jess provides an efficient way to communicate and reason with a high level of abstraction.

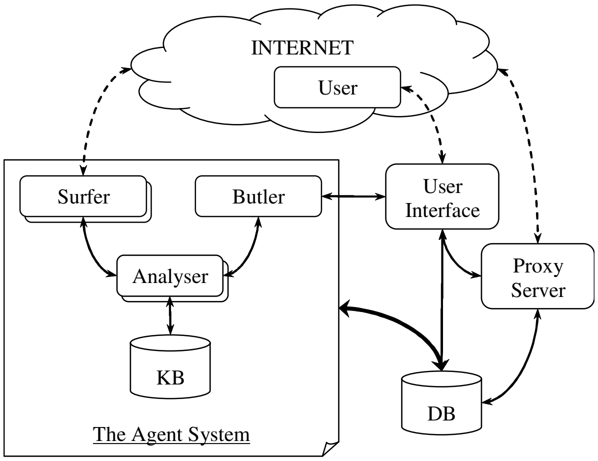


Figure 1. ISSIE Architecture

When a user wants to train the system to handle a new domain, it starts by adding a new task to the system using the user interface. When the training phase is complete and information about the navigation performed by the user and about the wanted extraction pieces is stored in the database, the agent system starts to work. The Butler agent is informed that the training phase is complete and it will send an examination request to an Analyzer agent for that training session.

The Analyzer agent will then start to examine the data from the training session. It will ask the Surfer agent to parse the requests and responses sent during the training phase. A semi-structured model of the pages are created by the Surfer, and they are they further analyzed by the Analyzer. The Analyzer works at a higher abstraction level than the Surfer. It never works with HTML or HTTP techniques, it only works with the the abstract model created by the Surfer. The advantage of this approach is that documents of other types than HTML can be analyzed by the Analyzer agent, as long as there are semi-structural information in the document.

The Analyzer agent uses the Jess logical engine and a knowledge base to handle the examination of the extraction process. The knowledge base consists of a set of rules and facts, which is used to decide which actions that the Analyzer shall take. See section 2.1.3 for more information about how this process works.

During the examination phase, when the agents shall repeat the extraction process, a set of web pages will need to be downloaded. When the Analyzer decides that a web page needs to be downloaded, it sends a download request to a Surfer agent. The Surfer agents have the abilities to communicate with web servers on the Internet and the necessary technical knowledge to create HTTP request and parse HTTP responses. It can also transform the HTML documents into a parse tree representation, i.e. the model that is later used by the analyzer.

2.1.1. The Semi-Structured Document Model

As stated earlier, the main type of information that is used by the ISSIE system in the extraction process is the semi-structured information. This is different from other types of information such as linguistic information, semantic information, and basic pattern matching. These other types of information are commonly used in other information extraction systems, e.g. named entity recognition, part of speech tagging, co-reference resolution, and use of semantical resources such as WordNet [13]. Linguistic and semantic information are currently not used by the system, but the addition of those types of techniques would improve the capabilities of the system. However, the use of linguistic information is not as appropriate for semi-structured text as for unstructured text. The text is often less grammatically correct and contains mostly choppy sentence fragments [4]. If semi-structured information exists in a document, that information can sometimes be sufficient by itself to complete an extraction task [4].

To be able to take advantage of the semi-structured information, the documents need to be transformed from the string representation to a tree representation. The system constructs a parse tree for each document where each node in the tree represents a block element[1] in the document. Also, links from one page to another page are considered to be nodes in this model. The edges between the nodes in the tree represent the parent-child relationships between the elements in the document.

The motivation for using only block-level elements is that the may represent an actual separation of text pieces, whereas in-line elements typically only represent changes in presentation. Of course, this may not always be the case and it may sometimes be preferable to also use in-line element to separated text pieces. Information about in-line element and linguistic information could be useful, but are currently not used by the system.

2.1.2. The Sibling Algorithm

The semi-structured information is used both for navigating through the document, locating the relevant, and to find all the siblings in a list. When a user adds a (simple) list during the training phase, he/she is supposed to provide the content for two pieces in the list. Using only the information about the content and the semi-structured information in the document, the system shall be able to locate all remaining siblings in the list.

The algorithm basically searches for the smallest possible paths to the siblings and store that path for each given sibling. These paths are then used to find all remaining siblings in the lists. The basic outline of the algorithm looks as follows for each given example:

1. Search the parse tree systematically for the first piece in the example and store that node in e.

2. Start from e, store parent node in p, and initialize up path $path_{up} \leftarrow \{p\}$.

3. Initialize list of siblings $siblings$ for all siblings (if any) for current example.

4. For each child c of p:

    (a) Add c to down path: $path_{down} \leftarrow path_{down} \cup \{c\}$.

    (b) Check content in c for sibling example match from $siblings$.

    (c) If match, store $path_{up}$ and $path_{down}$ as sibling paths for matching sibling and remove that sibling from $siblings$.

    (d) If $siblings$ is empty, terminate sibling search.

    (e) If no match, continue recursively to get children of c and go to step 4.a.

    (f) If $siblings$ is empty, terminate sibling search.

    (g) Remove c from down path: $path_{down} \leftarrow path_{down} \cap \neg\{c\}$.

    (h) Continue with next c.

2.1.3. The Examination Process

When the user has completed the training phase and thereby demonstrated to the system how to navigate and what to extract, the system shall examine the training data and try to repeat the extraction. The examination starts when the Butler agent sends an examination request to the Analyzer agent for the training session that was just completed by the user. The Analyzer now starts the examination. Here are the basic steps of the examination process:

[1] A block element is different from an in-line element in that that typically begin on a new line and represent some block of text, e.g. DIV and P elements.

1. Make sure the requests and responses have been parsed and that a parse tree has been built for each document. This is performed by the Surfer agent.

2. When the parse tree model has been built, the Surfer agent tries to locate the siblings for all lists that were given by the user. Each node that contains a wanted extraction piece is marked as an extraction node, including all siblings in a list, according to the sibling algorithm given above.

3. The Analyzer tries to repeat the extraction process by itself, using heuristic rules. It has three different plans to succeed with the extraction, and it starts with the simplest plan.

4. The first plan consists simply of requesting only the pages containing the extraction points. In some cases, this will work and it is therefore not necessary to request any other pages that exist in the training session. The Analyzer agent sends download requests to the Surfer agent.

5. If the first plan fails, it continues with the second plan that consists of requesting all pages containing form submittals. The motivation for such a plan is that it is common to need to authenticate in a web site before being able to obtain the wanted pieces. Also, if a search or similar type of filtering has been performed, it usually involves a form submittal.

6. If the second plan fails too, the Analyzer continues with the third plan. The third plan involves requesting all pages in chronological order from the training session, excluding duplicates.

7. If a plan succeeds and the wanted pieces are located, the Analyzer stores that plan in the database and the examination is completed. If no plan succeeds, the system responds to the user that it was unable to repeat the extraction.

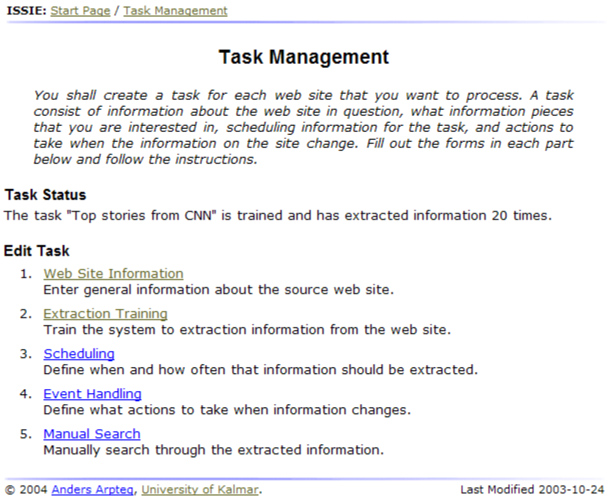
2.2. The User Interface

The user interface to the ISSIE system allows the user to manage the extraction tasks. The user can train the system to handle new tasks and modify existing tasks. The interface is a traditional web interface built using .Net. It is necessary for the interface to be simple enough so that users are not required to be computer experts.

Due to the space limitations of the paper, it is not possible to include any detailed information about the interface. However, since it is a traditional web interface, there is not much relevance to give any detailed information. A small screen shot is given for the page that manages a specific task in figure 2. From that page, it is possible for the user to give basic information about the task, to train the system, and modify other information for the task.

Table 1. List of experiments

Top news stories	Extract headlines of the current top news stories and the current top story from http://www.cnn.com/
Current scientific	Extract current scientific news stories headlines from http://news.google.com/
Video drivers	Extract video driver updates for a specific computer model from http://www.dell.com/



ISSIE: Start Page / Task Management

Task Management

You shall create a task for each web site that you want to process. A task consist of information about the web site in question, what information pieces that you are interested in, scheduling information for the task, and actions to take when the information on the site change. Fill out the forms in each part below and follow the instructions.

Task Status

The task "Top stories from CNN" is trained and has extracted information 20 times.

Edit Task

1. Web Site Information
   Enter general information about the source web site.
2. Extraction Training
   Train the system to extraction information from the web site.
3. Scheduling
   Define when and how often that information should be extracted.
4. Event Handling
   Define what actions to take when information changes.
5. Manual Search
   Manually search through the extracted information.

© 2004 Anders Arpteg, University of Kalmar. Last Modified 2003-10-24

Figure 2. User Interface: Task Management

3. Experiments

To evaluate the semi-structured list extraction hypothesis, a set of experiments was conducted using the ISSIE system. These experiments consist of extraction tasks such as extract the available driver updates for a particular computer model from a manufacturer's homepage. The list of experiments conducted is shown in table 1. The motivation for choosing these extraction tasks are that they are that the tasks represent interesting and relevant tasks, regardless of how advanced the web site and the extraction is.

The basic steps in each experiment are as follows:

1. The user creates a new task in the ISSIE user interface.
2. Basic information such as name of task and start URL are given by the user.
3. The user starts the training phase by making sure that the correct proxy settings are configured in the browser and then going to the start URL.
4. The user shall now navigate from the start URL to the pages containing wanted information pieces. It is possible for the user to for example provide username and password and fill out forms to obtain the information pieces.
5. When the user arrives at a page that contains wanted information pieces, the user shall copy text from those pieces and paste them into the "training" page in the ISSIE user interface. If a list of pieces shall be extracted, only two random items in the list needs to be copied. It is possible to specify if the pieces is part of a list or if it is a singleton in the ISSIE user interface.
6. When all information pieces have been found and examples copied to the user interface, the user stops the training by clicking a "training complete" button in the ISSIE interface.
7. The agents in the ISSIE system will now analyze the training session provided by the user and try to repeat the process. Information about the status is shown to the user.
8. If the agents were able to repeat the process and find the pieces provided by the user by themselves, the training is successful and the automated extraction can start. Otherwise, the user may need to provide additional information and re-train the system.

4. Results

The system was able to find the given extraction pieces and most of the times find the additional items in lists. There were some problems to correctly find all items in a list for some tasks, since the structure for list item was not always identical.

Here is some more detailed information about each experiment:

4.1. Top News Stories

The task to extract news stories from the CNN site is a common test for information extract systems. The task was very simple, take the top stories directly from the start page, which are located in a small box in the top right part of the start page. The system should not only extract the top stories headline, but also extract the current main top story that is located in a different place in the start page.

The two pieces "CIA: Tape likely is bin Laden" and "Blair sees wider role for U.N. in Iraq" where given for the top stories list and the piece "U.S. delays troop return from Iraq" was given for the main top story singleton.

There was no problem for the system to repeat the extraction and locate the additional items in the top stories list.

4.2. Current Scientific News Stories

The task consists of finding and locating the current scientific news from the Google news site. The start page was http://news.google.com/ and the scientific news can be reached with one click from the start page. This list is an example of a complex list. Some of the articles in the list of scientific news has a picture before the headline in a separate column of and other articles have no picture and therefore are the headline placed in a different column.

The pieces "Cassini quietly awaits ride in Saturn's orbit", "New texting speed record set", and "Juniper Serious About SAML" were given as examples of headlines. The two last pieces are siblings to the first piece and covers the two different types of articles in the list, i.e. with and without picture.

The system was able to determine the paths to both siblings and able to extract all siblings in the list.

4.3. Video Drivers

The purpose for this task is to be able to extract the list of video driver updates for a specific computer model from http://www.dell.com/. This is a rather advanced task since it involves a large amount of pages to navigate through, and it also requires form posts and cookie management to work properly. It is also spread across several web servers.

The task consist of starting at the start page, navigating to the support pages, submitting the service tag number in a form to retrieve updates relevant for a specific computer, and navigating to the video drivers page. There are in total nine clicks, 34 pages[2], and one form post to reach the video drivers page.

The two pieces "Video: ATI Mobility Radeon 9000, Driver, Windows XP, Multi Language, Inspiron 8500, v.7.80.4-021206a-6945c, A00" and "Video: nVidia GeForce4 4200 Go, Driver, Windows 2000, Windows XP, Multi Language, Inspiron 8500, Latitude D800, v.6.13.10.4258, A03" were given as examples of the video

[2] The number of pages are larger than the number of clicks since there are frames, redirects, and similar requests

drivers list. The system was able to navigate and locate the given pieces and locate the additional 11 drivers in the list.

5. Related Work

There exist other systems where the semi-structured information is used, for example [14, 15, 4, 16, 5]. The main idea in the wrapper toolkit by Ashish and Knoblock is to exploit the semi-structured information to facilitate the extraction task. The construction of a wrapper starts with identifying the relevant structure of a page, building a parser based on given structure, and finally adding communication capabilities to the wrapper to be able to find different sources of information and give the result to a mediator. A set of heuristic rules are used to identify sections and subsections in the web pages. These rules are basically regular expressions that exploit HTML knowledge to find the structure. In addition, heuristics such as font size are used to determine the hierarchical level of the structure. There is no training in the system, although the user is able to correct erroneous guesses through a graphical user interface. The heuristics basically employs pattern matching rules to identify sections and subsections, with assistance of HTML knowledge. The actual structural relationships present in the source pages are not used for the identified output structure.

The Rapper system uses the same techniques as in the Ashish system [14] and adds algorithms that employ linguistic knowledge. These extensions increase the cost of adapting the system to new domains, although they increase the accuracy for implemented domains. As stated in the paper, the construction of wrappers is a non-trivial task even with these tools. A significant amount of knowledge is still required to construct a wrapper.

The WYSIWYG Wrapper Factory [15] provides a very nice graphical user interface that allows the user to add extraction rules that takes advantage of the semi-structured information. However, there is no training in the system and the user still needs to be familiar to the advanced rule language used in the system.

6. Conclusion

As previously stated, the hypothesis in the paper is that a user-driven approach to semi-structured complex list extraction is possible and that it can in the long run lead to a large amount of extractable domains. There are several possible applications for this type of information extraction. For example, the user may simply want to receive notifications when some information pieces are changed, added, or removed from a site. A more advanced long term application would be to facilitate the goal of the Semantic Web. If the information on the Web should be machine understandable and not only machine readable, then the documents need to be transformed from the presentation format of HTML to more semantically encoded formats such as RDF and OWL [17, 18]. This type of information extraction services could assist in this transformation and make the information automatically available for machines as well as for humans.

A problem with previous implementations of the ISSIE system was that it was not able to handle complex lists that had different structures. By allowing the user to provide additional examples in the lists, and adding support in the system to handle multiple paths to the siblings, these more complicated lists can also be extracted.

Another problem that is more difficult to solve is how to manage multi-page tables. It is common to split a list into several pages, to make the list more manageable. The user of the ISSIE system still expects to extract all items in the list, not only the items in the first page. A future implementation of the system could possibly be improved to have multi-page lists, in addition to complex lists, normal lists, and singletons. The user would then need to give information by example of how to navigate to other pages in the list, in addition to how to extract the items in the list.

In general, the experiments were promising and the approach of using a proxy to monitor technical details that the user is unaware of is of great help. If compared to the reinforcement learning approach that should automatically navigate given some wanted text pieces, this approach is able to handle very advanced sites easily.

The current system allows users without domain expertise knowledge and without programming knowledge to quickly create an extraction task. The output from the system provides an XML document that can be managed by other computer systems. A possible application can be to encode additional semantic knowledge into the XML document and make it public. In that way, a large amount of domain can be made machine understandable and make the information available on the Internet more valuable.

Acknowledgment

This research was funded by the Swedish Knowledge Foundation and University of Kalmar. I also wish to thank Erik Sandewall for his encouragement and guidance throughout this project.

References

[1] N. Guarino, *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, ser. Lecture Notes in Artificial Intelligence. Frascati: Springer, July 1997, vol. 1299, ch. 8. Semantic Matching: Formal On-

tological Distinctions for Information Organization, Extraction, and Integration, pp. 139–170.

[2] H. Cunningham, “Information extraction: A user guide (revised version),” Department of Computer Science, University of Sheffield, Tech. Rep. CS-99-07, May 1999.

[3] J. Cowie and W. Lehnert, “Information Extraction,” *Communications of the ACM*, vol. 39, no. 1, pp. 80–91, 1996.

[4] S. Soderland, “Learning to extract text-based information from the world wide web,” in *Proceedings of the 3rd International Conference on Knwoledge Discovery and Data Mining (KDD-97)*, 1997.

[5] N. Kushmerick, “Wrapper induction: Efficiency and expressiveness,” *Artificial Intelligence*, vol. 118, pp. 15–68, 2000.

[6] A. Arpteg, “User-driven semi-structured information extraction,” in *4th International Conference on Intelligent Systems Design and Applications*, August 2004.

[7] ——, “Adaptive semi-structured information extraction,” Licentiate dissertation, Linköping university, Linköping, 2003.

[8] F. Bellifemine, A. Poggi, and G. Rimassa, “JADE — A FIPA-compliant agent framework,” in *Proceedings of the 4th International Conference on the Practical Applications of Agents and Multi-Agent Systems (PAAM-99)*, 1999, pp. 97–108.

[9] N. R. Jennings and M. Wooldridge, *Handbook of Agent Technology*. AAAI/MIT Press, 2000, ch. Agent-oriented Software Engineering.

[10] W. Grosso, H. Eriksson, R. Fergerson, J. Gennari, S. Tu, and M. Musen, “Knowledge modeling at the millennium – the design and evolution of protege,” in *Proceedings of the 12 th International Workshop on Knowledge Acquisition, Modeling and Mangement (KAW’99)*, Banff, Canada, October 2000.

[11] C. van Aart, “Java ontology bean generator for jade 3.0,” 2003, http://www.swi.psy.uva.nl/usr/aart/beangenerator/ (2004-04-19).

[12] H. Eriksson, “Using jesstab to integrate protégé and jess,” in *IEEE Intelligent Systems*, vol. 18, no. 2, 2003, pp. 43–50.

[13] C. Fellbaum, Ed., *WordNet: An Electronic Lexical Database*. Cambridge: MIT Press, 1998.

[14] N. Ashish and C. Knoblock, “Wrapper generation for semi-structured internet sources,” in *Proc. Workshop on Management of Semistructured Data*, Tucson, 1997.

[15] A. Sahuguet and F. Azavant, “Wysiwyg web wrapper factory,” 1999.

[16] D. Mattrox, L. J. Sligman, and K. Smith, “Rapper: A wrapper generator with linguistic knowledge,” in *Workshop on Web Information and Data Management*, 1999, pp. 6–11.

[17] World Wide Web Consortium, “RDF primer,” 1999, http://www.w3.org/TR/REC-rdf-syntax (2004-04-19).

[18] ——, “OWL Web Ontology Language Guide,” 2004, http://www.w3.org/TR/owl-guide/ (2004-04-19).

Web-supported Matching and Classification of Business Opportunities

Jing Bai [1], François Paradis [1,2], Jian-Yun Nie [1]

1. *DIRO*
Université de Montréal
C.P. 6128, succursale Centre-ville
Montréal, Québec, H3C 3J7, Canada

2. *Nstein Technologies*
75, Queen Street, Suite 4400
Montréal, Québec, H3C 2N6, Canada

{baijing, paradifr, nie}@iro.umontreal.ca

Abstract

More and more business opportunities are published on the Web; however, it is difficult to collect and process them automatically. This paper describes a tool and techniques to help users discovering relevant business opportunities, in particular, calls for tenders. The tool includes spidering, information extraction, classification, and a search interface. Our focus in this paper is on classification, which aims to organize calls for tenders into classes, so as to facilitate user's browsing. We describe a new approach to classification of business opportunities on the Web using language modeling (LM) approach. This utilization is strongly inspired by the recent success of LM in IR experiments. However, few attempts have been made to use LM for text classification so far. Our goal is to investigate whether LM can bring improvement to text classification. Our experiments are conducted on two corpora: Reuters containing newswire articles and FedBizOpps (FBO) containing calls for tenders (CFTs) published on the Web. The experimental results show that LM-based classification can significantly improve the classification performance on both test corpora, compared with the traditional Naïve Bayes (NB) classifier. In particular, it seems to have stronger impact on FBO than on Reuters. This result shows that LM can greatly improve classification on the Web.

1. Introduction

Finding and selecting business opportunities is a crucial activity for businesses, yet they often lack the resources or expertise to commit to this problem. To ease this task, many electronic tendering sites are now available. They usually follow either a centralizing approach, where information is received directly from the contracting authorities (for example, in the case of TED[1]), or an aggregation approach, where documents are collected from other sites (for example, SourceCan[2]). Although the centralizing approach allows to control the contents and richness of the information, it is difficult to apply to some domains where there is no recognized authority, and is often limited to one geographic area. Furthermore, additional information which might exist on the Web is ignored. On the other hand, with the aggregation approach it is difficult to extract and categorize relevant information, since documents do not follow a common form or model, and their contents can vary widely.

Business-related documents, in particular Calls for Tenders (CFTs), are typically classified according to an industry standard, for example, NAICS (North American Industry Classification System) or CPV (Common Procurement Vocabulary, for the European Union). Some CFTs are manually classified with these codes, whereas some others are not. A classification algorithm is a natural addition to organize and search and CFT into a browsable directory. It can also provide multi-code classification for conversion between standards or different versions of standards. However, automated classification is difficult on CFTs, especially when they are taken from the Web, where their contents can vary a lot and there can be a large number of unseen terms.

In this paper, we propose to improve the classification of CFTs using a language modeling approach. A language model (LM) refers to a set of probability estimates on a training corpus. It also uses smoothing to deal with the obtained-zero probability problem of unseen words in the corpus. In a classification context, LM is used to estimate the probability of a word within a class. We propose to use these estimates within the Naïve Bayes (NB) method.

The paper will be organized as follows. In Section 2, we will briefly describe the MBOI project. In Section 3, we describe our approach to text classification using language models. Section 4 presents the experimental design and results on the

[1] `http://ted.publications.eu.int/`

[2] `https://www.sourcecan.com/`

Reuters-21578 and FBO data sets respectively. Finally, Section 5 gives some conclusions.

2. The MBOI Project

The MBOI project (Matching Business Opportunities on the Internet) deals with the discovery of business opportunities on the Internet. In the first phase of the project we have implemented a tool to aid a user in this process. It includes spidering, information extraction, classification, and a search interface.

The information relevant to business opportunities comes from various types of documents: press releases, solicitation notices, awards, quarterly reports, etc. We are not so much interested in modeling these documents, however, but rather in extracting and organizing information that will help finding CFTs: not only information within the CFT, but also related to contracting authorities, prior clients, etc. This information is crucial for business decisions. For this reason, we will refer to the documents as evidence, from which the information can be inferred.

Figure 1 shows the information inference process. At the core of the model is the CFT synthesis, which combines evidence from various sites. For example, if two sites contain a French and English version of the same CFT, the synthesis will include relevant attributes (title and description) in both languages. Other characteristics such as submission and execution dates, classification codes, submission procedure, etc. will also be inferred from the call for tenders notices. Amendments can replace or add to some or all of the elements of the synthesis.

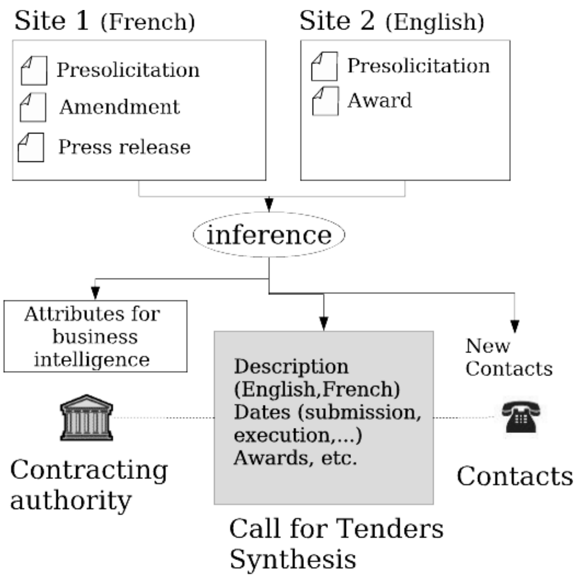


Figure 1: Information inference

Other information can add to the existing knowledge about contracting authorities and their contacts. These could later be used for business intelligence.

Since information can be extracted from several documents, there must be a strategy for the combination of evidence. Even for official documents such as call for tenders, there can be more than one version, published on the same site, or on several sites. Pairing these documents can be difficult if editors create their own solicitation numbers, sometimes without explicit reference to the contracting authority. We thus define a confidence measure on the inferred information. This confidence measures the validity of inference rules. It can also reflect the confidence of the source of the information: for example a contracting authority publishing its own documents can be deemed more trustworthy then an aggregator site.

Figure 2 shows a simplified example of a presolicitation notice and its amendment, regarding a contract for the office supplies of the Saskatchewan government. Both documents were fetched from the Merx site. From these documents, the system infers a synthesis with extracted information such as: publication and closing dates, title (both French and English), contact, etc. It also classifies the CFT: in this case, to NAICS code "418210" ("Stationery and Office Supplies Wholesaler-Distributors"). The synthesis is stored in an XML format inspired by xCBL (Common Business Language) and UBL (Universal Business Language) [5].

Presolicitation (on Merx):

Reference Number: CFAB4

Source ID: PV.MN.SA.213412

Published: 2003/10/08

Closing: 2003/10/28 02:00PM

Organisation Name: Saskatchewan Government

Title (English): Office Supplies

Title (French): Fournitures de Bureau

Description: The Government of Saskatchewan invites tenders to provide office supplies to its offices in Regina. The supplier is expected to start delivery on December 5, 2003, and enter an agreement of at least 2 years.

Contact: Bernie Juneau, (306) 321-1542

Amendment (on Merx):

Reference Number: CFAB4

Description: The start delivery date has been revised to January 5, 2004.

Figure 2: A call for tenders

Figure 3 shows the MBOI system architecture. There are two main processes: indexing, i.e., creating an index with the information inferred from the Web documents, and querying/browsing, which is the search interface for the user.

The first step of indexing is to collect documents from Web sites. We use a robot that can connect with a username and password (for sites with restricted access), look for URL patterns, fill out forms, and follow links of a given form. The next step is the inference of information, which includes information extraction and classification. Finally, an index is created and organized by fields of information (i.e., corresponding to elements in the CFT synthesis).

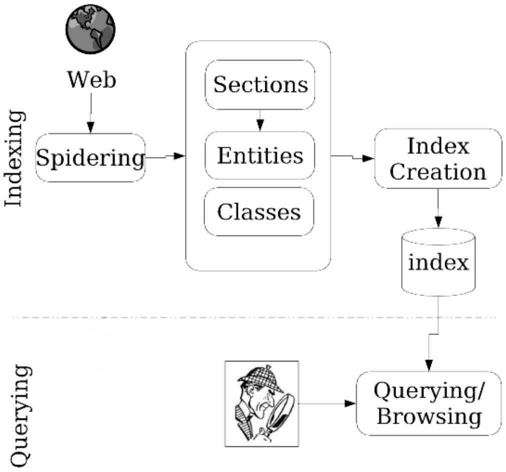
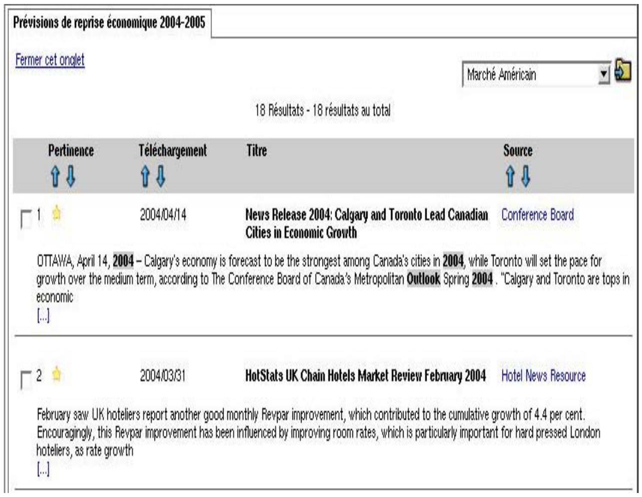


Figure 3. System architecture

The front-end to the system allows the user to search for CFTs by topic, date, class code, etc. or with an all-fields free text query. It also includes functionalities for browsing the class hierarchy, save the results in topic folders, etc. Figure 4 shows an example of results for a query about economic recovery. This is a saved query, i.e., one that has been defined by the user and is executed on a routine basis. This function is useful for a user who checks for a particular type of business opportunities on a daily basis.



Prévisions de reprise économique 2004-2005

Fermer cet onglet

Marché Américain

18 Résultats - 18 résultats au total

Pertinence ⇧⇩	Téléchargement ⇧⇩	Titre	Source ⇧⇩
□ 1 ☆	2004/04/14	**News Release 2004: Calgary and Toronto Lead Canadian Cities in Economic Growth**	Conference Board

OTTAWA, April 14, **2004** – Calgary's economy is forecast to be the strongest among Canada's cities in **2004**, while Toronto will set the pace for growth over the medium term, according to The Conference Board of Canada's Metropolitan **Outlook** Spring **2004** . "Calgary and Toronto are tops in economic
[...]

□ 2 ☆	2004/03/31	**HotStats UK Chain Hotels Market Review February 2004**	Hotel News Resource

February saw UK hoteliers report another good monthly Revpar improvement, which contributed to the cumulative growth of 4.4 per cent. Encouragingly, this Revpar improvement has been influenced by improving room rates, which is particularly important for hard pressed London hoteliers, as rate growth
[...]

Figure 4. Querying in MBOI

The indexing and retrieval processes used in MBOI use the classical IR approaches of vector space model, with some enhancements to deal with structures of CFTs (e.g., section, title, etc.). We will not describe these processes in detail. Instead, we will concentrate on the classification process of CFTs in which we use a new method based on the statistical language modeling approach.

3. Using Languahe Models for Text Classification

Language models have been successfully applied in many application areas such as speech recognition and statistical NLP. Recently, a number of studies have confirmed that language model is also an effective and attractive approach for information retrieval (IR) [6, 11]. It not only provides an elegant theoretical framework to IR, but also results in effectiveness comparable to the best state-of-the-art systems. This success has triggered a great interest in IR community, and LM has since been used to other IR-related tasks, such as topic detection and tracking [7]. However, until now, few attempts have been made to use language models for text classification although there is a strong relationship between IR and classification.

Text classification aims to assign text documents into one or more predefined classes based on their contents. Many machine learning techniques have been applied to automatic text classification, such as Naïve Bayes (NB), K-Nearest Neighbor and Support Vector Machines (SVM).

Indeed, classification shares several common processings with IR. It is then possible that LM can also bring significant improvement to classification. Our goal in using language models to classification is to investigate whether language models can also improve the performance of classification. In particular, we will first integrate NB with language models, because we can observe a strong similarity between them.

3.1 Naïve Bayes Classifier

Let us first describe the principle of Naïve Bayes classifier.

Given a document d and a set of predefined classes {… c_i, …}, a Naïve Bayes classifier first computes the posterior probability that the document belongs to each particular class c_i, i.e., $P(c_i \mid d)$, and then assigns the document to the class(es) with the highest probability value(s). The posterior probability is computed by applying the Bayes rule:

$$P(c_i \mid d) = \frac{P(d \mid c_i)P(c_i)}{P(d)} \qquad (1)$$

The denominator $P(d)$ in formula (1) is independent from classes; therefore, it can be ignored for the purpose of class ranking. Thus:

$$P(c_i \mid d) \propto P(d \mid c_i)P(c_i) \qquad (2)$$

In Naïve Bayes, it is further assumed that words are independent given a class, i.e., for a document $d = d_1,\ldots,d_m$:

$$P(d \mid c_i) = \prod_{j=1}^{m} P(d_j \mid c_i)$$

Formula (2) can then be simply expressed as follows:

$$P(c_i \mid d) \propto \prod_{j=1}^{m} P(d_j \mid c_i)P(c_i) \qquad (3)$$

In formula (3), $P(c_i)$ can be estimated by the percentage of the training examples belonging to class c_i:

$$P(c_i) = \frac{N_i}{N}$$

where N_i is the number of training documents in class c_i, and N is the total number of training documents respectively. $P(d_j \mid c_i)$ is usually determined by:

$$P(d_j \mid c_i) = \frac{1 + count(d_j, c_i)}{|V| + |c_i|}$$

where $count\,(d_j, c_i)$ is the number of times that term d_j occurs within the training documents of class c_i, $|V|$ is the total number of terms in vocabulary, and $|c_i|$ is the number of terms in class c_i. This estimation uses the Laplace (or add-one) smoothing to solve the zero-probability problem.

3.2 Language Modeling Approach in IR

Language modeling has been applied successfully in information retrieval [6, 11, 12] and several related applications such as topic detection and tracking [7]. Given a document d and a query q, the basic principle of this approach is to compute the conditional probability $P(d \mid q)$ as follows:

$$P(d \mid q) = \frac{P(q \mid d)P(d)}{P(q)} \propto P(q \mid d)P(d)$$

If we assume $P(d)$ to be a constant, then the ranking of a document d for a query q is determined by $P(q \mid d)$. The calculation of this value is performed as follows: We first construct a statistical language model $P(. \mid d)$ for the document d, called document model. Then $P(q \mid d)$ is estimated as the probability that the query can be generated from the document model. This probability is often calculated by making the assumption that words are independent (in a unigram model) in a similar way to Naïve Bayes. This means that for a query $q = q_1, \ldots, q_n$, we have:

$$P(q \mid d) = \prod_{j=1}^{n} P(w_j \mid d)$$

In previous studies, it turns out that smoothing is a very important process in building a language model [11]. The effectiveness of a language modeling approach is strongly dependent on the way that the document language model is smoothed. The primary goal of smoothing is to assign a non-zero probability to the unseen words and to improve the maximum likelihood estimation. However, in IR applications, smoothing also allows us to consider the global distribution of terms in the whole collection, i.e., the IDF factor used in IR [11].

Several smoothing methods such as Dirichlet, Absolute discount, etc., have been applied in language models. In Zhai and Lafferty [11], it has been found that the retrieval effectiveness is generally sensitive to the smoothing parameters. In our experiments on classification, we also observed similar effects.

3.3 Using Language Modeling Approach for Text Classification

If we compare Naïve Bayes with the general language modeling approach in IR, we can observe a remarkable similarity: the general probabilistic framework is the same, and both use smoothing to solve the zero-probability problem. The difference between them lies in the objects which a language model is constructed for and applied to. In IR, one builds a LM for a document and applies it to a query, whereas in NB classifier, one builds a LM for a class and applies it to a document. However, we also observe that in the implementation of NB, one usually is limited to the Laplace smoothing. Few attempts have been made in using more sophisticated smoothing methods.

As the experiments in IR showed, the effectiveness of language modeling strongly depends on the smoothing methods, and several smoothing methods have proven to be effective. Then a natural question is whether it is also beneficial in classification to use other sophisticated smoothing methods instead of the Laplace smoothing. In this paper, we will focus on this problem. As we will see later in our experiments, it will be clear that such a replacement can bring improvements to Naïve Bayes classifier. Another question we will examine is whether a LM classification approach will have similar impact on different types of documents.

3.3.1. Principle. The basic principle of our approach to text classification using language models is straightforward.

As in Naïve Bayes, the score of a class c_i for a given document d is estimated by formula (3). However, the estimation of $P(d_j \mid c_i)$ is different: It will be estimated from the language modeling perspective. First, we construct a language model for each class with several smoothing methods. Then $P(d_j \mid c_i)$ is the probability that the term d_j can be generated from this model. As smoothing turns out to be crucial in IR experiments, it is also necessary to carefully select the smoothing methods. In the next section, we will describe those that have been used in several IR experiments.

3.3.2. Smoothing Methods for Estimation. A number of smoothing methods have been developed in statistical natural language processing to estimate the probability of a word or an n-gram. As we mentioned earlier, the primary goal is to attribute a non-zero probability to the words or n-grams that are not seen in a set of training documents. Two basic ideas have been used in smoothing: 1) using a lower-order model to supplement a higher-order model; 2) modifying the frequency of word occurrences.

In IR, both ideas have been used. On the first solution, it is common in IR to utilize the whole collection of documents to construct a background model. This model is considered as a lower-order model to the document model, although both models may be unigram models. This solution has been useful for relatively short documents. Although a class usually contains more than one document, thus longer than a single document, the same problem of imprecise estimation exists, especially for small classes. Therefore, one can use the same approach of smoothing to classification. The second solution is often used in combination with the first one (i.e., one simultaneously use the collection model and change the word counts), as we can see in the smoothing methods described below.

Two general formulations are used in smoothing: backoff and interpolation. Both smoothing methods can be expressed in the following general form [12]:

$$P(w \mid c_i) = \begin{cases} P_s(w \mid c_i) & \text{w is seen in } c_i \\ \alpha_{c_i} P_u(w \mid C) & \text{w is unseen in } c_i \end{cases}$$

That is, for a class c_i, one estimate is made for the words seen in the class, and another estimate is made for the unseen words. In the second case, the estimate for unseen words is based on the entire collection, i.e., the collection model. The effect of incorporating the collection model not only allows us solving the zero-probability problem, but also is a way to produce the same effect as the IDF factor commonly used in IR (as shown in [11]).

In our experiments, we tested the following specific smoothing methods. All of them use the collection model.

- Jelinek-Mercer (JM) smoothing:

$$P_{JM}(w \mid c_i) = (1-\lambda)P_{ml}(w \mid c_i) + \lambda P(w \mid C)$$

which linearly combines the maximum likelihood estimate $P_{ml}(w \mid c_i)$ of the class model with an estimate of the collection model.

- Dirichlet smoothing:

$$P_{Dir}(w \mid c_i) = \frac{c(w, c_i) + \mu P(w \mid C)}{|c_i| + \mu}$$

where $c(w, c_i)$ is the count of word w in c_i, $|c_i|$ is the size of c_i (i.e., the total word count of c_i) and μ is a pseudo-count.

- Absolute discount smoothing:

$$P_{AD}(w \mid c_i) = \frac{\max(c(w, c_i) - \delta, 0) + \delta |c_i|_u \, P(w \mid C)}{|c_i|}$$

in which the count of each word is reduced by a constant $\delta \in$ [0, 1], and the discounted probability mass is redistributed on the unseen words proportionally to their probability in the collection model. In the above equation, $|c_i|_u$ is the number of unique words in c_i.

- Two-Stage (TS) smoothing [12]:

$$P_{TS}(w \mid c_i) = (1-\lambda)\frac{c(w, c_i) + \mu P(w \mid C)}{|c_i| + \mu} + \lambda P(w \mid C)$$

This smoothing method combines Dirichlet smoothing with an interpolation smoothing.

In the previous experiments of IR, it turns out that Dirichlet and Two-stage smoothing methods provided very good effectiveness. In our experiments, we will test whether these smoothing methods, when applied to text classification, bring similar impact.

4. EXPERIMENTAL EVALUATION ON CLASSIFICATION

4.1 Corpora

In order to compare with the previous results, our experiments have been conducted on the benchmark corpus of Reuters-21578, containing Reuter's newswire articles. We chose the ModApte split of Reuters-21578 data set, which is commonly used for text classification research today [9]. There are 135 topic classes, but we used only those 90 for which there exists at least one document in both the training and test set. Then we obtained 7769 training documents and 3019 test documents. The number of training documents per class varies from 2877 to 1. The largest 10 classes contain 75% of the documents, and 33% classes have fewer than 10 training documents.

In our experiments of finding business opportunities on the Web, we created a collection of CFT documents by downloading the daily synopses from the FedBizOpps (FBO) website, which are in the period from September 2000 to October 2003. This resulted in 21945 documents, which were split 70% for training and 30% for testing in our experiments. Notice that all the CFTs published on this site are manually classified using NAICS codes. NAICS codes are organized hierarchically, where every digit of a six-digit code corresponds to a level of the hierarchy. In order to reduce the class space, we only consider the first three digits in our current study. Although class hierarchy is an aspect that makes the classification of CFTs different from the general classification problem with flat classes, we will postpone this problem to a later study. That is, our current study will consider the set of classes at the same level. After removing the classes that do not included at least one document in both training and test set, we obtained 86 classes, 15312 training documents and 6627 test documents. The largest 10 classes contain 72% of the documents, and 30% classes have fewer than 20 training documents. We can see that the FBO collection has quite similar a distribution to the Reuters collection.

4.2 Performance Measure

For the purpose of comparison with previous works, we evaluate the performance of classification in terms of standard recall, precision and F_1 measure. For evaluating average performance across classes, we used macro-averaging and micro-averaging. Macro-averaging scores are the averages of the scores of each class calculated separately. Micro-averaging scores are the scores calculated by mixing together the documents across all the classes. Macro-averaging gives an equal weight to every class regardless how rare or how common a class is. On the other hand, micro-averaging gives an equal weight to every document, thus putting more emphasis on larger classes. In [9], it is claimed that micro-averaging can better reflect the real classification performance than macro-averaging. Therefore, our observations will be made mainly on micro-averaging F_1.

4.3 Naïve Bayes Classifier

To provide the comparable results of classification on Reuters-21578 corpus, we used the multinomial mixture model of Naïve Bayes classifier of the Rainbow package, developed by McCallum [3].

In NB classifier, feature selection is important. The effect of feature selection is to remove meaningless features (words) so that classification can be determined according to meaningful features. Several feature selection methods are commonly used: information gain (IG), chi-square, mutual information, etc. Information gain has shown to produce good results in [9]. The information gain of a word w is calculated as follows:

$$IG(w) = -\sum_{i=1}^{k} P(c_i)\log P(c_i) + \\ P(w)\sum_{i=1}^{k} P(c_i \mid w)\log P(c_i \mid w) + P(\overline{w})\sum_{i=1}^{k} P(c_i \mid \overline{w})\log P(c_i \mid \overline{w})$$

where $\overline{w}$ means the absence of the word w.

One can choose a fixed number of features according to their IG, or set up a threshold on IG to make the selection. The following table shows the classification results by NB without feature selection and with a selection of 2000 features according to IG. The number 2000 is suggested in [9].

Table 2 shows the classification results by NB without feature selection and with a selection of 12,000 features according to IG. The number 12,000 produced the best performance on FBO collection.

NB	miR	miP	miF$_1$	maF$_1$	Error
all features	0.6990	0.8668	0.7739	0.1838	0.00563
2K features	0.7145	0.8861	0.7911	0.3594	0.00520

miR: micro-averaging recall miP: micro-averaging precision

miF$_1$: micro-averaging F$_1$ maF$_1$: macro-averaging F$_1$

Table 1. Performance of NB on Reuters-21578 collection

NB	miR	miP	miF$_1$	maF$_1$	Error
all features	0.5144	0.5144	0.5144	0.1281	0.01129
12K features	0.5342	0.5342	0.5342	0.2572	0.01083

Table 2. Performance of NB on FBO collection

4.4 Language Modeling Approach

In the experiments using language models, we used the Lemur toolkit, which is designed and developed by Carnegie Mellon University and the University of Massachusetts [2]. The system allows us to train a language model for each class using a set of training documents, and to calculate the likelihood of a document according to each class model, i.e. $P(d \mid c_i)$. The final score of a class can then be computed according to formula (2).

4.4.1. Different Smoothing Methods. In our experiments, we used the four smoothing methods that are described earlier by varying the parameters. Table 3 shows the results by each method. No feature selection is made. The percentages in the table are the relative changes with respect to NB with no feature selection (Table 1).

Smoothing	miR	miP	miF$_1$	maF$_1$	Error
Jelinek-Mercer (λ=0.31)	0.7078	0.8778	0.7837 (+1.3%)	0.4659 (+153.5%)	0.00538
Dirichlet (μ=9500)	0.7051	0.8745	0.7807 (+0.9%)	0.3986 (+116.9%)	0.00546
Absolute (δ=0.83)	0.7118	0.8827	0.7881 (+1.8%)	0.4839 (+163.3%)	0.00527
Two-stage (λ=0.86,μ=6000)	0.7260	0.9003	0.8038 (+3.9%)	0.4214 (+129.3%)	0.00488

Table 3. Performance of LM on Reuters

As we can see, on Reuters-21578 corpus, the three first smoothing methods only lead to marginal improvements on micro-averaging F_1 over NB. On the other hand, Two-stage smoothing produces a larger improvement over NB.

The performances of different LMs on FBO collection are shown in Table 4.

Smoothing	miR	miP	miF$_1$	maF$_1$	Error
Jelinek-Mercer (λ=0.05)	0.5603	0.5603	0.5603 (+8.9%)	0.3725 (+190.8%)	0.01023
Dirichlet (μ=500)	0.5262	0.5262	0.5262 (+2.3%)	0.3486 (+172.1%)	0.01102
Absolute (δ=0.05)	0.5748	0.5748	0.5748 (+11.7%)	0.3791 (+195.9%)	0.00989
Two-stage (λ=0.05,μ=0)	0.5603	0.5603	0.5603 (+8.9%)	0.3725 (+190.8%)	0.01023

Table 4. Performance of LM on FBO

If we compare the three first smoothing methods (with their best performances shown in Tables 3 and 4), we can see that, the Absolute smoothing produced better performances than the other two smoothing methods on both corpora. Dirichlet smoothing produced the least improvements. Two-stage smoothing produced the largest improvement on Reuters. However, the phenomenon on the FBO collection is not the same. In the case of Two-stage smoothing on FBO, the best performance is obtained when μ is set to 0, i.e., we indeed use the Jelinek-Mercer smoothing. The differences of the smoothing methods on the two collections show that FBO has different characteristic than newswire articles, and they may require different classification methods.

Globally, our experiments show that using language models may improve classification effectiveness over Naïve Bayes on both corpora. This is true especially for macro-averaging F_1 which is much higher than with NB. The improvements on micro-averaging F_1 are more evident on the FBO collection than on Reuters-21578.

In order to test statistical significance of the changes of performance, we use the macro t-test [9], which compares paired F_1 values obtained for each class. It turns out that all the improvements obtained on both corpora with the four smoothing methods are statistically significant, with p-values < 0.001[3].

The comparison of the improvements on macro- and micro-averaging F_1 suggests that language models can bring larger improvements to small classes than to large classes. A possible reason is that our smoothing methods also combine the collection probabilities, instead of only changing the frequencies of words as in NB (Laplace smoothing). By modifying the frequency of words in Laplace smoothing, all the unseen words, either meaningful or not, will be attributed an equal probability. However, the smoothing methods with the collection model attribute different probabilities to unseen words according to their global distribution in the collection. Therefore, the latter probabilities can better reflect the characteristics of the collection and of the language. In our experiments, the addition of the collection model seems to benefit greatly small classes which have less training data and for which a heavy smoothing is required.

Another advantage of using the collection to smooth the class model is that the meaningless features that do not allow us to distinguish different classes are now "neutralized" with the collection model, in such a way that their differences across classes are weakened. This is equivalent to feature selection in the other classification methods. As we will see in Section 4.4.2, it turns out that feature selection is not necessary with LM. This confirms that smoothing has the same effects as feature selection.

The absolute level of performances on FBO is lower than that of Reuters. This suggests that the classification of CFTs, or more globally, the classification of business opportunities on the Web, is a more difficult problem than that for newswire articles. The main difference between them is that a CFT usually contains a very short description of the goods or services (one or a few sentences), which is the object of the call. The insufficient description makes it difficult to obtain a thorough characterization of the goods or service. On the other hand, the remaining parts, which take an important portion of the CFT, describe unessential elements for classification, such as the conditions of submission, the deadline, etc. These latter are not directly related to the classification by domain (although they may be useful for other purposes). By using the classical term weighting methods based on term frequency (or inverted document frequency), it is difficult to filter out the non-important parts of a CFT. These particularities make the global performances of classification on CFTs lower than for newswire articles.

4.4.2. Feature Selection with Language Model. Feature selection has been very useful for NB classifier. Would it produce a similar effect on language models? In order to

[3] A p-value lower than 0.05 is considered to be statistically significant at the 0.95 significance level.

answer this question, we conducted a series of experiments using different numbers of features selected according to information gain. The following table shows the results of doing feature selection on the four smoothing methods shown in Table 3.

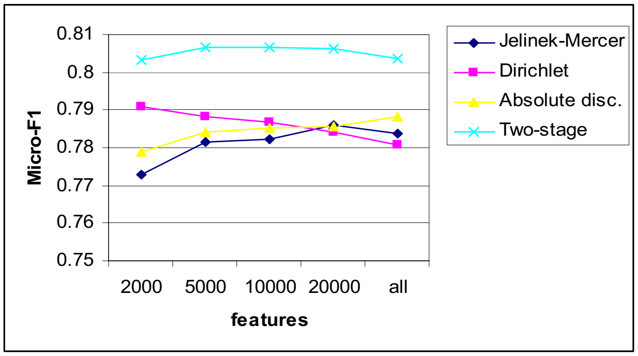


Figure 5. The effects of feature selection on Reuters

These results do not show significant performance improvement when we use feature selection, except for Dirichlet smoothing. On the contrary, for absolute smoothing and Jelinek-Mercer smoothing, the effect of feature selection is rather negative: We obtain lower performances if we select a subset of features. This conclusion seems contradictory to the results with NB, and counter-intuitive at the first glance. However, one can possibly explain this by the fact that, as the class model has been massively smoothed by the collection model, those non-discriminative features do not make a significant difference between documents with respect to a class. Therefore, the inclusion of such features in the calculation of the score does not hurt as much as in NB, which does not incorporate the collection model. This suggests that the consideration of the collection model in smoothing renders feature selection less necessary. Therefore, another important advantage of using LMs is that it can avoid the need for explicit feature selection.

5. Conclusion

We have described a tool to help the discovery of business opportunities on the Internet, and propose a new approach for the classification of such documents. The MBOI tool has been in use for a year and a half by our commercial partners, and deployed in several applications: as an aid for business opportunities watch for the St-Hyacinthe (Quebec) region, as a CFT search facility for the Canada's metal industry portal (NetMetal[4]), and as an "issue" or "thematic" watch for the Quebec travel industry. All have reported a significant improvement to their activities by using our system.

On classification, we used LM to enhance NB. In particular, the Laplace smoothing commonly used in NB is replaced by some other smoothing methods that integrate the collection model. Our experiments on Reuters-21578 and FBO collections have shown significant improvements over NB, especially on the macro-averaging F_1. On micro-averaging F_1, we also observed noticeable improvements, in particular, on FBO collection. This preliminary study did show that language models can contribute in improving text classification by NB.

Our comparison on two document collections show that language modeling approaches can be useful for the classification of both newswire articles and business opportunities on the Web, despite the differences between these documents. To further improve the classification performance of business opportunities, it will be necessary to study specific methods adapted to this type of data. In particular, we will have to deal with the problem with very short useful description in Calls for Tenders. We have noticed quite a bit of noise in the FBO documents in terms of irrelevant content, for example, pertaining to procedural instructions rather than the topic of the CFT. This is typical of Web documents, and therefore we think that it is quite encouraging that the improvement using LM was greater on FBO (a Web corpus) than on Reuters (a controlled test collection).

Our preliminary study is limited to the utilization of unigram models. We will investigate the integration of bigram language models for text classification in our future work. Other future works include: extending hierarchical classification, incorporating LMs into other classification algorithms, and using other types of features in classification (e.g., concepts, named entities as extracted using Nstein's tools).

ACKNOWLEDGMENT

This work has been carried out within a joint research project with Nstein technologies. We would like to thank Nstein and NSERC for their support.

REFERENCES

[1] S. T. Dumais, J. Platt, D. Heckerman and M. Sahami (1998). Inductive learning algorithms and representations for text categorization. *In Proceedings of ACM-CIKM98*, Nov. 1998, pp. 148-155.

[2] The lemur toolkit for language modeling and information retrieval. http://www-2.cs.cmu.edu/~lemur

[3] A. McCallum (1996). Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/~mccallum/bow

[4] A. McCallum and K. Nigam (1998). A comparison of event models for Naïve Bayes text classification. *In Proceedings of AAAI-98 Workshop*, AAAI Press.

[5] E. Dumbill. High hopes for the universal business language. *XML.com*, *O'Reilly*, November 7 2001.

[6] J. Ponte and W. B. Croft (1998). A language modeling approach to information retrieval. *In Proceedings of SIGIR 1998*. pp. 275-281.

[7] M. Spitters and W. Kraaij (2001), TNO at TDT2001: language model-based topic detection, *In Proceedings of Topic Detection and Tracking (TDT) Workshop 2001.*

[4] `http://www.netmetal.net/`

[8] Y. Yang (1999). An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, Vol. 1, No. 1/2, pp. 67–88.

[9] Y. Yang and X. Liu (1999). A re-examination of text categorization methods. *In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 42-49.

[10] Y. Yang (2001). A study on thresholding strategies for text categorization. *In Proceedings of SIGIR 2001*, pp 137-145.

[11] C. Zhai and J. Lafferty (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. *In Proceedings of SIGIR 2001*, pp. 334-342.

[12] C. Zhai and J. Lafferty (2002). Two-stage language models for information retrieval. *In Proceeding of SIGIR 2002*, pp. 49-56.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 37-44, 2004.

Facilitating Web-based Education using Intelligent Agent Technologies

Yang Cao and Jim Greer
ARIES Laboratory
Department of Computer Science, University of Saskatchewan
57 Campus Drive Saskatoon, SK. Canada S7N 5A9
{ yangc@athabascau.ca, jim.greer@usask.ca}

Abstract

Software agents will soon proliferate human organizations, education and society, helping users with information gathering, activity scheduling, email management, and individual and collaborative learning. This paper presents an intelligent Web-based educational system using multi-agent system technology and Web services technology. In this multi-agent architecture, each user is assigned with a personal assistant– software agent. In order to achieve teaching/ learning tasks, humans and agents need an effective way to interact. Two alternative approaches were developed for programmable agents in which a human user can define a set of rules to direct an agent's activities at execution time. This research also investigates concerns over user privacy and system security caused by agent programmability in an web-based interactive learning environment.

1. Introduction

As the demand for access to education grows and an increasing numbers of adults return to universities/colleges for continuing education and training [1], so grows the need for new technologies to facilitate learning. Online teaching and learning provide great opportunities to increase flexibility in time and location of study, in terms of availability of information and resources, synchronous and asynchronous communication and various types of interaction via the World Wide Web.

Agents have become popular additions to an interactive learning environments. In general, an interactive learning environment consists of the teachers and the fellow learners with whom the learner interacts during the learning process; the teachers and learners can be human or artificial companions. Besides teacher/lear ners, the learning environment also consists of a set of computer-based tools that can be used by the learner (i.e. educational software, communication tools), and the learning material that contains the topics the learner has to learn.

The agent-based approach is suitable for supporting Web-based education since relationships among learners, courses, and instructors last for a considerable period of time [2]. Due to the inherent distributed nature of Web-based learning, a Web-based educational environment can be enhanced by a set of software agents [3][4]. A lot of experimental research has shown that intelligent software agents have great potentials for reducing information workload and for automatically performing many knowledge/labor-intensive tasks for learners and educators [5].

An agent is known as a computer system that is "situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives" [6]. The agent's ability to play the role of a personal assistant arises from its autonomy, reactivity, and pro-activity properties. An agent with such properties could enter into negotiations, acting independently to help achieve the user's goals in an unpredictable environment, and communicate with the user. However, it is also these properties, particularly autonomy that raises significant challenges in human-agent interaction. The agent research community has focused on technologies for constructing autonomous agents and techniques for collaboration among agents. Little attention has been paid to supporting interactions between human and agent.

The issues in human-agent interaction may be more generally described by the following four categories [7]: delegating tasks and authority, instructing agents to act and react, sharing context, and dialogue issues. For example, questions arise as to how a user can successfully delegate a task to an agent, how agents acquire knowledge needed to understand a particular task and find a way to accomplish it, and how a system can deal with a disagreement between the user and his or her agent. Trust, user privacy and security issues have also become concerns in the design of agent-based systems.

This research has been focused on delegating tasks to an agent in a web-based supported learning and specifically within the bounds of a multi-agent online learning environment named I-Help.

The objectives of this research were to investigate how users behave when given the ability to program their agents, what are the users' concerns about their privacy, how agent-based systems can be built to protect users' privacy, whether the overall performance of the system will be affected with agent programmability, and whether agent programmability be better achieved by adding a full-fledged programming environment (like a rule based expert system shell) to the agent versus by adding a simpler customised and restricted rule system.

2. Background

Animated pedagogical agents [8] have been used in learning environments as artificial trainers. The pedagogical agents are animated characters that guide and encourage learners' study in computer-based learning environments. They interact with learners in a manner simulating the behavior of human tutors that includes a combination of verbal communication and nonverbal gestures. They can express both thoughts and emotions which are significant for human teachers. These pedagogical agents are not only knowledgeable about the topics being taught, but also have knowledge about pedagogical strategies and how to obtain relevant information from available resources such as the World Wide Web. One of the example pedagogical agents is STEVE, a virtual trainer for 3D environments [9]. STEVE can answer questions, monitor students' action, and advise learners when playing the role of a tutor as well as a learner's teammate. It provides more humanlike assistance than previous automated tutors could because of his animated body and interaction in the virtual world with students.

AUTOTUTOR and ATLAS are two other successful tutoring systems. AUTOTUTOR [10] is a fully automated computer tutor that has provided guidance for college students in a computer introductory course. AUTOTUTOR tries to comprehend student contributions and stimulate dialogues to guide students answering deep-reasoning questions. ATLAS [11] is a computer tutor for college physics that focuses on improving students' conceptual knowledge.

As the telecommunication infrastructures and the Internet grow, they provide great facilities for online delivering education and collaborative learning. Online learning is defined as Internet-enabled learning or e-learning, including any use of computers and the Internet to facilitate education [12]. Unlike the traditional distance learning, the success of the new online learning environment is not only just delivering the instructional materials but also providing a collaborative learning environment in the virtual learning community. One of the key elements for successful collaborative learning is peer-to-peer sharing of experiences [13] [14]. This provides a sense of belonging, a sense of feeling part of the community.

In the next section an agent based online collaborative learning environment, I-Help, is introduced, followed by an activities awareness issue in this learning environment.

3. I-Help system

The I-Help [15] system was developed in the Advanced Research in Intelligent Educational System Lab of the Department of computer Science, University of Saskatchewan, Canada. I-Help is designed to provide just in time help for students over the Internet. It is a "peer help" system where the students share their knowledge and exchange information with each other. That means people who receive help also give help [16]. There are two main components in the current I-Help system: the public discussion component and the one-to-one private discussion component.

3.1. Public discussion

Public discussion forums are also known as bulletin boards or newsgroups. In the public discussion forums, learners can post questions, discussion problems of common interest, reply to questions posted by others, read posting and search for posting according to author, concepts, keywords, etc. The public discussion component clusters user discussions around the courses in which they are currently enrolled. All the students who are taking a particular course share the same information including questions and answers within the various course forums. Each question or response to that question is called a posting which consist of a unique posting id and author name, etc.

The information about postings and the users' activities in the Public Discussions such as when a user reads a particular posting, when a user posts to a forum, etc. are recorded in the I-Help database.

3.2. One -to-One private discussion

In the one-to-one private discussion component, conversations are private and restricted to pairs of people. When a learner asks a question, an appropriate helper is recommended by the system. The system will match the student model with the models of other students, to find peers who are more suitable to provide help in a timely fashion. The helper is rated according to several factors, such as the knowledge level, availability, and eagerness to help, etc. Once the helper is selected, the helper and the helpee can start to communicate. The dialogues may be synchronous or asynchronous and many private discussions with different partners can proceed simultaneously.

The I-Help system is built on a multi-agent architecture where each person is augmented with a personal agent who acts on the user's behalf to manage the offering and getting of help. In particular, the personal agents are designed to monitor user activity, and to assist learners in locating help resources (both human helper and electronic help resources). Each personal agent keeps a model of its "owner" and this is used to find the best helpee-helper matches when negotiating help with other agents [17]. The user model information is obtained from the learners' self-assessment of knowledge level of the various topics, from short peer evaluations that occur at the end of a help session, and from monitoring student activities in both parts of the I-Help system. Users' activities which are used to measure student participation in I-Help include whether or not the student is currently or frequently online, how often a student reads/posts a message on the public discussion forum, and how often a student answers or replies questions/messages in the private discussion, etc. An agent negotiates with other agents on behalf of its user using a negotiation mechanism [18].

The Matchmaker agent is a coordinator agent that facilitates finding a best helpee-helper match. Matchmaker maintains profiles of the knowledge and some other characteristics of all the users in the system. Each user is able to change their help preference at any time. The user can specify the knowledge level for the various concepts that are relevant to the courses, the number of discussions he/she would like to process at once, about which topics or whom he/she will not help at all. As well the user can tell his/her agent how much he/she wishes to be paid for offering help and how much she is willing to pay for getting help.

A peer evaluation form is available for a learner to evaluate his/her partner after the help session completes. The evaluation includes whether the helper is helpful and knowledgeable on the topic they are working on. This information is stored in personal agent and maintained by matchmaker who uses it in subsequent matches.

3.3. Awareness issues in the current I-Help

Both I-Help prototypes have been used in computer science courses in the University of Saskatchewan. The students found the I-Help system useful and helpful. Most students responded that "reading postings helped their learning"; most found "answers received useful"; many found that "answering other people's questions helped in their own learning"[3].

I-Help users could send out help requests, read postings, or get replying from helpers. However, the WWW techniques and current version of I-Help do not address the problem of feeling “deaf”, “blind” and “alone” due to the lack of mechanisms to support awareness. In the current system there is no way or efficient manner for a user to know about other persons' presence, availability, willingness to interact, and other events happening in the I-Help environment. However, users want to know what is going on in their virtual community just as they would in any real society. For example, a user may want to know when another user logs in to the system, which posting attracts most of the people, and whether users have read a particular message, etc. These events could be used by agents to determine which person would be a good helper and most likely answer a question in a timely manner. Users could use this information to learn or infer about each other in order to cooperate in the learning community. A preliminary user study on activity awareness in I-Help demonstrates that awareness of other learners’ activities facilitates both individual learning and collaborative learning [19]. Other research [20][21] also show that in an online environment the participants' awareness of each other's activities is a critical feature when trying to build successful communities.

A human-agent interface should potentially be contrived to allow users to program their agents to obtain the activities information. Next section describes the design and implementation of programmable agents in I-Help.

4. System Design and implementation

4.1. Example scenarios

Figure 1 presents some examples of the usage of our Agent programming system. It illustrates how the end user programming environment enables different users to monitor others' activities in the I-Help world.

There are three types of users and they have different intentions using the system. Figure 1 includes one instructor, one tutor and two students. The instructor wants to know about common problems encountered by the students, the tutor wants to know whether there is a new question posted, and the two students need help from the instructor and the tutor. Each user can program his/her agent through a specific user interface about what he/she likes to watch and what action should be taken when a particular event happens. When these events happen, the agents will take appropriate actions according to the preference of their owners.

The student "A" may configure a rule to program his agent to send a special notification to the tutor within a half hour after the tutor reads his particular message in the public discussion forum. The rule looks like:

If *the tutor has read message 19765 within the past 30 minutes*

then *notify tutor with the subject "Can we talk?"*.

The student "B" might configure a rule to program her agent to notify her when the instructor signs in to the system. The rule looks like:

If *the instructor has logged in within the past 2 minutes*

then *notify me with the subject "The Instructor just signed in"*.

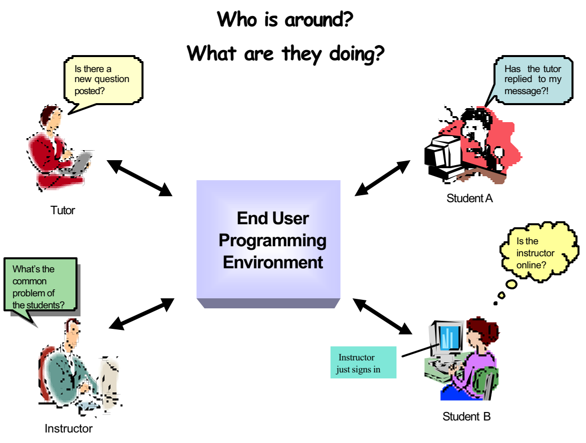

Who is around?
What are they doing?
Is there a new question posted?
Tutor
Has the tutor replied to my message?!
Student A
End User Programming Environment
What's the common problem of the students?
Instructor
Is the instructor online?
Instructor just signs in
Student B

Figure 1. Examples of usage of the end user environment

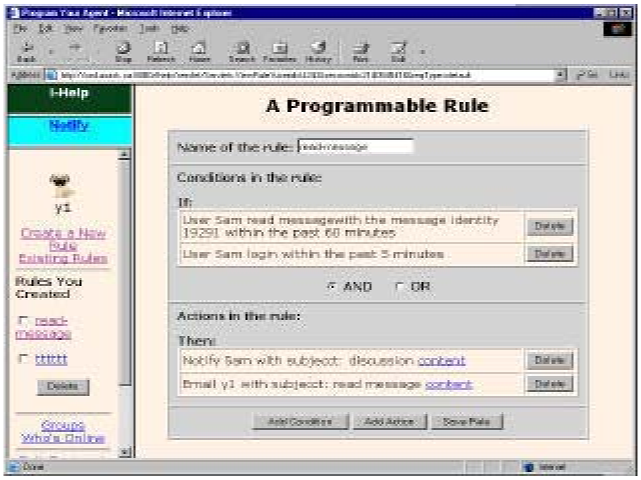
When the instructor signs in to the system, the system will inform student B. The tutor will get a notification message with the subject "Can we talk? after she reads message 19765.

The users are able to generate complex rules by combining several conditions and actions. See examples in the next section. The users can also write a rule to trigger another rule.

4.2. User Interfaces

In this research, two alternative approaches are employed to build user agent programming environments on top of the I-Help system. One approach is to add to each agent in I-Help a simpler customized rule system, which is called Agent Rule Management System (ARMS). Another approach is implementing CLIPS-based agents that involve connecting a rule based expert system shell to each personal agent in I-Help.

The primary user interface for a user to program his/her agent is the rule management interface which includes one notification signal bar named as Notify, an index frame with the names of the existing rules, and a rule editor frame (Figure 2).



A Programmable Rule

Name of the rule: read-message

Conditions in the rule:

If:

User Sam read messagewith the message identity 19291 within the past 60 minutes

User Sam login within the past 5 minutes

● AND ○ OR

Actions in the rule:

Then:

Notify Sam with subject: discussion content

Email y1 with subject: read message content

Figure 2. Rule Management Interface

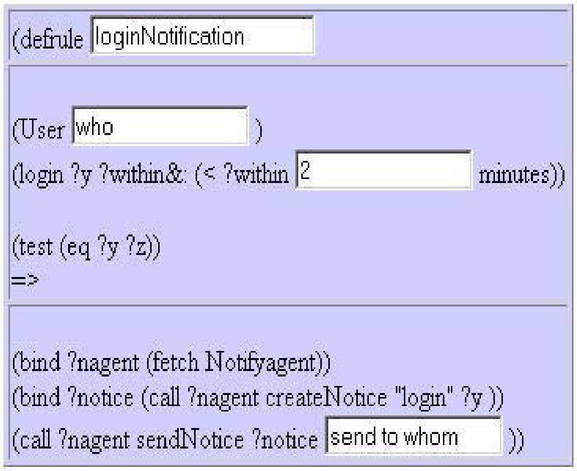
The notification signal (left upper) is used to notify a user when a new notification message is received. When a new notification message arrives, the notification signal bar will turn blue. Figure 2 shows that there is a new notification message for the user. The index frame (left part in Figure 2) enables a user to view the names of all the existing rules, to delete selected rules, to look at a particular rule, and to create a new rule. The actual generation and modification of the rules are performed in the rule editor (the right part in Figure 2), condition specification, and action specification interfaces. Once the rules are generated, they are displayed as an understandable pseudo-English sentence in the rule editor.

The rule management interface of the CLIPS-based rule environment is similar to Figure 2, which includes one notification signal bar named as Notify, an index frame with the names of the existing rules, and a rule editor frame.

There is a list of rule templates in the index frame of the rule management interface. Similar as ARMS approach, each rule contains three parts: rule name, a condition part, and an action part. A user is able to configure a rule by selecting and filling the value in a template. Figure 3 is a sample rule called loginNotification. The meaning for this template is

When a particular user logged in to the system within the past " 2 " minutes, then create a login notice with the information about his / her login status and send it to me or other users.

A user can specify who will receive the notification message when someone logs in at a particular time by filling the blanks in the templates (see Figure 3). In addition to selecting and filling the value in a template, the users are able to make complex rules by combining several events/actions as well as to define their own rules without using any functions provided by the system.



```
(defrule loginNotification

(User who )
(login ?y ?within&: (< ?within 2 minutes))

(test (eq ?y ?z))
=>

(bind ?nagent (fetch Notifyagent))
(bind ?notice (call ?nagent createNotice "login" ?y ))
(call ?nagent sendNotice ?notice send to whom ))
```

Figure 3. A CLIPS Interface for Login Notification Template

CLIPS permits users to code arbitrary rules to make their agents act in various ways. The full power of CLIPS would allow users to behave in ways that might compromises the system. For this reason, we decided to limit users to making CLIPS rules through template filling. There would be fewer syntax/run time errors when users are filling templates than when they are coding rules for themselves.

4.3. System architecture and implementation

Figure 4 represents a high level architecture for the I-Help end user programmable environment. The Rule Management module, Rule Cycle Detector modules are inside the box of Other Applications.

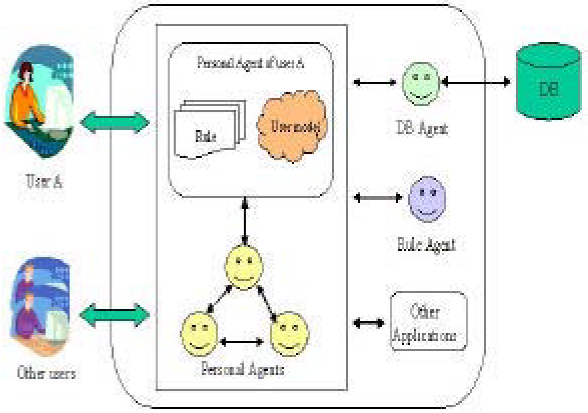


Figure 4. Architecture of I-Help End User Programming Environment

The function of each module is briefly introduced below:

- The main function of a PersonalAgent here is to communicate on behalf of its owner and this is achieved by collaborating with DBAgent, RuleAgent, and other applications. Each personal agent consists of a user model and a set of tasks to be performed. The functionality of the personal agents includes notifying the owner when specified conditions occur, delivering messages to other users or their agents, and responding to messages from other users or agents.
- DB Agent is an application agent that handles all writes and most reads from the database. The information about users' activities are stored and retrieved via DB Agent.
- RuleAgent is an agent that deals with managing the rule repository and detecting interactions among a set of rules.
- The interactions between a user and his/her agent are through a set of user interfaces which are composed of static and dynamically created html pages. Information is sent between the user and agent via Java Servlets.

- Other applications include Rule Management module, Rule Cycle Detector, and other Java components.

More information on structure of rules and system implementation can be found in [22].

5. Evaluations and results

5.1. Usability study on ARMS approach

An experiment on usability of the ARMS approach [22] was conducted with human subjects, to observe the behaviors of subjects during the experiments and analyze the questionnaire and rules generated by the subjects in terms of: Whether the users feel agent programmability is helpful; How easy/hard for a user to configure a rule and how they feel the difficulty of configuration; What kind of rules they would write, what they would watch, what kind of dangers to the system/ users would risk; What were the users' concerns on their privacy including personal information and activity information; and Whether the students had better performance in a learning task with agent programmability than no programmability support.

5.2. Comparative usability study

One of the objectives of this research was to evaluate and compare the strengths and weaknesses of the two approaches technically to see whether the agent programmability would be better achieved by adding a full programming environment CLIPS than a simpler customized rule system ARMS.

A one-hour comparative study on the comparisons of the CLIPS versus ARMS approach was conducted with ten human subjects who were selected from staff from various departments of Athabasca University and students from the University of Alberta. Some staff work in educational media development and some work in the computing centre. These people have experiences with online teaching, educational technology, and online course delivery techniques.

During the experiment, the subjects' activities included attending an introduction session, comparing two approaches, and completing an exit questionnaire on feelings about the system and security and privacy concerns.

In the introduction session, brief information was given on I-Help public discussion and private discussion forums and an explanation was given on how to use the systems. Each subject was given a demonstration of the agent programming environment in I-Help, which described what kind of activities agents can watch, how they can respond, and how to use the system, with both the ARMS and CLIPS user interfaces.

After the introduction, the users compared these two approaches by looking in detail at the interfaces of the CLIPS and ARMS approaches, filling in a form about their opinions on these two approaches, such as which approach is easy or hard to use, which is more or less powerful, and which is more or less secure, etc. Finally the users were required to take part in a structured interview session. During the interview, the author asked the subjects for their responses to a set of general questions, which included how they felt about the usefulness of the I-Help agent programming environment and whether surveillance issue and privacy concern might prevent them from using the system in future.

The questionnaire on system usability and the comparison as well as a record of interview were collected for analysis.

5.3. Result and discussion

We asked similar questions on usability and privacy concerns in the comparative experiment as the ones we did in ARMS usability study. In general the result is encouraging [22]. People would like to watch the login, read message, and send message activities of instructor/tutor and knowledgeable students and group members when they need help, want to discuss a question with others, or during a group discussion. None of the people indicated that they like to watch others just for curiosity. People indicated that security or privacy was not a big concern and it would not prevent them from using the system. However, similarly as the survey in ARMS usability study, concerns were raised by some users when people other than a tutor or a friend was watching them on certain events, such as send message, read message.

The majority of users felt the I-Help programmable agents would be very useful or useful to some extent and they would tend to use the system more than before or as same as now if programmable agents were available. No one said that it was useless or they would stop using it.

Table 1. (Appendix) shows the comparison result of the two approaches.

An interesting observation is that compared to the answers in the first ARMS usability study, more people would feel tempted to try to write some rules that could threaten the system or surveil other users if they had the power. This may be caused by their belief in the power of the CLIPS approach. In the interview, one user said she was very curious to know/see the power of CLIPS, and she would like to see how she could affect the system by writing her own rules.

6. Conclusions and future work

This paper presents two alternative systems were developed for programmable agents in which a human user can define a set of rules to direct an agent's activities at execution time, such as to communicate with other agents and to monitor the activities of other users and their agents. We reached the following conclusion based on the experiment result: (1) Agent programmability is able to support different users' needs and preferences (i.e. awareness of users activity) in the I-Help world. (2) The provision of agent programmability facilitates the participants accessing necessary resources (human and electronic) in their collaborative learning environment. (3) Agent programmability supports individual and collaborative learning by facilitating information exchange and enhancing communication among students within the virtual learning environment.

This research also provides a platform for investigating concerns over user privacy caused by agent programmability and how an online learning environment can be built to protect users' privacy. The result of the survey on users' privacy shows that people would like to expose more activity information to the public. However different degrees of privacy concern occur in the participants on different kinds of events in the learning environment. There is a desire that the users should have control over their agents to protect their privacy.

The future work for I-Help agent programming environment include making more system events available to users and developing programmable anti-spy agents that will enable a user to program his/her agent to detect surveillance activities of other agents, to notify the user, to take other actions, such as filter / block the information.

It is desirable to integrate the programmable agents with other interactive help facilities or e-learning applications, such as an instant messenger and the course delivery system.

7. References

[1] CIHE: The Council for Industry and Higher Education, Response to the joint consultation document from HEFCE and the Learning and Skills Council (2002), http://www.cihe-uk.com/partnershipsfor.htm

[2] Chan, T-W. (1995), Artificial Agents in Distance Learning, International Journal of Educational Telecommunications, 1(2/3), 263 -282.

[3] Greer J., McCalla G., Vassileva J., Deters R., Bull S., and Kettel L. (2001) Lessons Learned in Deploying a Multi-Agent Learning Support System: The I-Help Experience, Proceedings of AIED'2001, San Antonio, 410-421.

[4] Baylor, A. (1999). Intelligent agents as cognitive tools for education. Educational Technology, Volume IX(2), 36-41.

[5]Thaiupathump, C., Bourne, J., and Campbell, J. O. (1999) Intelligent Agents for Online Learning, JALN Vol.3, Issue 2.

[6]Wooldridge, M., and Jennings, N. R. (1995) Intelligent agents: Theory and practice. The Knowledge Engineering Review, 10(2), 115-152.

[7]Dickinson, L.(1998) Human-Agent Communication. http://www.hpl.hp.com/techreports/98/HPL-98-130.pdf

[8]Johnson, W. L., Rickel, W., and Lester, J.C. (2000). Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments. International Journal of Artificial Intelligence in Education 11(1), 47-78.

[9]Rickel, J., and Johnson, W. L.(1999) Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control. Applied Artificial Intelligence 13(4-5), 343-382.

[10]Graesser, A.C., Wiemer-Hastings, K., Wiemer-Hasting, P., Kreuz, R., and the Tutoring Resarch Group.(1999) AUTOTUTOR: A Simulation of a Human Tutor. Journal of Cognitive Systems Research 1(1), 35-51.

[11]Vanlehn, K., Freedman, R., Jordan, P., Murray, C., Osan, R., Ringenberg, M., Rose, C. P., Schulze, K., Shelby, R., Treacy, D., Weinstein, A., and Wintersgill, M. (2000). Fading and Deepening: The Next Steps for ANDES and Other Model-Tracing Tutors. In Intelligent Tutoring systems: Fifth International Conference, ITS 2000, eds. G.Gauthier, C. Frasson, and K. Vanlehn,. Berlin: Springer-Verlag. 474-483.

[12]Downs, S. (1998) The future of online learning. http://www.atl.ualberta.ca/downes/future/home.html

[13]Greer, J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., and Vassileva, J. (1998). The Intelligent HelpDesk: Supporting Peer Help in a University Course, in B.Goettl, H.Halff, C.Redfield, V.Shute (eds.) Intelligent Tutoring Systems, Proceedings ITS'98, San Antonio, Texas, LNCS No1452, Springer Verlag: Berlin. 494-503.

[14]Pressley, M., Wood E., Woloshyn, V, Martin, V., King, A., and Menke, D. (1992) Encouraging mindful use of prior knowledge: Attempting to construct explanatory answers facilitate learning, Educational Psychologist, 27(1), 91-109.

[15]Greer, J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., and Vassileva, J. (1998). The Intelligent

HelpDesk: Supporting Peer Help in a University Course, in B.Goettl, H.Halff, C.Redfield, V.Shute (eds.) Intelligent Tutoring Systems, Proceedings ITS'98, San Antonio, Texas, LNCS No1452, Springer Verlag: Berlin. 494-503.

[16]Greer J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., & Vassileva, J. (2000). Integrating Cognitive Tools for Peer Help in Computers as Cognitive Tools: The Next Generation, Susanne P.Lajoie (Ed.) Mahwah, NJ: Lawrence Erlbaum Publishers, 69-96.

[17]Vasseleva, J. Greer, G. McCalla, R. Deters, D. Zapata, C. Mudgal, S. Grant (1999) A Multi-Agent Approach to the Design of Peer-Help Environments, in Proceedings of AIED'99, Le Mans, France, 38-45.

[18]Mudgal, C., and Vassileva, J. (2000) Multi-agent negotiation to support an economy for online help and tutoring, in Proceedings of ITS'2000, Springer LNCS 1839, 83-92.

[19]Cao, Y., and Greer, J (2003). Supporting Awareness to Facilitate Collaborative Learning in an Online Learning Environment. Proceedings of the Computer -Supported Collaborative Learning (CSCL 2003), Bergen, Norway. 183-187.

[20] Jermann, P., Soller, A., and Muehlenbrock, M. (2001) From mirroring to guiding: A review of the state of art of technology for supporting collaborative learning. In Proceedings of the First European Conference of Computer-supported Collaborative Learning (Euro-CSCL). McLuhan Institute: University of Maastricht.
http://www.mmi.unimaas.nl/euro -cscl/Papers/197.pdf

[21]Schichter, J.H, Koch, M., and Xu, C. (1998) Awareness-The common link between Groupware and community support system. Community computing and support systems: Social interaction in networked communities. Berling:Springer-Verlag. 77-93.

[22]Cao, Y., and Greer, J.(2003) Agent Programmability in a Multi-Agent Learning Environment. Proceedings of the 11th International Conference on Artificial Intelligence in Education, Sydney, Australia. 8 pp.

Appendix

Table 1. Comparison on ARMS and CLIPS approach

1. How easy to program the agent (configure a rule):		
	ARMS (%)	**CLIPS (%)**
Easy to understand /operate without help	70%	20%
It's easy with a little help	30%	20%
It's confusing to understand/operate without help	0	60%
It's very hard to understand/operate even with help	0	0
2. For the tasks that are available in both approaches, the approach which the users were preferred to use:		
	ARMS(%)	**CLIPS (%)**
Prefer to use	90%	10%
3. Which approach the users thought have more power:		
	ARMS(%)	**CLIPS (%)**
Which has more power	20%	80%
4. How do you feel the risk of the system security or your own privacy?		
	ARMS(%)	**CLIPS (%)**
This approach is less secure	20%	60%
This approach is dangerous	0	30%

Self-Protection of Web Content

Hoi Chan, hychan@us.ibm.com Trieu C. Chieu, , tchieu@us.ibm.com

IBM T.J.Watson Research Center
19 Skyline Drive
Hawthrone NY, 10532

Abstract

In most Internet applications, there is little control on how to protect the data content once it reaches the client. Implementing centralized control for data content delivered to the client at the server side is complicated and requires frequent server and client interaction which may influence user experience negatively. This suggests that data contents embedded with self-protecting functions which run independently on the client side may be desirable. In this paper, we propose an approach utilizing existing browser and agent technology to enable content protection function by using agent embedded in the delivered content. We illustrate this approach by using a web mail application, in which the content of the display page is locked up automatically using embedded functionalities in the html page when no activities are detected for a period of time even when server connection is not available.

Introduction

As the complexity and use of web applications increases, building systems with agent technology[1,2,3] to perform various tasks autonomously becomes increasingly important. Much of the research and development on agents and its related technologies focuses on searching, mining, comparing, negotiating, learning and collaborating. Very little attention has been given to autonomic protection functions[4], especially to protect individualized data content once it reaches the client.

Typically, a server protects data access by disconnecting and logging out a user if no user activities are detected for a period of time. However, this does not prevent the current display page with sensitive information at the client workstation from being read by others when the workstation is unattended. To include functions which can protect or reconfigure the data content requires functionalities beyond what the typical browser can provide. These protection functions can be implemented, to a certain extent, by using a browser plug-in[5] which extends the functionalities of the browser to provide the necessary protection functions. However, in most of the cases, proprietary plug-ins need to be installed explicitly in a browser, and may not be readily available at the client side for the specific data content and applications. In addition, plug-ins in general offer pre-defined and generalized functionalities, it lacks the flexibility to configure itself dynamically to meet individualized needs.

In this report, we describe an approach to use intelligent agents embedded in the delivered content to provide functions specific to the application data or users. This embedded agent approach not only eliminates the need for plug-ins, it also greatly enhances its flexibility to provide functions tailored to specific user needs. In other words, different agents (such as applets[6]) with different protection mechanisms can be dynamically configured and included in the delivered web page based on user information and other criteria.

Data Content Embedded with Agent

In general, for web based application, a browser allows the user to select a link and retrieve another screen of information. The browser itself does not provide a lot of functionalities to manipulate the data. However, Java agent and plug-in technology allow extension of browser functions, and provide Java like functionalities in the browser environment. Therefore, it becomes possible to include a set of functions embedded in the data content which implements the various protecting functions by using agent technology. A Java[TM] agent, such as an applet, is a program written in Java programming language that can be included in a html page, much in the same way an image or text are included. When you use a Java technology-enabled browser[7] to view a page that contains an applet, the applet's code is transferred to your system and executed by the browser's Java Virtual Machine (JVM).

Typically, user activities and other browser parameters can be accessed through methods in JavaScript[8] in html pages. To enable an applet to access information from the current html pages, we need a mechanism to go beyond the boundary of Java Runtime Environment (JRE) of the applet and connect the applet with the JavaScript in the html pages. This is achieved by using the LiveConnect[9] facility, which is readily provided in most commercial browsers. Basically, this facility provides a netscape.javascript.JSObject class[10], which permits applets to work with JavaScript to enable a way to access the document-object-model (DOM)[11] of an html page.

Based on this facility, we have developed a design to provide a time out protection function embedded in web page. The root of the design is a Java class called *AbstractWebContentAgent* class. It provides a set of common functions to allow communication between Java applet and JavaScript to access the content of the html page. Figure 1 is a list of sample methods in the *AbstractWebContentAgent* class. Figure 2 and 3 show the basic Java applet code to access the html page through the use of JavaScript and LiveConnect facility. The *readContent* method is used to read the DOM of the html page, while the *writeContent* method is used to write data back to the original page. These two methods utilize the JSObject class to access the DOM of the html page.

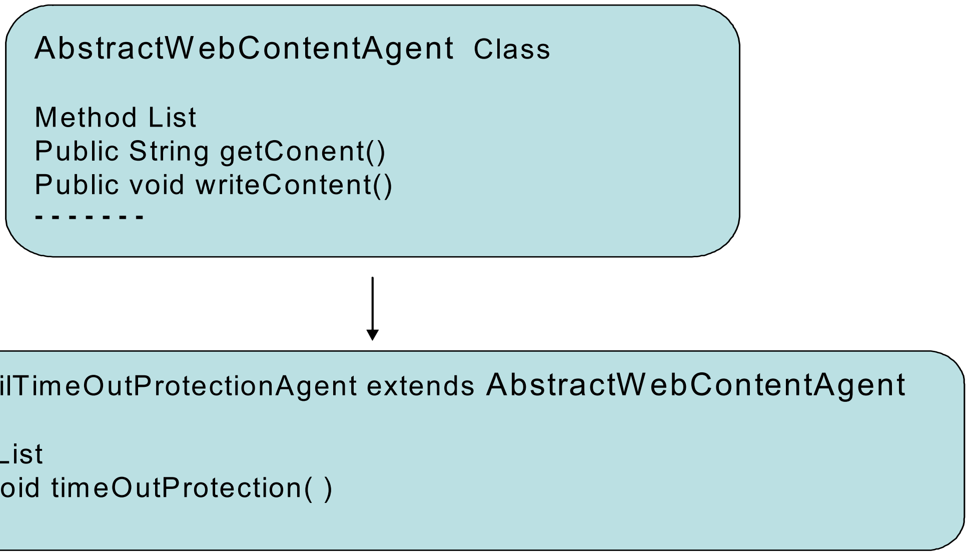

AbstractWebContentAgent Class
Method List
Public String getConent()
Public void writeContent()
- - - - - - -
WebMailTimeOutProtectionAgent extends AbstractWebContentAgent
MethodList
Public void timeOutProtection( )
……

Figure 1. methods provided in the AbstractWebContentAgent class and its extension - WebMailTimeOutProtectionAgent.

```
JSObject thisWindow = JSObject.getWindow(this);
JSObject document = (JSObject) thisWindow.getMember("document");
JSObject myMember = (JSObject) document.getMember(“myMember");
// get the member “myRef” as a string
String s = (String) myMember.getMember(“myRef");
```

Figure 2. Reading from the html page using JSObject

```
JSObject thisWindow = JSObject.getWindow(this);
JSObject document = (JSObject)thisWindow.getMember("document");
String htmlText = ”myText”;
args = new Object[ ] {htmlText};
document.call("write", args);
```

Figure 3. Writing to html page using JSObject

Protecting Function for Web Content

The protection scenario involves a company exposing sensitive mail information due to workstations frequently left unattended. The company wants to protect certain sensitive mail pages from displaying in client workstations when no user activity is detected for a certain period of time, even after server connection is disconnected. To achieve this, an autonomic protection function is needed which can react to user idling time. This function will lock out the current display page after a specified period of time if no user activities are detected and the user is required to enter the password again to return to the current page. To allow the mail page to be returned for viewing, the lock-up page should be stored locally and securely in order for the user to retrieve it, even when no connection to server is available.

Based on these requirements, one of the most flexible and cost effective approaches is to embed an agent, such as a Java applet, in selected mail pages. The applet performs the time-out protection function and utilizes the *WebMailTimeOutProtectionAgent* class (WMTOPA). This WMTOPA class extends the *AbstractWebContentAgent* class given above and allows the applet to gain access to the content of web mail page. through the *readContent* and *writeContent* methods.

Another function provided by this applet is to store the mail page locally after a time-out period is reached. To avoid security exposure, we need to encrypt the page before storing. To this end, we have developed a double encryption algorithm which uses a public/private key mechanism[12,13] together with the user password to encrypt and decrypt the mail content. The algorithm for the time-out protection and encryption mechanism at the client browser is illustrated in Fig. 4,5 and 6.

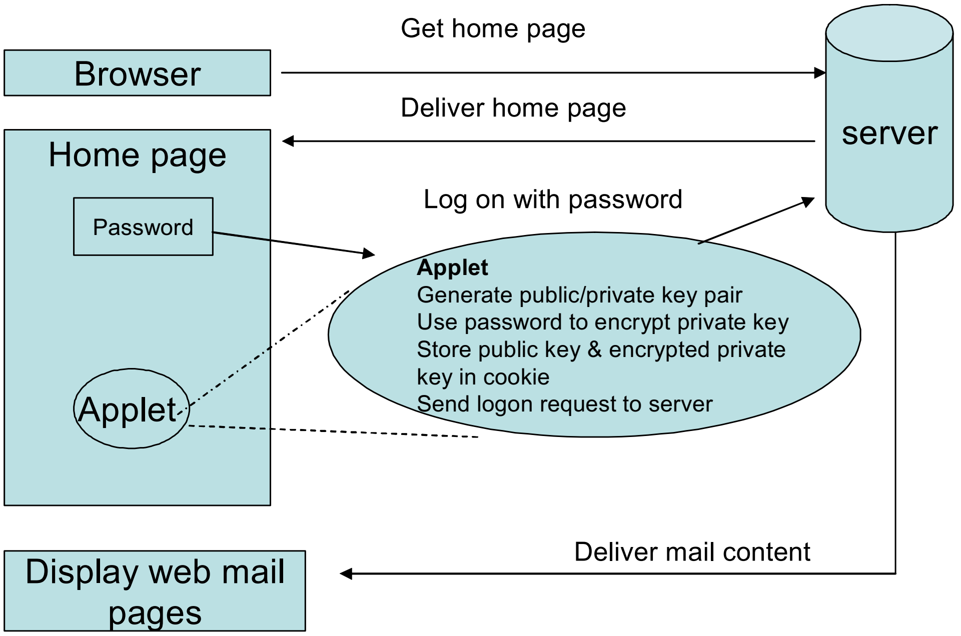

Get home page
Browser
server
Deliver home page
Home page
Log on with password
Password
Applet
Generate public/private key pair
Use password to encrypt private key
Store public key & encrypted private key in cookie
Send logon request to server
Applet
Display web mail pages
Deliver mail content

Figure 4. Mechanism to generate self-protecting behavior – locally generate encryption key

The operation sequence illustrated in Figure 4 for generating initial encryption keys is summarized below:

1. User requests mail home page.
2. Server delivers home page to user.
3. User enters a password to login. Applet generates a public/private key pair ($K_{private}$, K_{public}).
4. Applet uses the password entered by the user to encrypt $K_{private}$ to get $K_{encrypted\ private\ key}$.
5. Applet stores K_{public} and $K_{encrypted\ private\ key}$ in session cookie[14].
6. Send login request to the server.
7. Upon successful password verification, server delivers the requested mail page to browser.

The reason for the above operation is to avoid using the user password for future encryption/decryption of the mail page, which requires storing the unencrypted password locally. This may cause a security exposure. To address this issue, we use a double encryption algorithm with a public/private key mechanism, and encrypt the private key using user password during initial logon. By keeping public and encrypted private keys in the session cookie, we can use the public key to encrypt information when needed, and retrieve the encrypted private key for decryption when a user enters the same logon password.

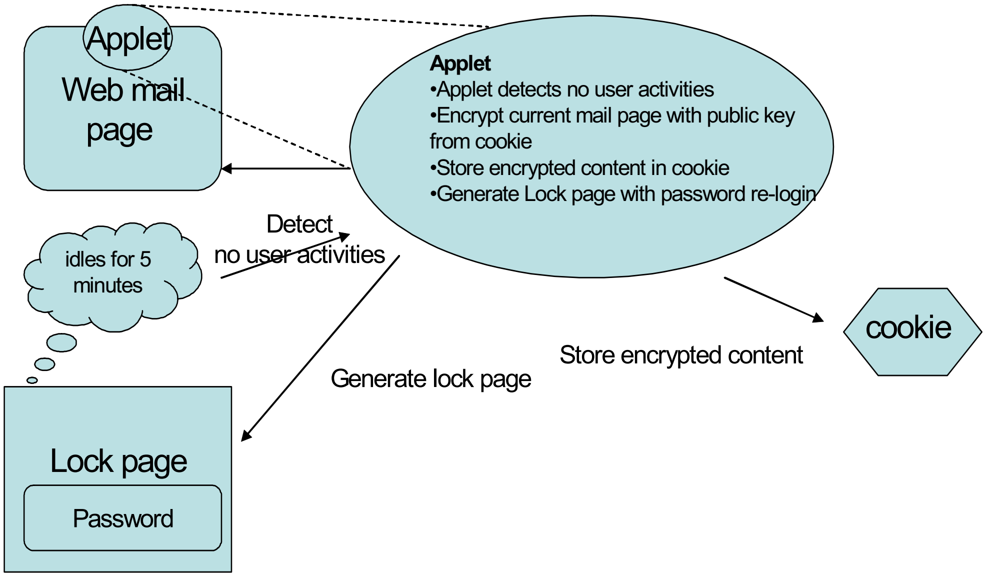

Applet
Web mail page
Applet
•Applet detects no user activities
•Encrypt current mail page with public key from cookie
•Store encrypted content in cookie
•Generate Lock page with password re-login
idles for 5 minutes
Detect no user activities
Generate lock page
Store encrypted content
cookie
Lock page
Password

Figure 5. Mechanism to generate self-protecting behavior – locally encrypt and store mail content, display Lock page with re-login

The operation sequence illustrated in Fig. 5 for locking and storing an encrypted mail page locally is summarized below:

1. Mail page is displayed on user's workstation.
2. If no user activities detected within a period of time (e.g. 5 minutes)
3. The current mail page is encrypted with public key K $_{public}$ retrieved from session cookie.
4. The encrypted content of the entire mail page is stored in session cookie
5. Generate a lock page with password unlock prompt

.

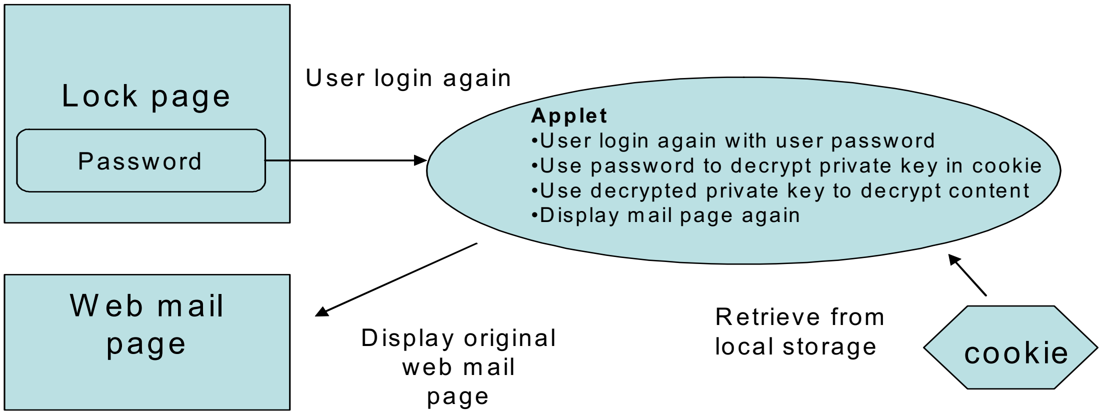

Lock page
Password
User login again
Applet
•User login again with user password
•Use password to decrypt private key in cookie
•Use decrypted private key to decrypt content
•Display mail page again
Web mail page
Display original web mail page
Retrieve from local storage
cookie

Figure 6. Mechanism to generate self-protecting behavior – locally retrieve, decrypt and display upon successful re-logon with password

The operation sequence illustrated in Figure 6 for retrieving and decrypted the web mail page is summarized below:

1. User enters password to unlock page
2. Applet retrieves $K_{encrypted\ private\ key}$ from session cookie and decrypt it with user password to get $K_{private}$.
3. Applet retrieves encrypted content from session cookie and decrypt it with $K_{private}$
4. Display original web page.

In the above example, we achieved storing the mail content locally and avoid security exposure by encrypting the information using a public/private key mechanism together with the user password. The generation of the encrypted private key $K_{encrypted\ private\ key}$ using the user password guarantees that the original key $K_{private}$ is securely stored locally. The original private key can be readily recovered when a user enters the password to unlock the page.

The important features illustrated by this example are the time-out and security functions that are provided by the attached applet agent in the html page. In general, different applets can be attached to different html pages depending on their data content and specific needs. It is also important to note that information is securely stored locally to satisfy the requirement that the user can still retrieve the locked page even where there is no server connection. This is achieved by using a double encryption technique using the pubic/private keys and the user password as the main entrance key.

Conclusions

In this report, we have described an implementation of an agent embedded in web content which provides a self-protection function for a delivered page without a browser plug-in even after server connection is disconnected. We have also described a double encryption scheme which allows secured mail content to be stored locally at the client side to avoid security exposure. The methodology described in this report provides a convenient way of adding autonomous functions[4] to web data contents at the client side, especially in situations where specialized plug-ins are not available. This approach and concept can be extended to enable other self-managing functions and personalized behaviors for web content using dynamic attachment of agents depending on data content, user information and other environment criteria in a client/server distributed environment. Despite many of its good points, this approach has its drawbacks, namely, its limitation by the size of the applet and the complexities of maintaining a repository of available applets.

References

[1] Intelligent Agents: Theory and Practice 12/2/99 - Mike Wooldridge and Nick Jennings, Intelligent Agents: Theory and Practice, Knowledge Engineering Review, v10n2, June 1995.

[2] Agent-Based Engineering, the Web, and Intelligence – Charles J. Petrie. December 1996 issue of IEEE Expert.

[3] Intelligent Agents in Cyberspace - 1999 AAAI Spring Symposium, http://www.aaai.org/Press/Reports/Symposia/Spring/ss-99-03.html.

[4] Jeff O. Kephart, David M. Chess, "The Vision of Autonomic Computing", Computer Journal, IEEE Computer Society, January 2003 issue

[5] Java Plug-in Component - http ://java.sun.com/j2se/1.4.2/docs/guide/plugin/

[6] Applet Resources - http://java.sun.com/applets/

[7] Java-enabled browsers - http://physics.syr.edu/courses/java/browsers.html

[8] JavaScript to Java Communication – http://java.sun.com/j2se/1.4.2/docs/guide/plugin/developer_guide/js_java.html

[9] LiveConnect/Plug-in Developer's Guide http://wp.netscape.com/eng/mozilla/3.0/handbook/plugins/

[10] JSObject – http://wp.netscape.com/eng/mozilla/3.0/handbook/plugins/doc/netscape.javascript.JSObject.html

[11] DOM – Document Object Model http://www.w3.org/DOM/

[12] Public Key Cryptography - http://en.wikipedia.org/wiki/Public_key

[13] Public Key Cryptography http://www.verisign.com/repository/crptintr.html

[14] Browser Session Cookie http://www.mach5.com/support/analyzer/manual/html/

On Selective Result Merging in a Metasearch Environment

Elizabeth D. Diaz, Arijit De, Vijay V. Raghavan
Center of Advanced Computer Studies,
University of Louisiana, Lafayette.
{edd8747, axd9142, raghavan}@louisiana.edu

Abstract

When a query is passed to multiple search engines, each search engine returns a ranked list of documents. The problem of metasearch is to fuse these ranked lists such that optimal performance is achieved because of the combination. There are two parts of the metasearch process. The first part is to select which search engine results are to be merged. The second part is the actual process of merging. In this paper we propose (1) a strategy for selective merging of results from a metasearch engine and (2) a heuristic for handling missing documents in result sets to be merged to improve the result-merging process Our experimental results will show that our proposed strategy for selection before merging coupled by incorporating it into the BORDA method improves the performance of merging.

1. Introduction

A metasearch engine is a system that supports unified access to multiple existing search engines. When a user submits a query to the metasearch engine, it selects a few promising search engines, from a larger set of underlying search engines, to which it will dispatch the query.

The search engines return results in the form of ranked lists. The metasearch engine extracts and selects results from the returned ranked lists, and merges the selected results into a single ranked list. In short, the metasearch engine needs to select search engines whose results (represented as ranked lists) with respect to a certain query need to be merged.

In this paper, we provide a strategy to select search engines whose results need to be merged. Our proposed strategy for selecting search engine results revolves around distances between ranked lists. Given a finite set of ranked lists, we perform computations such as distance among rankings and clustering of rankings. The results of these computations are clusters of rankings with respect to a given query. We utilize distance functions introduced in [6] for the clustering of search engine results in order to perform selective merging of rankings. The motivation for selecting search engines based on distances, is to ensure that we merge results that rank documents differently. In this way we are able to combine the diverse opinions of various search engines with respect to the way in which they rank documents. Another motivation is to remove redundancy between search engines at the time of merging.

A missing document is a document that has been retrieved by some search engines but not by all. A document might be missing from a ranked list if the search engine does not retrieve it, if the search engine does not index it or if the search engine does not cover the document. Our research focuses on the need to come up with one or more heuristic by which we can compute the position of each missing document in the ranked list where it is missing. By doing so we can insert missing documents into the ranked list and thereby obtain a more homogenous environment for merging.

We also focus our attention on comparing three new heuristics for handling missing documents in ranked lists that are returned by a search engine in response to a given query. In this paper we propose three heuristics (H1, H2 and H3) to handle missing documents.

The data sets used in our experiments were the TREC datasets, TREC 3, TREC 5, TREC 9 and Vogt. We used recall-based precision as the measure for comparing the effects that the three heuristics and the selection strategy had. Our strategy for selection and heuristics for handling missing documents were used in conjunction with the BORDA method.

As part of our experiments, we compared the performance when merging pre-selected search engines (based on our proposed selection strategies) using the method where missing documents were handled based on our proposed heuristics to the simple BORDA method based on the model proposed by Aslam and Montague [1]. BORDA with selection and missing document heuristics perform significantly better than the simple BORDA. Figure 1 is a block diagram of the representation of the metasearch process as envisioned by us.

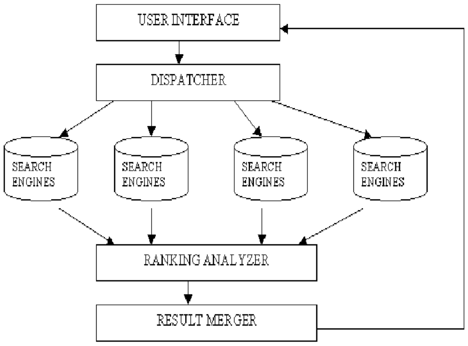


Figure 1: Block Diagram of the Metasearch process

The user interface captures the query from the user and the dispatcher sends the query to a series of search engines. Each search engine returns the results of the query in the form of ranked lists. These ranked lists are passed to the ranking analyzer. The ranking analyzer employs the selection strategy proposed by us to select the search engines whose results need to be merged. The ranked lists from these search engines are passed into the result merger. Missing document heuristics are applied in the result merger while the ranked lists are merged into a single final ranked list that is returned back to the user.

2. Related Work

Researchers and scientists working in the field of metasearch and distributed information retrieval have explored data fusion techniques for result merging. Thus a number of data fusion based models have been developed. To test the effects of our heuristics for handling missing document and strategy for selection of search engines before merging we use the basic BORDA model. In this section, we describe the BORDA model as proposed by Aslam and Montague [1].

2.1. Borda-Fuse Model & Weighted Borda-Fuse

Aslam and Montague proposed two models [1]. The first one is called Borda-Fuse model and it is based on a political election strategy named Borda Count. The Borda-Fuse works by assigning points to each document in each one of the lists to be merged. The number of points per documents depends on the rank position of the document, i.e.,, for a list of n ranked documents, the top document receives n points, the next document receives n – 1 and so on. The points assigned for a given document by different search engines are added up and the documents are ranked from highest to lowest according to the sum. The Borda count for document d_i is $\sum_k (n - r_{ik})$ where n is the number of documents and r_{ik} is the rank position of document d_i under search engine *k*. This model does not require training data or the RSVs and its algorithm is simple and effective. It has been shown that the Borda Count is optimal [7, 8] when compared to standard voting methods. However, in [4], it had been demonstrated that the Borda has limitations with respect to the Condorcet Principle, Condorcet Order and the Increasing and Decreasing Principles.

Their second method, the Weighted Borda-Fuse, is a weighted version of the Borda-Fuse. A weight w_i is assigned to the i^{th} search engine according to their performance. Weighted Borda-Fuse requires training to determine the best weights for the performance of the search engines. This method was shown to perform better than the Borda-Fuse.

3. Handling Missing documents

In this section, we describe a heuristic for handling missing documents. First we define the concept of positional values.

3.1. Positional Values

Positional Value: The positional value (PV) of a document d_i in the resulting list l_k returned by a search engine s_k is defined as $(n - r_{ik} + 1)$ where r_{ik} is the rank of d_i in search engine s_k and n is the total number of documents in the result.

3.2. Case of Missing Documents

Let PV_i be the positional values for a document d in the ith search engine. Let m be the total number of search engines. Let r be the number of search engines in which d appears. Let j denote a search engine not among the r search engines where d appears. Our heuristics are:

H1: For all j, $PV_j = \frac{\sum_{i=1}^{r} PVi}{r}$, i.e., average of the positional values of the document in the r search engines.

H2: For all j, $PV_j = \frac{\sum_{i=1}^{m} PVi}{m}$, i.e., the PV_j is the average of the positional values of the document in the m search engines where d appears.

H3: For all j, $PV_j = \min\{PV_i\}$ where $1 \le i \le r$, i.e., the minimum of the positional values of the document among the r search engines where d appears.

4. Proposed selection strategy

In the previous sections, we stated the heuristic for missing documents. Handling missing documents is an important part of the metasearch environment. However to improve the effectiveness of metasearch we can pre-select the search engines whose results we need to merge based on some strategies. In this section, we discuss our proposed approaches for selecting search engines.

4.1. Strategy of merging without selection

As the title suggests in this strategy we select search engines randomly. There is no specific strategy for selection.

4.2. Strategy of selective merging

In this section, we propose a method for selecting search engines based on the distances between the ranked lists obtained for a specific query. Distance computing measures are discussed in the next section. After distances are computed, we propose merging the search engine pair that has the maximum distance first. We call this "Farthest SE-Pair First" strategy. The rational behind, selecting the Search Engines that are farthest apart first, is to ensure that we merge results that rank documents differently. Thereby we are able to add variation to the results that are being merged.

4.3 Distance measures

To employ the "Farthest SE-Pair First" strategy we need to compute distances between the ranked lists returned for a given query by various search engines. By doing so we can measure the distances between search engines in the context of a particular query. We use the distance function proposed in [6] with some slight modification. Let R_1 and R_2 be two rankings. Let Δ_1 be the document set for ranking R_1. Let Δ_2 be the documents set for ranking R_2. Let $\Delta_3 = \Delta_1 \text{ I } \Delta_2$. Suppose that Ψ_1 and Ψ_2 are rankings of Δ_3.

If D and D' are in Δ_3 then the function is defined as

$$\delta_{\Psi_1,\Psi_2}(D,D') = \begin{cases} 0 & \rightarrow agree \\ 1 & \rightarrow onerankshigher, othertie \\ 2 & \rightarrow inverted \end{cases}$$

The ranking distance between Ψ_1 and Ψ_2, $d(\Psi_1,\Psi_2)$ can be defined by the expression

$$\frac{1}{|\Delta_3|(|\Delta_3|-1)} \sum_{(D,D')\in\Delta_3} \delta_{\psi 1,\psi 2}(D,D')$$

Once the distances have been calculated, the distance matrix can be defined.

Example: Calculating distance among rankings. Suppose we have 2 ranking list Ψ_1, Ψ_2, from two different search engines. These two rankings have been pretreated with some kind of heuristic (H1, H2, H3).

$$\Psi_1 = \begin{bmatrix} d_2 \\ d_1 \\ d_3 \\ d_4 \\ d_5 \end{bmatrix}, \Psi_2 = \begin{bmatrix} d_2 \\ d_5 \\ d_3 \\ d_1 \\ d_4 \end{bmatrix}$$

$\delta(d_2, d_1) = 0$, $\delta(d_2, d_3) = 0$,
$\delta(d_2, d_4) = 0$, $\delta(d_2, d_5) = 0$,

$\delta (d_1, d_3) = 2$, $\delta (d_1, d_4)= 0$,
$\delta (d_1, d_5) = 2$, $\delta (d_3, d_4)= 0$,
$\delta (d_3, d_5) = 2$, $\delta (d_4, d_5)= 2$,

$$d(\Psi_1, \Psi_2) = \frac{8}{5*4} = 0.4$$

4.4. Selection based on maximum distances: The “Farthest SE-Pair First” strategy

In this section, we provide an algorithm for the selection strategy mentioned in section 4.2.

Input: A set of search engines S. S = {$SE_i \mid \forall i$; 1 ≤ i ≤ n and n is the number of search engines underlying the meta search engine}.
Two sets PICKED and NPICKED (non picked).
PICKED = { $SE_i \mid SE_i$ ε S & SE_i has been selected for merging }. NPICKED = { $SE_i \mid SE_i$ ε S & SE_i has not been selected for merging }.
Distance matrix: DM is a matrix that contains the distance between the search engines based on the ranked lists obtained by querying it.
DM = { DM(i, j) | DM(i,j) is the distance between SE_i and SE_j }.
Number of search engines to be merged: k
Output: The set PICKED.

Stopping Condition: When the | PICKED | = k where | PICKED | is the size of the set PICKED.

Algorithm: Initially PICKED = ϕ. NPICKED = S. Select the element DM(i,j) of DM where DM(i,j) is maximum.
Select SE_i and SE_j for merging and place them in PICKED.
For each search engine SE_i in set PICKED, access its distance to every search engine SE_p that is in set NPICKED by referring to distance matrix, DM, and pick the DM(i,p) that has the maximum value over all i and p values. Remove SE_p from NPICKED and add it to picked. Ties are broken arbitrarily.
Repeat step 4 until the size of set PICKED is k.

5. Experiments

In this section, we describe the various aspects of our experiments.

5.1. Objectives

The objectives of our experiments were to (1) study the effect of our selection strategy on the BORDA method and (2) to explore how the heuristics for handling missing documents affected the performance of the BORDA method when used in conjunction with or without the selection strategy.

5.2. Procedure

In this section we describe experimental procedures for merging when no selection strategy is employed and when our proposed selection strategy is employed.

5.2.1 Experimental setup for “Randomly Select Search Engines”

Input: (1)A set of queries Q numbered 1 through n where n is the number of queries. (2) A set of search engines S. (3) Missing document heuristic to be applied. (4) Dataset to be used e.g., TREC3, TREC5 and TREC9.
Output: Average precision obtained by merging ranked list using BORDA.
Procedure: (A) For each query q in Q do the following procedure.
(1)Initialize a two dimensional matrix A[11]. Each element represents RB-precision values for a ranked list obtained by merging a certain number of search engines. Thus each element holds the value of RB-precision for a ranked list obtained when using a specific method to merge a specific number of search engines.

(2)Varying m from 2 through 12 do the following (a) pick m search engines randomly (b)pass the query q to the m search engines picked randomly and obtain results in the form of ranked lists.(c) Merge these ranked lists into one list using each of the methods BORDA, obtaining a single merged list called RBORDA. (d) Compute RB-precision for each of the merged lists for Recall values of 0.25, 0.5, 0.75 and 1.00. Average the RB-precision values thus obtained to consolidate them into a single average value. Thus we obtain a single value of precision, PRBORDA, for the list RBORDA . Let A[m-1]= PRBORDA. (e) Accumulate over m. Repeat steps a through d 50 times and average out the results.

(B) Accumulate over queries and then average by the number of queries.

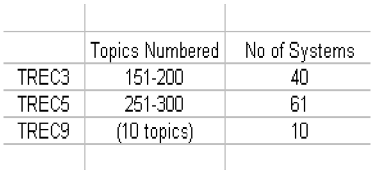
5.2.2 Experimental setup for "Farthest SE-Pair First"

The experimental procedure followed was the same. However instead of selecting search engines randomly (as in A(2).a), in this case we select search engines based on the strategy proposed in section 4.4. Thus, for a given m, only single search engines are considered.

5.3. Implementation

Our experiments were done using programs written in Visual Basic programming language that queried a Microsoft Access Database that held the data exported from TREC 3, TREC 5 and TREC 9 datasets.

For each experiment, we need to input the method name (in our case we have only one method), the strategy for selection and heuristic for selection.



	Topics Numbered	No of Systems
TREC3	151-200	40
TREC5	251-300	61
TREC9	(10 topics)	10

Table 1(a): Description of Data Sets.

5.4. Data Sets

Table 1 shows the particulars of the data sets TREC 3, TREC 5 TREC 9 and Vogt that are used. Each of the data sets has a specified number of systems that return up to 1000 documents when queried with a certain topic. There are 50 topics in each of the data sets. Each topic is analogous to a query and each system is analogous to a search engine. Thus, topics (queries) are passed onto a system (search engine). The search engines then return a set of documents in the form of a ranked list. Each document is either relevant (represented by 1), highly relevant (represented by 2) or irrelevant (represented by 3).

The comparative results of various experiments were tabulated. Each column in the represents a set of results obtained for a specific experimental case. Table 1(b) shows the symbols used in column headings in the tables to describe the experiments.

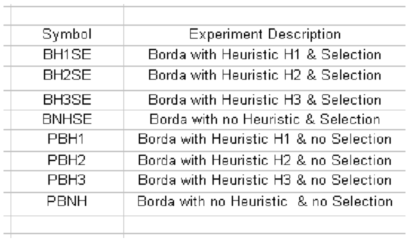
5.5. Performance Metrics

Our metric for measuring performance is Recall Based Precision. The detailed theory of Recall Based (RB) precision can be found in [5]. Recall based precision is used in case when there are a series of documents ranked in partial order and we need to find out the precision for various levels of recall.

The formula is shown below

$$\frac{x*n}{x*n+j+s*\frac{i}{r}}$$

where
(1) x is one of the standardized recall values i.e., 0.25, 0.5, 0.75, etc; (2) n is the number of relevant documents in the collection; (3) s is the number of relevant document wanted; (4) i is the number of irrelevant documents in the final rank; (5) r is the number of relevant documents in the final rank; (6) j is the number of irrelevant documents to get to s documents.
In this context, final rank is defined as the rank containing or completing the number of relevant documents as specified by s.



Symbol	Experiment Description
BH1SE	Borda with Heuristic H1 & Selection
BH2SE	Borda with Heuristic H2 & Selection
BH3SE	Borda with Heuristic H3 & Selection
BNHSE	Borda with no Heuristic & Selection
PBH1	Borda with Heuristic H1 & no Selection
PBH2	Borda with Heuristic H2 & no Selection
PBH3	Borda with Heuristic H3 & no Selection
PBNH	Borda with no Heuristic & no Selection

Table 1(b): Symbols describing experiments.

5.6.Comparison of missing document heuristics.

In this set of experiments, we have two cases
In case 1, we compare the performance of the BORDA algorithm, when each of the three heuristics for missing documents (H1, H2 and H3) proposed are applied in conjunction with the selection strategy.

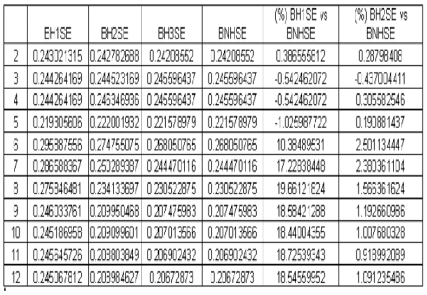
In case 2, we also compare the performance of the BORDA algorithm, when each of the three heuristics for missing documents (H1, H2 and

H3) proposed are applied without any the selection strategy.

5.6.1 Case 1

Tables 2(a), 2(b), 2(c) shows the performance of the BORDA algorithm for the data sets TREC 9, TREC 5, and TREC 3. The first column shows the number of search engines being merged. The second, third and fourth columns named BH1SE, BH2SE, BH3SE and show results when heuristic H1, H2 and H3 are applied. The fourth column named BHNSE represents results of experiments in which the selection strategy was employed but no heuristics was applied. Column 9,10,11 shows the improvement effects of heuristic H1, H2 and H3, respectively, in comparison to the case when no heuristic was applied.

TREC 9: Table 2(a) shows the results for TREC 9. From the table 2(a) the following observations can be drawn up: (1) Heuristic H1 improved upon the case of no heuristic by up to 18% in some cases. (2) Heuristic H2 improved by 2.5% in some cases. (3) Notice that Heuristic H3 did not effect the merging performance. The results are almost identical for the cases BNHSE and BH2SE (4). For each of the cases in which a heuristic is used and the case in which no heuristic is used the performance measure seem to go down as we vary the number of search engines from 2 to 5. Then the performance improves as we vary the number of search engines from 5 to 6. Beyond that if the number of search engines is increased the performance goes down.



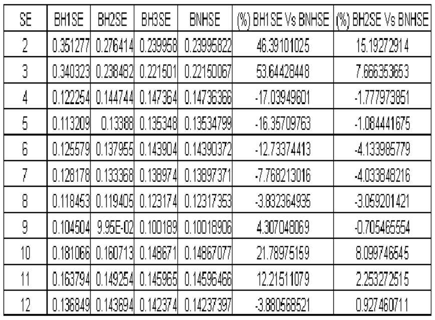
	BH1SE	BH2SE	BH3SE	BNHSE	(%) BH1SE vs BNHSE	(%) BH2SE vs BNHSE
2	0.243021315	0.242782688	0.24208552	0.24208552	0.386555812	0.28798408
3	0.244264169	0.244523169	0.245596437	0.245596437	-0.542462072	-0.437004411
4	0.244264169	0.245346936	0.245596437	0.245596437	-0.542462072	0.305582546
5	0.219305606	0.222001932	0.221578979	0.221578979	-1.025987722	0.190881437
6	0.295387556	0.274755075	0.268050765	0.268050765	10.38489531	2.501134447
7	0.286588367	0.250289387	0.244470116	0.244470116	17.22838448	2.380361104
8	0.275846481	0.234133697	0.230522875	0.230522875	19.66121624	1.566361624
9	0.246033761	0.209950468	0.207475983	0.207475983	18.58421288	1.192660986
10	0.245186958	0.209099601	0.207013566	0.207013566	18.44004355	1.007680328
11	0.245645726	0.208803849	0.206902432	0.206902432	18.72539343	0.918992089
12	0.245067812	0.208984627	0.20672873	0.20672873	18.54559952	1.091235486

Table 2(a): Comparing performance of Heuristics when selection strategy is employed for TREC 9

TREC 5: Table 2(b) shows the results for TREC 5. From the table 2(b) the following observations can be drawn up: (1) Heuristic H1 performs best effecting the performance of metasearch by about 46% in some cases. (2) The effect of Heuristic H2 once again is somewhat limited at about 7-8% (3) As in case of TREC 9 applying heuristic H3 has the same effect as applying no heuristic at all. The results are almost identical for the cases BNHSE and BH2SE (4) Overall performance tends to decrease with the increase in number of search engine results being merged till about 8 search engines after which the performance improves.

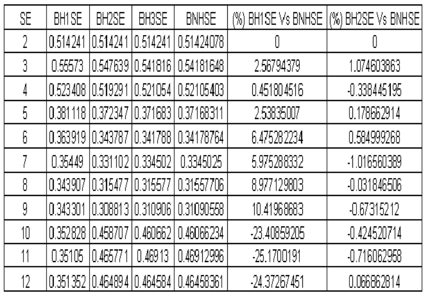
Heuristic H1 is most effective in improving performance of the merging algorithm. Heuristic H2 is less effective and H3 has no effect on the merging algorithm at all.

TREC 3: Table 2(c) shows the results for TREC 3. Our observations were similar to TREC 5. (1) Heuristic H1 performs fairly well when the number of search engines being merges is less that 10. (2) Heuristic H2 effects the performance nominally. In certain cases the effect is adverse and in some case the effect is positive. (3) Applying heuristic H3 has the same effect as applying no heuristic at all. (5) Overall performance tends to decrease with the increase in number of search engine results being merged until about 8 search engines after which the performance improves.



SE	BH1SE	BH2SE	BH3SE	BNHSE	(%) BH1SE Vs BNHSE	(%) BH2SE Vs BNHSE
2	0.351277	0.276414	0.239958	0.23995822	46.39101025	15.19272914
3	0.340323	0.238482	0.221501	0.22150067	53.64428448	7.666353653
4	0.122254	0.144744	0.147364	0.14736366	-17.03949601	-1.777973851
5	0.113209	0.13388	0.135348	0.13534799	-16.35709763	-1.084441675
6	0.125579	0.137955	0.143904	0.14390372	-12.73374413	-4.133985779
7	0.128178	0.133368	0.138974	0.13897371	-7.768213016	-4.033848216
8	0.118453	0.119405	0.123174	0.12317353	-3.832364935	-3.059201421
9	0.104504	9.95E-02	0.100189	0.10018906	4.307048069	-0.70546554
10	0.181066	0.160713	0.148671	0.14867077	21.78975159	8.099746545
11	0.163794	0.149254	0.145965	0.14596466	12.21511079	2.253272515
12	0.136849	0.143694	0.142374	0.14237397	-3.880568521	0.927460711

Table 2(b): Comparing performance of Heuristics when selection strategy is employed for TREC 5

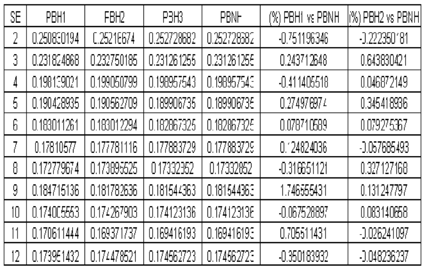


SE	BH1SE	BH2SE	BH3SE	BNHSE	(%) BH1SE Vs BNHSE	(%) BH2SE Vs BNHSE
2	0.514241	0.514241	0.514241	0.51424078	0	0
3	0.55573	0.547639	0.541816	0.54181648	2.56794379	1.074603863
4	0.523408	0.519291	0.521054	0.52105403	0.451804516	-0.338445195
5	0.381118	0.372347	0.371683	0.37168311	2.53835007	0.178662914
6	0.363919	0.343787	0.341788	0.34178764	6.475282234	0.584999268
7	0.35449	0.331102	0.334502	0.3345025	5.975288332	-1.016560389
8	0.343907	0.315477	0.315577	0.31557706	8.977129803	-0.031846506
9	0.343301	0.308813	0.310906	0.31090558	10.41968683	-0.67315212
10	0.352828	0.458707	0.460662	0.46066234	-23.40859205	-0.424520714
11	0.35105	0.465771	0.46913	0.46912996	-25.1700191	-0.716062958
12	0.351352	0.464894	0.464584	0.46458361	-24.37267451	0.066862814

Table 2(c): Comparing performance of Heuristics when selection strategy is employed for TREC 3

5.6.2. Case 2

Table 3(a), 3(b), 3(c) shows the performance of the BORDA algorithm for the data sets TREC 9, TREC 5, and TREC 3. The first column shows the number of search engines being merged The second, third and fourth columns named PBH1, PBH2, PBH3 and show results when heuristic H1, H2 and H3 are applied. The fourth column named PBNH represents results of experiments in which the selection strategy was employed but no heuristics was applied. Column 9,10,11 shows the improvement effects of heuristic H1, H2 and H3, respectively, in comparison to the case when no heuristic was applied.

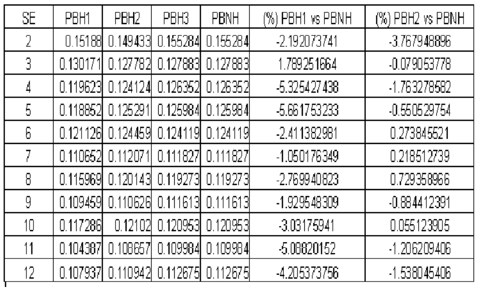


SE	PBH1	PBH2	PBH3	PBNH	(%) PBH1 vs PBNH	(%) PBH2 vs PBNH
2	0.250830194	0.25216674	0.252728682	0.252728682	-0.751196346	-0.222350181
3	0.231824868	0.232750185	0.231261255	0.231261255	0.243712648	0.643830421
4	0.198139021	0.199050799	0.198957543	0.198957543	-0.411405518	0.046872149
5	0.190428935	0.190562709	0.189906735	0.189906735	0.274976974	0.345418936
6	0.183011261	0.183012294	0.182867325	0.182867325	0.078710589	0.079275367
7	0.17810577	0.177781116	0.177883729	0.177883729	0.124824036	-0.057685493
8	0.172779674	0.173895525	0.17332352	0.17332852	-0.316651121	0.327127168
9	0.184715136	0.181782636	0.181544363	0.181544363	1.746555431	0.131247797
10	0.174005553	0.174267903	0.174123136	0.174123136	-0.067528897	0.083140658
11	0.170611444	0.169371737	0.169416193	0.169416193	0.705511431	-0.026241097
12	0.173951432	0.174478521	0.174562723	0.174562723	-0.350183932	-0.048236237

Table 3(a): Comparing performance of Heuristics when no selection strategy (random selection) is employed for TREC9

TREC 9: Table 3(a) shows the results for TREC 9. Table 3(a) show how each missing document heuristic effects the performance when no selection strategy is employed before merging.
From the table, we clearly observe that heuristic H1 and H2 have only slight positive effect on the process of merging. Heuristic H3 has virtually no effect on performance.

TREC 5: Table 3(b) shows the results for TREC 5. Table 3(b) show how each missing document heuristic effects the performance when no selection strategy is employed before merging. In case of TREC 5 the performance is adversely effected when missing documents are handled using heuristic H1 and H2. In case of H1 the effect is as significant as 5% in some cases. In case of heuristic H2 the effect is almost negligible.



SE	PBH1	PBH2	PBH3	PBNH	(%) PBH1 vs PBNH	(%) PBH2 vs PBNH
2	0.15188	0.149433	0.155284	0.155284	-2.192073741	-3.767948896
3	0.130171	0.127782	0.127883	0.127883	1.789251664	-0.079053778
4	0.119623	0.124124	0.126352	0.126352	-5.325427438	-1.763278582
5	0.118852	0.125291	0.125984	0.125984	-5.661753233	-0.550529754
6	0.121126	0.124459	0.124119	0.124119	-2.411382981	0.273845521
7	0.110652	0.112071	0.111827	0.111827	-1.050176349	0.218512739
8	0.115969	0.120143	0.119273	0.119273	-2.769940823	0.729358966
9	0.109459	0.110626	0.111613	0.111613	-1.929548309	-0.884412391
10	0.117286	0.12102	0.120953	0.120953	-3.03175941	0.055123905
11	0.104387	0.108657	0.109984	0.109984	-5.08820152	-1.206209406
12	0.107937	0.110942	0.112675	0.112675	-4.205373756	-1.538045406

Table 3(b): Comparing performance of Heuristics when no selection strategy (random selection) is employed for TREC 5

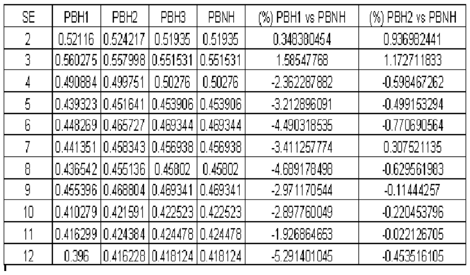
TREC 3: Table 3(c) shows the results for TREC 3. Table 3(c) show how each missing document heuristic effects the performance when no selection strategy is employed before merging. Results are almost identical to that of TREC 5.

5.7. Comparison of BORDA with and without selection strategy.

In this set of experiments, we compare the performance of the BORDA method when we employ a selection strategy before merging results to the performance of the BORDA method where no prior selection is done. In this case, no heuristics are employed for handling missing documents.

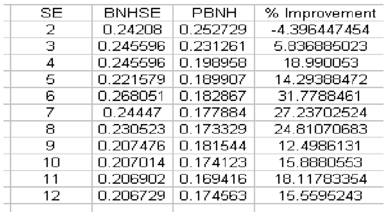
TREC 9: Table 4(a) shows the results when merging with and without selection. In this comparison, we do not apply any heuristics for handling missing documents. The performance is significantly better when our selection strategy is employed. The improvements when 6 search engines are merged are about 31%. Table 4(a) shows the improvements.

TREC 5: Table 4(b) shows the results for TREC 5. In this comparison, we do not apply any heuristics for handling missing documents. The performance is significantly better when our selection strategy is employed. In the best case, improvement is up to 35%. On the average 20% improvement is observed. Table 4(b) shows the improvements.



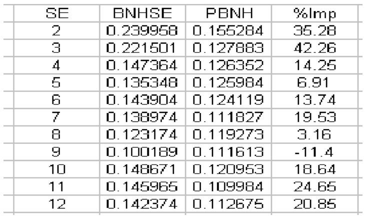
SE	PBH1	PBH2	PBH3	PBNH	(%) PBH1 vs PBNH	(%) PBH2 vs PBNH
2	0.52116	0.524217	0.51935	0.51935	0.348380454	0.936982441
3	0.560275	0.557998	0.551531	0.551531	1.58547768	1.172711833
4	0.490884	0.499751	0.50276	0.50276	-2.362287882	-0.598467262
5	0.439323	0.451641	0.453906	0.453906	-3.212896091	-0.499153294
6	0.448269	0.465727	0.469344	0.469344	-4.490318535	-0.770690564
7	0.441351	0.458343	0.456938	0.456938	-3.411257774	0.307521135
8	0.436542	0.455136	0.45802	0.45802	-4.689178498	-0.629561983
9	0.455396	0.468804	0.469341	0.469341	-2.971170544	-0.11444257
10	0.410279	0.421591	0.422523	0.422523	-2.897760049	-0.220453796
11	0.416299	0.424384	0.424478	0.424478	-1.926864653	-0.022126705
12	0.396	0.416228	0.418124	0.418124	-5.291401045	-0.453516105

Table 3(c): Comparing performance of Heuristics when no selection strategy (random selection) is employed for TREC 3



SE	BNHSE	PBNH	% Improvement
2	0.24208	0.252729	-4.396447454
3	0.245596	0.231261	5.836885023
4	0.245596	0.198958	18.990053
5	0.221579	0.189907	14.29388472
6	0.268051	0.182867	31.7788461
7	0.24447	0.177884	27.23702524
8	0.230523	0.173329	24.81070683
9	0.207476	0.181544	12.4986131
10	0.207014	0.174123	15.8880553
11	0.206902	0.169416	18.11783354
12	0.206729	0.174563	15.5595243

Table 4(a): Selection vs no (random) selection for TREC 9



SE	BNHSE	PBNH	%Imp
2	0.239958	0.155284	35.28
3	0.221501	0.127883	42.26
4	0.147364	0.126352	14.25
5	0.135348	0.125984	6.91
6	0.143904	0.124119	13.74
7	0.138974	0.111827	19.53
8	0.123174	0.119273	3.16
9	0.100189	0.111613	-11.4
10	0.148671	0.120953	18.64
11	0.145965	0.109984	24.65
12	0.142374	0.112675	20.85

Table 4(b): **Selection vs no (random) selection for TREC 5**

6. Conclusions

In our paper we have dealt with two problems pertaining to merging of results in the context of metasearch. The first one pertained to missing documents and second one was pertaining to selection of search engines that need to be queried before merging. We proposed three heuristics for handling missing documents and a strategy for selecting search engines to be merged based on the distances between the ranked lists of results they returned for certain queries. When merging search engines at random, our heuristics for handling missing documents had no effect on the performance of the merged list. However when applied in conjunction with the selection strategy the average precision of the resulting merged list was greatly improved. Selection before merging applied independently of missing document heuristics also resulted in significant improvements.

7. References

[1] J. A. Aslam, M. Montague, Models for Metasearch, Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, New Orleans, Louisiana, United States, September 2001, pp. 276-284.

[2] W. Meng, C. Yu, K. Liu, Building Efficient and Effective Metasearch engines, ACM Computing Surveys, March 2002, pp. 48-84.

[3] W. Meng, C. Yu, K. Liu. A Highly Scalable and Effective Method for Metasearc*h*, ACM Transactions on Information Systems pp. 310-335, July 2001.

[4] J. R. Parker, Multiple Sensors, Voting Methods and Target Value Analysis, Computer Science Technical Report, 1998, February 1, 1998, University of Calgary, Laboratory for Computer Vision, pp. 615-06.

[5] P. Bollmann, V. V. Raghavan, G. S. Jung, and L. C. Shu. On probabilistic notions of precision as a function of recall. *Information Processing and Management*, Vol. 28:291--315, May-June 1992.

[6] V. Raghavan, H. Sever, On The Reuse Of Past Optimal Queries, Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, July 9-13, pp. 344-350,1995.

[7] F. Roberts, *Discrete Mathematical Models*, Prentice Hall, Inc., 1976.

[8] F. Zachary, Lansdowne, Outranking Methods for Multicriterion Decision Making: Arrow's and Raynaud's Conjecture"; Social Choice and Welfare; Vol. 14, No. 1; January, 1997; 125-128; #2431.

Adaptation and Personalization in Web-based Learning Support Systems

Lisa Fan
Department of Computer Science, University of Regina, SK, Canada, S4S 0A2
fan@cs.uregina.ca

Abstract

In order to achieve optimal efficiency in a learning process, individual learner needs his/her own personalized assistance. For a web-based open and dynamic learning environment, personalized support for learners becomes more important. This paper demonstrates how to realize personalized learning support in dynamic and heterogeneous learning environments by utilizing Adaptive Web technologies. We focus on course personalization in terms of contents and teaching materials that is according to each student's needs and capabilities. To accomplish this, a conceptual model based on the Knowledge Structure is presented. Using the hierarchy and association rules of the concepts, we can organize courses and lessons as a multi-layer knowledge network, which has a reasonable classification and interdependent relations among the knowledge. With retrieval based on concept and association among the concepts, we propose a framework of knowledge structure based visualization tool for representing a dynamic learning process to support students' deep learning, efficient tutoring and collaboration in web-based learning environment.

1. Introduction

Web based learning offers many benefits over traditional learning environments and has becoming very popular. The web is a powerful environment for distributing information and delivering knowledge to an increasingly wide and diverse audience. Typical web-based learning environments, such as Web-CT [1], Blackboard [2], include course content delivery tools, quiz module, grade reporting systems, assignment submission components, etc. They are powerful integrated learning management systems (LMS) which support a number of activities performed by teachers and students during the learning process [3]. Students who study a course on the Internet tend to be more heterogeneously distributed than those found in a traditional classroom situation. Therefore, the learning material should be presented in a more personalized way.

In a web-based learning environment, instructors provide online learning material such as text, multimedia and simulations. Learners are expected to use the resources and learning support tools provided. However, it is difficult and time consuming for instructors to keep track and assess all the activities performed by the students on these tools [4]. Moreover, due to the lack of direct interaction between the instructor and the students, it is hard to check the students' knowledge and evaluate the effectiveness of the learning process. When instructors put together the learning material (such as class notes, examples, exercises, quizzes, etc) on-line, they normally follow the curriculum and pre-design a learning path for the students, and assume that all the learners would follow this path. Often this path is not the optimal learning sequence for individual learners, and does not satisfy the learner's individual learning needs. This is typically the teacher-centered "one size fits all" approach.

Not all students have the same ability and skills to learn a subject. Students may have different background knowledge for a subject, which may affect their learning. Some students need more explanations than others. Other differences among students related to personal features such as age, interests, preferences, emotions, etc. may also affect their learning [5]. Moreover, the results of each student’s work during the learning session must be taken into account in order to select the next study topics to the student [6].

By utilizing adaptive web technologies, particularly, Adaptive Educational Hypermedia (AEH) systems it is possible to deliver, to each individual learner, a course offering that is tailored to their learning requirements and learning styles [7]. These systems combine ideas from hypermedia and intelligent tutoring systems to produce applications that adapt to meet individual educational needs. An AEH system dynamically collects and processes data about student goals, preferences and knowledge to adapt the material being delivered to the educational needs of the students [8]. Since the learning process is influenced by many factors, including prior knowledge, experience, learning styles and preferences, it is important that the student model of an AEH system accommodates such factors in order to adapt accurately to student needs.

In this paper, we first provide an overview of concepts and techniques used in adaptive web-based learning support systems. Then we discuss and examine the use of student individual differences as a basis of adaptation in web-based learning support systems. This paper proposes a framework of knowledge structure based visualization tool for representing a dynamic and personalized learning process to support students' deep learning.

2. An Overview and the Basic Architecture of Web - based Adaptive Educational Hypermedia Systems

Adaptive hypermedia systems is one step forward of hypermedia-based systems. It combined the technologies of adaptive systems and hypermedia. The main purpose is to improve the usability of traditional hypermedia through the integration of intelligent techniques. It enables a system to arrange and present customized information and dynamic navigation support for Web-based learning material [8]. It has been shown in the literature that the efficient method of teaching is individualized teaching [9]. The ability to adapt and tailor the learning content to an individual learner's needs can significantly improve the teaching/learning process. Therefore, most of the developed adaptive hypermedia systems are applied in the field of education.

According to Brusilovsky [8], adaptive hypermedia systems can be defined as all hypertext and hypermedia systems that accommodate some user characteristics into the user model and apply this model to adapt various aspects of the system to the user. The major components of the system are the domain model, the user model and the ability to adapt the hypermedia using the user model (adaptation model). According to De Bra [10], an AHS builds a user model by observing the user's browsing behavior or by testing to determine what the user's background, experience, knowledge and interests are. These user characteristics are then used by the system to personalize the knowledge presentation. The presentation is adapted to the user model, and the user model is constantly updated as the user reads and interacts with the presentation. Figure 1 shows the classic loop of user modeling and adaptation in the system.

Adaptive hypermedia systems are the result of combining studies in the fields of hypermedia and user modeling. They represent an important research direction in adaptive systems based on user modeling [8]. Their main goal is to improve hypermedia functionalities through personalization. These systems not only allow users to browse and explore the learning material freely, but also are able to dynamically adapt the instructional sequence to the particular user knowledge level and learning goals. They provide intelligent guidance and support the user in acquiring knowledge [5]. Such system is able to adapt information and its presentation to each individual user's needs, and dynamically support the navigation through hypermedia material.

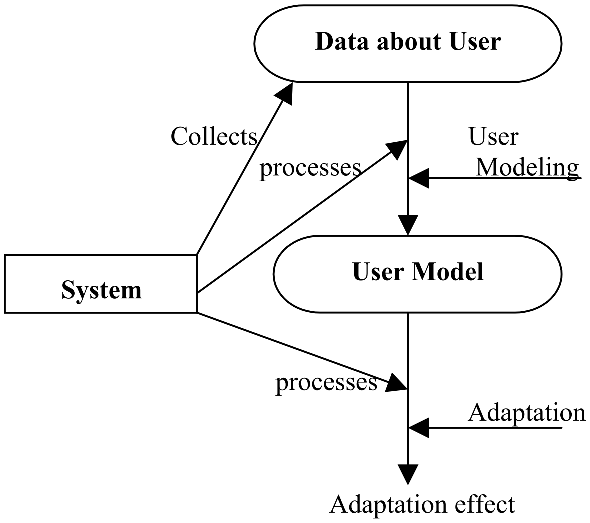


Figure 1. Classic "User modeling - adaptation" in adaptive systems [8]

Brusilovsky [5] presented the fundamental techniques and methods of how to achieve various adaptations. They are summarized as adaptive presentation and adaptive navigation support. Adaptive presentation is the techniques used to adapt the content of a web page based on user model. They include adaptive text presentation and adaptive multimedia presentation. Adaptive navigation support is the techniques used to modify the links accessible to the user at a particular time. They include: direct guidance, adaptive link sorting, adaptive link hiding, adaptive link annotation and map adaptation.

Web-based adaptive educational hypermedia (WBAEH) is one of the earliest and most popular applications of AHS. It is based on two earlier versions of adaptive educational systems: intelligent tutoring systems (ITS) and adaptive hypermedia systems (AHS). WBAEH actually combines two opposed teaching approaches: tutor-centered traditional AI based systems and the dynamic learner-centered browsing approach of hypermedia systems [11]. The basic components of the systems are domain model, the student model and the adaptation model. To be able to adapt itself to an individual student, the system has to be aware of the teaching domain, the individual students and their knowledge and has to monitor their learning progress. Therefore, beside adaptation model, the domain model and the student model are the most important parts of any adaptive system [12]. The relationship between all three models is shown in Figure 2.

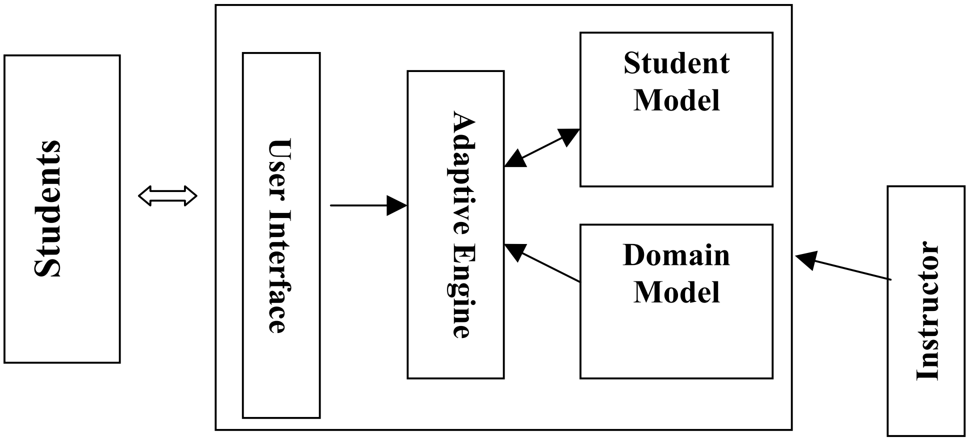


Figure 2. Architecture of Web-based Adaptive Educational System

2.1 Student model

The student model is the essential component in personalized learning. It is the student model that builds and maintains the system's understanding of the student. The learning process in a real adaptive educational environment is complex. A comprehensive student model should contain all features of the learner's behavior and knowledge that affect their learning and performance [13]. The adaptability of the system is influenced by the characteristics of the student contained in the student model. It is a complicated and challenging task to construct such a model containing all of student' features. In practice, most systems only consider those important and easy to implement features for use in design of the student model. Students' cognitive attributes such as emotions, social features have been ignored.

Table 1 shows some of the main student characteristics used for student modeling. The student model will build a student profile that stores information for each student. When constructing the student model, there are several things need to be considered: what characteristics of the student should be gathered into the model and how to collect them; how to represent the collected information in the system; and how to form and update the model. Information about the student in the student model can be categorized as domain independent and domain dependent, or divided into knowledge component and interfaces component; or as static and dynamic properties. Static properties of the student model [14] covers student's personal characteristics such as age, gender, background knowledge, student's preferences etc. These properties are normally gathered at the beginning of the learning process by using questionnaires and tests. The dynamic properties of the student model are the information about the student interaction with the system [15]. It includes student's knowledge, learning style, motivation, current goals, plans and believes, learning activities that have been carried out, objectives that have been achieved. The dynamic properties of the student model are collected from test results, problems solving behavior, browsing behavior, visited concepts, or time spent on page, total session time, navigation path, or searching for more information. This information is dynamically being gathered during the learning process and used for updating the student model. There are many different techniques can be used for constructing the student model [13]. Such as: Bayesian methods; Machine learning methods (rule learning, learning of probabilities, instance-based learning); Logic-based methods; Overlay methods; Stereotype methods.

Background	Experienced ; Non-experienced
Learning Style	Activist; Reflector; Theorist; Pragmatist (based on Honey-Mumford theory [15])
Knowledge	Novice; Intermediate; Expert;
Preference	Color; Size; Font
Goals	Global ; Local

Table 1. Student Model with individual differences

2.2 Domain model

The domain model contains a collection of learning materials that the adaptive hypermedia is intended to use as a resource. It supports composite concepts and concept relationships. The concepts are in a form of a hierarchical structure. For example, a course module in the system can be divided into a number of concepts with a particular value for each concept according to the level of difficulty. Each concept consists of several pages depending upon its complexity. The relationship between one concept and another can be expressed in terms of requirements such as pre-requisites. Students studying on a particular concept can undertake certain tests or quizzes at any time to evaluate their understanding for that concept. In order to move on to another higher-level course content, students must pass a threshold score that is defined in the relationship rules.

2.3 Adaptation model

The adaptation model describes the adaptation strategies. Most of the systems use rules to describe adaptation strategies. The adaptation rules specify how a page is presented to the student according to his/her own student model. Every time the student is assigned a score for a test/quiz, the student model will update his/her level of knowledge.

3. Adaptation and personalization support in WBLSS

Individuality means that a web-based learning support system [17] must adapt itself to the ability and skill level of individual student. Adaptive methods and techniques in learning have been introduced and evaluated since the 1950's in the area of adaptive instruction and the psychology of learning [18]. Adaptive instructional methods adapted the content of the instruction, the sequencing of learning units, the difficulty of units, and other instructional parameters to the students' knowledge. These methods have been empirically evaluated and shown to increase learning speed and to help students gain a better understanding through individualized instruction.

According to Brusilovsky [8], there are several goals that can be achieved with adaptive navigation support techniques, although they are not clearly distinct. Most of the existing adaptive systems use link hiding or link annotation to provide adaptive navigation support. Link hiding is currently the most frequently used technique for adaptive navigation support. The idea is to restrict the navigation space by hiding links that do not lead to the relevant pages. That means the pages are not related to the users current goal or they are not ready to be seen. Users with different goals and knowledge may be interested in different pieces of information and they may use different links for navigation. Irrelevant information and links just overload their working memories and screen [6].

De Bra [7] presented a course that uses a system they developed to track student progress and based on that, generate document and link structure adapted to each particular student. Links to nodes that are no longer relevant/necessary or links to information that the student is not yet ready to access are either physically removed or displayed as normal text.

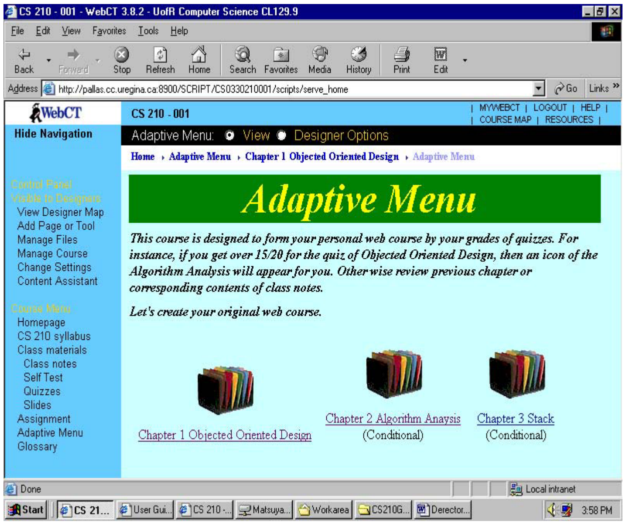
Da Silva et al [19] use typed and weighted links to link concepts to documents and to other concepts. The student's knowledge of each concept is used to guide him/her towards the appropriate documents.

Adaptation may be supported according to different student characteristics, including his/her preferences of browsing hyperlinks. Five main features are identified for maintaining adaptation [8]. They include student goals, student knowledge and familiarity with the domain, student qualification, experience in the hyperspace, and personal preferences. Most of the adaptive systems use knowledge representation and domain models and consider the student knowledge for providing adaptation. Student knowledge is often represented by an overlay model, which is based on the domain knowledge base. The domain knowledge base provides the structural description of the subject area. That represented as concepts and relations between them.

Most of WBAEH systems use complex and multi-layered semantic networks for representing domain knowledge. The semantic relations describe relations among the concepts. There are two common models widely used, namely overlay model and stereotype model. The overlay model keeps track of the student knowledge about every element of the domain knowledge base and covers the whole domain. The idea is to mark each knowledge item with a value calculated as student knowledge. The value could be binary, qualitative or quantitative. Overlay models are powerful and flexible. They contain information about students' familiarity with different topics. It can be simplified to distinguish several classes of group users. The drawback of overlay model is that it does not consider student's misconceptions. There may be stereotypes for every dimension considered in the domain (novice, intermediate, expert). The stereotype user model is simpler and easier to initialize and maintain comparing to overlay model. Sometimes the two models are combined. At beginning, the system classifies the user to some stereotypes in an interview or test, then after collecting enough information about student performance, the system switches to an overlay model.

4. The Proposed Framework Based on KSM

A primary concern for web-based learning support system is the design of an appropriate structure so that a student can easily and naturally find the most relevant information depending on his/her needs [17]. we presented a model of personalization that attempts to assist learners in their learning based on their assessment results on the learning materials. It provides feedback on their performance and suggests the most suitable learning content to be learned next. Figure 3 shows the sample screen shot of adaptive course material.



CS 210 - 001 - WebCT 3.8.2 - UofR Computer Science CL129.9
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites Media History Print Edit
Address http://pallas.cc.uregina.ca:8900/SCRIPT/CS0330210001/scripts/serve_home
Go Links
WebCT
Hide Navigation
CS 210 - 001
| MYWEBCT | LOGOUT | HELP |
| COURSE MAP | RESOURCES |
Adaptive Menu: View Designer Options
Home › Adaptive Menu › Chapter 1 Objected Oriented Design › Adaptive Menu
Control Panel
Visible to Designers
View Designer Map
Add Page or Tool
Manage Files
Manage Course
Change Settings
Content Assistant
Course Menu
Homepage
CS 210 syllabus
Class materials
Class notes
Self Test
Quizzes
Slides
Assignment
Adaptive Menu
Glossary
Done
Local intranet
Start CS 21... User Gui... CS 210 -... Matsuya... Workarea CS210G... Derector...
3:58 PM

Adaptive Menu

This course is designed to form your personal web course by your grades of quizzes. For instance, if you get over 15/20 for the quiz of Objected Oriented Design, then an icon of the Algorithm Analysis will appear for you. Other wise review previous chapter or corresponding contents of class notes.

Let's create your original web course.

Chapter 1 Objected Oriented Design

Chapter 2 Algorithm Anaysis
(Conditional)

Chapter 3 Stack
(Conditional)

Figure 3: Screen shot of the demonstration of the adaptive course material

In this paper, we propose a framework for integrating personalization and collaboration in web-based learning management environment. To support student-centric learning and to encourage students to actively engage in the learning process to construct their own learning and define their learning goal, knowledge structure map[20] is used as an effective learning aid for "Data Structure and Algorithm Analysis" course.

Knowledge structure map (KSM) is a method designed to produce a visual map of individual knowledge components comprising a study [21]. These components or map nodes are linked by lines that show learning dependencies. The map show clearly the pre-requisite knowledge needed for students in order to progress in the study. The nodes on the map provide access to learning material that allows the learner to acquire the particular piece of knowledge selected. A learning task can be made clear by using a KSM. The map will clearly show the learning goal, where to start and the paths to be taken. Figure 4 shows a sample knowledge structure map for the "Data Structure and Algorithm Analysis" course.

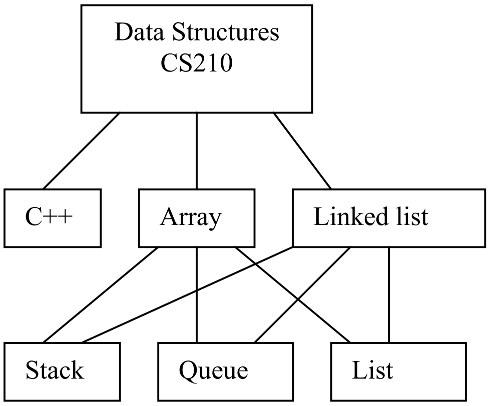
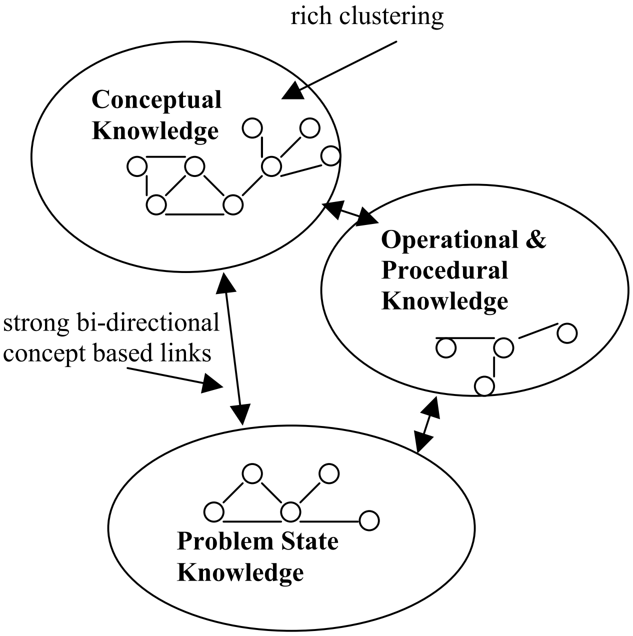


Figure 4. Sample partial knowledge structure map

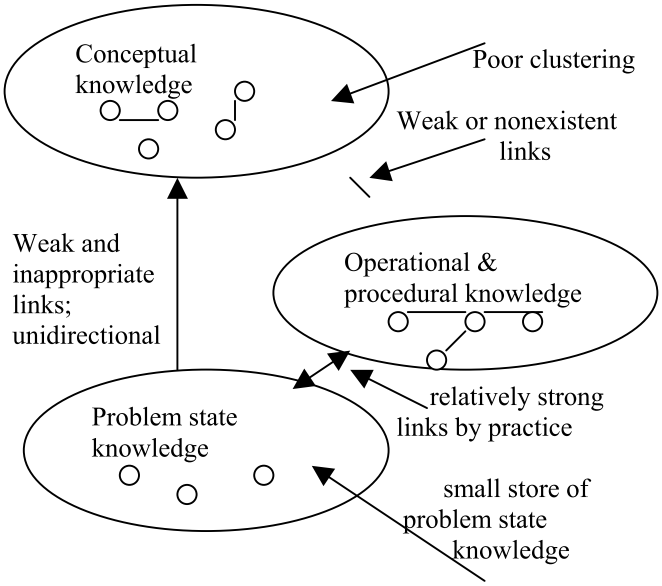
According to Ausubel [22], knowledge structure maps foster meaningful learning by teaching the connections among course concepts, and promote meaningful learning by encouraging students to generate their own connections between concepts.

Figure 5 and Figure 6 demonstrate the difference between an expert and a novice' knowledge structure.



rich clustering
Conceptual Knowledge
Operational & Procedural Knowledge
strong bi-directional concept based links
Problem State Knowledge

Figure 5. Expert's knowledge store [23]



Conceptual knowledge
Poor clustering
Weak or nonexistent links
Weak and inappropriate links; unidirectional
Operational & procedural knowledge
relatively strong links by practice
Problem state knowledge
small store of problem state knowledge

Figure 6. Novice's knowledge store [22]

From Figure 5 and Figure 6, we can see that the differences of an expert and a novice knowledge structure. The expert knowledge structure represents a strong linkage between conceptual knowledge and problem state knowledge, whereas the novice knowledge shows a weak linkage.

In the knowledge structure map, each node represents a piece of knowledge and the label in the box is the name of this knowledge. Links or arcs between nodes are unidirectional. A link shows that it is necessary for a student to know a particular piece of knowledge before it is possible for that student to have a full and detailed understanding of another piece of knowledge.

In order to provide an in depth understanding of the fundamental data structures and to motivate the students, a web-based adaptive course with analysis tool based on student learning styles has been proposed. The idea here is to create a tool for students and teacher to visualize and trace the learning process. Since viewing concepts in a different way can help to gain additional insight into knowledge, we see lots of opportunities in this new approach. Our main goal is to involve the students to take active role in planning his/her studies. The system should encourage student to go deeper into learning, to evaluate his/her studies, and to find out the most appropriate learning paths.

When the KSM is connected to student database, the system reflects the study process. All the concepts a student has learned can be marked with a certain color. Some comments can be added to the map. This will help students and the teacher to analyze the study process. For teachers who want to see how their students are doing, this allows quite efficient way to do it. The teacher can easily see her students' progress, strength and weakness and can help the student in future studies. Whenever a student adds a new concept to his/her personal knowledge store, the system suggests and recommends other concepts related to the concept to be added to the structure. With this feature, a student can build a logical map quickly and observe all the time the structure of his/her studies. The students' knowledge maps of the course can also be implemented and presented as animations. With an animation, the student can see all the time how the learning process is proceeding and which concepts are recognized to be similar. Personalized presentation based on each student knowledge level can be presented visually. The maps can be compared. This approach helps the students to get insight into the maps being compared. This makes it a reasonable platform for supporting students' collaboration. Students can compare his/her knowledge map with the instructor (expert), peers, or mentors. Through the process, student can view his learning process, and the dynamic changes of the student knowledge map will indicate how the student knowledge grows.

We can also construct dynamic and clickable knowledge structure maps by utilizing the web technologies. For example, student click on the node indicating “Stack”, it would give options for simulation of the algorithm; examples and demos of how to use it; or simple text file of the formal definition of stack. When suitable knowledge structure is designed, the system can be used for effective learning, tutoring, problem solving, or diagnosing misconceptions.

5. Conclusion

Our primary goal is to provide an adaptive learning support environment that will effectively accommodate a wide variety of students with different skills, background, and learning styles. The web offers dynamic and open learning environment. Based on the student-centered philosophy, personalization and adaptation are the important features for the Web-based learning support systems.

In this paper, we examined and discussed the adaptation and personalization in the web-based learning support system based on the student model. It is a challenging task to develop a fine-tuned WBLSS due to the uncertainty and complexity of the student model, particularly the students’ cognitive attributes.

In order to support the students' deep learning and understanding the difficult concepts about algorithms and data structures, we proposed a feasible framework by dynamically constructing knowledge structure map during the learning process. It can be visualized and clickable. Student can use his/her knowledge structure map to compare with the instructors or peers knowledge structure maps. When suitable knowledge structure is designed and constructed, the system can be used for effective learning, tutoring, problem solving, or diagnosing misconceptions.

6. References

[1] WebCT, WebCT home page, http://www.webct.com/.

[2] Blackboard Inc. Blackboard home page, http://www.blackboard.com/.

[3] Brusilovsky, P. KnowledgeTree: A Distributed Architecture for Adaptive E-learning. WWW 2004, May 17-24, 2004, ACM 104-111.

[4] Zaiane, O. R., "Building a Recommender Agent for e-Learning Systems", International Conference on Computers in Education (ICCE'02), Auckland, New Zealand, December 03-06, 2002, pp55-59.

[5] Nill, A., "Providing Useable and Useful Information by Adaptability", GMD – German National Research Center for Information Technology, Sankt Augustin, German, http://zeus.gmd.de/~nill/flexht97.html/

[6] Brusilovsky, P., Anderson, J., "An adaptive System for Learning Cognitive Psychology on the Web", WebNet 98 World Conference of the WWW, Internet & Intranet, Orlando, Florida, November 7-12, 1998, pp.92-97.

[7] Papandreou, C.A., Adamopoulos, D.X., "Modelling a multimedia communication system for education and training", Computer Communications 21, 1998, pp.584-589.

[8] Brusilovsky, P., "Methods and Techniques of Adaptive Hypermedia", User Modelling and User Adapted Interaction, Vol. 6, N2-3, 1996, pp. 87-129

[9] Gagne, R. M., "Principles of Instructional Design", Third Edition, Holt, Rinehart and Winston, New York, 1988.

[10] De Bra, P., "Teaching Through Adaptive Hypertext on the WWW." *International Journal of Educational Telecommunications.* August 1997, pp. 163-179

[11] Brusilovsky, P., "Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-Based Education Intelligent Tutoring Systems." Proceedings of 5th International Conference on Intelligent Tutoring Systems, ITS 2000, Montreal, Canada, June 2000.

[12] Brusilovsky, p., "Adaptive Hypermedia.". User Modeling and User Adapted Interaction, 2001. 11:p.87-110

[13] Wenger, E., "Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge", Morgan Kaufmann Publishers, 1987

[14] Jameson, A., "What Can the Rest of Us Learn From Research on Adaptive Hypermedia and Vice-versa?", 1999 http://w5.cs.uni-sb.de/~jameson/ahh/ahh-comment.html.

[15] Jameson, A., "User-Adaptive Systems: An Integrative Overview", 7th International Conference on User Modeling, Banff, Canada, 1999.

[16] Honey, P., Mufford, A., "The Manual of Learning Styles", P. Honey, Maidenhead.

[17] Fan, L., Yao, Y. Y., "Web-based Learning Support System", WI/IAT 2003 Workshop on Applications, Products and Services of Web-based Support Systems, October 2003, Halifax, Canada.

[18] Tennyson, R. D., Rothen, W., "Pre-task and On-task adaptive design strategies for selecting number of instances in concept acquisition". Journal of Educational Psychology, Volume 69, 1977, pp.586-592.

[19] Da Silva, D.P., Van Durm, R., Duval, E. & Olivi, H., "Concepts and Documents for Adaptive Educational Hypermedia: a Model and a Prototype", Second workshop on Adaptive Hypertext and Hypermedia, Ninth ACM Conference on Hypertext and Hypermedia, Pittsburgh, USA, June 20-24, 1998, pp.35-43.

[20] Novak, J. D., "The Theory Underlying Concept Maps and How to Construct Them". http://cmap.coginst.uwf.edu/info/

[21] Ausubel, D. P., "Educational Psychology", A Cognitive View, Holt, Rinehart & Winston, 1968.

[22] Gordon, J. L., "Creating Knowledge Structure Maps to Support Explicit Knowledge Management". Applications and Innovations in Intelligent Systems VIII. December 2000, by Springer. Pages 33-48.

[23] Gerace, W.J., "Problem Solving and Conceptual Understanding", http://umperg.physics.umass.edu/writings/online/.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 67-74, 2004.

An Integrated Approach to Discovery in Complex Information Spaces

Daryl H. Hepting and Cory Butz
Department of Computer Science, University of Regina
Regina, Saskatchewan, S4S 0A2, Canada
E-mail: {dhh,butz}@cs.uregina.ca

Abstract

As the amount of available data continues to increase, more and more effective means for discovering important patterns and relationships within that data are required. Although the power of automated tools continues to increase, we contend that greater gains can be achieved by coordinating results from a variety of tools and by enhancing the user's ability to direct the application of these tools. A system which can rely on multiple modalities for processing information has a distinct benefit in terms of user-confidence in the final results. We set forth an approach which permits a flexible, user-controllable model of the information space within which basic tools can be integrated. The analysis of data, whether it be through visualization or data mining, for example, is an exercise in problem-solving and any computer-based tool to support the analysis process should be designed to support problem-solving activities. The process by which a user can develop and interact with this model is described and the benefits of this approach are discussed. This integration can be extremely useful both for the development of new hypotheses regarding the data and for verification of existing hypotheses.

1. Introduction

Visualization in scientific computing continues to gain prominence as a tool for data analysis and comprehension, beginning with the landmark report of McCormick *et al.* [20]. In the modern era, this trend can trace its roots back to the beginnings of modern computing. Even before the advent of computer graphics, numerical simulations were an important tool for scientific insight. In the 1940's, John von Neumann [37] wrote about the potential for the use of computers in the study of differential equations. It is this potential of the computer as an experimental tool which caused Richard Hamming [9] to write "the purpose of computing is insight, not numbers" as the motto for his 1962 text, *Numerical Methods for Scientists and Engineers*. It is important to remember that insight is the goal of any computer-aided analysis activity, whether it be scientific visualization, data mining, or machine learning.

Although Jessup [12] contended that scientific visualization has the promise to democratize visual thinking, the capability to produce computer-generated visual representations alone is insufficient to realize this promise of aiding the achievement of insight for individuals. What is true for visualization is also true in a much more general sense for other forms of computer-based analyses. The mere existence of a capability is not sufficient to have it adopted and used successfully by all those who might gain insight with it. In general, tools must be created that allow access to the method without requiring the user to be an expert in the vocabulary associated with the details of the method. More generally though, tools should present the user with a representation of the context in which the method is invoked. Furthermore, these tools should enable the domain expert to work with the analysis tools, without being burdened by the need to learn a specialized vocabulary, to produce representations which are effective for the expert [32]. In the realms of scientific and information visualization, the *cogito* system has been a example of this paradigm [10].

Consider that any visual representation can be decomposed into *components*, each with their own *elements*. A component could be "graph type", with elements including "bar chart", "pie chart", "line chart", "scatterplot", and so on. Each visual representation can be denoted as an N-tuple, where e_i is an element of component C_i. In practice, not all N-tuples will correspond to valid visual representations because of incompatibilities between elements of different components. The Cartesian product of the elements from all the components forms the N-dimensional space of available visual representations

$$\langle e_1, e_2, \ldots, e_N \rangle \in C_1 \times C_2 \times \ldots \times C_N.$$

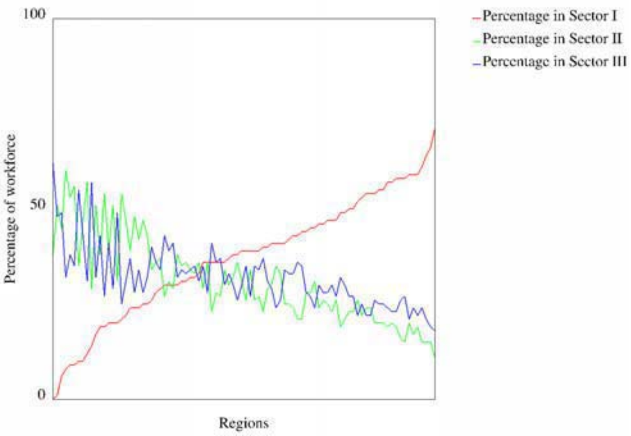


Figure 1: One possible visual representation of Bertin's data, constructed from components and elements that specify features including the graph type, the annotation, the sorting of the data, and the colours.

The space of available visual representations can be very large and it can be difficult to grasp the implications of all available combinations of elements. It is particularly clear in this type of situation that it is not possible to completely articulate all elements of a problem, *a priori*, to arrive at a solution. This fact only exacerbates the problem of selecting and specifying the individual elements in a visual representation. In the midst of so many combinations, it can be difficult to find a visual representation which is apposite. As Wegner [38] points out, interactive systems can offer a great deal more power in dealing with this situations, over algorithms alone. Researchers have observed that problems and solutions coevolve [6]. One usually does not, perhaps cannot, have a clear statement of the problem without an idea of the solution. Rittel [25] described this as follows: "you cannot understand the problem without having a concept of the solution in mind; and that you cannot gather information meaningfully unless you have understood the problem but that you cannot understand the problem without information about it." One must simply begin. The evolutionary nature of the design process is well-described by the model of evolutionary epistemology [31]. Allen [1] uses this formulation, based on Neill [21], for information seeking.

The *cogito* system supports each user in this process by providing external representations of the space of available alternative combinations and the means to manipulate it quickly and easily, through a predominantly visual interface [11]. Although the *cogito* system was developed as a means to support the scientific visualization process, the principles are amenable to other areas. The recent work by Grady *et al.* [8], which illuminates several important open problems with respect to the integration of data mining and visualization processes, echos key points of the design philosophy in *cogito*. Thus, we feel that our system coincides nicely with the architecture of web-based support systems suggested by Yao and Yao [44].

The rest of this paper is organized as follows. Section 2 describes the paradigm embodied by the *cogito* system. Section 3 describes methods for learning which can be incorporated within the *cogito* framework for discovery. Section 4 describes how the automated capabilities can be integrated within *cogito*. Finally, Section 5 presents some conclusions and directions for future work.

2. Design of the *cogito* system

The use of components and elements to describe particular visual representations is an adaptation of Bertin's [2] retinal variables which he used to systematically explore marks on a plane and how those marks could be used to construct diagrams, networks, and maps. Graphic communication in two dimensions has been thoroughly studied and the construction of visual representations within this realm is fairly well understood. For this reason, the example chosen to illustrate this presentation and to evaluate the prototype software is a two-dimensional graphing problem based on a small dataset from Bertin [2][page 100]. It provides a view of the French economy from the early 1960's. For each département in France, the data provides the workforce (in thousands of workers) for each of the three sectors (primary, secondary, and tertiary) in the economy; the total workforce (the sum of the three sectors); and the percentage of the workforce in each sector. Figure 1 presents a sample visual representation of this data.

Bertin [2] remarked that "to construct 100 DIFFERENT FIGURES from the same information requires less imagination than patience. However, certain choices become compelling due to their greater 'efficiency."' But the question of efficiency is closely linked to the task at hand and the user's experience with the elements of a visual representation, as Casner [3] describes. Although Bertin contends that meaning can be communicated fully through a graphic and its legend, the more widely accepted view is that communication and interpretation occur, or are influenced by things, outside this realm. For Winograd and Flores [39], this means that "the ideal of an objectively knowledgeable expert must be replaced with a recognition of the importance of background. This can lead to the design of tools that facilitate a dialog of evolving understanding among a knowledgeable community." This type of computer-based tool can be hard to construct.

Adapting the classification of Kochhar *et al.* [14], it is possible to distinguish manual, automatic, and augmented systems based on their relationship of human and computer.

Manual systems require the user to completely describe and control the operation of the visualization application. The space of alternatives available for exploration in these schemes is implicitly limited by the user's own experience. Systems exemplified apE (animation production Environment) [5] and AVS (Application Visualization System) [35], are collectively known as Modular Visualization Environments (MVE's). MVE's have come to prominence because they allow users to create complete visualizations from components connected using a visual dataflow model. DataDesk, the statistical graphics package first described by Velleman and Pratt [36] in 1989, provides a direct-manipulation interface to statistics and a good example of Tukey's Exploratory Data Analysis [34]. It builds on the idea that multiple, connected views of data can greatly enhance the power of data analysis. Graphical interfaces are seen as ways to specify "like this one, only different in the following ways." Insight is acknowledged as important. The Spreadsheet for Information Visualization (SIV) [4], based on work presented by Levoy [17], is a novel use of the spreadsheet programming paradigm that allows the user to explore the effect of the same operation on several related images.

Automated systems appear to the user as black boxes which are given input and produce output. The rationale behind them is that the number of alternative visual representations is so large that the user would be overwhelmed if he or she had to deal with the space in its entirety. In accepting this guidance from the computer, the user relies more on the computer for its application of design rules and gives up more freedom to exercise personal choices about what the visual representations will contain. In 1986, APT (A Presentation Tool) by Mackinlay [18] contributed a formalization of the design rules for two-dimensional static graphs, based on Bertin [2] and others. It was a prescriptive system because it chose graphics on the basis of expressiveness and effectiveness criteria. With BOZ in 1991, Casner [3] added information about the task to his presentation system and this resulted in a noticeable improvement in user performance with the graphs that his system generated.

Augmented systems aid the user by allowing certain well-defined tasks to be performed primarily by the computer, with the effect of increasing the capabilities of people to tackle complex problems. Because any articulation of a design is an ongoing process which is necessarily incomplete, it is important for the user to maintain some control. Rogowitz and Treinish [26] described a visualization architecture that allowed the user to choose a higher-level interaction with the visualization process, based on the invocation of appropriate rules. The VISTA (VISualization Tool Assistant) environment described by Senay and Ignatius [29] would analyse, as much as possible, the input data and suggest a visual representation to which the user could make modifications. The SageTools [27] system allowed users to work in the context of past graphics with the option to modify what had already been done. The Integrated Visualization Environment (IVE) [13] implemented the cooperative computer-aided design (CCAD) paradigm. It used a generative approach, in which the user could intercede after each iteration to select promising designs for further development. Design Galleries [19] worked to provide a good sampling of the range of alternatives. The user specified the means for comparison and the system worked off-line to generate and evaluate the images based on the user's specification and then displayed the results.

Rather than focus on the results produced by these visualization systems and attempt to answer whether a "best" visual representation can be decided for any context or any group of users, it is productive to look at the process by which these representations can be developed. According to Winograd and Flores [39], we can "create computer systems whose use leads to better domains of interpretation. The machine can convey a kind of coaching in which new possibilities for interpretation and action emerge."

Norman [22] describes the twin gulfs of execution and evaluation. With a goal in mind, a user experiences the gulf of execution in deciding which commands to execute in order to move from his or her present state to the goal state. Similarly, the gulf of evaluation is encountered when a user tries to reconcile an intermediate result state with the original goal state. An effective interface will minimize these gulfs, and for visualization tasks a visual interface is indicated.

In 1991, Sims [30] presented a method for the use of artificial evolution in computer graphics which employed both a genetic algorithm [7] and genetic programming [15]. Both of these "genetic" methods work by simulating the cellular-level processes of cross-over and mutation. The former does this as means to search a space whereas the latter works to transform the space itself. For Sims, the goal was to evolve images and textures. However, because it can be surprising to see images from different generations with no apparent connection between them, it can work to defeat the user's control. In 1992, Todd and Latham [33] also discussed a genetic approach to the generation of new images, theirs being more restrictive and controllable by not including genetic programming.

Even for small problems with relatively few alternatives, an exhaustive evaluation is almost always completely impractical. Instead, humans rely on heuristic search methods which are likely to find acceptable solutions in a reasonable amount of time. These search heuristics can be of two sorts, in general. If the problem is well-understood, local search techniques may be employed effectively. If the problem is new, a global search may be better suited to the exploration of alternatives.

The *cogito* software system was designed to address the shortcomings of traditional visualization tools. In particular, the system deals with the problem of articulation with a visual interface that provides non-verbal access to alternatives. With an incomplete articulation of the context, the iteration performed in selecting and evaluating candidate visual representations is crucial to the visualization process. The evaluation of visual representations can be done more effectively if the available alternatives are understood, and interaction is essential to accomplish this. The *cogito* system supports "combinatory play" by considering every visual representation to be the product of elements from each of several components and it relies on the user to choose these elements. Not only is this conception of components and elements familiar from Bertin, it also occurs in MVE systems like AVS [35] (Application Visualization System), and the toolkit philosophy of the Visualization ToolKit [28]. But, in *cogito*, the user does not choose these elements in isolation. Rather, he or she chooses between whole visual representations, each of which comprise particular elements.

The computer is well-suited to provide such external memory to support this decision-making process. Placed between manual and automatic systems, the design of *cogito* uses the computer to perform bookkeeping functions and allows the user evaluate and select. A traditional visualization system, with its need for expertise in programming, can separate the user from this important function. Whereas programming support is also required for *cogito*, the user and the programmer may work together to create the notion of the space of available representations and the user is still able to interact directly with the computer. Figure 2 illustrates this difference.

The *cogito* system provides, through views, the means to structure and examine the space according to a range of criteria. The user sees the current space, with the current organizational view, one screen at a time. Cells, which display individual visual representations and permit certain operations on them, comprise each screen. A schematic of one of these screens is shown in Figure 3. As the programmer and user define the space, it is also possible to use different organizational methods for the space of alternatives. In Figure 4, for example, one sees 3 different ways to organize a space with three dimensions. Using the terminology of Figure 3, the representatives $x_1 \ldots x_4$ in Figure 4(b) are formed by choosing sequentially from X and randomly from Y and Z.

The user indicates desirable elements or complete visual representations by non-verbal selection (done by clicking directly on the desired cell). Once the user is satisfied with the selections made on a particular space, a new space consistent with those selections is generated by a genetic approach which performs crossover operations amongst selected combinations. Successive generations can be used to either narrow or expand the search space (up to the size of the original), depending on the needs of the user. Additionally, an "image editor" is provided to directly make small changes. In this way, the space of all available visual representations can be navigated.

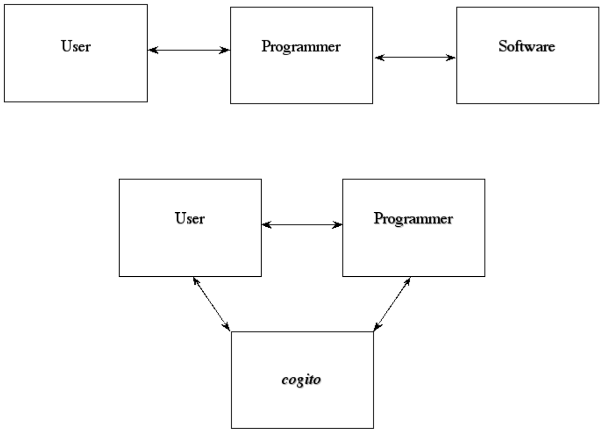


Figure 2: In the traditional model of interaction with visualization systems, the programmer mediates the user's experience with the software. The new model embodied in *cogito* allows the user to work directly with the software.

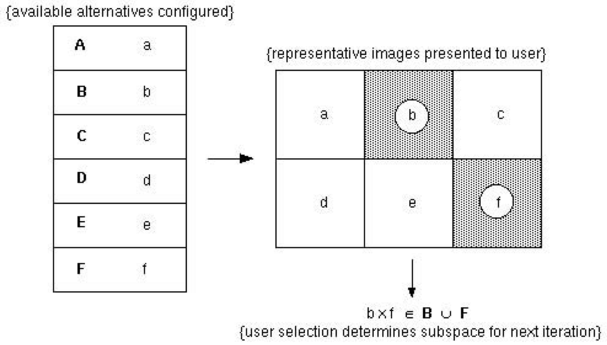

{available alternatives configured}
A a
B b
C c
D d
E e
F f
{representative images presented to user}
a
b
c
d
e
f
b×f ∈ B ∪ F
{user selection determines subspace for next iteration}

Figure 3: Schematic look at the interface: the space of available alternatives is grouped according to user-specified criteria. Each group (A – F) has a representative element (a – f) which is displayed to the user. The subspace for the next search iteration is based on the user selection (b and f).

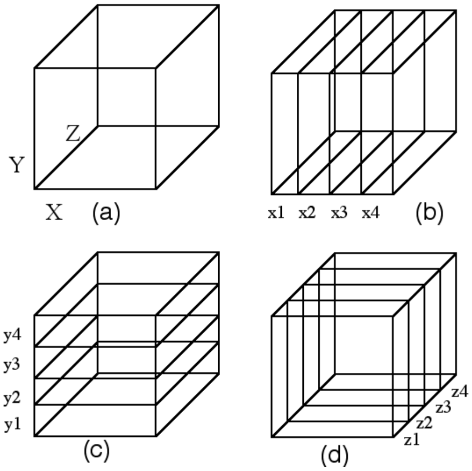


Figure 4: Consider a three-dimensional space, depicted in the top left, with axes X, Y, and Z. Organizing the space in terms of any of those 3 axes leads to the other states depicted. If elements in component X are chosen sequentially, those in Y and Z can be selected randomly to give a sense available options.

3. Computational Support for Discovery

A fundamental original emphasis for the *cogito* system was support for the user *without* directing the user. In very large and complex information spaces, this emphasis is perhaps impractical or even unwanted when considering experts in some domain of knowledge. There are two basic ways in which some direction can be given to the *cogito* user.

The first may emerge as patterns in usage of the *cogito* system over time. For the same data set and collections of available visual representations, it might be very instructive to know if previous users had selected certain visual representations in certain cases. This type of information can be garnered by multidimensional scaling [16], for example.

The second type of direction can derived from the data to be analyzed, where algorithms for learning can be applied. These learning methods are outlined in the balance of this section.

Learning from examples, known as inductive learning, is perhaps the oldest and best understood problem in artificial intelligence [43]. For our purposes, we may view this problem as indentifying the documents which a user is likely to be interested in. An inductive algorithm can, from a given sample of documents classified as relevant or nonrelevant, infer and refine decision rules.

Quinlan [24] proposed an inductive algorithm, called *ID3*, based on the statistical theory of information proposed by Shannon. Since it is essential to have an effective attribute selection criterion, ID3 uses the entropy function in selecting a suitable subset of attributes to construct a decision tree.

We illustrate this method using an example in [43]. In the following example, our problem of identifying the documents relevant to a user is synnonamous with the problem of a physician diagnosing patients. Consider the sample data in Figure 5 representing the diagnosis of eight patients by a physician. As already mentioned, the task at hand is to determine which attribute out of Height, Hair, and Eyes is the best classifier. By using the entropy function, it can be verified that attribute Hair is the best classifier. Thus, we construct the initial decision tree in Figure 6.

	Height	Hair	Eyes	Expert Classification
$o1$	Short	Dark	Blue	$-$
$o2$	Tall	Dark	Blue	$-$
$o3$	Tall	Dark	Brown	$-$
$o4$	Tall	Red	Blue	$+$
$o5$	Short	Blond	Blue	$+$
$o6$	Tall	Blond	Brown	$-$
$o7$	Tall	Blond	Blue	$+$
$o8$	Short	Blond	Brown	$-$

Figure 5: Sample data consisting of eight people diagnosed by a physician.

Any leaf node of the tree containing objects belonging to different expert classes requires further classification. In the initial decision tree, the leaf node for Blond needs refinement. Again, the objective is to determine which

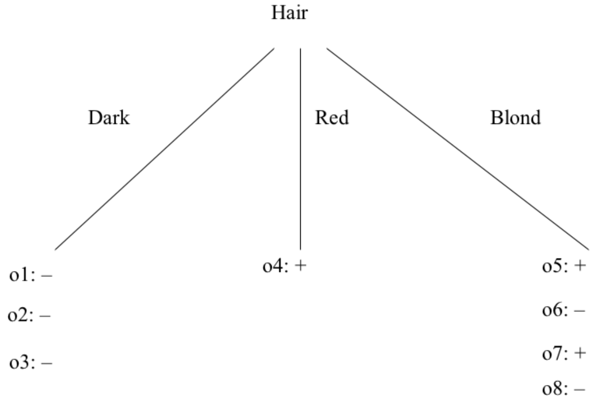


Figure 6: The initial decision tree with Hair as the root node.

of the remaining attributes best classifies the four objects $o5, o6, o7, o8$. It can be established that attribute Eyes is a better classifier than attribute Height. Thus, the initial decision tree in Figure 6 is refined as shown in Figure 7. Since each leaf node in the refined decision tree contains objects of the same expert class, no further refinement is necessary. Thus, given the sample data in Figure 5, the ID3 algorithm will produce the decision tree in Figure 7.

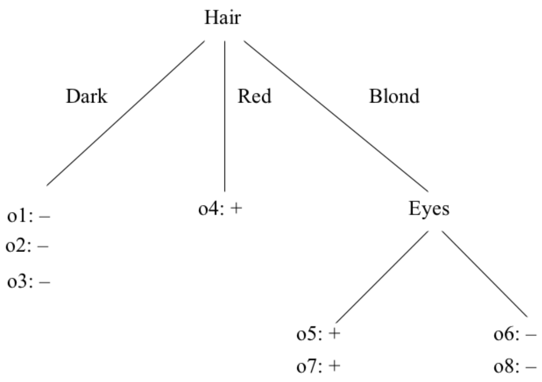


Figure 7: Refining the decision tree in Figure 6 by adding the node (classifier) Eyes.

The important point to remember is that, by providing the user with a decision tree tool, the user will be able to visualize how the attributes classify the sample documents. This information is useful as it can be used to assign a *weight* to the attributes (keywords) in future searches.

Yao [45] has recently argued that an information retrieval systems must provide a *variety* of tools to help a user understand collected data. In this section, we turn our attention to learning a probabilistic network from data. Such a tool is useful as it reflects the *probabilistic independencies* holding in the data.

Our implemented system [42] for learning a probabilistic network from data requires no à priori knowledge regarding the semantics of the attributes (variables) involved. The required input is simply a repository of observed data in tabular form. Our system is capable of learning conditional independencies from the data and outputs a probabilistic schema encoding all the discovered independencies. Due to lack of space, however, we assume the reader is familar with the notions of *probabilistic conditional independence* [41], *Markov networks* [40], and *learning algorithms* [42].

Given a distribution p_X on $X \subseteq R$, we can define a function as follows:

$$\begin{aligned} \mathcal{H}(p_X) &= -\sum_{t_X} p_X(t_X) \log p_X(t_X) & (1) \\ &= -\sum_{x} p_X(x) \log p_X(x), & (2) \end{aligned}$$

where t is a tuple (configuration) of X and $x = t_X = t[X]$ is a X-value, and $\mathcal{H}$ is the Shannon entropy.

Let $\mathcal{G} = \{R_1, R_2, \ldots, R_n\}$ be hypertree and $R = R_1R_2 \ldots R_n$. Let the sequence R_1, R_2, $\ldots R_n$ be a tree construction ordering for $\mathcal{G}$ such that $(R_1R_2 \ldots R_{i-1}) \cap R_i = R_{i^*} \cap R_i$ for $1 \leq i^* \leq n-1$ and $2 \leq i \leq n$. A joint probability distribution p_R factorized on $\mathcal{G}$ is a Markov distribution, if and only if

$$\mathcal{H}(p_R) = \sum_{i=1}^{n} \mathcal{H}(p_{R_i}) - \sum_{i=2}^{n} \mathcal{H}(p_{R_{i^*} \cap R_i}). \quad (3)$$

This theorem indicates that we can characterize a *Markov distribution* by an entropy function. We now demonstrate how we learn the dependency structure of a Markov distribution.

Initially, we may assume that all the attributes are probabilistically independent, i.e., there exists no edge between any two nodes (attributes) in the undirected graph representing the Markov distribution. Then an edge is added to the graph subject to the restriction that the resultant hypergraph must be a hypertree. The undirected graph of the Markov distribution with minimum entropy is being selected as the graph for further addition of other edges. This process is repeated until a predetermined *threshold*, which defines the rate of decrease of entropy between successive distributions, is reached. From the output hypertree, we can infer the probabilistic conditional independencies which are satisfied by the distribution.

Suppose we have a database D consisting of the observed data of a set of four attributes, $\mathcal{N} = \{a_1, a_2, a_3, a_4\}$, containing five tuples as shown in Figure 8. We have set the threshold to zero, the maximum size of a clique $\eta = 4$, and the maximum number of lookahead links to one. The output is the undirected graph $\{(a_1, a_2), (a_1, a_3), (a_1, a_4)\}$. By applying the separation method, we know the following CIs hold in the observed data $\{I(a_2, a_1, a_3a_4), I(a_3, a_1, a_2a_4), I(a_4, a_1, a_2a_3)\}$.

It should be mentioned that there are numerous methods for learning a *Bayesian network* [23] from data (see [42] for references). Since a Bayesian network is defined on a *directed acyclic graph*, the directionality of the edges may be interpreted as causality between variables. Thereby, one may choose to learn a Bayesian network as well as a Markov network from the sample data.

a_1	a_2	a_3	a_4
0	0	0	0
1	1	1	0
1	1	0	0
0	0	1	0
2	1	1	1

Figure 8: Observed data consisting of 4 attributes and 5 tuples.

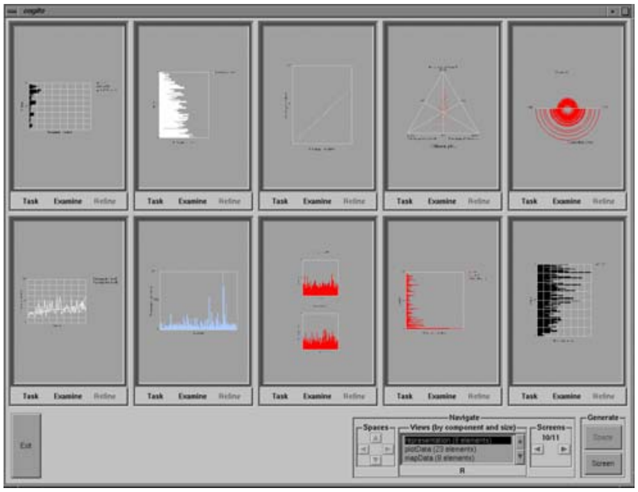


Figure 9: The interface to the *cogito* system. The interface displays a subset of available representations (sampled according to the selected organization of the search space), generated from the current data, with which the user can interact.

4. Integration

The *cogito* interface is shown in Figure 9. An integrated visualization and discovery tool will definitely help to address the issues of trust [32], as users can manipulate data and represent them in a variety of ways. The variety of methods available to the user will enable them to arrive at conclusions from several means.

Rather than lose sight of the forest for the trees, this approach allows users to examine the trees in the context of the forest, and to examine the forest at various levels of granularity, according to different criteria.

For visualization and discovery, we see two important advantages arise. Interesting patterns can be found through visualization which can be then coded within the discovery portion and similarly, patterns discovered by the learning algorithms can focus the interpretation efforts in the visualization stage. Thus, we feel our system coincides nicely with the architecture of web-based support systems suggested by Yao and Yao [44].

5. Future work

The modelling and representation of the space of available alternatives has proven to be important in many respects. Primarily, it provides each user with the means to explore in a safe, structured environment that can then act as a record of the whole decision process. The approach is even generalizable beyond the scope of the visualization and discovery. Work is now being done to use this paradigm in numerical experimentation. Consider using this approach to manage a numerical experiment where each parameter becomes a component and the values for that parameter become elements. We are pursuing other possible applications for this paradigm.

References

[1] B. L. Allen. *Information Tasks: Toward a User-Centered Approach to Information Systems*. Academic Press, 1996.

[2] J. Bertin. *Semiology of graphics : diagrams, networks, maps*. University of Wisconsin Press, 1983. translated by W. J. Berg.

[3] S. M. Casner. A task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics*, 10(2):111–151, April 1991.

[4] E. H. Chi. Principles for information visualization spreadsheets. *IEEE Computer Graphics and Applications*, pages 30–38, July/August 1998.

[5] D. S. Dyer. A dataflow toolkit for visualization. *IEEE Computer Graphics and Applications*, pages 60–69, July 1990.

[6] G. Fischer and B. Reeves. Beyond intelligent interfaces: Exploring, analyzing, and creating success models of cooperative problem solving. *Journal of Applied Intelligence*, 1:311–332, 1992.

[7] D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading, MA, 1989.

[8] N. Grady et al. Integrating data mining and visualization processes. In U. Fayyad, G. Grinstein, and A. Wierse, editors, *Information Visualization in Data Mining and Knowledge Discovery*, pages 299–303. Morgan Kaufmann, 2001.

[9] R. Hamming. *Numerical Methods for Scientists and Engineers*. McGraw-Hill, 1962.

[10] D. H. Hepting. *A New Paradigm for Exploration in Computer-Aided Visualization*. PhD thesis, Simon Fraser University, 1999. Ph.D. Dissertation.

[11] D. H. Hepting. Towards a visual interface for information visualization. In E. Banissi, editor, *Proceedings of the Sixth International Conference on Information Visualization*, pages 295–302. IEEE Computer Society, 2002.

[12] M. E. Jessup. Scientific visualization: Viewpoint on collaborations of art, science, and engineering. *SIGBIO Newsletter*, pages 1–9, February 1992.

[13] S. Kochhar, M. Friedell, and M. LaPolla. Cooperative, computer-aided design of scientific visualizations. In *Proceedings of Visualization '91*, pages 306–313, 1991.

[14] S. Kochhar, J. Marks, and M. Friedell. Interaction paradigms for human-computer cooperation in graphical-object modelling. In S. MacKay and E. M. Kidd, editors, *Proceedings of Graphics Interface '91*, pages 180–189, 1991.

[15] J. R. Koza. *Genetic Programming : On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems)*. MIT Press, 1992.

[16] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–29, 1964.

[17] M. Levoy. Spreadsheets for images. In A. Glassner, editor, *Computer Graphics: SIGGRAPH 94 Proceedings*, pages 139–146, 1994.

[18] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, 1986.

[19] J. Marks et al. Design Galleries: A general approach to setting parameters for computer graphics and animation. In *SIGGRAPH '97 Conference Proceedings*, pages 389–400, 1997.

[20] B. H. McCormick, T. A. DeFanti, and M. D. Brown. Visualization in scientific computing. *Computer Graphics*, 21(6), November 1987.

[21] S. D. Neill. The reference process and the philosophy of karl popper. *RQ*, pages 309–319, Spring 1985.

[22] D. A. Norman. *The Psychology of Everyday Things*. Basic Books, New York, 1988.

[23] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Francisco, California, 1988.

[24] J. Quinlan. *Machine Learning: the Artificial Intelligence Approach*, chapter Learning Efficient Classification Procedures and Their Application to Chess End Games. Tioga Press, Palo Alto, 1983.

[25] H. W. J. Rittel. Second-generation design methods. In *Developments in Design Methodology*, pages 317–327. Wiley and Sons, 1984.

[26] B. E. Rogowitz and L. A. Treinish. Data structures and perceptual structures. *SPIE*, 1913:600–612, 1993.

[27] S. F. Roth, J. Kolojejchick, J. Mattis, and M. Chuah. Sagetools: An intelligent environment for sketching, browsing, and customizing data-graphics. In *Proceedings CHI'95 Human Factors in Computing Systems*, pages 409–410. ACM Press, 1995.

[28] W. J. Schroeder, K. M. Martin, and W. E. Lorensen. The design and implementation of an object-oriented toolkit for 3D graphics and visualization. In *Proceedings of Visualization 96*, pages 93–100, 1996.

[29] H. Senay and E. Ignatius. A knowledge-based system for visualization design. *IEEE Computer Graphics and Applications*, 14(6):36, 1994.

[30] K. Sims. Artificial evolution in computer graphics. In R. J. Beach, editor, *Computer Graphics: SIGGRAPH '91 Conference Proceedings*, pages 319–328. ACM Press, 1991.

[31] P. Skagestad. Thinking with machines: Intelligence augmentation, evolutionary epistemology and semiotics. *Journal of Social and Evolutionary Systems*, 16(2):157–180, 1993.

[32] K. Thearling et al. Visualizing data mining models. In U. Fayyad, G. Grinstein, and A. Wierse, editors, *Information Visualization in Data Mining and Knowledge Discovery*, pages 205–222. Morgan Kaufmann, 2001.

[33] S. Todd and W. Latham. *Evolutionary art and computers*. Academic Press, London, 1992.

[34] J. W. Tukey. *Exploratory data analysis*. Addison-Wesley, 1977.

[35] C. Upson et al. The application visualization system: A computational environment for scientific visualization. *IEEE Computer Graphics and Applications*, 9(4):30–42, 1989.

[36] P. F. Velleman and P. Pratt. A graphical interface for data analysis. *Journal of Statistical Computation and Simulation*, 32:223–228, 1989.

[37] J. von Neumann. Recent theories of turbulence. In A. Taub, editor, *Collected Works of John von Neumann*, volume 6, pages 437–472. MacMillan, New York, 1963.

[38] P. Wegner. Why interaction is more powerful than algorithms. *Communications of the ACM*, 40(5):80–91, 1997.

[39] T. Winograd and C. F. Flores. *Understanding Computers and Cognition*. Ablex, Norwood, New Jersey, USA, 1985.

[40] S. Wong and C. Butz. Constructing the dependency structure of a multi-agent probabilistic network. *IEEE Transactions on Knowledge and Data Engineering*, 30, Part A(6):785–805, 1999.

[41] S. Wong, C. Butz, and D. Wu. On the implication problem for probabilistic conditional independency. *IEEE Transactions on Systems, Man, and Cybernetics*, 30, Part A(6):785–805, 2000.

[42] S. Wong, C. Butz, and Y. Xiang. Automated database scheme design using mined data dependencies. *Journal of the American Society for Information Science*, 49(5):455–470, 1998.

[43] S. Wong, W. Ziarko, and R. Ye. Comparison of rough-set and statistical methods in inductive learning. *International Journal of Man-Machine Studies*, 24:53–72, 1986.

[44] J. T. Yao and Y. Yao. Web-based support systems. In *Proceedings of the WI/IAT 2003 Workshop on Applications, Products, and Services of Web-based Support Systems*, 2003.

[45] Y. Yao. Information retrieval support systems. In *Proceedings of the 2002 IEEE World Congress on Computational Intelligence*, pages 1092–1097, 2002.

Web-based Agricultural Support Systems

Yuegao Hu* Zhi Quan
College of Agronomy and Biotechnology,
China Agricultural University
Beijing , 100094, P. R. China
{huyuegao, lemonquan}@cau.edu.cn
*the correspounding author

Y. Y. Yao
Department of Computer Science
University of Regina
Regina, Saskatchewan, Canada S4S0A2
yyao@cs.uregina.ca

Abstract

Agriculture is a complicated system, related to a wide range of environments, which is difficult to deal with perfectly. Web-based agricultural support system (WASS) has been proposed to applicable support agricultural activities, which combines web technologies and agricultural systems. In this paper we analyze the basic characters of the web-based agricultural support system and then describe the functionalities of the system.

1. Introduction

Advances in computer technologies have profoundly influenced the use of computerized support in various activities [1]. With the unlimited growth of the Internet and ever expansion of information on the Web, we have come to a new information era [2]. The Web provides new medium for storing, presenting, gathering, sharing, processing and using information. The benefits of the Web technology have been shown as following: [1].

1. The Web provides distributed infrastructure for information processing.
2. The Web is used as a channel to discuss one of the most popular support systems, DSS.
3. The Web can deliver timely, secure information and tools with user friendly interface such as Internet Explorer and Netscape.
4. The Web has no time or geographic restrictions. Users can access the system at any time, any place.
5. Users can control and retrieve results remotely and instantly.

With the rapid development of web technology, Computerized support systems are emerging more and more diverse groups, such as learning support system [3], education support system [4], research support system [2,5], etc.

Agriculture plays a vital role in human development. In the developing countries, agriculture must multiply its productivity for food security and keep the people from undernourishment or outright famine; in the industrial countries, agriculture must continue to increases its productivity to provide enough raw material for the textile, plastics, and other industries, and fulfill the need for expanding populations [6].

There are many definitions of agriculture:

"Agriculture means the science or art of cultivating the soil, growing and harvesting crops, and raising livestock. The art of making land more productive is practiced through the world—in some areas by methods not far removed from the conditions of several thousands of years ago, and other areas, with the aid of science and mechanization, as a highly commercial type of endeavor." [7]

"Agricultural science, the science dealing with farm production, including soil cultivation, water control, crop growing and harvesting, animal husbandry, the processing of plant and animal products, engineering, economics, and other related matters. The agricultural industry that is the focus of study includes farming, concerned with production; service industries, concerned with making or supplying machinery, buildings, fertilizers, and pesticides; and the first purchasers of farm products, such as processors, distributors, and marketing boards." [7]

"Agriculture is the systematic raising of useful plants by human management. Food production is the main reason for agriculture, but cultivated plants also furnish substances useful as textile fibers, dyestuff, medicines, and ornaments. Gathering wild plants for food or other purpose is not agriculture. In a broad sense, ‘agriculture’ often includes animal husbandry.” [8].

“Agriculture encompasses production of food, fiber, wood products, horticultural crops, and other plant and animal products and includes: financing, processing, marketing, and distribution of agricultural products; farm production supply and service industries; health, nutrition, and food consumption; the application of science; the use and conservation of land and water resources; development and maintenance of recreational resources; related economic, sociological, political, environmental, and cultural characteristics of the food and fiber system.”[9]

We can concluded from above discussion that agriculture is a complicated system, closely related with natural systems and social systems. Agricultural system exchanges substance, energy and information with natural system, and has great effect on society progress, natural environment. In order to understand the agricultural system and the relationship with human beings, Hu Yuegao (2000) proposed that the agricultural system be comprised of three subsystems: production-subsystem, management-subsystem and research-subsystem, and each one can be divided into more specific subsystems, as shown in figure 1[10].

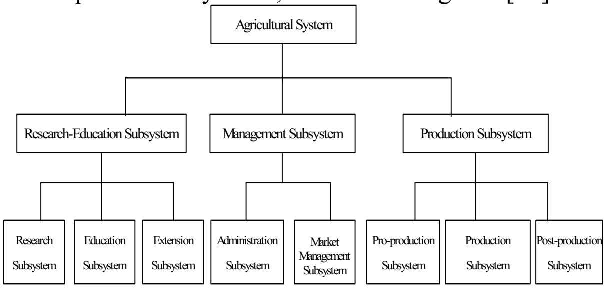


Figure 1. The structure of agricultural system

Due to the complexity of agricultural system, we found it very difficult to deal with it correctly and perfectly. We have confronted series of challenges due to the frequent and unexpected fluctuation, shortage of resources within systems:

- Lack of resource: water, arable land, forestry resource, energy, and fertilizer;
- Ecology degradation: soil degradation;
- Pollution: water pollution, soil pollution, air pollution;
- The weather warmer;
- Gap between poverty and rich enlarged;
- The frequency of agricultural disaster;

Generally speaking, the agricultural system encompasses more than namely education system, research system, learning system etc. We aim to study the issues and challenges brought on by the Web technology for various support systems and try to find out how applications and adaptations of existing methodologies on the Web platform can benefit our decision-makings and various activities in agriculture.

This paper briefly summarize the initial and basic ideas about WASS. We will focus on specific objectives: we will discuss the basic characters of the agricultural support system in section 2; we will give a depiction about the function of the web-based agricultural support system in section 3.

2. The Basic Characters of the Support System

I try to discuss several interrelated characters one by one as the follows:

2.1. Research-education subsystem:

2.1.1. Agricultural research subsystem. Agricultural research focuses on more diverse objectives than other science research, including to crop production, animal husbandry, water management, soil cultivation, pesticide /herbicide application, nutrition of nitrogen, etc. The research model proposed by Yao [5] is suggested applicable to agricultural research subsystem, we lay out the whole research process into 7 phrases:

Idea-generating phase. The phase aims to identify a study topic of interest. It may also be referred as the preparation or the exploration phase. Literature search and reading plays important roles in this phase.

Problem-definition phase. The objective is to precisely and clearly define and formulate study question from general observation generated from the previous phase. Problem definition involves careful conceptualization and abstraction. Precisely defined problem renders. It easier to find related and solved problems, as well as potential solutions.

Procedure-design/planning phase. The objective is to make a workable research plan by considering all issues involved, such as expected findings and results, available tools and methodologies, experiments designs, system implementation, time and resource constraints, and so on. This phase deals with planning and organizing research at strategic level.

Observation/experimentation phase. The objective is to observe real world phenomena, collect data, and carry out experiments. Depending on the nature of the research disciplines, various tools and equipment, as well as different methods, can be used.

Data-analysis phase. The objective is to make sense out of the data collected. So we can extracts potentially useful information from data. Statistical software packages can be used.

Results-interpretation phase. The objective is to build rational models and theories that explain the results from the data-analysis phase. It is necessary to investigate how the results help answer the research question, and how this answer contributes to the knowledge of the field. The connections to other concepts and existing studies may also be established.

Communication phase. The objective is to present the research results to the research community. Communication can be done in either a formal or an informal manner. Books and scientific journals are the traditional communication media. Web publication is a new tool of communication. Oral presentation at a conference, or discussion with colleagues, is an interactive means of communication.

2.1.2. Agricultural education subsystem. Agricultural education means training people to produce, process, and distribute food or fiber, and spreading scientific and technical information related to all phases of such work. It strives to help the people of the world improve the quantity and quality of products indispensable to human life. Agricultural education covers different levels from children's class in village schools to graduate study in universities [6]. Education and training are widely acknowledged as important contributors to national economic development and social well-being [11].

In general, agricultural education is divided into three level: higher agricultural education, for example education in universities and institution; vocational agricultural education, including various kinds involved knowledge and skills in agriculture; and the agricultural training for the adults or the youth. So a agricultural support system should fulfill all such needs.

2.1.3. Agricultural extension subsystem. The agricultural extension subsystem plays key role for agricultural development. The main function of extension is to disseminate useful information, including the research results in agriculture, home economics, and related subjects. As well as to helps families to apply such knowledge to real problems at farm, home, and community level [12]. Such function are shown as the follows [9, 13]:

First, it is medium between the agricultural research institutions, universities and farmers;

Second, it fills the gap between agricultural technology into real practice;

Third, it transfers the skills and knowledge to the farmers as to improve their living standard and agricultural practices;

Forth, it helps farmers to make decision.

Firth, it helps extension agents or organizations effectively and efficiently identify the goal and which decision it tries to help its farmers,

Last, extension managers can effectively deal with administrative affairs.

2.2. Production subsystem

Agricultural production subsystem is the core and basis component of agricultural system, which can be further divided into three levels: pre-production system, production system, and post-production system.

2.2.1. Pre-production and post-production subsystem. The agriculture pre-production subsystem includes all various departments, which provide production material and service for agriculture. The main tasks include the manufacture and maintainents of farm machineries and other agricultural facilities; the production of chemical products such as fertilizers and pesticides, the production of agricultural construction materials, and supplementary materials, the production of agricultural transportation facilities, the processing of seeds and feed; the circulation, transportation, information and finance service, and etc.

The agriculture post-production subsystem deals with processing the primary products such as grain, oil, food, feed,etc.

2.2.2. Production subsystem. Production subsystem is a main component, which directly supplies food to human being and raw material to industry. It is comprised of five parts: planting, forestry, animal husbandry aquaculture, etc. The production may effected greatly by soil, weather, water etc. Farmers need to overcome all kinds of constraints due to resources limits, etc., in order to get higher yield and better quality products. A combination the web-technology and agricultural expert system or agricultural decision support system will be very helpful to farmers so that they can get to under certain latitude and soil type. Which is suitable for specific crop, how to control the insects, what kinds of feedstuff to be feed on the livestock etc.

2.3. Agricultural management subsystem

2.3.1. Agricultural administrative subsystem. Agriculture administration is a concept that the government and the ministry of agriculture should formulate guidelines, provisions, plans, strategic decisions, and policies of agriculture development and be responsible for carrying out policies for different purposes, such as, production, distributing, financial, credit, labor, etc.

2.3.2. Agricultural market-management subsystem. Though the transformation from planning-economy to market-modulated economy has taken place since 1980,s, some major conflicts have occurred. One most outstanding contradiction is that circulation channels of the primary products can't meet the demands of market. Market-management by the governmental macro-manipulation can help to stock and protect the crucial primary products which related to the national economy and the people's livelihood. For example, under the market economy, some crucial primary products may overstock largely due to the years' bumper harvest, quantity and quality problems. The price will decrease too much once the primary products can't sell successfully, which leads to the loss of farmers' interest to produce in next year. Under such condition, the government should generate the protective prices to ensure the farmers' essential income. On the other hand, when the farmers are faced with serious natural disasters that they reject to sell their agricultural products, some agricultural products will be in serious shortage and result in panic buying and high-rising prices, which can't be accepted by the consumers. In order to deal with such problems, governments should have enough stock of crucial agricultural products to stabilize prices in the market.

It is obvious that good management will be of great benefit not only to nation but also to farmers and others. So WASS should be able to help decision makers for better solution.

3. Functionalities of the web-based agricultural support system

In order to support a large spectrum of agricultural activities, WASS must be flexible and has many functionalities. This section summarizes the functionalities and required computer technologies.

3.1.Decision support or expert system:

There are many factors, which can affect the agricultural activities. It's not an easy thing to deal with all kinds of agricultural problems effectively and correctly no matter to the producers or the governors or others related. Agricultural decision support system can serve as an important and very useful tool for farmers and decision-makers for solution to various problems. They can reach an optimal decision based on many considerations. For example, farmers can get information about what kind of crop should be grow under different soil type and how to choose the crop varieties, how to fertilize, how to irrigate, how to prevent the diseases and insects, etc.

3.2. Collaborative work support:

Collaborative work support provides a sound environment where all experts for agriculture in different areas can work together virtually, and significantly promote agriculture development.

Collaboratory is one kind of collaborative work support, which is an open meta-laboratory that spans multiple geographical areas with collaborators interacting via electronic means [14]. It gives a good chance to scientists to share research instruments, data and information, to exchange experiences, and to accelerate the development and dissemination of knowledge [14].

Audio/video conferencing is another kind of collaborative work support. The virtual conferencing greatly supports interaction between scientists, farmers, governors, extension agents and any others who are engaged in agriculture, and it provide a friendly environment to communicate with each other. Lots of the agricultural problems can be communicated and solved effectively and efficiently with such conference. And the audio/video conferencing can act as a virtual classroom for agricultural education too.

Chat room is another component of the collaborative work support, which will facilitate the communications between the users. In the agricultural support system there are various chat rooms in accordance with the difference of subsystem, for examples education chat room for education subsystem, extension chat room for extension subsystem etc. The users who want to communicate with the extension agents can enter the extension chat room.

Bulletin Board System (BBS) is integrated into the Collaborative work support system. The same as the chat room, it is comprised of education BBS, the extension BBS and so on, so the users can easily keep a track of previous discussion contents in which they are interested.

Furthermore, e-mail is a essential tool suitable for exchanging information too.

3.3. Information support

Information support includes information collection, management, retrieve or searching, exchange of for agricultural usage.

Agricultural production is closely involved with many factors a great matter, such as soil, precipitation, temperature, altitude, price of the products, transportation etc. So it is very essential to continuously collect information in different aspects and construct database, which is easy for searching and reuse later on.

Good searching support is very important for scientists, farmers, governors, and others. The scientists can find information of interest efficiently by researching support [2, 5]. With searching support, the farmers can get to know information about crops varieties, livestock, price of agricultural products etc. The extension agents can collect new information of agricultural technology by searching.. The governors and other decision makers can also benefit greatly from the searching support..

Exchange of information allows users to experiences, skills, data etc, thus to promote agriculture development. Researchers can upload research papers, others can share such information by downloading this; and government or administrator can publicize agricultural policies, rules; Extension agents can disseminate and popularize new technology through the system, at the same time farmers can keep up with the progress of new agricultural technology.

4. Conclusion:

Agricultural system is a complicated huge-system, comprised of three subsystems namely research-education subsystem, production subsystem and management subsystem, and there are distinct different characters for each subsystem.

Web-based agricultural support systems are based on the combination of agricultural science and computer science. By synergizing computer technology and agricultural science, we examine the characteristics of agricultural support systems with focus on the assembling and integration of existing computer systems to agricultural support system. Some preliminary and scattered ideas on the topic were discussed. The WASS may play a significant role in agriculture development in future.

Web-based agricultural Support Systems will be a very important research topic in the domain of Web Intelligence. Web-based agricultural support system can be used by researchers, producers, farmers and decision makers, etc., for various activities. Web-based technologies make the WASS easy to use and access.

The functionalities of the WASS are decision support, collaborative work support, and information support.

References

[1] Yao, J.T. and Yao, Y.Y., Web-based support Systems, *Proceedings of WSS'03*, 2003, pp. 1-5.

[2] Tang, H., Wu, Y., Yao, J.T., Wang, G.Y. and Yao, Y.Y., CUPTRSS: A Web-based Research support System, *Proceedings of WSS'03*, 2003, pp. 21-28.

[3] Lisa Fan and Yiyu Yao, Web-based Learning Support Systems, *Proceedings of WSS'03*, 2003, pp. 43-48.

[4] José M Parente de Oliveira and Clovis Torres Fernandes, A Framework for Adaptive Educational Hypermedia System, *Proceedings of WSS'03*, 2003, pp. 55-62.

[5] Yao, Y.Y., A Framework for Web-based Research support Systems, *Proceedings of the Twenty-sventh Annual International Computer Software and Applications Conference,* Dallas, USA, November, 2003, IEEE Computer Society Press, pp.601-606.

[6] Anderson, Robert S., *The Encyclopedia Americana*, Americana Corporation, c1980, vol. 1. pp.342.

[7] Gwinn, Robert P., *The New Encyclopaedia Britannica,* Encyclopaedia Britannica, Inc., c1993, v1, pp. 156.

[8] Anderson, Robert S., *The Encyclopedia Americana,* Americana Corporation, c1980, vol. 1. pp.353.

[9] http://www.dpi.state.wi.us/dpi/dlsis/let/agindex.html

[10] Hu Yuegao, *Agricultural Development Principle*, China Agricultural University Press, Beijing, China, 2000, pp.139-155.

[11] Sue Kilpatrick，Education and Training: *Impacts on Farm Management Practice,* http://www.crlra.utas.edu.au/files/discussion/2000/D03-2000.pdf

[12] Anderson, Robert S., *The Encyclopedia Americana*, Americana Corporation, c1980, vol. 1. pp.345.

[13] A.W.van den Ban, Agricultural Development; Opportunities and Threads for Farmers and Implications for Extension Organizations, *The Journal of Agricultural Education and Extension,* 1999,vol. 6. no.3. pp.145-156

[14] Xiaorong, Xiang, Yingping, Huang, Gregory Madey, Steve Cabaniss, AWeb-based Collaboratory for Supporting Environmental Science Research, *Proceedings of WSS'03*, 2003, pp. 29-26.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 81-87, 2004.

Estimating Size of Search Engines in an Uncooperative Environment

Surendra Karnatapu, Karthik Ramachandran, Zonghuan Wu,
Biren Shah, Vijay V. Raghavan, Ryan Benton
The Center for Advanced Computer Studies
University of Louisiana at Lafayette
{skk0487, kxr3869, zwu, bshah, raghavan, rgb8817}@cacs.louisiana.edu

Abstract

The number of documents that are indexed by a search engine is referred to as the size of the search engine. The information about the size of each underlying search engine is essential for any metasearch engine to conduct search engine selection, result merging and a few other processes. Thus, effectively estimating the size of search engines is important for a metasearch engine that incorporates multiple autonomous search engines. In this paper, we propose an algorithm that achieves better accuracy compared to the other existing methods for estimating the size of search engines, without losing efficiency. Compared to the Sample-Resample approach, which is the best-known approach in literature, our technique also shows much better tolerance to unfavorable environments.

1. Introduction

One of the major observations in distributed information retrieval in the past few years has been that no single search engine indexes the entire web or even a large portion of it [7]. This observation has led to the development of integrated tools to create metasearch engines, which are built on a number of individual search engines. With each of the search engines indexing certain part of the web, a metasearch engine can achieve better coverage by concurrently searching many search engines. But to be able to do so, the builders of the meta-search engine must address two key issues. These are 1) acquiring information about each of the search engines (*Resource Description*), and 2) selecting a subset of the resources (underlying search engines) for a given query (*Resource Selection*). The metasearch engine then merges the ranked results returned by the different search engines before presenting it to the user (*Result Merge*). Statistical approaches have been widely used for addressing the above issues in current metasearch engine systems. One essential piece of information required by such approaches is the size of each of the underlying search engines, i.e. the number of documents indexed by each of the search engines. Usually, it is difficult for a metasearch engine to obtain this information when search engines are not cooperative and hence do not provide the required information. In such situations, it becomes necessary to develop techniques that can estimate the size of search engines.

A few methods have been proposed to estimate the size of a search engine in uncooperative environments and they will be briefly reviewed in section 2. In section 3, we propose an approach for Boolean search engine systems that provides higher estimation accuracy than the best available method with comparable efficiency. In section 4, we explain our experimental setup. In Section 5, we explain our results, analyze them and highlight key reasons as to why our approach scores over the others. Finally, in Section 6, we conclude and present the future work, which is a sketch of our effort to further validate and improve our approach.

2. Related Work

To the best of our knowledge, there are three algorithms that have been proposed for estimating the size of a search engine. They are Interval Estimation based on Sample Data [1], Capture-Recapture [2], and Sample-Resample [3]. In this section, we briefly review the three approaches.

The Interval Estimation based on Sample Data [1] technique uses a pair of independent query terms, say (t_1, t_2), to estimate the size of a search engine. The number of documents containing either of the terms (t_1 or t_2) and the number of documents containing both the terms (t_1 and t_2) are found by sending distinct queries to the search engine. The estimate is then computed using probabilistic independence criterion. However, the problem of finding independent terms is not trivial and was not discussed in the paper. The author manually created a list of term pairs; the two terms in each pair are assumed to be independent. To achieve even reasonable accuracy, averaging the estimate values over a number of individual estimates becomes necessary. This degrades the efficiency of the technique.

The Capture-Recapture [2] technique assumes there are two (or more) independent samples from a population. Let N be the population size, A be the event that an item is included in the first sample, which is of size n_1, B be the event that an item is included in the second sample, which is of size n_2, and m_2 be the number of items that appears in both samples. The probabilities of events A and B, and the relationship between them, are shown below.

$$\mathrm{P}(A) = \frac{n_1}{\mathrm{N}} \tag{1}$$

$$\mathrm{P}(B) = \frac{n_2}{\mathrm{N}} \tag{2}$$

$$\mathrm{P}(A|B) = \frac{m_2}{n_2} \tag{3}$$

Since the two samples are assumed to be independent,

$$\mathrm{P}(A|B) = \mathrm{P}(A) \tag{4}$$

Thus, the population size is estimated as

$$\mathrm{N}_{est} = \left(\frac{n_1 * n_2}{m_2} \right) \tag{5}$$

The above technique was applied to estimate the size of a search engine by sending random queries to the search engine and then sampling from the result documents (ids). Since this method depends heavily on the number of probe queries and the sample documents to achieve good accuracy, a large number of sample queries need to be sent; hence, the technique might not scale well for large search engines [3]. Also, the technique estimates the search engine size using formula (5), based on the assumption that the two samples that are being chosen are independent. However, as mentioned in [3], documents with large number of terms, and greater diversity of terms, are more likely to be retrieved by queries and hence, some document ids in the two samples are likely to be redundant. This may result in violation of the independence condition, which would affect the accuracy of the size estimation.

The Sample-Resample [3] algorithm is the best-known method in the literature, in terms of accuracy and efficiency. It uses terms from the resource description to actually query the database. The resource description (created using the query based sampling technique [4]) is assumed to be present before the estimation is done. The basic assumption behind this technique is that, if the resource description is a good representation of the document collection (i.e. the appearance statistics for each term in the representative corresponds to those in the actual database), the probability of finding a term in the resource description is equal to the probability of finding the term in the database. In other words, let

D be the number of documents in the actual database[*],

D_T be the number of documents containing term t in the actual database,

D_R be the total number of documents in the resource description (size of the resource description), and

D_{RT} be the number of documents containing term t in the resource description,

The Sample-Resample technique assumes that, for any given term t, the condition

$$\frac{D_T}{D} = \frac{D_{RT}}{D_R} \tag{6}$$

holds. Thus, the number of documents in the document collection can then be estimated as follows:

$$D = \frac{(D_T * D_R)}{D_{RT}} \tag{7}$$

The result is averaged over a number of sample queries.

In the ideal case, when the resource description is a very good representative of the actual database, the above assumption is valid because the document frequency of term t in the resource description is proportional to its document frequencies in the actual database. However, in a real life scenario, it is impractical to expect the condition to be satisfied for all terms so that the estimation accuracy would depend on the terms that are chosen to query the database. It is not trivial to find terms that are proportionately represented in the resource description and the actual database, since the search engine is a black box to the meta-search engine. However, experiments show that this method achieves better accuracy than the capture-recapture method. Also, one advantage of this technique is it requires very few queries [3] (as low as five) to probe the database. Thus, its efficiency (the time taken to estimate the size of a search engine which directly depends on the number of probe queries) is much better than the other two techniques mentioned previously.

3. Independence Controlled Sample Size Estimation

In this paper, we propose an approach called *independence controlled sampling,* shown in Figure 1. Our approach makes an assumption different from the one made by the Sample-Resample approach. Our assumption is that the resource description is a good sample of the

[*] In this paper, the terms "database" and "search engine" are used interchangeably.

```
1. while (stopping condition not reached)
   1.1. Select a pair of terms from the resource description such that the two terms are independent (The exact method used to check the independence is explained later in Section 3.1).
   1.2. Query the Document Collection for finding the following values: $D_1$, $D_2$ and $D_{1\cap 2}$
   1.3. The estimate obtained from this pair is computed as $Est_i = \left(\frac{D_1 * D_2}{D_{1\cap 2}}\right)$   (8)
   1.4. Corrected Estimate ($Est_{ic}$) = Correction Factor (CF) * $Est_i$, where,
        CF = (resource description estimate / resource description size)
           = $\frac{\left(\frac{D_{R1} * D_{R2}}{D_{R1\cap R2}}\right)}{D_R}$   (9)
2. The final estimate is averaged over $n$ individual estimates
   $Est_f = \frac{\left(\sum Est_{ic}\right)}{n}$
```

Figure 1. Search Engine Size Estimation

actual database, as far as the relationship between the terms within the resource description is concerned. That is, terms that are independent in the actual database are also independent in the resource description. Similar to all three approaches mentioned above, we assume that the search engine provides information about the number of documents that match a given query. Based on the above assumptions, our method selects a pair of independent terms from the resource description. The selected terms are then sent to the database, individually and in conjunction, and the number of returned result documents are recorded. The size estimate can then be calculated, by applying probabilistic independence criterion on these numbers.

Some of the terminology used in the algorithm in Fig 1 is mentioned below:

D_1 is the number of documents containing term t_1 in the actual database

D_2 is the number of documents containing term t_2 in the actual database

$D_{1\cap 2}$ is the number of documents containing terms t_1 and t_2 in the actual database

D_R is the number of documents in the resource description

D_{R1} is the number of documents containing term t_1 in the resource description

D_{R2} is the number of documents containing term t_2 in the resource description

$D_{R1\cap R2}$ is the number of documents containing terms t_1 and t_2 in the resource description

There are three points in Figure 1, we would like to elucidate further:

1. In Step 1.1, the algorithm finds two candidate terms that, if independent, can be used to estimate the size of the actual database. We propose two methods to control the independence of the terms namely: (1) the independence criterion in the descriptive statistics and (2) the inferential statistics-based chi-squared test. Once the independent terms are obtained, they are used to estimate the size of the actual database. The two methods will be discussed in section 3.1.
2. The second issue is about an appropriate stopping condition to be used. The stopping condition can either be a simple one like, a predetermined number of term pairs; or, to achieve more accuracy, a slightly complicated condition, which takes into account factors such as the convergence of the average estimate at every iteration. (In this paper, we used the simple condition for our experiments with the number of term pairs fixed at 5).
3. The third issue is a so-called *correction factor* applied to the final estimate in Step 1.4. Since we are using probabilistic statistics to find the independent terms, the two terms thus found may still not be truly independent. This could introduce an error in the final estimate value. A correction factor is used to reduce this error. Since the resource description is assumed to be a good sample of the actual database, the percentage error in estimating the size of resource description and the actual database is assumed to be the same. The two terms found to be independent are used to estimate the size of the resource description. Since the actual size of the resource description is a known piece of information, the correction factor can be computed as the ratio between the estimated size and the actual size of the resource description.

The accuracy of both techniques (Sample-Resample and Independence Controlled Sampling) depends on the faithfulness of the resource description (in a faithful resource description, the probability that a term appears in a document in the resource description should equal the probability that it appears in a document in the actual database). The faithfulness of a resource description cannot be guaranteed in an uncooperative environment. As illustrated in [4], it depends on several factors such as the initial query term, number of query samples, number of documents stored at each stage and so on. Thus, size estimation methods that depend on resource description faithfulness could be critically impacted by unfaithful resource descriptions. By choosing term independence for estimation and by using a correction factor from the resource description estimate to account for the independence error, our technique is much more flexible and robust to fluctuations in the resource description quality. On the other hand, the accuracy of the Sample-Resample technique is tightly coupled to the faithfulness of the resource description and hence is affected to a greater extent by fluctuations in resource description faithfulness. The above points will be explained in more detail in section 5.1.

3.1. Finding Independent Terms

In this paper, we have used two techniques to check term independence. The first one, which is more primitive, uses the simple descriptive statistics based independence criterion to check if the two terms are independent. For any two terms t_ι and t_2, the independence criterion is specified as follows:

$$\left|P(t_1 \cap t_2) - P(t_1) * P(t_2)\right| < \mu \qquad (10)$$

where, $P(t_1)$ is the probability that a document picked randomly from the sample contains term $t_{1.}$

$P(t_2)$ is the probability that a document picked randomly from the sample contains term $t_{2.}$

$P(t_1 \cap\ t_2)$ is the probability that a document picked randomly from the sample contains term t_1 and $t_{2.}$

μ is a threshold which is set to a low value.

Table 1. Sample Contingency Table.

		t_2		
		0	1	
t_1	0	f_{00}	f_{01}	*Row_0_total*
	1	f_{10}	f_{11}	*Row_1_total*
		Col_0_total	*Col_1_total*	*Sample_Size*

The second technique is the inferential statistics-based chi-squared test for independence [8]. The relationship between the variables being tested for independence is represented using a contingency table. A sample contingency table is shown above. The chi-squared test of independence is done as follows:

1. State the null and alternative hypothesis. The null hypothesis states that the two terms are independent whereas the alternative hypothesis states that the two terms are not independent.
2. The contingency table (Table 1) is then populated with the observed frequency values (f_{ij}) (number of documents) for each of the cells, from a sample chosen randomly. The subscripts in the observed frequencies denotes the presence or absence of the particular term for example, the value f_{10} would denote the number of documents having term t_1 but not term t_2, the value f_{11} would denote the number of documents with both terms and so on.
3. The next step is to assume independence and compute the expected frequency (e_{ij}) (which must be greater than or equal to five) values from the observed frequency values as follows:
$$e_{ij} = \left[\frac{(Row\_i\_Total)*(Colomun\_j\_Total)}{Sample\_Size}\right] \qquad (11)$$
4. After finding the expected frequency and observed frequency values, the test statistic (χ) is found using the formula:
$$\chi^2 = \sum_i \sum_j \frac{(f_{ij} - e_{ij})^2}{e_{ij}} \qquad (12)$$
The summation is done over all the rows and columns.
5. Once the test statistic is found, based on the required level of significance, it is compared with the test statistic value at the required level of significance (from the chi-squared distribution table). If the obtained statistic is greater, the null hypothesis is rejected, else it is accepted.

4. Experimental Setup

We have conducted experiments to compare the performance of our algorithm with the Sample-Resample algorithm, since the Sample-Resample algorithm performs best among all three existing algorithms that were described in section 2. We also present the results for our technique without the correction factor, in order to illustrate the importance of the correction factor. For the purpose of comparison, we used the same two test beds that were used in [3] which are described below:

1. **Trec-123-100col-bysource**: The test bed contains 100 small databases from the TREC-123 collection. The sizes of the databases are not skewed and the databases themselves are organized by source and publication date.
2. **Trec123-10col:** This test bed was created to test the effectiveness of algorithms on larger databases. For this purpose, ten large databases were created as explained below: The Trec-123-100col-bysource collection was first sorted alphabetically. The first large database was created, by combining every tenth database of Trec-123-100col-bysource starting with the first. The second large database was created, by combining every tenth database starting with the second and so on.

We simulate a search engine (using Boolean Retrieval Model) on every database in each of the test beds.

4.1. Building Resource Description

Resource Descriptions were built for each of the databases in the two test beds using query-by-sampling technique proposed in [4]. Thus, query terms were selected randomly and submitted to a search engine, and the top four documents were retained. This process was carried on until *300* documents were accumulated in the resource description. Resource Descriptions are generally judged based on their *goodness*, which is a complex and abstract notion, difficult to measure. Faithfulness of a resource description is only one of the many factors that contribute towards goodness. In our experiments, we use goodness of a resource description rather than the faithfulness because the goodness of a resource description is a more formal way of evaluating a resource description. Our intuition is that, a good resource description would more or less be a faithful representative of the actual database whereas the vice versa need not be true. Hence, measuring the performance of the estimation algorithms on the basis of goodness instead of faithfulness would yield more comprehensive and reliable results. One measure that has been widely used for measuring resource description goodness is the *Collection Term Frequency ratio* or the *ctf* ratio. It was suggested by [4] for measuring the goodness of a resource description. Essentially, the *ctf* ratio gives a measure of the number of terms in the database that are covered by the resource description. The *ctf* ratio is computed as below:

$$ctf\ \text{ratio} = \left(\Sigma_{i\in v^r}\ ctf_i \right) / \left(\Sigma_{i\in v}\ ctf_i \right) \qquad (13)$$

where ctf_i is the collection term frequency of the term '*i*' (Number of occurrences of the term in the database), *v* is the vocabulary in actual database and v^r is the vocabulary in the resource description. A larger *ctf* ratio denotes better coverage of terms and hence a better resource description.

5. Experimental Results and Analysis

The Mean Average Error Ratio (*MAER*) measure proposed in [3] was used to compare the accuracies of the estimates obtained by the Sample-Resample and Independence Controlled Sampling techniques. The MAER is computed as follows:

$$MAER = mean\left[\frac{(Actual\_Size\_of\_SE) - (Estimated\_size\_of\_SE)}{Actual\_Size\_of\_SE}\right]$$

As can be seen from Table 2, the estimates obtained by the Independence Controlled Sampling approach with correction factor applied is more accurate (at least *10%* better in terms of *MAER*) than the ones obtained by the Sample-Resample for both large databases and small databases. Even without using the correction factor, the Independence Controlled Sampling method obtains better estimates than the Sample-Resample method as can be seen from Table 3. However, the improvements are less significant.

We also studied the effect of the optimality of the resource descriptions on the accuracy of the estimates. For this purpose, the databases were grouped based on the *ctf* ratios of their resource descriptions and the *MAER* for each of these groups was computed. For example, all resource descriptions with *ctf* ratios in the range *0.9* to *1.0* were grouped under one category, those in the range *0.8* to *0.9* fell in another category and so on. The *MAER* for each of the categories for the two methods was then plotted against the *ctf* ratios. Figure 2 shows that the effect of goodness of resource descriptions on Sample-Resample is much more serious whereas its effect on our method (with or without using the correction factor) is less dominant.

Table 2. Accuracy of estimation algorithms based on Mean Average Error Ratio

Collection Method	Trec-123-100col-bysource	Trec123-10col
Sample-Resample	0 .316	0.378
Independence Controlled Sampling (Using independence criterion)	0. 191	0.274
Independence Controlled Sampling (Using chi squared test)	0.192	0.238

Table 3. Accuracy of estimation algorithms based on Mean Average Error Ratio (without CF)

Collection Method	Trec-123-100col-bysource	Trec123-10col
Sample-Resample	0 .316	0.378
Independence Controlled Sampling (Using independence criterion)	0.288	0.294
Independence Controlled Sampling (Using chi squared test)	0.286	0.352

We analyze the above results in detail from two aspects; which are effectiveness (accuracy of estimation), and, efficiency (number of probe probing queries sent to the database).

5.1. Effectiveness

As far as accuracy is concerned, it can be seen from Table 2 that our method outperforms the Sample-Resample method.

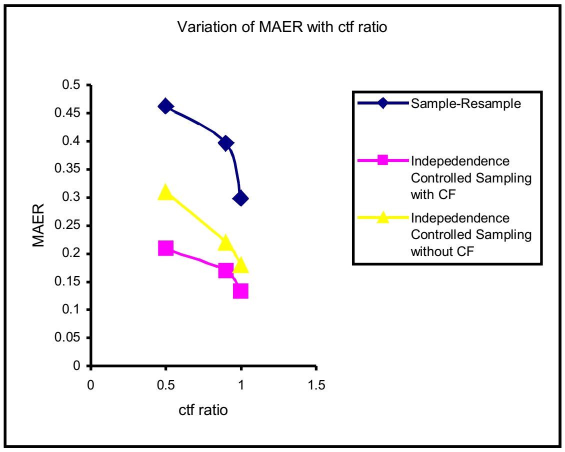


Figure 2: Variation of Mean Average Error with *ctf* ratio.

From Figure 2, we can further observe that the accuracy of the estimate given by the Sample-Resample algorithm depends a great deal on how good the resource description. If *ctf ratio* is taken as the criteria for judging a resource description, then, for smaller databases, resource descriptions generally record most of the terms present in the search engine and hence are fairly accurate. However, as the database size grows, it becomes difficult to build good resource descriptions and the assumption, the document frequency for most terms are same in the resource description and the actual database', becomes weaker and less convincing. As can be seen from Figure 2, for *ctf ratio* values close to 1, both algorithms have very low *MAER,* whereas, at lower *ctf ratio* values, *the* estimates obtained by Sample-Resample begin to deteriorate, while, the deterioration of the estimates for the Independence Controlled Sampling technique is comparatively less severe. Furthermore, as can be seen from Table 2, using the chi-squared technique to test the term independence yields better estimates than the primitive independence criterion test plotted for large databases. A major advantage of our technique is that it is less affected by the quality of a resource description as can be seen from Figure 2 where in, the accuracy of the technique is good even when the ctf *ratio* is low.

The Sample-Resample technique obtains the estimates using statistics from both the resource description and the actual database. Because of this, the technique has no way of finding the error in their estimate in case their assumption is not met. Also, the Sample-Resample technique tends to underestimate the actual database size because, as mentioned in [3], the actual database contains a large vocabulary and the percentage of documents containing a sampled word tends to be overestimated.

On the other hand, the Independence Controlled Sampling method applies term independence to the resource description and finds ‘qualified’ terms that can then be used to estimate the size of the actual database. The use of term independence facilitates adjusting the final value (*Est$_i$* in Figure 1) with a correction factor obtained by finding the error in estimating the size of the resource description (Note that computation of this error requires only the resource description but nothing from actual database). This error is then applied to the final value as a correction. As can be seen from Table 2 and Table 3, the correction factor introduces a significant improvement in the size estimates. This is because, if the size of the resource description is wrongly estimated, then, it is reasonable that the estimates obtained for the actual database will also differ proportionally. Hence, the use of term independence for estimation gives us an intuitive means for correcting certain unbalanced estimates. By combining both term independence control and application of the correction factor, our method provides the all-important robustness, not obtainable by the Sample-Resample method. In other words, compared to the Sample-Resample approach, our approach has better “toughness” or “tenacity” to the environment (in terms of the resource description that is available).

5.2. Efficiency

The improved efficiency of our approach and the Sample-Resample approach, as compared to the other two techniques, is due to the fact that they make effective use of the resource description to choose sample query terms, provided that the resource description is a good representative of the document collection of the search engine. The number of sample queries required by our approach is almost the same as that required by the Sample-Resample approach. Since the cost of querying the search engine is dominant while the local computation costs (i.e. the computation done on resource descriptions) are negligible, it is reasonable to consider the efficiency of our method to be the same as that of the Sample-Resample approach.

6. Conclusion & Future Work

We propose an efficient and effective search engine size estimation technique that outperforms the existing techniques namely: Interval Estimation, Capture-Recapture and Sample-Resample approaches. This technique takes advantage of the use of resource description to minimize the number of sample queries to be sent to the search engine. It achieves better accuracy by applying a mechanism to select statistically independent term pairs to be used to query search engines and through a mechanism that corrects estimates using data derived from the resource description. All in all, the effect of a sub optimal resource description is less dominant on the Independence Controlled Sampling method as compared to the Sample-Resample method.

In future, we look forward to extending our study to search engines that apply Vector Space Model since only Boolean retrieval model was used in the current experimental setup. We will then further validate our approach by applying it on real time search engines such as university search engines, Google, and AltaVista.

7. Acknowledgement

This work is supported in part by the IT Initiative of the State of Louisiana to Lafayette.

8. Reference

[1] Yuxi Chen, “Statistical Methods to Estimate the Sizes of Search Engines”, Technical Report, Computer Science Department, State University of New York at Binghamton, USA,

[2] King-Lup Liu, Clement Yu, Weiyi Meng, Adrian Santoso, C. Zhang, “Discovering the Representative of a Search Engine”, Proceedings of Tenth ACM International Conference on Information and Knowledge Management, Atlanta, Georgia, USA, 2001, pp.577-579.

[3] Luo Si and Jamie Callan, “Relevant Document Distribution Estimation Method for Resource Selection”, Proceedings of 26th annual international ACM SIGIR Conference on Research and development in information retrieval, Toronto, Canada, 2003, pp. 298-305.

[4] Jamie Callan and Margaret Conell, “Query Based Sampling of Text Databases”, ACM Transactions on Information Systems (TOIS), New York, USA, 2001, pp. 97-130.

[5] Jamie Callan and Margaret Conell, “Automatic Discovery of Language Models for Text Databases”, Proceedings of the ACM SIGMOD Conference, Philadelphia, Pennsylvania, USA, 1999, pp. 479-490.

[6] Jamie Callan, “Distributed Information Retrieval”, In W.B. Croft, editor, Advances in Information Retrieval., Kluwer Academic Publishers, Boston, USA, 2000, pp. 127-150.

[7] Michael K. Bergman, “The Deep Web: Surfacing Hidden Value”, Journal of Electronic Publishing, University of Michigan Press, Michigan, USA, 2002, pp. 72-78.

[8] Shusaku Tsumoto, “Statistical Independence as Linear Independence”, Electronic Notes in Theoretical Computer Science, Department of Medical Informatics, Shimane Medical University, School of Medicine, Shimane, Japan, 2003, Volume 82, Number 4, 12 pages.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 88-93, 2004.

PPDN — A Framework for Peer-to-peer Collaborative Research Network

Vlado Keselj and Nick Cercone

Faculty of Computer Science, Dalhousie University
E-mail: {vlado,nick}@cs.dal.ca

Abstract

PPDN, Push-Pull Distribution Network, — a proposal for a novel framework for peer-to-peer collaborative research network is presented. Some requirements not addressed by the currently proposed systems are discussed, and we show how these issues are addressed in our framework. The framework is based on a distributed approach and the concept of semantic web. The collaborative network is represented as a graph, with push and pull edges. The nodes can act as autonomous or semi-autonomous agents, implementing different policies.

1 Introduction

The Internet is continuously maturing from its early years of exciting but somewhat mechanical and static applications and protocols toward a more flexible and more intelligent network. Although the traditional means of communication and information sharing on Internet, such as e-mail, WWW, or Usenet, still require further research to address the problems such as spam, authentication, and information privacy, we can say that their scope and usage are well-understood. Under this umbrella of traditional methods, we could add search engines, database interfaces, e-mail lists, and web-based forums. The new level of integration and collaboration includes the so-called groupware applications, peer-to-peer systems, and similar kinds of distributed systems.

From a vast area of different application domains we limit our domain to the web-based research support systems. To give a motivation for such system, we list some of the activities from the life of a typical researcher X that are not well supported currently:

- easy access to relevant publications and to corresponding meta-data (e.g., BibTeX entry),
- keeping track of X's publications, in X's own database, using it to generate a Web list, in her/his CV, grant applications etc.,
- passing publications or their metadata to the research group(s) web sites, selectively, to co-author, collaborators, organizational web site, wider research community, research search engines, and similar,
- receiving information about new publications, conference announcements, calls for papers (CFPs), software releases, books, and similar.

While an obvious item of exchange described above is a publication, there are several different types of information that require similar kind of dissemination:

- publications and publication metadata,
- software and software metadata,
- conference calls for papers (CFP), and
- links, web resources, and web services.

Additionally, in order for our application to be useful and to be used, the experience has shown that the following requirements also need to be satisfied:

Low maintenance: The researchers are usually happy to share their contributions, but they refuse to put any significant work into preparing meta-data and system maintenance [6].

Non-centralized: Non-centralized solutions do not scale very well. They also represent a one-size-fits-all approach, which hardly fit in a wide domain such as scientific research.

Flexible: Setting elaborate and rigid standard and frameworks in advance would be premature. It is hard to predict future requirements, and complex standards require time to learn and train. It is desirable for a standard or framework to be learnable incrementally—learn only as much as you need. Such flexibility would provide an environment for emerging standards and solutions.

Under flexibility, we also assume connection flexibility. Instead of a rigid distributed system depending on real-time communication among peers, we put forward a network for information dissemination using push and pull communication links.

2 Related Work

We divide the related work into two groups: the centralized repositories and peer-to-peer (P2P) systems.

Centralized repositories. The centralized research repositories are available to the scientific community for several years now. Some of them are CiteSeer[1] since 1999, DBLP[2], CS BibTeX[3], CompuScience[4], CoRR The Computing Research Repository or arXiv[5], NZ-DL[6], Zentralblatt MATH[7], and MathSciNet[8].

While they have proved to be invaluable to the research community, showing that they do scale up to certain nontrivial amount of publications[9], these centralized sites also confirmed weaknesses of the centralized approach. They are limited in scalability, for user connection as well as for submissions. The user completely depends on the connection to the site: so if the site is too busy, or simply down, the system is unusable. A user is not provided with software to maintain his own database of publications.

A new solution is needed, but still we would like to make the centralized repositories part of it. One step in this direction is CiteSeer's compliance with the OAI—the Open Archives Initiative protocol for metadata harvesting[10].

Peer-to-peer systems. Several P2P projects to support research are described recently.

Werlen 2003 [6] presents the DFN—the German Research Network, which is a non-profit organization that provides research infrastructure in Germany. The focus of the project is on the search capability in indexing and gathering scientific information. It is a peer-to-peer network that uses JXTA[11] open search protocol. An important fact noted in [6] that the messages in the network are much more efficiently exchanged if the network is organized around 'super-peers' or hubs, so that the small-world phenomenon can be exploited, i.e., the routes from peer to any other peer are always short. Another important observations are that researchers do not want to invest any significant amount of time to prepare data, and the open networks are prone to spam data, i.e., material inappropriate for the network.

Haase and Siebes 2004 [4] discuss peer selection in peer-to-peer networks with semantic topologies. The focus of this paper is on finding a peer in a peer-to-peer network that has relevant information for our query. Instead of the traditional approach where a query is broadcasted to all peers, they propose that peers advertise their expertise, which is organized into a semantic network. The approach resembles multi-agent systems proposed for distributed information retrieval about ten years ago, e.g., see [5].

The focus of our approach is different being focused on an approach of selective information dissemination instead of active peer querying, however an important commonality with [4] is the domain of application. Haase and Siebes [4] consider the case study of bibliographic metadata about publications, which is included in our target domains. The common ontology used in [4] is the Semantic Web Research Community Ontology (SWRC) [2].

Ahlborn *et al.* [**?**] discuss how an existing peer-to-peer system Edutella could be reused to provide OAI repositories with search capability.

Very recently, BIBSTER[12]—an open source P2P system for managing, searching and sharing bibliographic metadata from BibTeX files was announced [3]. The system is implemented in java on top of the JXTA platform. It provides search capability by routing the query to peers. Bibster is an application based on technology that combined Semantic Web and P2P technologies. It does not have centralized control.

3 Problem Specification

As we saw in the previous section, we could roughly divide the existing approaches to the problem of web-based research support into two groups: (1) centralized publication repositories like CiteSeer and arXiv, and (2) new distributed approaches such as Bibster, which somewhat resemble the multi-agent systems for information retrieval being proposed several years ago [5]. While we find the both of these approaches useful, we would like to offer a new peer-to-peer approach called *Push-Pull Distribution Network* (PPDN) to address certain applicative approach. The PPDN framework is designed to address the following issues:

- The weaknesses of centralized sites are well-known [3]: centralized server, which can be a single point of failure, it does not scale well with the number of users nor data items, complete dependence on direct network connection to the server and on its bandwidth and delay.

[1] http://citeseer.ist.psu.edu/

[2] http://dblp.uni-trier.de/

[3] http://liinwww.ira.uka.de/bibliography/index.html

[4] http://www.zblmath.fiz-karlsruhe.de/COMP/quick.html

[5] http://arxiv.org/archive/cs/intro.html

[6] http://www.nzdl.org/

[7] http://www.emis.de/ZMATH/

[8] http://www.ams.org/mathscinet/search

[9] DBLP announced recently that they reached 520,000 papers.

[10] http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm

[11] http://www.jxta.org

[12] http://bibster.semanticweb.org

The existing P2P approaches have other issues:

- They typically require significant effort from users involved w.r.t. maintaining the peer system and keeping it on-line.
- While in a centralized system we rely on one host, which is normally reliable, in a P2P system we rely each time on different hosts, with expected higher probability that one of them may be off-line.
- P2P system relies on distributed querying in search for relevant publications. Routing, dividing, and merging such queries is a complex problem, it wastes bandwidth due to a lot of redundancies, and may have longer delay than the centralized approach. It is reported that such systems may produce too many queries if the network topology is not carefully designed [4].

These issues are addressed in the PPDN approach in the following way:

- PPDN is a distributed approach that does not require a centralized server. The users can with little effort keep their PPDN nodes, connect and disconnect them in a flexible way without disrupting significantly the system as a whole.
- We delegate the issue of searching and querying to reliable and high-performance servers, which are part of the PPDN network. These are equivalent to 'superpeers,' or hubs, as called recently. The long experience with information retrieval on the Internet provides arguments that a very distributed approach to information retrieval would not perform favorably compared to strong and reliable single-site search engines.
- The issue of relying on some peers to be on-line in a typical peer-to-peer system in a moment when we need information is addressed in PPDN by using the push and pull transfer of information. Rather than waiting for the moment when we need information, we focus on information dissemination, so that by the time we need information, it is available either locally or it is stored in a search engine repository. Thus, the system reliability is improved. In our prototype system, we rely on the e-mail protocol, SMTP, as the transport protocol, which further improves system robustness, since SMTP transfer can be performed over relays, not requiring that a sender and a recipient are on-line in the same time. The search and retrieval task is left to a centralized repositories which are part of PPDN.
- The network is semi-autonomous, allowing users by creating forwarding policies to create sub-networks, networks of trust, and to avoid spam.

```
X-DBWorld-Message-Type: conference/announcement
X-DBWorld-Name: iiWAS2004
X-DBWorld-Start-Date: 27-Sep-2004
X-DBWorld-Location: Jakarta; Indonesia; Asia
X-DBWorld-Deadline: 23-Jul-2004
X-DBWorld-Call-For: papers, demos, reports,
X-DBWorld-Web-Page: http://www.iiwas.org/conf...
```

Figure 1. DBWorld Example

PPDN Description. A PPDN is a network of nodes, i.e., a directed graph with two kinds of edges: *push* and *pull*. Each site is a semi-autonomous site since it can automatically forward or store received data items, or it can be moderated by a user. The data items are transfered through the edges using a transport protocol. In our prototype, we are use SMTP and CGI as the transport protocols. In a *push* connection, the sender initiates data transfer; e.g., buy sending an e-mail through a link; while in a *pull* connection, the receiver initiates transfer, for example, by accessing a web site, running a CGI script, or sending a query by e-mail. In our prototype, the pull connections are implemented using the CGI protocol.

4 Examples

The PPDN framework is not a completely new idea: it is more a matter of gluing and merging existing pieces than designing and launching a new paradigm from the scratch.

Example 1. The members of the DBWorld mailing list[13] may have noticed that the information about conference announcements is encoded using the RFC 822 standard into headers of the list e-mail messages. This represents an elegant example of a transition from natural-language-only to semantic web style of informing. An example is given in Figure 1

Example 2. The second example is taken from the arXiv mailing list [14] and it is shown in Figure 2. This is example of an e-mail list used to disseminate publication information in a well-formated, but still user-readable style. The formatting is similar to Example 1, following the style of the RFC 822 headers. The arXiv mailing list is integrated with the CoRR repository of the publications with a search interface.

Example 3. The third example presents an e-mail message used to disseminate information about new links available at the ACL NLP/CL Universe[15]. The ACL Universe

[13] http://www.cs.wisc.edu/dbworld/

[14] http://arXiv.org

[15] http://perun.si.umich.edu/˜radev/u/db/acl/

```
--------------------------------------------...
 Submissions to:
Computational Complexity

 received from  Thu  1 May 03 20:00:02 GMT  t...
--------------------------------------------...
\\
Paper: cs.CC/0305035
Date: Mon, 19 May 2003 16:02:54 GMT   (3kb)

Title: P is not equal to NP
Authors: Craig Alan Feinstein
Comments: The body is less than 2 pages and e...
  recently submitted to the SIAM Journal of D...
Subj-class: Computational Complexity
ACM-class: F.1.3
\\
  The question of whether the class of decisi...
solved by deterministic polynomial-time algor...
the class of decision problems that can be so...
polynomial-time algorithms (\textit{NP}) has ...
first formulated by Cook, Karp, and Levin in ...
prove that they are not equal by showing that...
solves the SUBSET-SUM problem must perform at...
\lfloor\frac{n}{2} \rfloor}$ computations for...
${\rm O}(n^2)$, where $n$ is the size of the ...
\\ ( http://arXiv.org/abs/cs/0305035 ,  3kb)
```

Figure 2. The arXiv Example

web site contains a hierarchy of links with descriptions, which is browseable as well as searchable. The hierarchy is encoded using attributes ‘cat1,’ ‘cat2,’ ‘cat3,’ and ‘cat4.’ The format is similar to previous examples, being text-based and having attributes and values paired at each line.

5 PPDN Framework

PPDN. Push-Pull Distribution Network (PPDN) is directed graph with two kinds of edges: *push* and *pull* edges. For two vertexes a and b, there may exist two edges (a, b), one push and one pull edge. The nodes can be regarded as information repositories. The direction of edges describe information flow. In a push edges, the information transfer is initiated by the source node, while in a pull edge the transfer is initiated by the destination node.

The transfers are either triggered by an event, they are invoked periodically, or they are invoked manually.

Node structure. The structure of a node in the network is shown in Figure 4. The information is received through in-edges and disseminated through out-edges. The edges are grouped into channels. For example, a channel is a list of e-mail addresses to be informed about new items. In the prototype we use plain text files for site archives, but one could use any database engine as well. The suggested organization of the archive is into the personal archives and cache archives.

```
--------------------------------------------...
Additions to the ACL NLP/CL Universe:
June 7 - October 20, 2003
--------------------------------------------...

link_id L000002988
url     http://cf.hum.uva.nl/computerlinguist...
title   Amstelogue\'99 - Workshop on Dialogue
author
cat1    CONFERENCE
cat2    1999
cat3    5
cat4
email
annotation      May 7-9, 1999, University of ...
date_added      Wed Jul 23 12:26:50 EDT 2003
date_indexed
```

Figure 3. The ACL NLP/CL Universe Example

The moderating module acts as a semi-autonomous agent. It can be configure to automatically store or forward received data items, wait for user approval, or drop them, based on a set of rules. The rules depend on the incoming or outcoming channel, but they can also be arbitrary regular expression-based rules on data items.

A typical scenario is the following: A researcher X would set up his own PPDN site. A department, research groups, projects, and collaborators would also have defined sites. A site typically would have a Web interface to produce a list of items.

Communication Issues. There are several communication issues that needs to be addressed in a PPDN network:

access control: If our site is source pull site, we may need a protocol to restrict access to the site based on channel. This can be solved in various ways based on the transport protocol. In our case, the CGI access is regulated through htaccess method.

authentication: If we are the receiver push site, we need a way to authenticate the sender. Since the transport protocol for push edges is SMTP, we use GPG (or PGP) public key signatures for authentication.

encryption: If we need encrypted transport, so that a third party cannot observe data transfer, in case of SMTP a GPG/PGP-based encryption is used, and in case of CGI, HTTPS protocol is used.

Transport protocol. In the prototype we use SMTP transport protocol for push, and CGI for pull edges. However, a whole slew of alternative transport protocols is available: SOAP, web services, scp and ssh, ftp, being among them.

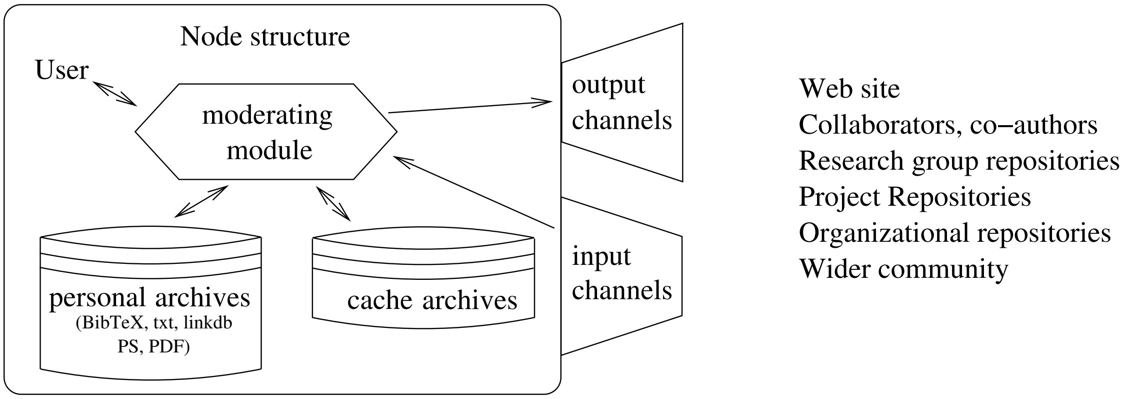


Figure 4. Node structure

Data items. The following data items are exchanged in our prototype:

- publication metadata,
- conference CFPs and announcements,
- software metadata,
- links (e.g., resources, web services),
- e-mail list metadata,
- publications, and
- software.

The set of items and their ontology is not rigidly defined, so in this prototype stage, the network can be used even as a distributed e-mail list—where there is no a centralized server, but each user can decide to have his own redistribution list.

There is an issue of infinite forward loops, which is resolved by keeping MD5 digests of passed data items in the cache archives, and dropping the ones that are repeated.

Encoding The standard encoding schemes used in similar semantic networks are XML and OIL. While we intend to provide a compliant translation into these standards, the prototype is based on a simpler encoding scheme, similar to the RFC 822 e-mail headers standard and YAML standard. An example of encoding of a CFP is given in figure 5. Several data items are separated by blank lines. Within a data item, each line starts with an attribute ending with a colon (:). A line may be continued by starting the next line with space or tab, or by ending the the line with backslash (\).[16] If a binary data needs to be encoded, an encoding standard, such as BASE64 is used. This is needed usually when large data items, such as papers or software is passed.

```
Type: conference/announcement/cfp
Name: WSS'04
Full-name: The Second International Workshop on
           Web-based Support Systems
Comments: In conjunction with 2003 IEEE/WIC/ACM
          International Conference on Web Intelligence
Location: Beijing, China
URL: http://www2.cs.uregina.ca/~wss/wss04/
Due: 20-Jul-2004
Start-Date: 20-Sep-2004
```

Figure 5. CFP Example

Policies. There are four kinds of policies defined for a moderating module in a node:

receiving policy: defining whether a data item will be received at all from a channel,

storing policy: defining whether a received data item will be stored in the cache archive,

sending policy: defining whether a new data item in the archive will be sent to a channel, and

forwarding policy: defining whether a received data item will be forwarded (even if not stored in the archive.).

The policies are rule based, taking into account the receiving channel, sending channel, and based on regular expression matching on data items. There are three policy results: (1) free, i.e., passing the data item, (2) blocked, i.e., dropping (deleting) the data item, and (3) moderated, i.e,, storing a data item in a waiting queue, waiting for users decision.

[16] The difference is that a line ending with backslash, the backslash will be removed and this is a way to encode a new-line character within an attribute value. In a line continued only by space or tab in the next line, the new-line character is removed.

6 Conclusion

We presented design and a prototype implementation of the PPDN—Push-Pull Distribution Network—framework for peer-to-peer research collaboration support. The current systems were discussed and it is demonstrated how PPDN addresses a new problem specification. The framework prototype is being implemented in Perl and it will be made open-source.

Future Work. The future work includes beta testing with a group of collaborators and network of PPDN sites. A potential issue with a PPDN network is that if the nodes only periodically do forwarding and the time period is very long, or if the forwarding policy is moderating and the users do not attend their moderating duty frequently, then a significant delay in information dissemination could be experienced. This could be explored by running simulation experiments, which is a part of our future plans.

References

[1] The ACM topic hierarchy. WWW. Accessed Jul 2004 http://www.acm.org/class/1998.

[2] The semantic web research community ontology (SWRC). WWW, 2004. Accessed Jul 2004 http://ontobroker.semanticweb.org/ontos/swrc.html.

[3] P. Haase, J. Broekstra, M.Ehrig, M. Menken, P.Mika, M. Plechawski, P. Pyszlak, B. Schnizler, R. Siebes, S. Staab, and C. Tempich. Bibster — A semantics-based bibliographic peer-to-peer system, November 2004.

[4] P. Haase, R. Siebes, and F. Harmelen. Peer selection in peer-to-peer networks with semantic topologies. In *International Conference on Semantics of a Networked World: Semantics for Grid Databases*, Paris, 2004. http://bibster.semanticweb.org/publications/haase_04_peer.pdf.

[5] V. Keselj. Multi-agent systems for Internet information retrieval using natural language processing. Technical Report CS-98-24, University of Waterloo, 1998. ftp://cs-archive.uwaterloo.ca/cs-archive/CS-98-24/.

[6] R. Werlen. DFN science-to-science: Peer-to-peer scientific research. In *Proceedings of the Terena Networking Conference, TNC 2003*, Zagreb, Croatia, 2003.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 94-101, 2004.

The Design of an Integrated Web Mining and Usability System

Ringo Wai-kit Lam, Chun-hung Li, Chun-kit Chui
Department of Computer Science
Hong Kong Baptist University
chli@comp.hkbu.edu.hk

Abstract

Web usability is an important and sometimes controversial research area. We review the different approaches to web usability and illustrate that the factors influencing Web usability are often incompletely analyzed. We proposed an integrated system for web mining and usability study where four core modules are designed to address the fundamental issues in usability analysis. The integrated approach allows a totalistic view of the web usability and facilitates analysis across different modules. As an example to cross modules analysis, we apply association rule mining from the link structure obtained from web mining module to automatically discover menus and structures in a web site. Furthermore, such mining tools allow the decoupling of the design-based link structure from the contextual-based link structure.

1. Introduction

Usability test is always regarded as a costly and time exhaustive process but it is becoming prevalent and more important as many of our business operations and social activities are processed and completed on the Internet directly.

Usability evaluation is usually conducted in a number of ways, e.g. user testing, heuristics evaluation and automatic tool analysis. Since usability evaluation is very expensive, automatic tools were developed in the last few years to help the web designer evaluate the web site. However, such automatic tools still cannot replace the value of testing with the actual users.

An integrated system that enables Web Mining and Usability analysis (**Webmius**) is proposed in this paper. The system is composed of four modules that perform the function of the task-based usability evaluation, the web design and layout analysis, the web structure mining and the semantics inference.

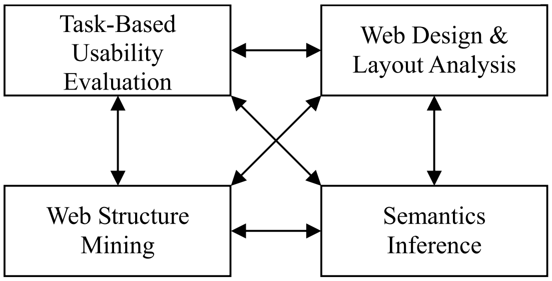


Fig.1 Web Mining and Usability System

Conventionally, web usability analysis is focused on the function of either one or two of the modules we implement. For example, for task-based usability evaluations, studies include [26, 27]. Design and layout analysis includes [17]. Web structure mining has [2, 21]. For semantics inference, there are [6, 9, 10].

The four modules shown in Fig.1 are integrated to tackle the usability evaluation. Except the task-based usability evaluation module, the other three modules may run automatically. The task-based usability evaluation module requires actual user involvement although simulated users like Bloodhound [7] and MESA [24] have been developed, the simulated users are still not mature. On the other hand, metrics are being developed for the other 3 modules. These metrics try to capture the web design parameters, such as color, font, layout, image size, etc. Some of the metrics are similar to [17] but our metrics are also emphasized on the hyperlink and menu structure as well as their positions.

Concerning the design and the layout, research study [14] found that the search time of a hierarchical and labeled layout was faster. Besides, McCarthy et al. [23] reported that users rapidly adapted to an unexpected screen layout and the internal consistency of a web site was the most important. Ken Hinckley [12] extended Fitts' Law to consider the IBM ScrollPoint and the IntelliMouse Wheel. Their experimental approach revealed a crossover effect in performance versus distance, with the Wheel performing best at short distances but the ScrollPoint performing best at long distances.

This paper first describes the background of usability metrics and testing tools. Then the characteristics and the design of our system are discussed. Our Webmius platform is described in section 5. Finally, an approach using association rule mining to remove the menu is introduced to simplify the web site structure for further analysis.

2. Usability Metrics and Evaluation Tools

There are many metrics and evaluation tools. Bobby, A-Prompt, WebSat, 508 Accessibility Suite and WebTango were developed to enable designers and evaluators to verify a page or a web site according to the guidelines [4]. Their main problem is that the guidelines are hard coded in the tool.

Ivory summarized the page characteristic metrics of the other studies in [15] that generally classified the metrics into Page Composition, Page Formatting and Overall Page. Besides, the EvalWeb project http://lis.univ-tlse1.fr/evalweb/ tried to create a framework to organize the guidelines to help people structure the guidelines and use them for design and evaluation of web sites. Guidelines were classified into 5 categories : (1.) Design rules, (2.) Ergonomic algorithms, (3.) Style guides, (4.) Compilations of guidelines and (5.) Standards.

Usability.gov that is maintained by the U.S. Department of Health and Human Services (HHS) also put over 50 Web design and usability guidelines on their web site. IEEE has a Std 2001-1999 [http://www.computer.org/cspress/CATALOG/st01117.htm] which defines recommended practices for web page design and implementation for intranet/extranet environments. The American's National Institute of Standards and Technology (NIST) [25] on the other hand developed a prototype tool WebSAT for researching usability rules. It allowed either to use its own set of usability rules or those of the IEEE Std 2001–1999.

Apart from the metrics or guidelines for the usability test, there are over 50 evaluation tools developed [16]. One of the major tools is WebTango [17]. Under the WebTango project, the 157 measures were categorized into 9 major types, (1.) text elements, (2.) link elements, (3.) graphic elements, (4.) text formatting, (5.) link formatting, (6.) graphic formatting, (7.) page formatting, (8.) page performance and (9.) site architecture. The assessment was separated into site-level and page-level. However, the site-level assessment was not comprehensive enough, it only reflected the total number of pages, the breadth and the depth traversed by the crawler.

To mine the user behavior from the task-based usability evaluation, logging tools are commonly implemented. A recent project which embeds logging design is TEA [26]. It is an open source project that develops a client-side proxy to capture the client-side events and feedback to the analysis server. WebRemUSINE [27] is similar to TEA but no client-side proxy is set up. WebRemUSINE's logging tool is able to capture the browser's event by event handler scripts. Unfortunately, the scripts are not persistent. Each page of the site has to include the script and all events are communicated to the applet that sends the server back with all the logged events at the end of the session.

WebQuilt [13] was built by the User Interface Research Group at the University of California at Berkeley. A tailor-made Java-based proxy server was developed to record the link structure of the web pages. However, the WebQuilt has its shortcomings, it cannot handle Flash, Java Applet or any kind of web page which has links or redirects created dynamically by JavaScript and other browser scripting languages cannot be handled. As a consequence, the JavaScript generated pop-up windows and DHTML menus popular on many web sites are not captured by the WebQuilt.

Heer et. al. [11] introduced an evaluation by building user profiles and combining users' navigation paths with features, such as page viewing time, hyperlink structure, and page content. They tried to find how well these features contributed to the clustering process in real world and to evaluate whether the clustering algorithms correctly categorized the user sessions so that the real user's behavior might be determined from the web log.

They used WebQuilt proxy-based logger [13] to capture all of the user sessions, therefore their system might suffer from the drawback of WebQuilt's deficiency. On the contrary, our work extends the work done by [11] to include a more comprehensive set of features and system design. Besides, we also consider the hyperlink and menu structure, and their positions in our system.

3. Web Site Structure

Conventionally, a web site structure may be evaluated with the Card Sorting technique; however, this technique is difficult to implement for a large and information-centric web site. There are a number of commercial and free evaluation tools available. Most of the tools are based on the user logging results that are stored in the proxy server or through the embedment of client-side programming code to capture the client's events.

A recent publication by Miller and Remington [24] pointed out that the structure of linked pages (the site's information architecture) has a decisive impact on the usability. Previous studies including Shneiderman [29], and Larson and Czerwinski [21] also provided suggestions on how to create the best structure.

Larson and Czerwinski [21] found that users took significantly longer time to find items in a structure with depth than breadth. They compared a three-tiered, eight-links-per-page (8 x 8 x 8) structure with two-tiered, 16 and 32 links per page structures (16 x 32 and 32 x 16).

Bernard [2] had an important contribution to the analysis of hypertext structure. He devised a metric called Hypertext Accessibility Index (HAI) to model the informational accessibility of a particular hypertext structure compared to other alternative structures. The metric was based on the Entropy theory. It explained what Larson and Czerwinski [21] as well as Kiger [20] found about the navigation time of shallow and deep structure in quantitative terms.

Although Bernard's HAI metric is useful for web structure comparison and it considers the level of depth as well as the number of hyperlink at each node, it cannot be used easily in practical because a web site structure also depends heavily on the content and the distribution probability of the information goal. The location and the design of the hyperlink also affect the navigation time.

To consider other factors on the navigation time of the web site, an entropy approach may be employed. Kao et al. [19] proposed the LAMIS method with the entropy analysis to distill the information of a web site. Rather than defining the probability term as the normalized feature frequency in the page set, we may define a probability term for the hyperlink. The probability value will depends on the position, size, menu structure, description, color and the information goal, etc. Empirically, the probability is directly linked to the transition probability of the web log.

A study on the use of entropy theory to merge the web site content was performed by Chen et. al. [6]. The merge depends on the mutual information of term w_i and w_j in the sub-tree as well as their counts. A similar approach will also be applied to our web site structure analysis in order to combine the web site content to reduce the navigation time.

In a study on web site structure, some approaches were developed to restructure the hyperlinks according to the users' browsing behavior. To achieve such goal, collaborative filtering, Markov model, Longest Repeating Subsequences (LRS) [28] or data mining techniques are commonly used. Based on the Markov model, Jenamani et al. [18] proposed several algorithms to examine (1.) the most accessed pages, (2.) the company's interest, (3.) the visitor's interest pages, (4.) the current visitor's interest pages, (5.) the customized index generation algorithms. However, our system objectives are not to provide a dynamic hyperlink structure, we will concentrate on the study of building a static optimal web site structure.

A web site is viewed as a combination of a set of pages and sub sites as shown in Fig.2. The sub-site structure may be much different from its parent web site. For example, in our department, professional short-term training courses under http://www.comp.hkbu.edu.hk/~training/ are stored in a separate directory and follow a different navigation menu structure than the departmental web site. Such design is very common in a medium to large scale web site. Therefore, to analyze a web site structure, sub sites should be isolated and analyzed in its entity.

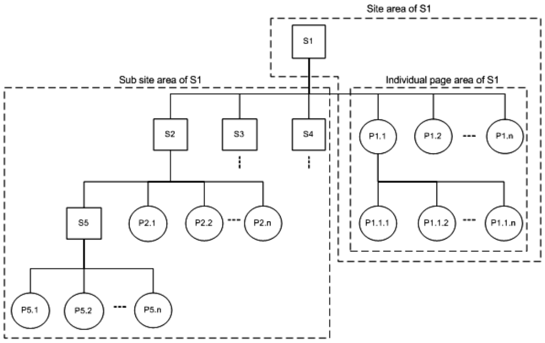


Fig.2 Web Site Structure Model (□ : Site, S1 is the parent web site, S2, S3, etc are the sub sites; ○ : Page, P1.1 refers to the first page in S1, P1.2 is the second)

4. Web Design

Web site layout is composed of many objects, such as menu, buttons, text area, image and graphics, etc. Because the users have their expectations of where web objects are located, the layout design has an influence on the navigation time. Table 1 summarizes what Bernard empirically found in the past few years [1]. These design guidelines will be embedded into the Web Design and Layout Analysis Module of our system.

Web Object	Expected Position
Internal web link	Upper left side
External web link	Right side or lower left side
“Back to home” link	Top-left
Search engine	Top-center
Ad banner	Top
Login / register button	Upper-left
Shopping cart	Top-right
Help	Upper-right
Links to merchandise	Left upper-center
Account / order button	Upper-right
Links with summaries	Most usable
Lists	Best be bulleted
Menu	Index menu accessed faster
Categorical menu	Superior in search performance
Menu links with summary text	More preferable

Table 1 : Web object and expected position

In a study on color usage, Meister and Sullivan [22] reported on the relative legibility of seven colors as a function of symbol size. It was found that white, yellow and red symbols were more easily read than the others. It was also revealed by researchers Shurtleff [30] and Durrett [8] that symbol identification accuracy was best for white and for colors near green and yellow, blue on red was slightly worse.

5. The Webmius Platform

5.1. Task-Based Usability Evaluation Module

The task-based usability evaluation module consists of a front-end web server, a back-end proxy server and a database.

In the front-end, the module has an administration area and a tester area (Fig.3). Inside the administration area, task questions and answers are input by the administrator, a set of target web page locations where the answers can be found are also input to the platform. The target locations are needed to determine whether the user has entered the target answer page before answering the question. If the user does not really enter the target page, the answer will be ignored in the analysis.

In the back-end, user activities are logged into the proxy server which is capable of capturing the query string content in the URL. Therefore, the proxy server may know what dynamic content is requested by the user. However, the access log of the proxy server contains a lot of noise such as image files and multimedia files. To remove such noisy records in the proxy server, the access log is filtered by the module and the filtered access data will be stored in the database.

To start the user test, the platform will send the participant an email which contains a total of 15 tasks, 5 tasks each web site. The tasks are randomly selected from a task pool.

After the user clicks on the corresponding hyperlink of the email, a pop-up window will be opened to start the test. Proxy server configuration is needed to set up in the user’s browser. It can be detected by examining whether a designated page requested from the user’s browser recorded in our proxy server. As the access log of the proxy server has a record of the designated page, the browser has been configured to connect to the proxy server. If the system cannot find a record in the access log, the browser is not configured properly with connection to our proxy server. In this case the platform will issue a reminder and send help procedures to guide the user to set up a connection to our proxy server.

As the proxy server is configured correctly, the user starts to click on the first task, a new window will be opened to show the tested web site. The user may browse into the web site and find the corresponding answer. After the answer is found, the user has to go back to the task window to input the answer and submit the form. Our platform will compare the proxy record to the user’s answer. If the user enters the answer without going to the corresponding webpage, the answer will be recorded but it is marked to indicate the discrepancy. There is a give-up button for each task to let the user skip a task.

After the user completes all tasks, the user has to answer a questionnaire to comment on the web site. The purpose of the questionnaire is to gather the user’s comment on the web site. These data will be analyzed later together with the task record to examine any correlation.

The user may log in to the platform to read the statistics of the web evaluation of all participants. In the later stage of the platform development, the user may even suggest a web site for evaluation and provide a corresponding set of questions and answers in the platform.

After all the users have completed the test, the administrator may export the user statistics to a text file for further analysis.

5.2. Web Design & Layout Analysis Module

After all the evaluations are completed, the data will be analyzed using the metrics stated in [17] together with the following metrics :

	Metric	Description
1	Number of hyperlinks	Only visible and internal hyperlinks except the In-Page ones are counted.
2	Hyperlink level	The level that the hyperlink resides is recorded. Level 1 is the root level which is directly visible. If there is pop-up menu or tree menu, level 2 is the first pop-up menu, level 3 is the second pop-up menu.
3	Hyperlink position	• Based on the top left position of the hyperlink • For pop-up menu or tree menu, the hyperlink position is the absolute position measured based on all other hyperlink groups closed but the interested hyperlink group opened.
4	Hyperlink size	• Text – The area covered by the text. (±5 pixels tolerance) • Graph - Image size
5	Word counts	• English : Number of words • Chinese : Number of characters • English and Chinese mix : Number of English words + Number of Chinese characters
6	Word count inside and near the hyperlink	• English : Number of words • Chinese : Number of characters • English and Chinese mix : Number of English words + Number of Chinese characters • For image hyperlink with text content, find the word count based on the text displayed. • For image hyperlink with alt, find the word count inside the alt.

Table 2 : Web Design and Layout Analysis Metrics

For each metric, a weight factor associated with the probability of the click is determined by the combined results of task-based usability evaluation module, the web structure mining module and the semantics inference module.

5.3. Web Structure Mining Module

The web structure mining tool is embedded with a spider which captures the whole web site structure. After downloading the web pages, the hyperlinks of each web page will be indexed. Based on our heuristic hyperlink analyzer, the web page will be segmented into 2 major areas, the hyperlink area and the content area. Other than analyzing with the metrics, the hyperlink type is diagnosed. Seven hyperlink types listed in Table 3 are classified.

	Hyperlink Type	Description
1	External link	Hyperlink pointing to the web site of different subdomains or domains
2	Internal link	
	- In-page link (Self loop) (index 1)	The content and the effect of such hyperlink will be ignored in the analysis
	- Intra-directory links (index 2)	Inside the same directory
	- Up link (index 3)	Point to parent directory
	- Down link (index 4)	Follow the file directory structure, point to the immediate child directory
	- Across links (index 5)	All links within a host that are not of the other types
	- Download link (index 6)	Link for downloading image, file, etc. Regarded as leave

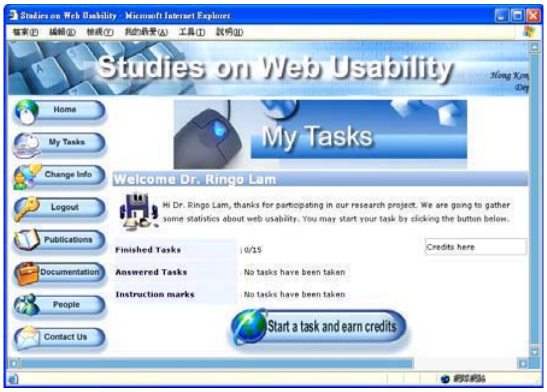
Table 3 : Hyperlink Types

External link will be ignored in further diagnosis because it points to a different web site. On the other hand, the hyperlink type is useful to identify the page characteristics, four page types can be discriminated, they are :

- Home page : is the first page of a set of pages. Its distance to other pages should be small but the number of links should not be large [3]
- Index page : is referred as TOC, i.e. table of contents page, and has a higher number of outlinks.
- Reference page : is like glossary, contains references, and has higher number of inlinks.
- Content page (leave)

The understanding of the page type is fundamental to the web site structure mining. For example, index page is designed for navigation purpose. Information searching should not return the index page.

Besides, the text around the hyperlink is an important metric in analyzing the web page relationship. For example, it was proposed by Chakrabarti [5] to help finding the authority, that is the reference or the content page.

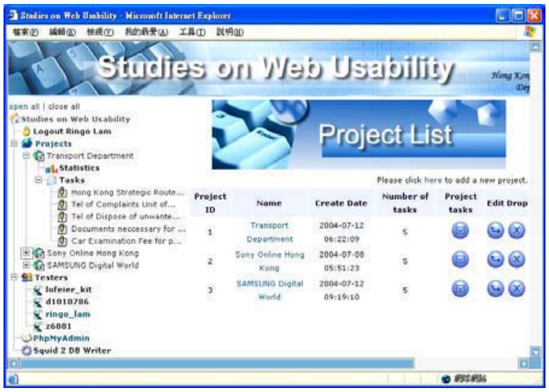

Studies on Web Usability - Microsoft Internet Explorer
檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)
Studies on Web Usability
Hong Kon
Dep
Home
My Tasks
Change Info
Logout
Publications
Documentation
People
Contact Us
My Tasks
Start a task and earn credits

Welcome Dr. Ringo Lam

Hi Dr. Ringo Lam, thanks for participating in our research project. We are going to gather some statistics about web usability. You may start your task by clicking the button below.

Finished Tasks	0/15	Credits here
Answered Tasks	No tasks have been taken	
Instruction marks	No tasks have been taken	

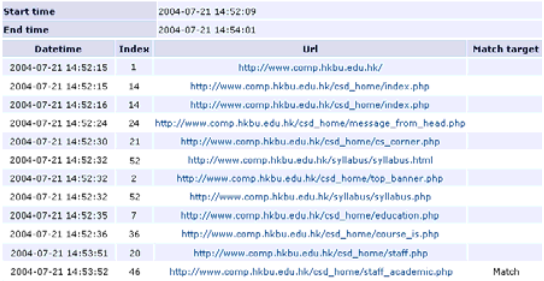
(a.) User test area



Please click here to add a new project.

Project ID	Name	Create Date	Number of tasks	Project tasks	Edit Drop
1	Transport Department	2004-07-12 06:22:09	5		
2	Sony Online Hong Kong	2004-07-08 05:51:23	5		
3	SAMSUNG Digital World	2004-07-12 09:19:10	5		

(b.) Administration area



Start time	2004-07-21 14:52:09		
End time	2004-07-21 14:54:01		
Datetime	**Index**	**Url**	**Match target**
2004-07-21 14:52:15	1	http://www.comp.hkbu.edu.hk/	
2004-07-21 14:52:15	14	http://www.comp.hkbu.edu.hk/csd_home/index.php	
2004-07-21 14:52:16	14	http://www.comp.hkbu.edu.hk/csd_home/index.php	
2004-07-21 14:52:24	24	http://www.comp.hkbu.edu.hk/csd_home/message_from_head.php	
2004-07-21 14:52:30	21	http://www.comp.hkbu.edu.hk/csd_home/cs_corner.php	
2004-07-21 14:52:32	52	http://www.comp.hkbu.edu.hk/syllabus/syllabus.html	
2004-07-21 14:52:32	2	http://www.comp.hkbu.edu.hk/csd_home/top_banner.php	
2004-07-21 14:52:32	52	http://www.comp.hkbu.edu.hk/syllabus/syllabus.php	
2004-07-21 14:52:35	7	http://www.comp.hkbu.edu.hk/csd_home/education.php	
2004-07-21 14:52:36	36	http://www.comp.hkbu.edu.hk/csd_home/course_is.php	
2004-07-21 14:53:51	20	http://www.comp.hkbu.edu.hk/csd_home/staff.php	
2004-07-21 14:53:52	46	http://www.comp.hkbu.edu.hk/csd_home/staff_academic.php	Match

(c.) Filtered access record

Fig. 3 : Task-Based Usability Evaluation Module

5.4 Semantics Inference Module

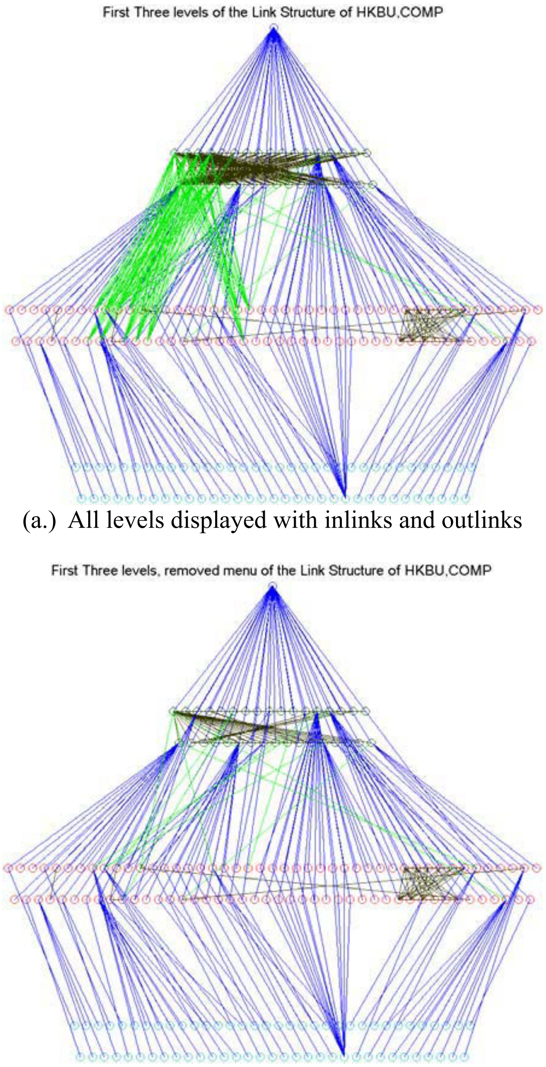
The semantics information from the web page and in particular the descriptive labels for hyperlinks are important for successfully analyzing the web design and usability. We plan to develop word analysis and machine learning tools in this module.

6. Web Design Layout Discovery via Association Rule Mining

In this section, we will briefly introduce a method of using the web mining for layout discovery. Traditionally, web site design and layout are often recovered manually or aided by authoring tools such as Macromedia Dreamweaver. In this part, we present our results on the recovery of web design layout via Web mining. The visualization of link structures of any non-trivial site is almost impossible due to the large and complex link structures of the sites. Figure 4(a) shows the partial link structure of the first three levels in the web site of Computer Science dept. of HKBU, where the links from the third level to other lower levels are omitted. The dense structure is very difficult to analyze visually and furthermore the links itself are generated from two sources where one type of links originates from the menu structure of the web page and one type of links originates from the actual content of the page.

As the association pattern of links are different in menu links, we propose the use of association rule mining to process the results we obtain in the web structure analysis. In the several web sites we studied, the association rules extraction is very successful in retrieving the top menu system and the sub-site structures of the web sites. Figure 4b shows the link structure after the links originated from menu is removed.

On the other hand, we may find that there are some sub-site structures in our departmental web site. The obvious one is located at the area bounded near 80 – 100 in Fig.5.


First Three levels of the Link Structure of HKBU,COMP
First Three levels, removed menu of the Link Structure of HKBU,COMP

(a.) All levels displayed with inlinks and outlinks

(b.) All levels with menu removed and displayed with inlinks, outlinks

Fig. 4 : Link structure of the Computer Science dept. at the Hong Kong Baptist University (Blue line : From the upper to the lower level, Green line : From the lower to the upper level)

The x axis denotes the page with the inlink and the y axis refers to the page with the outlink. There are some horizontal lines in Fig.5. They refer to pages with a number of links pointing to the other pages. In general such pages are usually navigation pages. The vertical lines are usually related to the menu page. Because menu is consistent and almost ubiquitous throughout the web site, there are many pages with links to the menu page.

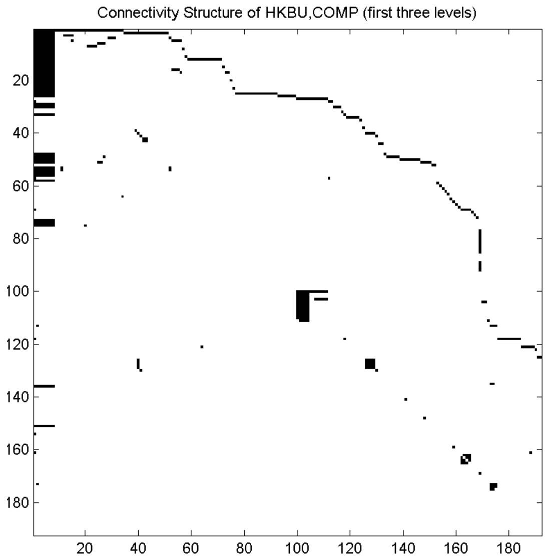


Fig.5 : Transition matrix of our web site

7. Conclusions

We have reviewed the different approaches to web usability study and proposed an integrated web mining and usability analysis system. The system is composed of four modules: the task-based usability evaluation, the web design and layout analysis, the web structure mining and the semantics inference. The task-based usability module has been completed and is being user tested, the web structure mining and design and layout analysis module are partially completed. The preliminary result on web structure mining shows how web mining can provide important design and layout analysis for a web site.

References

[1] Bernard, M. L., “Optimal Web Design”, *http://psychology.wichita.edu/optimalweb/print.htm*

[2] Bernard, M. L., “Examining a Metric for Predicting the Accessibility of Information within Hypertext Structures”, *Dissertation Thesis*, Wichita State University, 2002

[3] Botafogo, R. A., Rivlin, E., Shneiderman, B., “Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics.” *ACM Transactions on Information Systems*, Vol. 10, No. 2, 1992, pp.142 – 180.

[4] Brajnik, G. “Automatic Web Usability Evaluation: What Needs to be Done?”, *Proc. of 6th Human Factors and the Web Conference*, Austin, Texas, June 2000.

[5] Chakrabarti, S., Dom, B., Raghavan, P., Rajagopalan, S., "Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text", *Proc of 7the World Wide Web Conference*, 1998

[6] Chen, Z., Liu, S., Liu, W., Pu, G. and Ma, W., "Building a Web Thesaurus From Web Link Structure", *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, 2003, pp.48-55.

[7] Chi, E. H., Rosien, A., Supattanasiri, G., Williams, A., Royer, C., Chow, C, Robles, E., Dalal, B., Chen, J., Cousins, S., "The Bloodhound Project: Automating Discovery of Web Usability Issues Using the InfoScent™ Simulator", *CHI 2003*, Ft. Lauderdale, Florida, USA, April 5–10, 2003.

[8] Durrett, H.J., *Color and the Computer*. Orlando, FL: Academic Press, Inc, 1987.

[9] Glover, E. J. , Tsioutsiouliklis K., Lawrence S., Pennock, D. M. and Flake, G. W., "Using Web Structure for Classifying and Describing Web Pages", *WWW2002*, Honolulu, Hawaii, USA, May 7-11, 2002.

[10] Halkidi, M., Nguyen, B., Varlamis, I. and Vazirgiannis, M., "Thesus: Organising Web Document Collections based on Semantics and Clustering", *Journal on Very Large Databases, Special Edition on the Semantic Web*, November, 2003

[11] Heer, J. and Chi, E. H., "Separating the Swarm: Categorization Methods for User Sessions on the Web", *CHI2002*, Minneapolis, Minnesota, USA, April 20-25, 2002.

[12] Hinckley, K., Cutrell, E., Bathiche, S., and Muss, T., "Quantitative Analysis of Scrolling Techniques", *CHI 2002*, Minneapolis, Minnesota, USA, April 20-25, 2002.

[13] Hong, J. I., Heer, J., Waterson, S., and Landay, J. A., "WebQuilt: A Proxy-based Approach to Remote Web Usability Testing", *ACM Transactions on Information Systems*, Vol.19, Iss.3, 2001, pp.263-385.

[14] Hornof, A. J. and Halverson, T., "Cognitive Strategies and Eye Movements for Searching Hierarchical Computer Displays", *CHI 2003,* Ft. Lauderdale, Florida, USA, April 5-10, 2003.

[15] Ivory, M. Y., Sinha, R. R. and Hearst, M.A., "Preliminary Findings on Quantitative Measures for Distinguishing Highly Rated Information-Centric Web Pages", *Proceedings of the 6th Conference on Human Factors and the Web*, 2000.

[16] Ivory, M. Y. and Hearst, M. A., "The State of the Art in Automating Usability Evaluation of User Interfaces", *ACM Computing Surveys*, Vol.33, No.4, 2001, pp.470 – 516.

[17] Ivory, M. Y., "An Empirical Approach to Automated Web Site Evaluation", *Journal of Digital Information Management*, Vol.1, No.2, 2003, pp.75-102.

[18] Jenamani M., Mohapatra P. K. J., and Ghose S., "Online Customized Index Synthesis in Commercial Web Sites", *IEEE Intelligent Systems*, Vol.17, No.6, 2002, pp.20-26.

[19] Kao, H., Lin, S., Ho, J., Chen, M., "Mining Web Informative Structures and Contents Based on Entropy Analysis", *IEEE Transactions on Knowledge and Data Engineering*, Vol.16, Iss.1, 2004, pp.41 – 55.

[20] Kiger, J. I. , "The Depth / Breadth Tradeoff in the Design of Menu-Driven Interfaces.", *International Journal of Man-Machine Studies*, Vol.20, 1984, pp.201-213.

[21] Larson, K., & Czerwinski, M. , "Web page design: Implications of memory, structure and scent for information retrieval", *CHI'98: Human Factors in Computing Systems*, New York: ACM Press, 1998, pp.25-32.

[22] Meister, D., & Sullivan, D.J. "Guide to Human Engineering Design for Visual Displays" Bunker-Ramo Corp., 1969

[23] McCarthy, J, Sasse, M.A. & Riegelsberger, J., "Could I Have the Menu Please? An eye Tracking Study of Design Conventions", *Proceedings of HCI2003,* Bath, UK, Sep 8-12,, 2003.

[24] Miller, C. S. and Remington, R. W., "Modeling Information Navigation : Implications for Information Architecture", *Human-Computer Interaction*, Vol.19, No.3, 2004.

[25] National Institute of Standards and Technology, "Overview WebSAT", http://zing.ncsl.nist.gov /WebTools/WebSAT/overview.htm, 2001.

[26] Obendorf, H., Weinreich, H., Hass, T., "Automatic Support for Web User Studies With SCONE and TEA", *CHI 2004,* Vienna, Austria, ACM, April 24–29, 2004.

[27] Paganelli, L and Paterno, F., "Automatic Reconstruction of the Underlying Interaction Design of Web Applications", *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, 2002, pp.439 – 445.

[28] Pitkow, J. E. and Pirolli, P. "Mining Longest Repeated Subsequences to Predict World Wide Web Surfing.", *Second USENIX Symposium on Internet Technologies and System, 1999.*

[29] Shneiderman, B., "Designing the User Interface", Strategies for Effective Human-Computer Interaction, 3rd ed, Reading, MA : Addison-Wesley, 1998.

[30] Shurtleff, D.A., *How to Make Displays Legible*, La Mirada, CA: Human Interface Design, 1980.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 102-110, 2004.

Web-based Support Systems for Sustainable Communities

W.N. Liu J.T. Yao L. Fan Y.Y. Yao X.D. Yang

Department of Computer Science
University of Regina
Saskatchewan, Canada S4S 0A2
{liuwe200,jtyao,fan,yyao,yang}@cs.uregina.ca

ABSTRACT

This paper studies Web-based support systems for sustainable communities. A sustainable community is a community that respects the needs of both nature and future generations. A sustainable community activity is a collaborative process of solving environmental, economic and societal problems. We present an architecture of Web-based support systems for sustainable communities. The architecture is based on multi-tier, component-based structure. Each component provides distinct information services to support sustainable community activities. Users can access these services through standard Web browsers anytime, anywhere. It is argued that a Web-based support system can provide comprehensive and extensible services for diversified sustainable community activities.

1 INTRODUCTION

The notion of sustainable development emerged in 1987 as the overriding goal for human activities [23]. Sustainable communities seek well-balanced social, economic and environmental development strategies based on human responsibility to respect the needs of both nature and future generations [8]. Sustainable communities are also about the participation of all community members in sustainable community activities. A sustainable community activity is a collaborative process of solving environmental, economic and societal problems [9].

Public-led sustainable community activities must be managed to secure the transition to sustainable development [15]. Management support systems facilitate the management of sustainable community activities. The term of management support systems refers to the application of information technologies to support various management tasks [21]. For the diversity of sustainable community activities and participants, we use a more general term, computerized support systems, to replace the notion of management support systems. Computerized support systems for sustainable communities are used to facilitate the creation, discovery, management, distribution, exchange and presentation of sustainable community information. However, most of computerized support systems for sustainable community activities are devoted to individual activity, such as decision support systems [2, 5, 8, 22]. Due to the overlap between sustainable community activities in the content, different computerized support systems may overlap each other with respect to the services that they provide.

Sustainability issues are of multi-disciplinary, multi-agency, and multi-sector in nature [15]. The collaboration between community members is usually accomplished through classical media, such as print, audio, video and face-to-face contact, etc. The interaction between community members often suffers from financial, spacial and temporal constrains. In addition, policies do not always meet the needs of all community members.

The information technologies behind various computerized support systems for sustainable communities are well suited to take the advantage of the World Wide Web. The software architecture of Web-based applications is an essential shift of classical thick client/server architecture. A typical Web-based application consists of a Web site, application servers and data management servers. The Web site works as a presentation service provider. The data management servers store and supply the data needed by the Web site and the application servers. Information technology instruments work as application service providers. They are independent and reusable functional components. Users can access these application services through standard Web browsers anytime, anywhere. A Web-based support system for sustainable communities is an integration of relevant information technology instruments in the software architecture of Web-based applications, which provides comprehensive and extensible services for various sustainable community activities.

In addition, the Web has stimulated many communication tools for worldwide problem-solving collaboration and sustainable knowledge transmission. Web-based communication tools are user-friendly and multi-styles. They are less spacial-, temporal- and financial-restricted than classical media. Web-based communication tools play key roles in attracting community members to participate in sustainable community activities through the Internet. It is easier for community members to be involved in sustainable community activities by choosing preferable Web-based communication tools. As particular information instruments, these communication tools can be integrated into Web-based support systems for sustainable communities.

The organization of this paper is as follows: we first study sustainable community activities in the section 2. The participants of sustainable communities and the information technology instruments supporting sustainable communities are identified in the section 3. We outline an architecture of Web-based support systems for sustainable communities in the section 4. The functional modules of Web-based support systems for sustainable communities are elaborated in the section 5.

2 SUSTAINABLE COMMUNITY ACTIVITIES

Essential sustainable community activities include forecasting, public consultation, planning, decision making, implementation, and measuring progress. Each activity consists of some sub-activities or a series of procedural steps. These activities may overlap each other in the content. For example, both forecasting and decision making involve simulation activities. Sustainable community activities can occur individually, but they usually interact with each other to form more complex activities. The interaction between sustainable community activities is shown in the Figure 1 which is adapted from [10].

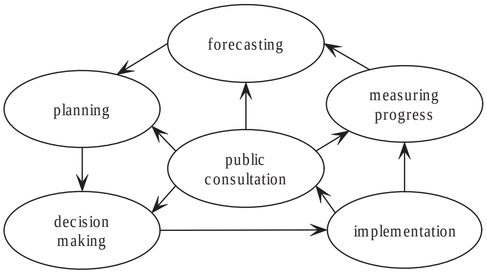


Figure 1: The interaction between essential sustainable community activities.

2.1 Forecasting

Forecasting activities are to identify the socioeconomic values that a community seeks to attain [6]. Sustainable development strategies often derive from forecasting activities. In a forecasting process, mathematical and physical models may be used to simulate the future environmental and socioeconomic situations. At the end of a forecasting activity, a clear view about the future community development is formed as the overall goal which may be impacted by the public consensus.

2.2 Public Consultation

Public consultation activities are the prevalent form of public participation in sustainable community activities. They usually involve the following sub-activities: public education, provision of background information, recruitment of participants, establishment of communication channels, coordination of consultation activities, recommendation and negotiation [19]. The first four activities are the preparations for the public recommendation and negotiation. The recommendation and negotiation are the core of public consultation activities.

2.3 Planning

Planning activities identify community problems and creates solution variants. A planning activity involves the following procedural steps: project preparation, problem identification, solution identification, and plan assessment. In the project preparation phase, the social, economic and environmental information of a community is collected. The problem identification is to find out issues from the collected community information. The identification of solutions to above community issues relies on the cooperation of stakeholders and the public consultation. The community background information, community problems and the corresponding solutions constitute a strategic plan.

Project preparation

The detailed planning procedure is formulated. A training program for planning and decision-making is given. Planners gather as much community information as possible, and compile it into a background information document [17]. Planners also need to identify who can influence the planning and to what extent [10]. Several advisory panels are established. The members are elected from stakeholders. The background information document are distributed to the stakeholder representatives for comment [17].

Problem identification

The overall goal is decomposed into subgoals [6]. Each subgoal is about a specific sustainability theme. Subgoals are highlighted as issues which may hinder the development from meeting the overall goal.

Solution identification

The planners and stakeholder representatives cooperate to create plausible solution alternatives for every identified problem. A solution is an ordered set of actions. Each solution is associated with necessary indicators. These indicators indicate what has been done, and how affected objects are responding. All issues associated with a specific subgoal, and their solution variants are assembled into a modal plan [6]. All modal plans constitute a strategic plan.

Plan assessment

The strategic plan is reviewed by the stakeholder representatives, and then revised based on the suggestions from the review processes [17]. At the end of this phrase, the strategic plan is submitted to the decision makers.

2.4 Decision Making

The decision making lays a practical scheme as the policy with legal validity. A decision making activity involves the following procedural steps: barrier and conflict identification, strategy formulation, strategy assessment and strategy judgement. The barrier and conflict identification examines the feasibility of above strategic plan. The strategy formulation and judgement are the key steps towards a sustainable policy. In a decision making process, simulation instruments may be used to forecast the impact of the policy on the community and neighboring regions.

Barrier and conflict identification
Decision makers identify barriers to the implementation of the strategic plan. These barriers involve legal, institutional, financial, political, cultural and technological obstacles. Decision-makers also need to examine potential conflicts hiding in the strategy plan. Both barriers and conflicts are called constraints of the strategic plan. Within a given set of constraints, decision-makers establish a priority of solving identified problems.

Strategy formulation
Decision makers find out compatible sets of solutions in which solutions can reinforce or compensate each other in meeting their respective objectives. In every compatible set, solutions are arranged in a order by which more desirable overall performance can be achieved. The integration of solution helps to reducing barriers to implementation, and it is likely to be more effective than selecting any one solution on its own [10]. An integrated package of solutions is called a strategy. A strategy is a approach to achieve the overall goal. Usually, the range of solutions and the of ways in which they are combined can lead to more than one strategy.

Strategy assessment
Since the evidence available on the effects of introducing a new strategy is often incomplete, a number of scenarios or mathematical models are developed by experts to simulate the potential impact of individual strategy variants [10]. Stakeholder representatives also assess the strategy variants against the full set of overall goal and subgoals.

Strategy judgement
A priority list of selected strategy alternatives feasible to meet the overall goal is generated. The authorities choose one as the policy with legal validity.

2.5 Implementation and Measuring Progress

Community members also become involved in the implementation of policies. The implementation is usually monitored within a specified time frame [6]. Based on a set of accepted performance indicators, regular assessment reports indicate whether identified problems are being overcome or whether new issues are emerging. Revisions of the policies are made regularly according to the effect, the experience and the public consensus.

A forecasting activity, a strategic planning activity, a decision making activity and public consultation activities make up a sustainable community policy-making activity. A sustainable policy starts from a forecasting activity. Based on the overall goal formed in forecasting activities, planning activities identify community problems and alternative solutions to produce a community strategic plan. Decision making activities make critical judgements on the strategic plan to formulate the policy with legal validity. All sustainable community activities may be impacted by consensus. The effect or outcome of sustainable development affects public opinion in turn.

3 THE SUPPORT ENVIRONMENT OF SUSTAINABLE COMMUNITIES

The support environment of sustainable communities consists of human participants and information technology instruments. Human participants are the main body of sustainable community activities. Information instruments are specific software programs, such as database search engines and data ming tools, etc. They provide information and communication services for human participants of sustainable communities.

3.1 Human Participants

Participants of sustainable communities can be classified into coordinator, planning body, advisory body, decision making body, implementing body, and monitoring body as shown in the Figure 2. The coordinator is responsible for supervising sustainable community activities. The planning body usually consists of several mixed working panels. Each panel works on a specific aspect of community issues, but it should make its interest match to the overall goal. The decision making body may be governmental authorities or legislatures [12, 16]. The implementing body takes charge of implementing community development strategies.

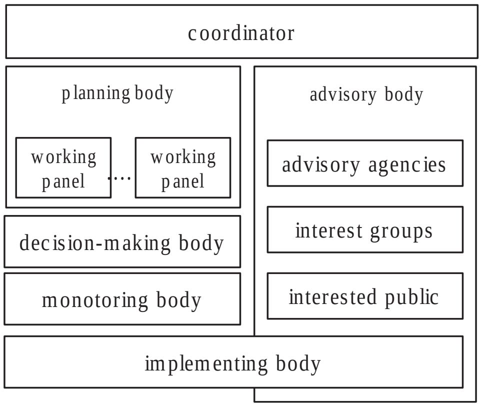


Figure 2: The human participants of sustainable communities.

The advisory body involves stakeholder representatives of a wide range of interests. The advisory body can be further divided into advisory agencies, interest groups and interested public. Advisory agencies include management branches in all governmental levels. Interest groups include public sector, private sector, aboriginal people and local residents. Their interests are likely to be affected by sustainable community activities. The implementing body is also an interest group. The interested public refers to the general public who shows a high degree of interest and willingness to participate in sustainable community activities. The monitoring body works as a permanent management structure to oversee the result or outcome of sustainable community activities.

3.2 Information Instruments

The information technology instruments supporting sustainable communities can be roughly classified into computer-mediated communication tools, data management tools, as well as knowledge acquisition and presentation tools.

Computer-mediated communication tools
Computer-mediated communication tools run over the Internet. They provide synchronous and asynchronous communication services for community members.

Data management tools
Part of sustainable community information comes from increasing electronic data sources over the Internet. The electronic data is managed by either file management systems or database management systems. These management tools provide basic data storage service and data retrieve service. However, data management tools can not interpret data or discover knowledge behind data.

Knowledge acquisition and presentation tools
Information-theoretic tools and geographical information systems (GIS) can help users discover potentially useful information, identify problems, create solutions and make decisions. Information-theoretic tools include data mining tools, reasoning tools and expert systems, etc.

Information instruments used to work as either command line applications or thick client/server applications. Unfortunately unfriendly interfaces and complex software/hardware deployment of these applications may prevent ordinary community members from participating in sustainable community activities. Web-based support systems can fill the gap between human participants and information instruments.

4 THE ARCHITECTURE OF THE WEB-BASED SUPPORT SYSTEMS FOR SUSTAINABLE COMMUNITIES

A Web-based support system for sustainable community can be depicted according to its software architecture and application architecture, respectively.

4.1 Software Architecture

The software architecture of a Web-based support system for sustainable communities is a multi-tier, component-based structure. The structure can decrease system complexity and enhance extensibility. There are technical specifications on the software architecture, such as Sun J2EE [20] and Microsoft Dot-NET [13]. The multi-tier, component-based structure is shown in the Figure 3 which is adopted from [20].

The software architecture is divided into four tiers: Web client tier, Web service tier, application service tier and information service tier. A Web client is a standard Web browser. The Web browser renders the Web pages published by the front-end Web server in the Web service tier. Users access back-end application services through the Web pages.

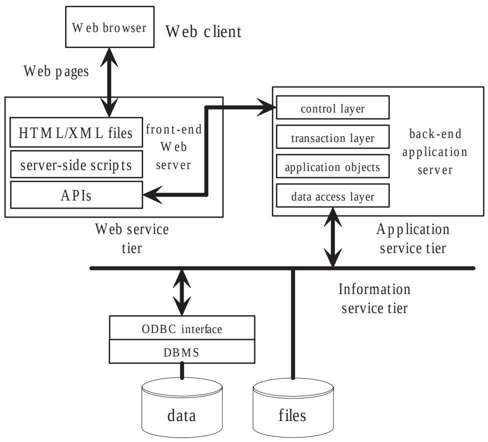


Figure 3: The software architecture of the Web-based support system for sustainable communities.

The application service tier can be further divided into control layer, transaction layer, application object layer and data access layer [20]. The control layer takes charge of the access control of application services. The transaction layer manages user inputs and sends them to the application object layer for processing. The application object layer contains a set of independent and reusable application components. Each component provides a specific application service. The data access layer is responsible for handling the data stored in the information service tier. The information service tier consists of database servers and file servers. It manages the raw data needed by the Web-based support system.

4.2 Application Architecture

The application architecture of Web-based support systems for sustainable communities defines functional modules. Information instruments are integrated into relevant functional modules to support specific sustainable community activities. Each information instrument works as an application service bundled with Web service. Therefore users can access it through standard Web browsers. The combination of functional modules can be used to support complex sustainable community activities, such as planning and decision making activities. In the section 5, we will elaborate these functional modules.

All modules are also organized into several subsystems with respect to their duties. Therefore the application architecture can be depicted according to these subsystems shown in the Figure 4. An Internet portal is the interface or entrance to a Web-based support system. The management subsystem provides management services for sustainable community activities. The communication subsystem mediates the

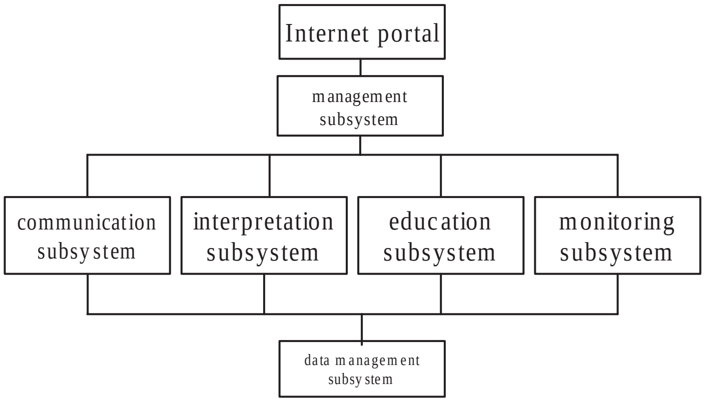


Figure 4: The application architecture of Web-based support systems for sustainable communities.

communication and collaboration between participants. The interpretation subsystem is dedicated to explaining and presenting sustainable community information for participants. The education subsystem is used to enhance the public participation consciousness and to transmit sustainable community information. The data management subsystem manages sustainable community information. The monitoring subsystem takes charge of measuring sustainable community progress.

The application architecture is mapped into the software architecture. The Internet portal is in the Web service tier; the data management subsystem is in the information service tier; the other subsystems are in the application service tier.

4.3 Web-based Resource Sharing

From the aspect of resource utilization, a Web-based support system possesses local human resource, local application resources and remote resources. The local human resource consists of participants in a specific sustainable community activity. The local application resources include sustainable community information and application services. They are usually located in a local-area network (LAN) and connected to the Internet through the front-end Web server. A Web-based support system can fully utilize remote resources over the Internet to compensate local resource shortage. The Web-based distributed resource utilization is shown in the Figure 5.

Correspondingly a Web-based support system for sustainable communities can be divided into human resource layer, local application resource layer and remote resource layer. The resource sharing can be implemented by the access to remote application service providers as well as the communication between participants and remote experts through the Web. If necessary, sustainable community information is translated into meaningful and easy-understood forms by either intellectualized application services or professionals who handle the interpretation of questions and computer output.

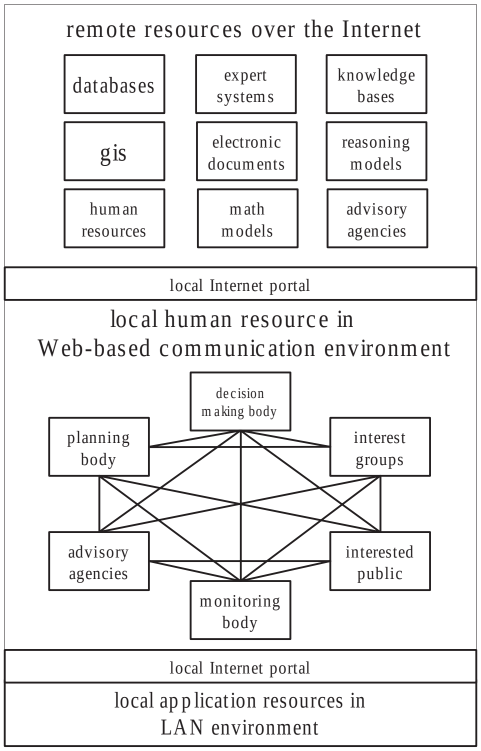


Figure 5: Web-based distributed resource utilization of sustainable communities.

4.4 The Impact of Web-based Support Systems on Sustainable Communities

Web-based support systems for sustainable communities take full advantages of the Web and Web-based distributed computing pattern. Their characteristics can be summarized as: ubiquitous and user-friendly access, integrated and extensible application services, worldwide resource sharing, and less cost of deployment.

- *Ubiquitous and user-friendly access*. The biggest benefit of Web-based support systems might be their Web interfaces [18]. Web interfaces are actually interconnected Web pages which are accessible for any device with a Web browser almost anywhere in the world. Web interfaces can be presented in multimedia style; hence they make users easily understand the content of a support system and use it without training.

- *Integrated and extensible application services*. In a Web-based support system, an information instrument works as an independent application service. These application services have distinct characteristics in function and style. They can reinforce each other to accomplish greater benefits. For instance, two planners can cooperate to prepare a plan draft through electronic white-board, meanwhile they can exchange their ideas through audio service. All application services can be flexibly uploaded to or offloaded from application servers to adapt to different sustainable community activities.

- *Worldwide resource sharing*. Through Web links embedded in Web interfaces, a Web-based support system can be seamlessly integrated with other Web-based support systems to provide more support services for local sustainable community activities.

- *Less cost of deployment*. Since Web-based support systems distribute information and deliver application services through the Web, and Web browsers are freely available for every major computer platform, few client software needs to be distributed [4].

Web-based support systems for sustainable communities can convert sustainable community activities into online community activities to some extent. Since online community activities can occur at less cost, Web-based support systems may therefore stimulate more community members to participate in sustainable community activities. Community members can freely choose preferable application services provided by Web-based support systems to accomplish their participation in a personal style. We may say that a Web-based support system for sustainable communities lays the foundation of online democracy. On the other hand, reusable application services and Web-based resource sharing maximize the availability and utilization of community resources, so the development of a Web-based support system for sustainable communities is also a sustainable community activity.

5 FUNCTIONAL MODULES

The functional modules making up a Web-based support system for sustainable communities are defined in this section.

5.1 Internet Portal

An Internet portal hosts a hierarchy of static and dynamic Web pages through which users can access application services. It is embedded with inner and external search engines. The inner search engine takes charge of retrieving local resources. The external one takes charge of retrieving remote resources. The external searching service may be provided by Internet service providers. For instance, with the Google Web APIs service, software developers can query more than four billion Web pages directly from their own computer program.

5.2 Management Subsystem

The subsystem includes project management module, stakeholder management module and access control module.

Project management

The module is used to help coordinators to manage sustainable community activities. Its functions may include creating project plans, scheduling tasks, tracking progress, managing cost as well as assigning and levelling hardware resources. By regularly pushing community activity progress reports to the Web, the module allow stakeholders to trace how their critiques are used.

Stakeholder management

The module is used to facilitate the recruitment of participants in a sustainable community activity. It maintains a database of stakeholder information. Initiators or organizers of the community activity evaluate stakeholder information to identify who can influence the community activity, and to what extent. The stakeholder database can help participants to rapidly find out proper advisory agencies and professionals against a specific problem. The stakeholder database may be accessed by remote Web-based support systems to accomplish human resources sharing. The module may also include a Web-based classroom for training participants.

Access control

The module provides integrated security administration services for the Web-based support system.

5.3 Communication Subsystem

The communication subsystem includes computer conferencing module and electronic polling module.

Computer conferencing

The module mediates users across the Internet to hold a teleconference, access a common database, or work on a common application process by using a series of Web-based communication tools. These communication tools may involve Email service, mailing list, chat room, audio/video-conference, electronic white board and asynchronous discussion board. Email is used to support personal communication. Mailing lists are used to support group discussion among stakeholders who share a common interest.

Planners, decision makers or experts can initiate a chat or audio/video meeting. However, the remote synchronous meeting preparation is not piece of cake. Initiators first use Email to schedule participants. And then they use Email to distribute meeting materials. They also need to check the status of each material submitted at the start of meeting. Finally, some speaking protocols and facilitation may need to be established to insure the smooth implementation of the synchronous meeting.

As a particular teleconferencing application, electronic white boarding is used to support the cooperative research between experts on a specific topic. Web-based asynchronous discussion board is becoming a prevalent communication platform, which allows all stakeholders to participate in consensus forming processes without spacial and temporal limits. In Web-based discussion board, users can express their views through text, audio, video, flash, or their combination. The Web-based discussion board uses a matrix to manage topics [14], where one dimension is specific sustainable themes, and another dimension is interest groups. For each of the topic cells, there is a separate mailing list.

Electronic polling
Electronic polling is database-oriented Web application. It is used to collect public opinion on a specific topic or problem. Any authorized stakeholder can create a poll or topic. Anyone can view the subject and the statistical result for the poll, but initiators can determine who can be entitled to vote. The presentation of voting results only comes true after the expiration date in order not to affect public opinion.

5.4 Interpretation Subsystem

The interpretation subsystem includes spatial modelling module, data presentation module and intelligent reasoning module.

Spatial modelling
The module may be used to predict the effect of a policy or understand the dynamics behind community changes. It consists of model constructor, model repository and simulation driver. Its duty may include unit model development, model archiving and reuse, integration of multiple spatial representations, simulation, data access and visualization, and visualization of remote simulation [11].

Data presentation
The module is used to help users better understand sustainable community information. It involves two groups of tools. The first group consists of data warehousing tools, OLAP tools, data ming tools and data visualization tools. Its function may involve:

- cleaning and repairing noisy, erroneous, missing and irrelevant data,
- selecting data relevant to analysis task,
- transforming selected data into forms appropriate for mining,
- integrating relevant heterogenous data into a data file,
- extracting data patterns, associations, changes, anomalies and significant structures from the consolidated data,
- identifying the truly interesting knowledge based on interest measures,
- and presenting the mined knowledge to the user by graph or animation.

The second group of tools are GIS toolkit which is used to geoprocess geography-oriented data and render the results in the form of dynamic charts and maps. Geography-oriented data comes in three basic forms: spatial data, tabular data and image data. These data can be uniformly managed by advanced object-relational database systems.

Intelligent reasoning
The module generates explanations on how and why particular conclusions have been drawn from sustainable community information. It consists of knowledge bases, information theoretic-based reasoning tools, and communication tools. Knowledge bases store the domain expert knowledge captured by knowledge engineers. Reasoning tools may include rule-based reasoning, fuzzy logic, Bayesian networks, case-based reasoning, connexionist reasoning, evolutionary computing, qualitative reasoning, constraint satisfaction, and model-based reasoning [2], etc.

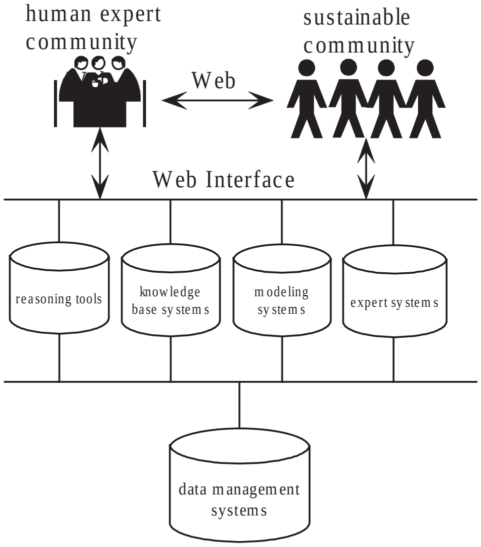


Figure 6: The combination of machine intelligence and human intelligence.

However, the inherent complexity and inflexibility of intelligent tools need a human expert community to work as a key element in the support system. The expert community consists of experts on various fields related to sustainable development. Depending on what users want, they convert question into a suitable form for a computer system and interpret the generated data in a meaningful way for users. These experts need to be enough flexible to distinguish different aims and requirements of various persons. The module provides communication tools or call the computer conferencing module to facilitate the interaction between experts and users. The combination of computer output and human judgement can shorten the time in which decisions can be made and improve the consistency and the quality of the decisions. The combination of human intelligence and machine intelligence is shown in the Figure 6.

5.5 Public Education Subsystem

The subsystem is used to enhance public participation consciousness by deepening public understanding of sustainable development. A virtual library archives electronic educational materials on sustainable development. It also maintains Web links to other sustainable Web sites. The subsystem may include Web-based instruction (WBI) or distance learning application, which can deliver online personalized tutoring via the Web. The application dynamically generates learning materials based on students' past performance.

5.6 Data Management Subsystem

The subsystem is responsible for the storage, retrieve, exchange and dissemination of electronic data. The data needed by sustainable community activities can be classified into structured data and unstructured data. Structured data are well formed and fit into relational rows and columns. Structured data is managed by relational database systems. Unstructured data does not necessarily following any format or sequence. Unstructured data is managed by file management systems. Unstructured data can also be organized into self-describing terms called semistructured data [1].

5.7 Monitoring Subsystem

The subsystem is used to collect sustainable development indicators. It regularly or real time receives data from remote data sources. The collected data is verified and stored in database. The module may periodically analyze accumulated data and produce monitoring reports. If some pre-established critical level is reached, the module will sound a warning.

6 CONCLUSIONS

This paper studies Web-based support systems for sustainable communities. Essential sustainable community activities are studied. Computerized support systems can facilitate sustainable community activities, however, these support systems are hardly extensible with respect to the diversity of sustainable community activities. On the other hand, human participants are the main body in sustainable community activities. Information technology instruments provide information services for human participants, but the complexity of these tools may frustrate public participation. Web-based support systems are a feasible solution to above problems.

We outline an architecture of Web-based support systems for sustainable communities. The architecture is based on multi-tier, component-based structure. Information technology instruments are integrated into relevant functional modules. Each information instrument works as an independent application service. Users can access it through standard Web browsers anytime, anywhere. A Web-based support system can combine various application services to support diversified sustainable community activities. Community members can choose preferable application services to accomplish their participation in sustainable community activities.

ACKNOWLEDGEMENTS

Funding for this research is partially provided by Centre for Sustainable Communities of the University of Regina.

REFERENCES

[1] Abiteboul, S., Buneman, P., Suciu, D., "Data on the Web: From Relations to Semistructured Data and XML", Morgan Kaufmann Publishers, USA, 2000.

[2] Cortés, U., Sànchez-Marrè, M., Ceccaroni, L., R-Roda, I., Poch, M., "Artificial Intelligence and Environmental Decision Support Systems", *Applied Intelligence*, 13:77-91, 2000.

[3] Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M., White, B., "Web Engineering", *Journal of Web Engineering*, 1(1):3-17, 2002.

[4] Ginige, A., "Web Engineering: Managing the Complexity of Web Systems development", *Workshop on web engineering*, Proceedings of the 14th international conference on Software engineering and knowledge engineering, 721-729, 2002.

[5] Haagsma, I.G., Johanns, R.D, "Decision Support Systems: An Integrated Approach", *Environmental Systems*, 2:20-34, 1994.

[6] Idaho Transportation Department (ITD), "Decision Process", *Idaho Transportation Plan (ITP)*, 1995.

[7] Yao, J.T., Yao, Y.Y., "Web-based Support Systems", *WI/IAT 2003 Workshop on Applications, Products and Services of Web-based Support Systems*, 1-7, 2003.

[8] Kaempf, C., "Decision Support Systems (DSSs) for Environmental management: Web-based Communication Modules to Enhance Public Participation", Society for Technical Communication (STC) Proceedings, 1-6, 2001.

[9] Lachman, B.E., "Linking Sustainable Community Activities to Pollution Prevention: A Sourcebook", Critical Technologies Institute, RAND, 5-11, 1997.

[10] May, A.D., "Developing Sustainable Urban Land Use and Transport Strategies: A Decision Makers' Guidebook", Procedures for Recommending Optimal Sustainable Planning of European City Transport Systems (PROSPECTS), European Commission, 7-37, 2003.

[11] Maxwell, T., Costanza, R., "Developing Understanding of Ecological Economic Systems", RAND Workshop on Complexity and Public Policy, 14-15, 2000.

[12] Metro Regional Center, “Transportation planning decision-making process”, http://www.metro-region.org/library_docs/trans/trans_process.pdf, 2001.

[13] Microsoft, “Application Architecture for .NET: Designing Applications and Services”, Microsoft Corporation, 2002.

[14] Moor, A.D., “Information Tools for Susainable Development: Enabling Distributed Human Intelligence”, *Journal of Failure & Lessons Learned in Information Technology Management*, 2(1):21-31, 1998.

[15] National Round Table on the Environment and the Economy (NRTEE), “Environmental Quality in Canandian Cities: The Federal Role”, National Library of Canada Cataloguing in Publication, 2003.

[16] San Diego City Hall, “Land Development Procedures”, *San Diego Municipal Code*, Chapter 11, http://clerkdoc.sannet.gov/Website/mc/mc.html, 2000.

[17] Saskatchewan Environment, “Forest Land Use Planning”, http://www.se.gov.sk.ca/forests/landuse, 2000.

[18] Shim, J.P., Warkentin M., Courtney J.F., Power D.J., Sharda R., and Carlsson C., “Past, Present, and Future of Decision Support Technology”, *Decision Support Systems*, 33:111-126, 2002.

[19] Sinclair, A.J., “Public Consultation for Sustainable Development Policy Initiatives: Manitoba Approaches”, *Policy Studies Journal*, 30(4):423-443, 2002.

[20] Sun microsystems, $Java^{TM}$2 Platform Enterprise Edition Specification v1.4, 2003.

[21] Turban, E., Aronson, J.E., Bolloju, N., Decision Support Systems and Intelligent Systems, Prentice Hall, 2001.

[22] United Nations University International Institute for Software Technology(UNU-IIST), “Decision Support systems for Sustainable Development Experience and Potential: A Position paper”, UNU-IIST Macau Workshop, 1996.

[23] World Commission on Environment and Development (WCED), “Our Common Future”, Oxdord University Press, 1987.

Configuring Java-Based Web Application Development Environment for an Academic Setting

Ritesh Mehra, Satya K Gandham, Zonghuan Wu, Vijay V.Raghavan
Center for Advanced Computer Studies,
University of Louisiana, Lafayette
{rxm3304, skg7478, zwu, raghavan}@cacs.louisiana.edu

Abstract

In this paper, we analyze the characteristics and constraints of a typical academic environment for web application development. A set of Java-based web technologies and tools are introduced and reviewed for such an environment. The motivation behind this work is to provide comprehensive resource for university faculty members and students about emerging technologies and available tools to facilitate rapid development of web applications.

1. Introduction

In this paper we present a comprehensive view of available resources in terms of technologies and tools for building Java based web application environment in academic settings. In situations like, setting a new research lab or deciding upon a suitable technology to use for developing web applications, project supervisors and students often find several, if not too many, alternatives to evaluate. At times, this becomes an extremely confusing exercise. For instance, which tools and technologies to explore and why, how cost-effective such tools are and what is the learning curve involved etc. In this paper, keeping our focus on Java based web development applications, we review a set of Java based tools and technologies to address the common issues encountered by students and faculty members when making such a choice. In section 2 of this paper, we analyze the characteristics and constraints of a typical academic environment and present a set of relevant Java based web development technologies for developing robust and scalable web applications. In rest of the paper, section 3, we review some of the popular Java based tools capable of building efficient and cost-effective web applications rapidly.

2. Characteristics and Constraints of a Typical Academic Environment

In this section, we review some of emerging Java technologies like EJB, Struts, JSF and Tiles in context of their relevance to academic community. We also discuss the relevance of extreme programming to academic community. To start with, we review academic preferences for open source software, personal preferences of student and faculty and some of the common project requirements.

2.1 Preference for Open Source Software

Universities usually have preference for open source software solutions. This is evident from the recent resolution approved by University of Buffalo, State University of New York stating that direct, unmediated and unfettered access to information is fundamental and essential to scholarly inquiry, academic dialog, research, advancement of research methods and academic freedom. Even industry has shown great interest in promoting open source software solutions. It is mainly because of open source policies that Sun's J2SE & J2EE API standards have been adopted and promoted by some of the leading software vendors, such as BEA, IBM, Apache and Oracle etc. In addition to adhering to standard specifications, some of these vendors like Apache [28], SourceForge [29] etc. are now offering open source free software solutions for numerous other Java based web applications as well.

Another reason for preferring Java based open source software is due to the fact that universities/colleges usually have tight budgets to invest in licensed software. Therefore, one of the goals is to minimize extra investments on tools the equivalents of which may as well be available free on Internet. Moreover, universities usually need to develop only non-commercial prototypes for establishing the research ideas and do not require the extensive feature support of licensed software. Even in cases when commercial license is necessary, due to the wide vendor support available for Java technologies, universities can explore a wide range of tools with varying prices to choose from.

2.2 Student Preferences

Usually university student work on research/class projects only part time during their academic semesters, as such they prefer to use tools that are easy-to-manage, easy-to-configure, freely available on Internet (may be for limited duration) and can quickly do their job. At the same time, students also want to get hands on working experience in emerging technologies and latest tools to enhance their skill sets. As a result of this preference, students tend to learn and implement new technologies on their research projects.

For instance, Java based MVC (model-view-controller) design pattern, Struts [1], help students in meeting the exact expectations mentioned above.

2.3 Faculty Preferences

Since students work on research/class projects only for limited hours during their course of graduation, there is always a need to maintain the projects properly documented. Documentation is often required when starting a new project or renovating an existing one. As a result, project supervisors look for tools that are easy-to-access, easy-to-manage and are capable of capturing different formats of design and documentation. Design tools like ArgoUML [3] offer extensive support for drawing different types of design diagrams free of cost. Similarly, API documentation tools like, Javadoc [31], can automatically generate HTML based API documentation from java doc comments written inside the source code files.

In terms of selecting students for their projects, professors or project supervisors usually do not have many options to find domain experts having special skill sets. And by the time students become productive for the project, they are already close to finishing their graduation and leaving the school. Also research-oriented projects are generally dynamic in nature. Quite often, research ideas change in the course of development thereby impacting original design and application functionality significantly. All such observations indicate the necessity of embracing and employing the principles of '*Extreme Programming*'. By involving both developers (students) and customers (external or internal) in every phase of project, extreme programming provides the flexibility to cope with frequent redesign and re-factoring. Since the technique is more suitable for small size teams working on frequently changing projects, it provides an excellent option for project supervisors to consider. It also helps in maintaining a continuous learning environment within the team thus making it even more relevant to students. Many of the Java web application technologies like Tomcat [4], ANT [5], JUnit [6], XDoclet [7], Cactus [8] etc. support the principles of extreme programming.

2.4 Project Specific Requirements

Security and integrity of web applications is becoming increasingly important. With increase in online monetary transactions over Internet, it is evident that web applications can no longer compromise on web security. Even behind firewalls, web applications are juicy targets for cross-site scripting, URL manipulation, complex SQL insertion attacks, and more. Malicious users can subvert basic role-based security and have their way with the source code. It is also observed that university web servers are more vulnerable than industry servers. Sun offers "Java Web Services Developer Pack (Java WSDP)" a free integrated toolkit that can be used to build, test and deploy XML [2] applications, web services, and web applications with the latest web service technologies and standard implementations. Fine-grained web service security can be implemented using XML digital signatures, encryption, Java Authentication and Authorization Service (JAAS), and the Java Cryptography Extension (JCE). This may be very useful for academic projects that require comprehensive web service security.

Nowadays, one of the principal goals for building research oriented web applications is to ultimately promote the ideas (to industry or within academia) through technology transfer. At the same time, prototype systems are expected to deliver industry equivalent quality. Web services are often brought into such situations to expose existing functionality of web application. Sun's J2EE EJB [9] technology addresses this need by supporting distributed, transactional, secure and portable applications based on Java technology. Besides, it offers many other enterprise application features such as load balancing, clustering, resource pooling and caching. The EJB API specifications are publicly available and application servers like JBoss [10] provide free support for EJB containers.

Using the combination of EJB and JBoss, academic communities can develop robust & secure web applications free of cost with a little investment on learning the EJB technology.

At times, projects might need intellectual property protection for securing the confidential parts so that source code remains unexposed even after distribution of application. There are a number of free Java obfuscators available on Internet that are capable of securing the source code while making the application publicly available and still keeping it platform independent.

2.5 Rapid Development

Web development technologies and tools need to foster rapid development of research ideas. For instance, ‘integrated development environments’ can be used in coding, debugging and testing phases of development to speed up the process. Some of the freely available ‘integrated development environments’ for developing JAVA based web applications like Eclipse [11], NetBeans, and JDeveloper etc. can be used as building platform to develop the research ideas quickly and easily.

Similarly, Java based technology, Tiles, helps in managing the HTML layout structure across different web pages of the application. It provides a better control on the layout of web pages, reduces code duplication, avoids HTML frameset problems and increases the overall speed of development.

Tiles work hand in hand with JSP and Struts to avoid code repetition by using a common layout template shared among all the web pages. Struts do a clean separation of presentation, view and business layers by implementing MVC architecture using XML configuration files, Java classes and resource bundles. Struts also provide tag libraries and classes to support JSP, Tiles, JSF [12], message internationalization and automatic form validation etc.

Another emerging technology, Java Server Faces (JSF) offers user-friendly interface to build HTML oriented GUI controls and their associated event handlers. Students of various skill levels can quickly build web applications by: assembling reusable UI components in a page; connecting these components to an application data source; and wiring client-generated events to server-side event handlers.

Usually universities have easy access to multiple platforms e.g. Unix systems, Win NT systems, Linux systems, Win 2000 machines, Solaris machines etc. This kind of infrastructure support allows them to develop and test platform independent web applications. One of the build tools that can automate the process of deploying Java based web applications on any platform is “ANT”. Academic communities can use ANT to quickly deploy and test web applications on multiple platforms free of cost.

3. Available Tools for Developing Java Web Applications

There are numerous Java based web development tools available on Internet as freeware. Each of them has their own advantages and disadvantages. As such, there can be several possible combinations for setting web development environment configuration. However, the choice is left on students and project supervisors to select the relevant ones. Although freeware usually do not guarantee extended support for bugs and issues reported during their usage, they still offer an excellent option for academic communities to quickly develop and test their research ideas without making any investment.

In this part of the paper, we present a list of popular Java based freeware tools categorized as per distinct phases of software development lifecycle. For each category in the list, we start with a brief introduction about the category, followed by a small description of the tools and conclude with a comment on their academic relevance to students and/or project supervisors.

3.1 Designing and Modeling Tools

Designing is an important phase for any project. A well-designed project can significantly reduce the possibility of functional and logical errors in the application.

One of the open source design tools available on Internet is ArgoUML [3]. ArgoUML is a designing and modeling tool similar to “Rational Rose” in many aspects except that it is available for free. It can run on any platform and can support various diagrams such as Class, State Chart, Activity, Use Case, Collaboration, Sequence, Deployment diagrams etc. It also provides features to generate skeletal code in Java, C++ and Php and supports internationalization. For students and project supervisors, this is an excellent option to consider before going for “Rational Rose”. As it provides extensive support for UML based design patterns, it can save a huge investment on licensed version of similar design tool while meeting all the major project requirements.

3.2 Development Tools

Development is usually the next important phase after design. This phase requires a combination of multiple development tools such as web servers, integrated development environments, refactors, and beautifiers to develop and manage code.

3.2.1 Development Web Servers

Web servers can be broadly classified into two categories, development servers and deployment servers. Two popular open source free web servers are Tomcat and JBoss. Whereas Tomcat is an open source free development server, JBoss is an open source free deployment server.

Tomcat is a Java Servlet Container that is used in the official reference implementation of Servlet and JSP technology. Students generally use it for developing Java web applications.

3.2.2 Integrated Development Environment

Integrated development environment (IDE) tools are the starting blocks for building web applications. They usually serve as single unified platform to access and manage other tools integrated in them. For instance, Eclipse is a popular, open source, freely downloadable, integrated development environment providing a universal toolset for web development. Plugins like VSS plugin (for source control), Tomcat launcher plugin (for web server), Easy Struts plugin (for struts support), XML Editor plugin (for XML editing) and SWT/Swing Designer plugin (for drag-and-drop GUI support) can be easily integrated with eclipse through its generic plugin support API.

Students can easily integrate other web development tools like Tomcat, VSS etc. into Eclipse as per their choice and configure Eclipse as a single point of access for controlling different parts of web application environment.

3.2.3 Applet Development

The Java Abstract Window Toolkit (AWT) can do much more than what HTML can do in a browser. Using applets, AWT [13] can be used to draw figures, build images at run time and support actions and event handling. Since applets execute within the client's browser, they can dynamically generate and display graphs using browser's in-built JVM.

Many tools have been built upon this technology to support dynamic images in web pages. These tools generally provide features to customize HTML controls (creating text boxes and combo boxes with fairer look). However, students need to be careful when using applets and applet development tools because sometimes browser settings enable "only trusted applets". Browser on this account can discard even a normal applet, which is not trusted. Moreover, making an applet a trusted applet may require extra efforts in terms of obtaining trust certificates etc.

3.2.4 Refactors

Refactoring [25] is a disciplined technique for restructuring an existing body of code, altering its external structure without changing its internal behavior. Each refactoring step executes a series of small transformations to produce a significant restructuring. System is tested after every refactoring step. Since the process is based on incremental transformations & testing, it reduces the possibility of system failure or undesired change in functionality.

Code refactoring allows restructuring of source code so that original functionality remains unaltered. For instance, when renaming a variable to better reflect its usage, all occurrences of original variable in the entire application require update. Also, while extracting and moving a block of code into a separate function for efficient code reuse, extra efforts are required to ensure that new errors are not introduced.

One of the free, open-source, auto-refactoring tools available on Internet is RefactorIT [26]. It can automatically update all references in source code whenever a variable, method or a class is refactored and easily integrate with most of the available IDEs like Eclipse, Netbeans etc. It also detects unsafe throw and catch clauses, hidden static methods, unused variable assignments and loose nested blocks. JEdit [27] is yet another tool for refraction that offers search and replace functionality in addition. For students, such tools offer an excellent option to easily manage refraction in source code when integrated with IDE.

3.2.5 EJB Code Generators

These tools are useful for advanced applications that involve extensive usage of EJB components.
XDoclet, for instance, is an open source, EJB code-generating engine that enables attribute-oriented programming for Java by adding metadata (attributes) in special JavaDoc tags. It is particularly useful for maintaining large number of EJBs where a single EJB spans across seven or more files. As it is capable of generating source code files using standard templates and attribute information, it can help in rapid and continuous integration of web components into the web applications. However it requires Ant support for build process. This tool may be of relevance to students only when their web application project involves extensive usage of EJB components.

3.2.6 Code Beautifiers

Many projects spend a hefty amount in code maintenance. Maintenance procedure becomes a redo if the programmer decides to rewrite the code just because the earlier code lacked readability**.** Code beautifiers help in making the code more readable by adhering to standard coding recommendations.

One of the free, code beautifier tool, Jalopy [14], automatically layouts any valid Java source code according to some widely configurable rules to meet certain coding style. It also checks if the program adheres to some standard coding style for braces, white space handling, indentation and intelligent line wrapping etc.

Another free, code beautifier tool, CheckStyle [15], automates this process of checking code standards. It is highly configurable and can support multiple coding standards. When integrated with ANT, it can check JAVADOC comments, name conventions (in regular expressions), headers, imports (checks to see that no import statement ends with *), class design and duplicate code etc.

These tools are especially useful for new students joining in the team to become familiar with recommended coding standards.

3.3 Testing Tools

Testing of web applications is possible at different levels like unit testing, functional testing, integration testing, load testing, regression testing, performance testing etc. Before investing in any particular testing tool, it is advisable to first evaluate the actual test requirements of web application in terms of test importance, effort estimation and software cost.

3.3.1 Unit Testing

There are various testing tools for Java based web applications that are available free on Internet. Unit testing tools, like JUnit, are based on the concepts of extreme programming and emphasizes on developing test cases in terms of expected results and test fixtures parallel with code. JUnit is an open source, testing framework used for writing and running repeatable unit tests. However, effective usage of JUnit requires programming discipline and patience on part of student because of the extra efforts involved in developing test cases besides code.

3.3.2 Debuggers

Debugging is a process of finding and fixing errors by inserting breakpoints in the source code and verifying the correctness of a variable or state of a process. Most of the Java debugging tools are built upon standard ‘Java Platform Debugger’ API specifications. JSwat [16] is a freely downloadable debugging tool with colored interface and graphical panel for threads, stack frames, visible variables etc. It provides features for debugging applets, JSPs, Servlets and J2ME applications. It can insert breakpoints with conditions and monitors and to debug application in either graphical or normal console mode.

Omniscient Debugging [17] is another free debugging tool written in Java that allows rewinding and retrieving previous values of variables without inserting breakpoints. The debugger backtracks by recollecting recorded "time stamps" of Java Byte Code to determine previous values of the objects, variables and method calls. Students can use these tools to quickly debug Java based web application.

3.3.3 Integration Testing

Integration testing tools are helpful when integrating sub modules, sub-components into the main application. They can be used at every step of the integration process.

Cactus is one of the free integration-testing tools that can test server side Java code like Servlets, EJBs, Tag Libraries, and Filters. It tries to minimize the overall integration cost by spreading integration testing into development in a more automated way. This tool may be useful to academic communities when their web-applications are highly component-oriented and involves considerable integration efforts.

3.3.4 Test Coverage

Test coverage tools are used to highlight sections of source code that are uncovered and untested by the test cases. They help developers in creating better unit tests.

One of the free, open-source, code coverage tools is Clover [18]. It discovers sections of code that inadequately tested by the unit tests. This then feeds back into the testing process to improve tests. When integrated with Eclipse and JUnit, it can make an excellent configuration for supporting extreme programming concepts. Students can find this useful only when they are using one of the testing tools in their applications.

3.3.5 Load Testing

Tools for load testing are used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load conditions. One of the open-source testing tools for measuring server performance is Apache-JMeter [19]. It is used to test performance both on static and dynamic resources such as static files, Java Servlets, CGI scripts, Java objects, databases, FTP servers etc. This tool may be used by students even for simpler web applications to get an approximate estimate on application's load handling capacity.

3.4 Deployment Tools

These tools are generally used for deploying fully developed and tested web applications on production servers/deployment servers. Build tools like ANT automates the tedious process of linking and deploying applications on deployment servers.

3.4.1 Deployment Web Server

In terms of caching at memory level, deployment servers usually offer better support for hosting advanced web applications and are relatively more advanced than simple development servers, like Tomcat.

JBoss is an open source free Application Server that can be used for deploying any application built upon J2EE technologies. It can be logically separated into two parts, one part including web container and other including EJB container. For the web container part, JBoss can use either Tomcat or Jetty (it uses Tomcat by default). With EJB comes the ability to incorporate JMS (messaging technology). JBossCache is a feature that provides option to cache the transaction data for enterprise applications. It gives an excellent option to academic community as it provides most of the advanced features of equivalent commercial server with no investment cost.

3.4.2 Build Tools For Deployment

Large projects often involve multiple programmers developing separate modules for different parts of application. Building is a process of bringing together multiple modules (may be from CVS), compiling them, generating documentation, managing files and finally deploying on possibly different platforms.

One of the free, Java based, cross platform, build tool is "ANT". Compared to other build tools like "GNU Make", it is simpler and easier to use. Instead of a model that is extended with shell-based commands, Ant is extended using Java classes. Instead of writing shell commands, the configuration files are XML-based, calling out a target tree where various tasks get executed. Each task is run by an object that implements a particular Task interface.

This is a useful tool for students and project supervisors when their application requires cross platform development, testing and deployment. Other than cross platform builds, it also allows integration with other tools like XDoclet etc. Though the tool requires a small learning phase on behalf of students, but its knowledge is worth the efforts.

3.5 Release Tools - Obfuscators

By default, compiled byte code contains a lot of debugging information: source file names, line numbers, field names, method names, argument names, variable names, etc. This information makes it straightforward to de-compile the byte code and reverse engineer the entire programs. At times this calls for source code security.

Obfuscators are tools that remove such debugging information and replace all names by meaningless character sequences, making it much harder to reverse-engineer. They further compacts the code as a bonus. YGuard [20] & ProGuard [21] are some of the free obfuscators available on Internet.

These obfuscating tools can be of significant importance to academic communities when considering intellectual property protection of source code and simultaneously publishing the binary class files on Internet for promoting technology transfer.

3.6 Maintenance Tools – Issue Trackers

Issue tracking tools such as IT-Tracker, AT-Project etc provide easy bug-monitoring system across multiple projects and user bases and are can be easily integrated with different IDEs.

In general, these tools are of more relevance to academic project supervisors when the projects requires cross-team support and maintenance by monitoring bugs and user change requests etc.

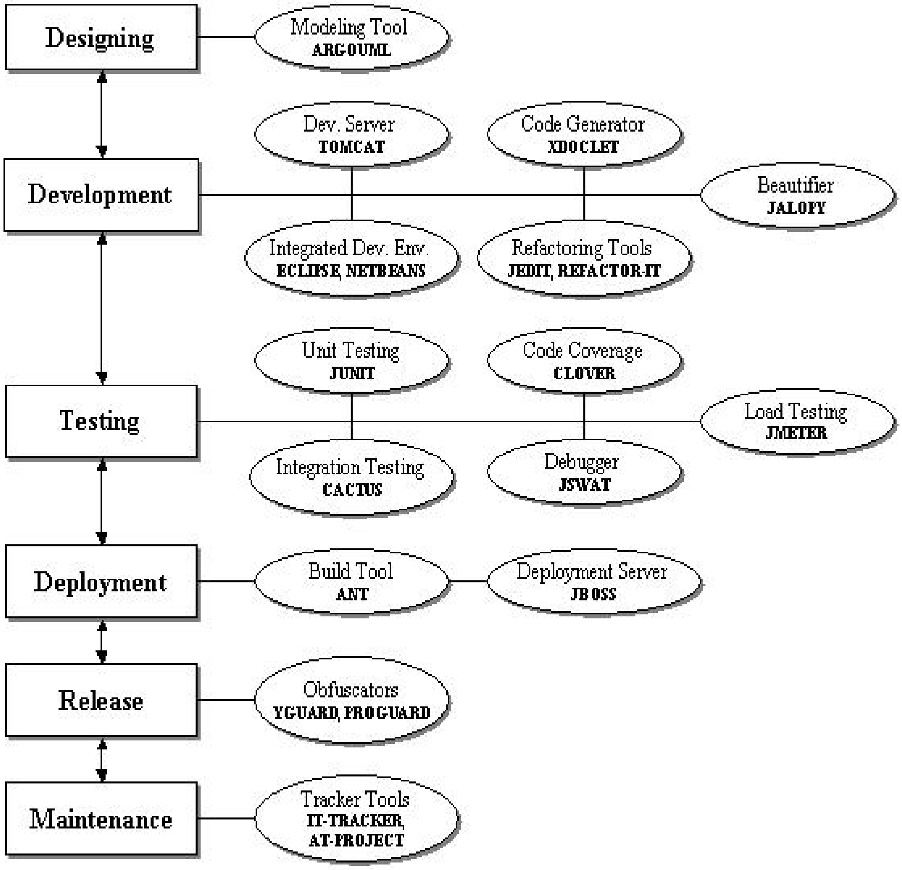


Figure 1: Tools categorization into different phases of application development lifecycle

4. Conclusions

In this paper, we presented various open source Java technologies and tools that can be downloaded from Internet for free and can help university faculty members and students to configure a robust, secured and advanced Java web development environment cost effectively.

In section 2, we analyzed the characteristics and constraints of a typical academic environment. We also presented several emerging Java web development technologies like EJB, Struts, JSF, Tiles and XML and linked them with the needs and preferences of academic communities. In section 3, we presented a list of popular Java based freeware tools categorized into phases of web application development lifecycle. Figure 1 summarizes the tools discussed in section3.

Acknowledgements This work is supported in part by the IT Initiative of the State of Louisiana to Lafayette

5. References

[1] Strut, http://www.coreservlets.com/Apache-Struts-Tutorial/Understanding-Struts.html#Pros
[2] XML, http://www.w3schools.com/xml/default.asp
[3] ArgoUML, http://argouml.tigris.org/
[4] Tomcat, http://jakarta.apache.org/tomcat/index.html
[5] ANT, http://ant.apache.org/
[6] JUnit, http://www.junit.org/index.htm
[7] XDoclet, http://xdoclet.sourceforge.net/
[8] Cactus, http://jakarta.apache.org/cactus/index.html
[9] Ed Roman, Scott Ambler, Tyler Jewell, Mastering Enterprise JavaBeans, 2nd edition.
[10] JBoss http://www.jboss.com/index.html
[11] Eclipse, http://www.eclipse.org/
[12] JSF, http://java.sun.com/j2ee/javaserverfaces/index.jsp
[13] ANT, http://www.oreilly.com/catalog/javawt/book/

[14] Jalopy, http://jalopy.sourceforge.net/
[15] CheckStyle, http://checkstyle.sourceforge.net/
[16] JSwat, http://www.bluemarsh.com/java/jswat/
[17] Omniscient debugger, http://www.lambdacs.com/debugger/debugger.html
[18] Clover, http://www.cenqua.com/clover/
[19] JMeter, http://jakarta.apache.org/jmeter/index.html
[20] YGuard, http://www.yworks.com/en/products_yguard_about.htm
[21] ProGuard, http://proguard.sourceforge.net/
[23] IT-Tracker, http://www.cowsultants.com/
[24] AT-Project, http://www.atreides-technologies.com
[25] Refactoring, http://www.refactoring.com/
[26] Refactor-IT, http://www.refactorit.com/
[27] JEdit, http://www.jedit.org/index.php?page=features
[28] Apache, http://jakarta.apache.org/
[29] SourceForge, http://sourceforge.net/
[30] EJB, http://java.sun.com/products/ejb/
[31] Javadoc, http://java.sun.com/j2se/javadoc/

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 119-126, 2004.

Enhancing Interoperability services in the U-campus Digital Library Project

Morales-Salcedo Raul, Hiroaki Ogata and Yoneo Yano
Department of Information Science and Intelligent Systems, Faculty of Engineering,
The University of Tokushim. Japan
[raulms, ogata, yano]@is.tokushima-u.ac.jp

Abstract

Digital libraries have the potential to not only duplicate many of the services provided by traditional libraries but to extend them. Basic finding aids such as search and browse are common in most of today's digital libraries. But just as a traditional library provides more than a catalog and browseable shelves of books, an effective digital library should offer a wider range of services. Using the traditional library concept of special collections and the concept of virtual spaces, in this paper we propose that explicit creating collections using virtual spaces in the digital library –virtual collections- can benefit both the library's students and teacher's contributions and increase its viability. We first introduce the concept of a virtual collection, outline the costs and benefits for defining such collections, and describe an implementation of collection-level metadata to create virtual collections for learning proposes in a distributed digital library. We conclude by discussing the implications of virtual collections for enhancing interoperability and sharing across digital libraries, such as those that are being developed as part of the Ubiquitous Campus project (U-campus).

1. Introduction

Most of the digital library research and development to date has centered on issues related to the technology and content of digital libraries [10]. This work has focused on issues such as developing effective ways to digitize and store resources, how to efficiently deliver resources over the network, providing ways to search for resources, and how to enable digital libraries to interoperate.

These are fundamental issues to be sure, but to be viable in the long run a digital library must be more than a collection of digital objects that can be efficiently stored and transported. Just as the traditional library evolved to provide services to make its contents more accessible to its users, the effective digital library must develop a range of services to assist its users in finding, sharing, cataloging, understanding, and using its contents. Moreover, in its digital form the library has the potential to not just emulate traditional libraries in the services it provides but to improve and extend them by capitalizing on advantages inherent in the medium.

One important area where the digital library can extend the services it provides beyond that of the traditional library is in integrating and highlighting user contributions. With the exception of especially unique or noteworthy contributions, the traditional library is rarely eager to receive resource contributions outside of its usual channels, as the effort needed to catalog and integrate outside contributions into a physical library is substantial. Digital libraries, on the other hand, are more often willing to receive contributions. It has been demonstrated that a combination of minimal submission data and basic verification procedures can result in high-quality digital library contributions with low rejection rates [6] [12]. Such contributions enhance the value of the digital library by increasing its size and diversity and the process of cataloging and integrating contributed resources into a digital library often requires less effort.

However, the aspects that make digital libraries built from user contributions valuable—diversity of content, potential for large growth—also create potential drawbacks. For example, search and browse facilities enable users to find learning resources based on features such as author, subject, or keywords, but as a digital library grows, finding specific resources of interest among the entire collection can become more difficult. At the same time, the prominence of a given contributor's contributions becomes diminished as the library grows.

One way to help users find resources of interest in a digital library while ensuring that contributors receive recognition is to borrow a concept that has long been part of traditional libraries: the special collection. By defining and making available virtual collections we believe the digital library can extend the specific collection model and—at a modest cost—provide benefits to both its users.

2. Collections

Traditional libraries often contain, in addition to their main holdings, special collections. In these settings a special collection is generally defined as a group of related materials that is given some form of special treatment. The special treatment might be due to the rare or delicate nature of the materials (rare books or antique maps, for example), or because the library wants to highlight the materials in some way (collected material of some classes or specific areas).

In contrast to traditional libraries, the special or collections of digital libraries can be much more fluid. Where the holdings of a traditional library are physically constrained to a single space and a single ordering, resources in a digital library can be distributed across many servers, can be owned by different universities or organizations, and can be displayed in many different orderings and arrangements. As suggested in [4], however, even a broad definition of a collection in the context of digital libraries can be ambiguous. It can, for example, be influenced by the point of view of those making the definition. Defining sub-collections can be even more flexible as there are many possible factors that can suggest how sub-collections can be formed. A sub-collection can be defined by including all those resources that share a topic or other significant attribute (the collection of all japanese, spanish or english language classes), those contributed by a specific organization (the resource collection of the university of foreign languages), or those used for a specific purpose (all resources used for the online course of languages).

These sub-collection examples are instances of collections that cannot be easily replicated in traditional libraries. They are made possible by exploiting advantages the digital environment inherently provides: objects can exist in multiple collections, collections with the same objects grouped in different ways can co-exist, collections can be created dynamically and exist for varying amounts of time. They become virtual collections and as such—in contrast to the traditional library—enable a digital library to provide a limitless number of sub-collections based on a wide range of features.

3. Benefits of Virtual Collections in a Ubiquitous Campus

Although it is common for traditional libraries to create and maintain special collections, many digital libraries do not attempt to provide a similar service. Most digital libraries do create the most basic of virtual collections—the result set dynamically created from a search request or category browsing—but rarely do they explicitly create and promote the sort of virtual collections described above. By providing access to users to this kind of virtual collections using internet, wireless and mobile technologies allows them to interact with the digital library environment for learning activities anywhere at anytime.

A digital library that is available anywhere at anytime containing virtual collections helps its users in several ways. Firstly, it provides permanency; users never lose their work unless it is purposefully deleted. A new user who may be intimidated by a digital library's search interface or the number of results returned by a query might be better introduced to the digital library through the more easily exploreable partitioned set of resources in a virtual collection. In addition, all learning activities and processes are recorded continuously. A directory of the virtual collections contained by a digital library, as shown in Figure 1, can provide a good introduction and overview of the library's contents to new or casual users.

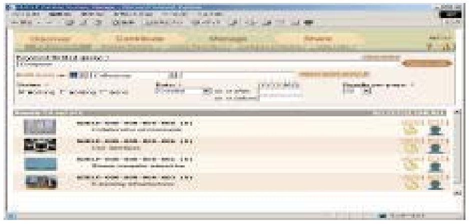


Figure 1. Virtual collections available in the U-campus digital library project

Associating resources with virtual collections enables those resources to be found more easily, either by browsing the contents of a highlighted virtual collection (launched from a page such as that in Figure 1) or through standard search and browse interfaces. Figure 2 shows how virtual collections are available from the browse page of the U-campus Digital Library Project, a distributed digital library of learning resources.

By proving accessibility, users have access to the digital library's virtual collections from anywhere. Adding virtual collections to search facilities, such as that of the U-campus digital library of educational resources shown in Figure 3, enables a user to perform a standard search but restrict it to a specific virtual collection, which could provide a more manageable and higher-quality result set than by searching the entire digital library, therefore wherever user are, they can get any information immediately.

Looking at the use of the digital library from a "learning-oriented perspective" [9], other benefits to the user stem from a more productive use of time. In [11] it is suggested that sub-collections can facilitate learning by isolating a group of related content and enabling a user to focus on those resources. Defining virtual collections makes it easier for users to find and work with such groupings of related content, either through a listing of available collections as in Figure 1, or by a "related resources" link based on virtual collection associations and tied to specific resources. Additionally, the virtual collection description might include links to related information outside the digital library, thus guiding users to more materials for their learning.

In most cases those who contribute resources to digital libraries are not directly recognized, yet digital libraries often depend largely on contributions for the content they provide. In a University library for example, teachers usually put supplementary material of their classes in reserve, making these "special collections" available to all students for a certain period of time. It is, therefore, in the best interests of the digital library to find ways to encourage new and repeat contributions. Virtual collections can recognize contributors in several ways. First, they provide an alternative distribution outlet. Users often have collections in which they have invested effort in creating and would like to see used more widely (for example, supplementary multimedia material of a language class at the university). Because a digital library will generally have a much larger base of regular users than contributors, contributing the collection gives the contributor's resources more exposure.

Virtual collections can not only help improve a set of resources and support their distribution for learning proposes, but can also offer basic infrastructure of services. In some cases, such as with the U-campus Project where resources are quite large (video and audio files), contributing resources enables the contributor to share resources without the overhead of storing and managing them, while retaining an association with them. If a contributor owns a large number of resources, this is a significant benefit itself, and one that has been taken advantage of by several contributors at the U-campus project.

Figure 2. Virtual collections as browse choices

Finally, if the digital library shares information about resource usage, either directly to its contributors or as is increasingly common, through most recommended, the contributor can gauge the relative demand of his contributions. This is helpful not only to contributors and the users of the digital library, but also "helps new contributors to understand what is considered a good item" [7].

Figure 3. Virtual collections search criteria

4. Implementing Virtual Collections

The benefits of virtual collections do not come without a price, of course. For a digital library to be able to easily create and remove virtual collections, to associate resources with different virtual collections in a flexible way, and to help users find and use the virtual collections, the library must have a structured approach to representing these collections. Moreover, to make creating such collections practical, this approach should also strive to minimize the costs associated with creating virtual collections.

In the remainder of this paper we describe an approach to implementing virtual collections in personal and group spaces based on our research in creating learning collections for a digital library in the U-campus project. We first define personal and group spaces in terms of learning activities and then review the current research related to representing collections in digital libraries for learning proposes and describes the costs and benefits of different types of metadata used to represent these collections. We then describe how we used and extend this information to define a collection-level schema for Educative Virtual Collections (EduVC) digital library and discuss practical issues related to implementing the schema in the U-campus project.

4.1. Virtual Spaces

We define personal space as a virtual area that is generated, owned and maintained by users to persistently keep resources objects or references to resources which are relevant to a task or set of tasks the user needs to perform within the learning processes. Personal spaces may thus contain digital documents in multiple media, personal schedules, visualization tools, and user agents. Resources within personal spaces can be pre-assigned according to the user's role. For example, a research user would have access to research-specific topic materials, visualization tools and interfaces to communicate (video, audio or text based chat) with his/her colleagues. Agents may be available for recommending virtual collections or library materials that are relevant to the research topic and the personal space could be enriched by the agent's suggestions.

Similarly, we define group spaces as virtual areas in which users can meet to conduct collaborative activities synchronously or asynchronously. These virtual areas are created dynamically by a group leader or facilitator who becomes the owner of the space and defines who the participants will be. Group spaces can be generated automatically when a number of users have been detected to have similar user profiles or interests around a given topic or task. In addition to direct user-to-users video-communication, users should be able to access virtual collection's materials and make annotations on them for every other group participants to see.

4.2. Metadata

Metadata is a key element of any library, traditional or digital. Metadata is used by libraries to describe and organize item-level resources and by users to search and browse the library. It consists of a set of elements that describes a resource. Collection-level metadata performs a similar function for collections and is used in traditional libraries for discovery across collections.

Work on collection-level metadata from several fields including archives, museums, libraries, and the Internet is relevant to the design and implementation of virtual collections for learning proposes. As outlined in [13], each field defines collections differently and has different standards governing collection description. The past few years have seen a movement to create a standard for collection description that is informed by, yet transcends, the fields from which it is derived. There are existing technologies, standards and ongoing initiatives for collection-level metadata for learning proposes. The alliance of remote instructional authoring and distribution networks for Europe [1], Dublin Core initiative [3], IMS global learning consortium [5] and, IEEE learning object metadata working group [15] are the most important initiatives dealing with metadata for computerized learning. Work in UK, USA and MEXICO has had also resulted in the formulation of goals for collection-level metadata and the definition and development of schemas to describe collections.

Based on work with the eLib working group on Collection Level Descriptions, the RIDING Clump Project created a searchable database of collection descriptions to provide information about what was available in its libraries [2]. The purpose of its scheme was to describe any type of collection—physical or virtual (electronic), networked or otherwise. RIDING collection metadata should allow users to discover, locate and access collections, search across multiple collections and allow software to provide services based on user preferences.

The Research Support Libraries Program (RSLP) Collection Description Project developed a model allowing all the projects in its program to describe collections in a consistent, machine readable way [14]. The RSLP builds upon the RIDING goals above by requiring that collection metadata allow the refinement of distributed searching approaches based on the characteristics of collections.

The UDLA Digital Library project in the Universidad de las Americas - Puebla, Mexico [16] is a research and development digital library project focused on provide access to special collections such as antique books, newspapers and historical documents. UDLA project allows to navigate, to visualize and to consult this kind of special collections identified by static metadata.

Several themes emerge from this survey of requirements. First, it highlights the importance of establishing standardized collection level metadata schemas that can effectively describe and manage a diverse set of collections and their metadata. Second, it argues that the schemas must support a number of functional library services that enable users to access collections and items, to search for materials, and to comprehend and use them effectively for learning proposes.

4.3. Types of Metadata

One challenge to creating collection-level metadata noted in the literature is the potentially high cost of production. Metadata can be automatically-generated or human-created [10] with the latter clearly imposing more significant costs in terms of human effort and time. In the context of collections, [4] describes two types of roughly corresponding metadata: "inherent" metadata, or information that can be extracted from the resource objects themselves, such as total number of objects or total file-size of the collection; and "contextual" metadata, or metadata which involves human judgment to create, such as a textual description of a collection of resources.

There are significant advantages to utilizing inherent and template-based metadata as much as possible. Because it can be generated automatically or based in general information, metadata has minimal costs associated with its creation and maintenance and can be updated on a regular or automated schedule. In contrast, human-created metadata is time-consuming, error-prone, costly to create, and more likely to be inconsistent. A person assigned to create metadata may only perform this task on an occasional, as-needed basis, and it may be a lower priority task than others for which that person is also responsible. Inconsistencies in metadata assigned to resources can arise due to variations in a given cataloger's judgment over time and because different catalogers may make varied judgments in cataloging resources.

There are drawbacks to relying solely on inherent and template-based metadata to define virtual collections, however. A risk in complete automation is the loss of many of the benefits of creating virtual collections, and also the application of metadata is mainly limited to content. Our first observation is that such a risk and application of metadata can not describe dynamic objects such as multimedia elements. Metadata can not influence multimedia content itself, because metadata usually contain universal and widely applicable description of objects. Contextual metadata is important because it enables us to give some character and cohesiveness to the virtual collection. Human created metadata is thus vital for articulating the scope, intent, and function of a particular collection, attributes that are likely to make the collection easier to locate. The use of metadata and human judgment in selecting resources to be included in a virtual collection has other benefits. Virtual collections can be described in terms of expected use in addition to being characterized by the terms they actually contain. It can be described in a dynamic way in order to facilitate the I/O behavior of a dynamic element. Resources can be more carefully chosen for inclusion in a virtual collection, with consideration of expected use, resulting in a more concise collection of high-quality resources that is easier to for the user to search, visualize and browse.

However, it is important to recognize, that a collection-level schema that relies heavily on contextual and dynamic metadata is relatively costly to implement and thus less likely to be maintained in the long term. A more viable approach is to define a schema for virtual collections that balances the costs and benefits of each type of metadata. In short, a cost-effective schema should include useful inherent and template-based metadata, supplemented by contextual and dynamic metadata that captures human judgments of a collection's nature and the selection of criteria for inclusion in the collection. From our point of view, the use of both inherent and contextual metadata schemas requires a new sort of metadata that includes useful, inherent, dynamic and template-based metadata supplemented by human judgments to facilitate the behavior of dynamic elements in digital collections.

5. Virtual Collection in U-campus Project

As stated in the specification of the IEEE's learning object metadata and according to [15], "a learning object is defined as any entity, digital or non-digital, which can be used, re-used or referenced during technology-supported learning". Examples of learning objects include multimedia content, instructional content, instructional software and, software tools referenced during technology-supported learning. In a wider sense, learning objects could even include learning objectives, persons, libraries, universities, organizations or events. A learning object is not necessarily a digital object; however the reminder of this article will focus on learning objects that are stored in the digital library's virtual collections.

The U-campus digital library that we are developing contains learning items and accepts contributions from anyone, subject to review before being made publicly available. Substantial collections of resources have been contributed to the digital library by a single person or organization. These collections include a group of english learning material that has been digitalized as part of second language learning program, a collection of images and related information of computer supported collaborative learning and, substantial contributions of video from some recorded japanese TV programs of english.

Sub-collections resources of the U-campus digital library can be found through various searching and browsing mechanisms. However, for reasons discussed earlier, we felt that creating virtual collections within personal and group spaces to represent the contributed sub-collections would benefit both the contributors and the users of the digital library. Specifically, our primary motivations for developing virtual collections were to highlight the work of authors/creators who contributed a critical mass of materials on a topic, to streamline the creation of item level metadata, to customize learning objects and to provide users with another way of accessing, storing, visualizing and understanding the items available in the digital library within personal and group spaces.

5.1 Defining a Collection-Level Schema

IEEE description schema was chosen by EduVC in U-campus project because the set of elements was universal (it wasn't created to meet the needs of a specific digital library), yet provided the flexibility for customization, if needed. The IEEE schema was also selected because it is similar to the Dublin Core schema [3]. Dublin Core is a common item-level schema used by many digital libraries, which would facilitate mapping elements and exchanging data. Previous work from the IEEE and the Collection Description Focus [6] resulted in thorough documentation, which facilitated understanding and implementing the schema in a relatively short amount of time. Other projects currently use the IEEE schema to describe large, unrelated, relatively static physical collections in a digital environment. Our contribution is to use and extend the IEEE schema using inherent and contextual static and dynamic metadata to describe "born digital-learning" objects of varying granularities, with varying relationships and at varying stages of collection growth in a digital library.

EduVC formulated requirements used to select IEEE elements, and, more generally, to measure the success of implementing the IEEE schema. Useful collections descriptions can be created dynamically in personal spaces by identifying a subset of elements relevant to users, by ensuring that metadata is complete within a collection description and consistent across collections and by presenting descriptions in an easily understood interface.

Low-cost metadata creation can be accomplished by harvesting metadata automatically (template-based metadata), by requiring the collection creator, rather than a cataloger, to describe their collections and by providing an efficient cataloging tool.

Using the IEEE schema, collection descriptions were created for EduVC digital library and its sub-collections. It was important to identify the metadata source (item-level record, collection creator, subject-area reviewer) to track the cost of creating static metadata. By starting with the complete schema we identified elements that aided understanding the collection (especially for learning proposes). This process also identified elements, which were not included in the collection record interface and the collection cataloging tool being developed. The resulting subset of elements met our requirements: collection descriptions could be created and extended with minimal cost while providing sufficient information to aid discovery.

5.2 Implementation of the Collection-Level Schema

The IEEE schema contains lots of elements. Originally, U-campus digital library project implemented a subset of thirteen. After another iteration of testing and design, we extended these thirteen to twenty-one IEEE elements. The element subset records of information about collections were implemented as template-based metadata (catalog id, description, access policies, relationships to other collections, and collection owner contact information. The template-based subset was chosen because the initial cataloging process consistently yielded data for these elements. The subset also matched the types of data reflected in item-level records. This provided users with consistent information between object (item) and collection.

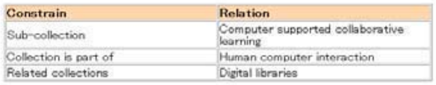
During the initial implementation, some elements were not included in the subset type. Because the IEEE schema has been used to describe physical collections, the developers created a controlled vocabulary to distinguish between collection types. We used the type element during the initial cataloging process, but found that some collections were not often of the same type, so the same vocabulary terms were used repeatedly with no distinction. Also, the terms would have to be explained to collection contributors and users, which could be a barrier to cataloging in personal or group spaces using collections. Recently, however, U-campus digital library project has incorporated the type element into its schema as a means to distinguish between virtual collections created for different spaces and purposes, such as collections organized around a specific personal space contributor and collections containing resources from different personal space contributors (group spaces) intended for a special purpose, such as a test collection.

Vocabulary for Type, which classifies collections by curatorial environment, content or policy, EduVC created a new vocabulary more appropriate to its resources.

The cost of creating collection-level metadata can be reduced by automatically populating template-based fields in the collection description. In EduVC, "manually-entered" metadata is provided by the collection creator via a cataloging tool or by a subject-area reviewer when the collection-level metadata is examined. "Automatically populated" template-based metadata is derived from querying specific fields of item-level resources within the collection. In our case, collection descriptions are completely comprised of contextual metadata that is manually entered either at the item or collection level. Currently three fields in the subset can be automatically populated with template-based metadata from the item-level description. For collection description to be cost-effective the cost of item-level metadata creation must be minimized and more fields in the collection description must be dynamically populated. Implementing a cataloging tool within personal and group spaces in a usable interface for collection contributors is another way to reduce the cost of creating collection-level metadata. In the prototyped collection cataloging tool shown in Figure 5, metadata for other fields will be supplied in drop-down menus with standardized vocabulary or text boxes that can be modified and extended by collection contributors or reviewers dynamically. This will ensure consistency in collection description.

Also, a well-designed interface with clear instructions should minimize the cost of metadata creation in terms of a contributor's time. For example, when a collection record is rendered in XML, the elements retain their IEEE attributes; however, field names were changed on the interface (IEEE attribute "Concept" becomes "Keyword"; "Super Collection" becomes "Collection is Part Of"). EduVC hopes to pass the majority of the cataloging costs on to its collection contributors as a trade-off for having the collection publicized and also incur some cost through the involvement of the subject-area reviewer as they error-check metadata and recommend changes.

One aspect of metadata creation that U-campus digital library contributors and EduVC subject-area reviewers share is identifying the relationships between collections and expressing them through the relational fields (Contains Sub-collections, Collection is Part Of, Related Collections). These relationships can be applied to collections of varying sizes and granularity, as in Figure 4, which shows the relational fields of EduVC and the U-campus digital library project.



Constrain	Relation
Sub-collection	Computer supported collaborative learning
Collection is part of	Human computer interaction
Related collections	Digital libraries

Figure 4. Collection relationship

As collection-level metadata becomes widely used, we believe the relational attributes will be essential not only for discovering resources within personal and group spaces repositories, but also across digital libraries. However, the larger and more distributed the digital libraries become, more difficult will be for users to find valuable resources and the (often small) collections they need to represent in their personal or group spaces. By explicitly representing not only a wealth of virtual collections, but also the relationships among them, regardless of their physical location or collection-level metadata schema we need to improve the navigability of digital library.

6. Conclusion

The collection-level metadata schema that we have developed and extended has started to be tested in our U-campus digital library project enabled us to define virtual collections that benefit both library users and collection-providers in several ways. But in a broad sense, the most important beneficiary may be the U-campus digital library itself.

Virtual collections encourage us to see a digital repository not as unitary structure, but as virtual and modular representation of learning objects that comprise many sets of resources, some small and others large, some separate and others overlapping, some stable and others transient, some defined by the library managers and others extended dynamically by library users. We think this is a compelling perspective. In fact, large scale digital libraries are increasingly adopting just such a modular structure.

There are a number of reasons this perspective could be attractive to other projects. In the first place, as we have noted, it is often costly to create metadata. Object level metadata is the most costly of all, since it describes the "atomic" digital learning objects in a collection. However, it is often unnecessary to incur this cost: for example, when all members of a set of learning objects are similar, item descriptions are redundant. In such cases, collection-level descriptions will be more cost-effective than object-level metadata. However, a given repository may have some distinct objects, as well as sets of similar components and services. This means that descriptions of resources in a collection should neither be fixed at a low level of granularity (object-level metadata) nor or at a high-level (complete collections), but must change as needed. In other words, a cost-effective way of describing a collection will require the flexibility of virtual collection metadata schemas within personal and group spaces such as the one we have presented here.

The EduVC metadata-based framework also addresses the customization of learning objects within personal and group spaces. Having explained the extensions and challenges of metadata, we describe our implementation in U-campus digital library project. Technical details concerning the transaction of collection-level descriptions among federated repositories will also need to be worked out, if metadata is going to be shared across a distributed personal and group spaces in the U-campus digital library at low cost. Fortunately, many of the protocols that have been tested for learning object-level metadata should also apply straightforwardly to collection-level schemas as well. For example, the Metadata harvesting protocol [8] enables personal and group space's collection to provide easily their metadata to services providers. By agreeing on a standard collection-level metadata schema it should be as simple for virtual repositories to exchange collection information as it now is for them to share learning object records.

7. References

[1] Alliance of Remote Instructional Authoring and Distribution Networks for Europe. Available at: http://www.ariadne-eu.org

[2] Brack, E.V., Palmer, D. and Robinson, B. "Collection level description – the RIDING and Agora experience". *D-lib magazine*, September 2000.

[3] Dublin Core Metadata Initiative. Available at http://www.dublincore.org

[4] Hill, L. L., Janee, G., Dolin, R., Frew, J. and Larsgaard, M. "Collection metadata solutions for digital library applications". *Journal of the American Society of Information Science*, 50(13), p. 1169-1181.

[5] IMS Global Learning Consortium, In. "IMS Learning Resource Metadata Specifications" Available at http://www.imsproject.org/metadata/

[6] Jones, P. "Open(source)ing the doors for contrubutor-run digital libraries". *Communications of the ACM*, 44(5), May, 2001, p. 45-46

[7] Johnston, P. and Robinson, B. "Collection convergence, the work of the collection description focus". *Ariadne*, 29, October 2001.

[8] Lynch, C. "Metadata harvesting and the open archives initiative". *ARL Bimonthly Report 217*, August 2001.

[9] Levy, D. M. and Marshall, C.C. "Going digital". *Communication of the ACM*, 38(4), p. 77-84, April 1995.

[10] Marshall, C.C. "Making metadata". *In proceedings of the third ACM Conference on Digital Libraries*, p.162-171, June 23-26, 1998. Pittsburg, Pennsylvania.

[11] Morales-Salcedo, R., Yoneo, Y. and Ogata, H. "Hyzone: diversifying resources in learning spaces via personalized interfaces". *In proceedings of the International Conference on Web-based Education*, February 16-18, 2004. p. 37-42. Innsbruck, Austria.

[12] Morales-Salcedo, R., Yoneo, Y., Miyoshi, Y. and Ogata, H. "Collaborative spaces in a distributed digital library". *In proceeding of the International conference on Computers in Education*. Hong Kong, China. p. 126-129, 2003.

[13] Powell, A. (ed). "Collection level descriptions: a review of existing practice". *D-lib Magazine*, September 2000.

[14] Powell, A., Heaney, M. and Dempsey, L. "RSLP collection description". *D-lib Magazine*, September 2000.

[15] The IEEE Learning Object Metadata. "WG12: Learning Object Metadata". Available at http://ltsc.ieee.org/wg12/

[16] The Universal Digital Library for All "UDLA". Available at http://biblio.udlap.mx.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 127-136, 2004.

On The Evaluation of Adaptive Web Systems

Hossein Sadat and Ali A. Ghorbani
Intelligent and Adaptive Systems Group
Faculty of Computer Science, University of New Brunswick
Fredericton, NB, E3B 5A3, Canada

Abstract

Adaptive Hypermedia Systems *(AHS) affect the way most of the Web-based applications are developed and used. One of the application domains affected by AHS is the domain of Web-based Support Systems (WSS), which is part of the more general online information systems domain. Since the very beginning of the research on AHS, different systems have been developed for different domains and application areas (such as educational systems, online information systems, etc.). Besides some general features, which are expected to be present in any software system, AHSs have their own extra features that make them different from non-adaptive systems. Through a careful study of a large number of adaptive hypermedia systems, a hierarchy of primary features based on which one can evaluate different adaptive web systems is proposed. An evaluation weighting scheme is also proposed for the given features. A comparative analysis is carried out for 4 major adaptive hypermedia systems (AHA!, InterBook, SETA and SeAN). The results of our detailed evaluation put AHA! at the top followed by SETA, SeAN and InterBook, respectively.*

1 Introduction

Adaptive hypermedia systems are those systems that change their behaviour according to their context. *Adaptive* refers to the ability of the system to change its responses in reaction to runtime environment, users, and other parameters. Adaptiveness affects all types of the Web-based systems, one of which is Web-based Decision Support Systems (WSS) category. WSSs are online information systems that provide decision-making information to its users (decision makers) based on available data, information, or knowledge. An adaptive WSS helps its users to quickly locate and find information they are looking for. It also maximizes a user's ability in finding relevant information by exploiting their historic usage (access) behaviour and other environmental contexts.

During the recent years, quite a few adaptive hypermedia systems have been developed. In addition, the current state of research in the adaptive hypermedia/Web-based systems suggests that a lot of new systems are being and will be developed. These systems have different domains, though some of them may be general enough to be applied to various domains. The point that needs to be taken into consideration is that although adaptive system may be developed for different domains, there are general functionalities, qualities and features expected to be present in them. Most of the systems focus on some specific areas, such as user modelling or data mining. Some of the them are not really systems, but algorithms or useful tools to be used in the development of AHSs. However, a complete and usable adaptive hypermedia/Web-based system is supposed to have all the important adaptive features as well as general software-related features. For instance, if a system proposes a new model for the adaptive hypermedia applications, then the corresponding methodology for the development of the suggested model should also be proposed. Otherwise, the new model cannot be practically applied.

There are some efforts to develop a general framework for AHS that offers all the functionalities, qualities, tools and methodologies. It would be useful if there were an evaluation framework to realize the features and functionalities of an AHS. In this paper, we propose such a criteria based on a hierarchy of features.

The rest of this paper is organized as follows. In the next section, we briefly review all the systems developed in AHS research. Section 3 presents a hierarchy of features that covers various aspects of an AHS. In Section 4, based on the proposed features, we evaluate four adaptive Web-based systems. Finally, the results and conclusions of the present study are summarized in Section 5.

2 Adaptive Hypermedia/Web-based Systems

In order to define an evaluation framework, it is helpful that the previous efforts toward building adaptive Web systems be reviewed. In fact, such evaluation criteria would be the result of studies on the features those systems support and what they need to support (but don't support).

Table 1 shows a list of the projects in the area of AHS, including frameworks, prototypes (application), methodologies, models, and other kinds of software systems. We have identified these systems based on their domain and type (framework, prototype, etc.).

In the following sections, a short description of each of these systems is given:

ACT-R [12] is an electronic bookshelf, based on InterBook, which has been developed to support learning ACT-R, a theory in cognitive psychology.

ADAPTS [13] is an electronic performance support system that integrates an adaptive diagnostics engine with adaptive access to supporting information.

AHA! [10, 9, 11]is an adaptive website framework, which has been used to implement some adaptive website in educational area.

AHAM [25] is a model for adaptive hypermedia applications, based on Dexter reference model, which divides an AHS into domain model, user model, teaching model, and adaptation engine.

ALPHANET [44] aims to build a learning environment that offers intelligent personalization capabilities and addresses the problem of effective adaptive learning for individual learners.

AOAI [57] is an adaptive interface between a Web search engine and a user.

Arthur [22] is a Web-based instruction system that provides adaptive instruction.

AST [51] (Adaptive Statistics Tutor) is an adaptive courseware on the WWW.

BASAR [53] is an agent-based framework in which the agents filter information, initiate communication, monitor events, and perform tasks. The agents rely on usage profiles to adapt their assistance to specific users.

Broadway V1 [54] is a WWW browsing advisor reusing past navigation from a group of users.

CHEOPS [40] is an educational server-side package that can provide a requested page based on the user profile and history.

DI2ADEM [46] (Diffusion of Information with Interactive and ADaptive Environment in Medicine) aims at designing an interactive and adaptive environment, intended to improve the diffusion of medical knowledge.

ELM-ART [48] is a web-based course system to support programming in Lisp. In fact, ELM-ART is a hyperbook with two additional features: adaptive navigation support and intelligent problem solving support.

Fab [6] is a test-bed for comparing different adaptation techniques and it is based on agent technology. It's an automatic recommendation service that adapts to its users over time.

Table 1. Adaptive Hypermedia Projects

Project	Domain	Type
ACT-R	Educational	Prototype
ADAPTS	Miscellaneous	Prototype
AdaptWeb	Educational	Framework,Prototype
AHA!	General	Framework, Prototype
AHAM	General	Methodology
ALFANET	Educational	Miscellaneous
AOAI	Miscellaneous	Miscellaneous
ARNIE	Miscellaneous	Miscellaneous
Arthur	Educational	Prototype
AST	Educational	Prototype
BASAR	General	Framework
Broadway	General	Prototype
CHEOPS	Educational	Miscellaneous
DI2ADEM	Online Info. Sys	Prototype
ELM-ART	Educational	Prototype
Fab	General	Miscellaneous
GAHM	General	Model
GAS	General	Miscellaneous
GRAS	Miscellaneous	Algorithm
Hera	General	Methodology
HYPERADAPTER	Educational	Prototype
HyperAudio	Multimedia	Prototype
HyperContext	General	Model
ILEX	Online Info. Sys	Prototype
InterBook	Online Info. Sys	Framework
iWeaver	Educational	Framework
KBS HyperBook	Online Info. Sys	Framework
MASPLANG	Educational	Framework
METIORE	Multimedia	Miscellaneous
MMA	Online Info. Sys	Prototype
NetCoach	Miscellaneous	Framework
PageGather	General	Algorithm
Peba-II	Online Info. Sys	Prototype
PEGASUS	General	Framework, Model
PersonalWebWatcher	General	Miscellaneous
PowerBookmarks	Miscellaneous	Miscellaneous
PUSH	Miscellaneous	Prototype
PVA	Miscellaneous	Miscellaneous
RATH	Educational	Prototype
RLATES	Educational	Prototype
SeAN	Online Info. Sys	Framework
SETA	e-Business	Framework
SKILL	Educational	Framework
SmexWeb	Educational	Framework
SQLTutor	Educational	Prototype
SWAN	Miscellaneous	Framework
TANGOW	Educational	Framework
UWE	General	Methodology
VALA	Educational	Miscellaneous
WEAR	Educational	Miscellaneous
WEBMINER	General	Miscellaneous
WebWatcher	General	Miscellaneous
XAHM	General	Model, Framework

GAHM [45] is a formal approach to the modelling of personalizable, adaptive hyperlink-based systems.

GAS [50], Group Adaptive System, is a collaborative environment that provides the interface and tools for a group of people to share their browsing experience.

GRAS [28] (Gaussian Rating Adaptation Scheme) is a new personalization algorithm for hypermedia databases, which combines content-based and social filtering.

Hera [26, 7] is a design methodology aiming at automated generation of adaptive hypermedia presentations.

HYPADAPTER [24] is an adaptive hypertext system designed to individually support exploratory learning and programming activities in the domain of Common Lisp.

HyperContext [52] is a new model for adaptive hypertext. HyperContext achieves adaptation of the information and hyper-links through explicit context.

ILEX [43, 34], the Intelligent Labeling Explorer system, uses NLG (Natural Language Generation) technology to generate descriptions of objects displayed in a museum gallery.

InterBook [49] is a tool for authoring adaptive textbooks on the web.

iWeaver [56] is an interactive web-based adaptive learning environment, which aims to create an individualized learning environment that accommodates specific learning styles.

KBS Hyperbook [41] is a framework for designing and maintaining open, adaptive hypermedia hyperbooks in the Internet.

MASPLANG [32] is focused on the utilization of intelligent agents in online learning environments.

METIORE [14] is a Personalized Information Retrieval system that keeps a user model based on objectives.

MMA [20] (Mars Medical Assistant) uses a combination of user, situation, and task models to create virtual hypertext structures by piecing together medical information components.

NetCoach [55] is an authoring-system, which allows to create adaptive and individual course modules without programming-knowledge.

PageGather [47] is an algorithm to find collections of related pages at a web site, relying on the visit-coherence assumption.

Peba-II [35] is an on-line animal encyclopedia, which produces descriptions and comparisons of animals as world wide web pages.

PEGASUS [17] (Presentation modelling Environment for Generic Adaptive hypermedia SUpport Systems) is a generic presentation system for adaptive educational hypermedia that is highly independent from domain knowledge representation and application state management.

PersonalWebWatcher [37] (PWW) is a search assistant based on WebWatcher. The main difference is that PWW establishes a user model for individual users and recommends to the users based on these learned models.

PowerBookmarks [31] is a Web information organization, sharing, and management tool, which parses metadata from bookmarked URLs and uses it to index and classify the URLs.

PUSH [19] is an adaptive help assistant for users of SDP(the documentation of a software development method).

PVA [18], Personal View Agent, is a system that can automatically organize a personal view by learning the users interests, and adapt the personal view to the users changing interests.

RATH [23] is an adaptive tutoring WWW software prototype combining a mathematical model for the structure of hypertext with the theory of knowledge spaces from mathematical psychology.

SeAN [3] is a an adaptive system for personalized access to news.

SETA [5, 4, 2] is a prototype toolkit for development of adaptive Web stores.

SKILL [42] provides the students a collaborative and adaptive learning environment utilizing new web technologies proposed by W3C.

SmexWeb [1] (Student Modelled Exercising on the Web) is an adaptive web-based tutoring system, which implements a user model considering cognitive and knowledge aspects as well as general abilities of the students.

SQL-Tutor [36] is a knowledge-based teaching system, which supports students learning SQL and can adapt to the needs and learning abilities of individual students.

SWAN [21] (Adaptive and Navigating Web Server) aims at designing adaptive web servers for on-line multimedia information systems about nautical publications.

TANGOW [16] is a tool for developing Internet-based courses, which facilitates the construction of adaptive learning environments for the Web.

UWE [30] is a UML-based Web engineering approach .

VALA [33] focuses on developing a learning architecture with user interface adaptability that provides a personalized learning environment for each learner.

WBI [8] (pronounced WEB-ee) is a multi-agent system that organizes agents on a users workstation to observe user actions, proactively offer assistance, modify web documents, and perform new functions.

WEAR [39] is a Web-based authoring tool for the construction of Intelligent Tutoring Systems (ITSs) in Algebra-related domains, such as physics, economics, chemistry, etc.

WEBMINER [38] is a system for pattern discovery from WWW transactions.

WebWatcher [27] is a program, which guides the user of a website through different pages of that website based of a set of given interests of that user.

XAHM [15] is an Adaptive Hypermedia Model based on XML, which is relatively more expressive than other models, in defining the domain and adaptation models.

3 The Features and Their Rationale

In order to establish how a system is adaptive and how many important aspects of an adaptive web system it supports, a set of features must be defined. These features include adaptation-related features, software quality features, software engineering features and technology features. These features are described in the following subsection.

3.1 Runtime Features

Runtime features are the most important factors in evaluating a system. They can describe how a system behaves and how it accomplishes its objectives. They also describe how well the system can be used, either as a black box or as a reusable and/or extendable set of components or library.

Adaptation Dimensions: The adaptation may take place based on various information about the user, runtime context parameters, and other related information. Generally, the following context information might be used in the adaptation process [15]:

(a) *User/User Community (Group)*: This specifies if the user preferences and browsing behaviour are taken into account in the adaptation. Also, user communities might be taken into account when the system is responding to an individual user. These user communities (groups) are usually extracted and updated by mining techniques referred to as *group mining*.
(b) *Environment*: This feature, realizes the consideration of external environment status, like time, location, etc.
(c) *Technology*: Technology feature, targets the different capabilities and characteristics of the client terminal or network. For example, a system can adapt the pages for delivery to a mobile client.
(d) *Unexpected Events*: There are other kinds of information that could be taken into account in adaptation process. In a system with more than one actor, for instance, at some point, one of the actors may impose some constraints on the system which affects other actors. To be more specific, imagine an online course system, with teacher, student and administrator, as three kinds of users (actors).

Adaptation Features: *Adaptation features* target the different aspects of adaptation in adaptive hypermedia systems as described in [29].

(a) *Navigation*: Direct guidance, link sorting, link hiding, link removal, link disabling, link annotation
(b) *Content*: Fragment addition/removal (conditional inclusion of fragments), fragment's level-of-detail support, fragment generation, fragment annotation
(c) *Presentation*: There are three different presentation adaptation: presentation adaptation based on user model in which the system decides how to present certain components to the user, based on the user model (e.g. the order of the information fragments); presentation adaptation based on environment where external environment can change the presentation format (e.g. language, location of the client, time); presentation adaptation based on technology: not all kinds of media or graphical elements can be used on all kinds of client machines (e.g. tailoring the presentation for handheld devices).

Authoring: Authoring features, describe how a system supports the development of an adaptive web site (if it is a framework or model), or the integration of its components into a system (if it is a technique or library), or any other useful task that can be automated. We consider simplicity, hierarchical development support (different roles for different levels of development), expressiveness, and other useful tools supported, as important authoring features.

Usability: This feature is concerned with how easy it is for the user to use the system and how much effort is required to learn, operate and interact with the system. For example, the way a system gathers user's feedback, affects usability. If the system requires that the user give explicit feedback, at some point the user might get tired or not feel comfortable using the system. The point that is of a great importance here is that, one of the goals of adaptive systems, is to help users find their way in the system and use the system easily. So, it doesn't seem logical to compromise the usability for the adaptability/adaptivity.

Security: All the systems, especially multi-user web-based software systems, must have some mechanisms to ensure that the system is secured, that is, no unauthenticated person may use the system and also, no unauthorized action may be accomplished in the system by a user. This concern is more evident in adaptive hypermedia systems, since the system services change in accordance to users and environment.

Privacy: Whatever mechanism a system uses to provide adaptivity to a user, it should not violate the user's privacy. So, it is considered as an important feature in a system. As an example, a system that uses Cookies to keep track of user sessions is somehow violating the user's privacy.

Performance: As adaptive systems usually use AI techniques and algorithms to do adaptation, it seems reasonable to consider the performance of such systems as a characterizing feature, since AI algorithms are most often time-consuming. For instance, automated reasoning in First Order Logic is a very slow procedure, though may be effective for some problems. If it is used in a system, then the performance of the system would be much slower than a system that doesn't use it.

Scalability: The AHS should be scalable in relation to both the content size and the number of users.

Generality: This feature determines if a system is general enough to be used in multiple application domains. Some systems are general to some extent. For instance, they can be used to author online information systems, whether an online help system or an online course. However, they cannot be used to develop an e-Business Web system. It is obvious that the more general a system is, the more difficult it's development is. Therefore, the efforts of a system to provide a general adaptive Web system should be taken into account.

Cost: The cost of a system may be calculated regarding various parameters, including the minimum required technology to run the system, the start-up time of the system (the time required to get the system up and running).

3.2 Technology

This category of features, capture the use of different technologies in the projects. For example, there are a lot of AI algorithms and techniques that can be integrated into an adaptive system. These techniques can range from information extraction algorithms (data mining) to the intelligent agents application in the system architecture.

Mining Techniques: An AHS may apply mining techniques to extract useful information. These mining techniques include usage mining, content mining, and structure mining.

Agent-based Features: The use of agent technology is considered as a distinguishing feature in software systems in general, and adaptive web-based systems, in particular. On one hand, using multi-agent architectures helps systems achieve a set of goals, such as distribution, high-level communication, and problem solving. On the other hand, these architectures have their own issues and problems to be addressed. Through this feature, the extent to which the agent technology is exploited in a particular system, is realized.

Page Synthesis: Page synthesis is the dynamic generation of Web pages. The adaptive Web systems usually have different degrees of synthesis. That is, some of them have pages stored somewhere, and change them on demand, whereas some of them produce the pages from a set of data in the database. Page synthesis feature is considered a noticeable feature since it captures the power of the adaptation in the system as well as the number of free parameters that are controllable in the adaptation process. There are different levels of synthesis, which might be used in an AHS: natural language generation, template-based page generation, and totally dynamic page generation.

3.3 Software Engineering

This category contains features that address the software development process.

Portability: It is very important that how portable an AHS is. For example, since most of the adaptive systems are based on server-side web development, it really matters if they can be ported to different Web servers, such as Windows-based or Linux-based servers.

Extendability: This feature addresses how the system can be extended across different dimensions. It is desirable that a system be extendable by new algorithms, techniques or functionality. As an example, suppose that a new mining algorithm needs to be integrated into the system. How much effort is needed to do this integration?

Flexibility: The way the system can be customized and changed to meet different configurations for an application, determines the flexibility of the system. It is an important feature that the system be flexible enough to be tailored easily and effectively. For example, in an application, there might be no need for group mining. Then, the efforts needed to remove this feature from the system determines its flexibility.

Maintainability: As a software quality attribute, maintainability is considered important, especially in the context of AHS; some AHSs have an adaptation model, which consists of some rules or programs. It is a concern that how the maintainability of the system is affected, regarding this extra model in the system. If a problem is detected in the system, how difficult it is to find the source of the problem and to change the required parts of the system, regrading the extra models that the system has.

Support for Design Models and Methodologies: If a system proposes a design process or a modelling procedure, then it's considered a design feature. It is desirable that an AHS be model-driven. For such a system, we are interested in determining how the system supports the design of different models, that is, the domain model, the user model, and the adaptation model.

Implementation: This category deals with the features related to the implementation issues of the system. We consider the programming language used in the development (the popularity, simplicity, etc.) and the platform on which the system works on.

Support: This section addresses the features that evaluate the way a project is supported for later developments or use:
(a) *Documentation*: is there any documentation about the system functionality?
(b) *Running Prototype*: is there any sample application that shows how the system works?
(c) *Continuing*: is the project still under development and research?

Open Source: If a project is open source, everybody can look at the source code and learn a lot. Some developers and researcher may change the code in some specific direction. In this way, it may become a test-bed for many other projects in the field. So it is a very important feature.

4 Selected Systems for Evaluation

We have carried out detailed evaluations on 4 AH systems: AHA!, InterBook, SETA, and SeAN. There is no concrete reason why we have selected these four out of the huge list of Table 1, however, we had some factors in mind (such as, documentation, implementation, etc.) when choosing them. AHA! is a rather general system mostly used in educational domains. InterBook is a general tool for developing online books. SETA is a framework to build adaptive Web stores. SeAN is an adaptive news system. The following is a summary of some of their features and functionalities.

4.1 AHA!

AHA! is a general adaptive hypermedia framework used in educational domains. In this framework, the domain is modelled through concepts and relationships between them. Concepts can be related to a resource (page or fragments, for instance). AHA! adapts the pages based on the user model. In AHA! the adaptation model and the domain model are interwoven. There are two tools provided to facilitate the authoring: Graph Editor, which is a high-level tool for defining concept relations, and Concept Editor, which is relatively low-level and used for rule definition. However, the tools generate XML files that can be edited manually. The AHA! has a predefined page structure in the sense that the pages are not synthesized, however, the author can include conditional fragments so that if some conditions hold, the fragment is not shown. No tool is provided for creating pages. The author has to use XHTML to define the pages. AHA! has content adaptation (conditional fragments) and link adaptation (link coloring). AHA! is an ongoing project. AHA! uses Java language and Servlet technology and it is platform independent. This project is open source and there is a quite good documentation online.

4.2 InterBook

InterBook is a tool for authoring adaptive textbooks on the Web. It uses a domain model of concepts and a user model to provide adaptivity. It provides two major parts, the glossary and the indexed textbooks. The glossary is the structured hierarchy of the domain. The textbooks are indexed so that each unit has a set of related concepts and the role of that concept. In addition to regular navigation support (back and forward, etc), InterBook provides an adaptive set of links between the textbook and the glossary based on the current user's knowledge. Also it provides visual cues about each link (adaptive annotation) and direct guidance about the suggested next place the user should visit. Another kind of direct guidance is used to provide prerequisite-based help for the user. Since the system knows the prerequisite relationships between concepts, when the user has difficulty understanding a concept or solving a problem, the system can suggest the unit that contains the concepts that are the prerequisite concepts of the difficult unit. InterBook is implemented based on CL-HTTP Web server using LISP language. The authoring has different stages and it uses pre-existing tools. It seems that in the hyperbook area, InterBook is a dominant tool for developing adaptive online books, since it has the tools and the server for serving the books online. However, it cannot go beyond this domain, hence, it is not considered a general AHS.

4.3 SETA

SETA is a prototype toolkit for development of adaptive Web stores. It exploits a multiagent-based three-tier software architecture, and is designed to allow building different Web stores by authoring tools, that is, all the domain-dependent knowledge about users and products can be configured by tools.

SETA dynamically generates the pages of a Web store catalog and selects the content of the pages based on the user's interests and familiarity with the products. Also the system sorts the available items for a product class based on

Table 2. The evaluation features and their corresponding weights.

Features			AHA	Interbook	SETA	SeAN
Run-time	Adaptation Dimensions(1)	User/Group(out of .75)	.5	.5	.5	.5
		Environment(.1)	0	0	0	0
		Technology(.1)	0	0	0	0
		Unexpected Events(.05)	0	0	0	0
	Adaptation Features(3)	Navigation(1)	.4	.5	.2	.2
		Content(1)	.5	0	.8	.8
		Presentation(1)	0	0	0	0
	Authoring(1)		.6	.8	.8	0
	Usability(1)		High	High	High	High
	Security(1)		Medium	Low	-	-
	Privacy(1)		Low	High	-	-
	Performance(1)		High	Medium	Medium	Medium
	Scalability(1)		Medium	Medium	Medium	Medium
	Generality(1)		High	Low	Low	Low
	Cost(1)		Low	Low	Medium	Medium
Technology	Mining Techniques(.4)	Usage Mining(.2)	0	0	.2	.2
		Content Mining(.1)	0	0	0	0
		Structure Mining(.1)	0	0	0	0
	Agent-based Features(.2)		0	0	.2	.2
	Page Synthesis(.4)		0	0	.1	.1
Software Engineering	Portability(1)		High	Medium	High	High
	Extendability(1)		High	Medium	Medium	High
	Flexibility(1)		High	Medium	Medium	Low
	Maintainability(1)		Medium	Low	High	Low
	Design(3)	User Model(1)	1	1	1	1
		Domain Model(1)	.5	1	1	1
		Adaptation Model(1)	.5	0	0	1
	Implementation(1)	Language(.5)	.5	.2	.5	.5
		Platform(.5)	.5	.5	.5	.5
	Support(3)	Documentation(1)	1	.5	.3	.2
		Prototype(1)	1	.5	.5	.5
		Continuing(1)	1	0	-	0
	Open Source(1)		1	1	0	0
Overall Normalized Result			.55	.37	.50	.47

the user's preferences. During a session, the system monitors the user's selections to figure out her needs for product functionalities and recommends potentially interesting product classes.

SETA system is developed using JDK 1.2 and uses the Apache Web Server.

4.4 SeAN

SeAN is a an adaptive system for personalized access to news. This system has a three-tier multi-agent architecture that is inherited from SETA project. SeAN has three goals: first, to select news topics relevant to the user. Second, to present an appropriate level of detail of the news based on the user model and third, to provide advertisement most relevant to the page and the user. SeAN uses a structured hierarchy to represent news (domain model). In fact, each news is considered as a composite entity having several attributes that define its components. For example, title, abstract, full text, author, pictures, video. Based on this representation, different levels of detail can be used for news according to the user model. This system relies on a modular and compositional approach to user modelling. SeAN has been implemented using Java.

5 Evaluation

Table 2 shows the evaluation results for the systems reviewed in the previous section. The value inside the parentheses in front of each feature indicates the maximum value for that feature. The values in each level of the feature hierarchy have been normalized and used in the higher level category. Due to the lack of evidence, some features are not assigned any values. These features have not been used in the normalization process. For the features for which we have used linguistic terms in Table 2, the corresponding numeric values[1] are used in the computations as well as in the normalization process. After normalization, each high-level category (i.e, Run-time, Technology, and Software Engineering), has a value between 0 and 1. Note that in our evaluation the *Run-time* and the *Software Engineering* features are considered to be more important than the *Technology* features. We gave the latter a weight of 0.2 compared to a weight of 0.4 for the other two categories. Taking these weights into account, we normalized the values and obtained a final value between 0 and 1 for a system. This value gives an estimate of the overall support of an AH system for the set of proposed features.

The results of our (subjective) evaluation of the above 4 adaptive hypermedia projects are summarized in Table 3. AHA! has received the highest value. This is mainly because of AHA!'s generality and software engineering considerations. SETA and SeAN are ranked very close because they use the same software architecture. InterBook is a useful system for adaptive online books, however, it doesn't have AHA!'s generality or software engineering considerations.

Table 3. The results of a (subjective) evaluation of 4 adaptive hypermedia projects.

Project Name	AHA!	SETA	SeAN	InterBook
Eval. results	.55	.50	.47	.37

6 Conclusions

Adaptiveness is becoming one of the most important features of hypermedia/Web-based systems. Web-based support systems, as part of the Web-based systems family, benefit from the advantages of AHS technology.

The features hierarchy presented in this paper gives a general evaluation framework to compare adaptive web systems regardless of their domains. The proposed hierarchy along with a weighting scheme made it possible for us to evaluate a number of adaptive hypermedia systems. The results of our comparative analysis of 4 major adaptive hypermedia systems (AHA!, InterBook, STEA and SeAN), show that AHA! is the winner. This is mainly due to the fact that AHA! is a fairly general AH system. Our evaluation results put SETA, SeAN and InterBook in the second, third and fourth positions, respectively.

Through careful study of the proposed features for an AHS, one may notice the weak points of the system under study and look for ways to compensate them. We are currently developing an adaptive hypermedia/Web-based framework, considering all these features.

[1] High: .9, Medium: .5, Low: .2

References

[1] F. Albrecht, N. Koch, and T. Tiller. Smexweb: An adaptive web-based hypermedia teaching system. *International Journal of Continuing Engineering Education and Life-Long Learning, Special Issue on Intelligent Systems/Tools in Training and Life-Long Learning*, 2001.

[2] L. Ardissono, C. Barbero, A. Goy, and G. Petrone. Adaptive web stores. In *Agents'99 Workshop: Agents for Electronic Commerce and Managing the Internet-Enabled Supply-Chain, Seattle, WA, May 1999*, pages 9–13.

[3] L. Ardissono, L. Console, and I. Torre. An adaptive system for the personalized access to news. *AI Commun.*, 14(3):129–147, 2001.

[4] L. Ardissono, A. Goy, R. Meo, G. Petrone, L. Console, L. Lesmo, C. Simone, and P. Torasso. A configurable system for the construction of adaptive virtual stores. *World Wide Web*, 2(3):143–159, 1999.

[5] L. Ardissono, A. Goy, G. Petrone, M. Segnan, L. Console, L. Lesmo, C. Simone, and P. Torasso. Agent technologies for the development of adaptive Web stores. *Lecture Notes in Computer Science*, vol. 1991:194–??, 2001.

[6] M. Balabanovic. An adaptive web page recommendation service. In W. L. Johnson and B. Hayes-Roth, editors, *Proceedings of the First International Conference on Autonomous Agents (Agents'97)*, pages 378–385, New York, 5–8, 1997. ACM Press.

[7] P. Barna, F. Frasincar, G.-J. Houben, and R. Vdovjak. Methodologies for web information system design. In *International Conference on Information Technology: Computers and Communications*, pages 420–?, 2003.

[8] R. Barrett, P. P. Maglio, and D. C. Kellem. How to personalize the web. In *Proceedings of the Conference on Human Factors in Computing Systems CHI'97*, 1997.

[9] P. D. Bra. Design issues in adaptive web-site development. In *Proceedings of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW*, 1999.

[10] P. D. Bra and N. Stash. Aha! adaptive hypermedia for all. In *the SANE 2002 Conference, Maastricht*, pages 411–412, 2002.

[11] P. D. Bra, N. Stash, and B. D. Lange. Aha! adding adaptive behavior to websites. In *Proceedings of the NLUUG Conference, pp. n-n+10, Ede, The Netherlands, May 2003*.

[12] P. Brusilovsky and J. Anderson. Act-r electronic bookshelf : An adaptive system to support learning act-r on the web. In *The 3rd World Conference of the WWW, Internet, and Intranet, WebNet'98*, pages 92–97, 1998.

[13] P. Brusilovsky and D. W. Cooper. Domain, task, and user models for an adaptive hypermedia performance support system. In *Proceedings of the 7th international conference on Intelligent user interfaces*, pages 23–30. ACM Press, 2002.

[14] D. Bueno, R. Conejo, C. Carmona, and A. David. Metiore: A publications reference for the adaptive hypermedia community. In *Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 480–483. Springer-Verlag, 2002.

[15] M. Cannataro, A. Cuzzocrea, and A. Pugliese. Xahm: an adaptive hypermedia model based on xml. In *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, pages 627–634. ACM Press, 2002.

[16] R. M. Carro, E. Pulido, and P. Rodrguez. Designing adaptive web-based courses with tangow. In *In proceedings of the 7th International Conference on Computers in Education, ICCE'99, Chiba, Japan, November 4 - 7, 1999. V. 2, pp.697-704*.

[17] P. Castells and J. A. Macas. An adaptive hypermedia presentation modeling system for custom knowledge representations. In *World Conference on the WWW and Internet (WebNet2001)*, 2001.

[18] C. C. Chen, M. C. Chen, and Y. S. Sun. PVA: a self-adaptive personal view agent system. In *Knowledge Discovery and Data Mining*, pages 257–262, 2001.

[19] F. Espinoza and K. Hk. A www interface to an adaptive hypermedia system. In *PAAM (Practical Applications of Agent Methodology), April 1996, London*.

[20] L. Francisco-Revilla and F. M. S. III. Adaptive medical information delivery combining user, task and situation models. In *Intelligent User Interfaces*, pages 94–97, 2000.

[21] S. Garlatti, S. Iksal, and P. Kervella. Adaptive on-line information system by means of a task model and spatial views. In *2nd Workshop on Adaptive Systems and User Modeling on the WWW*.

[22] J. E. Gilbert and C. Y. Han. Adapting instruction in search of a significant difference. *Journal of Network and Computer Applications (1999) 22*, 22, 1999.

[23] C. Hockemeyer and D. Albert. The adaptive tutoring system RATH. In M. E. Auer and U. Ressler, editors, *ICL99 Workshop Interactive Computer aided Learning: Tools and Applications*, Villach, Austria, 1999. Carinthia Tech Institute.

[24] H. Hohl, H.-D. Bcker, and R. Gunzenhuser. Hypadapter: An adaptive hypertext system for exploratory learning and programming. In *Spec. Iss. on Adaptive Hypertext and Hypermedia, User Modeling and User-Adapted Interaction 6 (2-3), 131-156.*, 1996.

[25] P. D. B. Hongjing Wu, Geert-Jan Houben. Aham: A dexter-based reference model for adaptive hypermedia. In *Proceedings od the 10th ACM Conference on Hypertext and Hypermedia, Darmstadt, Germany, 1999*, pages 147–156.

[26] G.-J. Houben. HERA: Automatically generating hypermedia front-ends. In *EFIS*, pages 81–88, 2000.

[27] T. Joachims, D. Freitag, and T. M. Mitchell. Web watcher: A tour guide for the world wide web. In *IJCAI (1)*, pages 770–777, 1997.

[28] T. Kahabka, M. Korkea-aho, and G. Specht. GRAS: An adaptive personalization scheme for hypermedia databases. In *HIM*, pages 279–292, 1997.

[29] M. Kilfoil, A. Ghorbani, W. Xing, Z. Lei, J. Lu, J. Zhang, and X. Xu. Toward an adaptive web: The state of the art and science. In *the 1st Annual Conference on Communication Networks and Services Research (CNSR 2003)*, pages 119–130, 2003.

[30] N. Koch and M. Wirsing. Software engineering for adaptive hypermedia applications? In *Third Workshop on Adaptive Hypertext and Hypermedia, Sonthofen, Germany, July 13-17, 2001*.

[31] W. Li, Q. Vu, E. Chang, D. Agrawal, Y. Hara, and H. Takano. Powerbookmarks:a system for personalizable web information organization, sharing, and management. In *In Proceedings of the Eighth International World-Wide Web Conference, May 1999.*

[32] J. Marzo, C. Pea, M. Aguilar, X. Palencia, A. Alemany, M. Valls, and A. Joh. Adaptive multiagent system for a web-based tutoring environment agentcities.net project ist-2000-28384 - final report.

[33] A. Metcalfe, M. Snitzer, and J. Austin. Virtual adaptive learning architecture. In *IEEE International Conference on Advanced Learning Technologies (ICALT'01)*, 2001.

[34] M. D. Micko. Demonstration of ilex 3.0.

[35] M. Milosavljevic and J. Oberlander. Dynamic hypertext catalogues: Helping users to help themselves. In *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia (HT'98)*, Pittsburgh, PA, USA, 20-24 1998.

[36] A. Mitrovic and K. Hausler. Porting sql-tutor to the web. In *ITS'2000 workshop on Adaptive and Intelligent Web-based Education Systems, pp. 37-44, 2000*.

[37] D. Mladenic. Personal webwatcher: Design and implementation, 1996.

[38] B. Mobasher, N. Jain, E. Han, and J. Srivastava. Web mining: Pattern discovery from world wide web transactions, 1996.

[39] M. Moundridou and M. Virvou. Wear: A web-based authoring tool for building intelligent tutoring systems.

[40] A. Negro, V. Scarano, and R. Simari. User adaptivity on www through cheops, 1998.

[41] W. Nejdl and M. Wolpers. Kbs hyperbook - a data-driven information system on the web.

[42] G. Neumann and J. Zirvas. Skill - a scalable internet-based teaching and learning system. In *Proceedings of WebNet 98, World Conference on WWW, Internet and Intranet AACE, Orlando, Fl, November 7-12 1998*.

[43] J. Oberlander, M. O'Donnell, C. Mellish, and A. Knott. Conversation in the museum: experiments in dynamic hypermedia with the intelligent labelling explorer. *The New Review of Hypermedia and Multimedia*, 4:11–32, 1998.

[44] O.C.SANTOS, E.GAUDIOSO, C.BARRERA, and J.G.BOTICARIO. Alfanet: An adaptive e-learning platform. In *2nd International Conference on Multimedia and ICTs in Education (m-ICTE2003), 2003*.

[45] J. Ohene-Djan, M. Gorle, C. P. Bailey, G. B. Wills, and H. C. Davis. Is it possible to devise a unifying model of adaptive hypermedia and is one necessary? In *AH2003: Proceedings of the Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems, Fourteenth Conference on Hypertext and Hypermedia, pages 167-183, Nottingham, UK, August 26 2003. ACM*.

[46] R. Pagesya, G. Soulaa, and M. Fieschia. Diadem: an adaptive hypermedia designed to improve access to relevant medical information.

[47] M. Perkowitz and O. Etzioni. Adaptive web sites: Automatically synthesizing web pages. In *AAAI/IAAI*, pages 727–732, 1998.

[48] E. S. Peter Brusilovsky and G. Weber. Elm-art: An intelligent tutoring system on world wide web. In *Intelligent Tutoring Systems. Lecture Notes in Computer Science,Springer Verlag, Berlin, Vol. 1086*, pages 261–269, 1996.

[49] J. E. Peter Brusilovsky and E. Schwarz. Web-based education for all: a tool for developing adaptive courseware. In *Computer Networks and ISDN Systems, Vol. 30, Nos. 1–7*, pages 291–300, 1998.

[50] V. Scarano, M. Barra, P. Maglio, and A. Negro. Group adaptive system project (gas). In *2nd International Conference on Adaptive Hypermedia and Adaptive Web Based System, Malaga, Spain*, 2002.

[51] M. Specht, G. Weber, S. Heitmeyer, and V. Schch. Ast: Adaptive www-courseware for statistics. In *Workshop Adaptive Systems and User Modeling on the World Wide Web, Sixth International Conference on User Modeling, Chia Laguna, Sardinia*, 1997.

[52] C. Staff. Hypercontext: Using context in adaptive hypertext. In *Proceedings of the International and Interdisciplinary Conference on Modeling and Using Context , 97, 243–255.*, 1997.

[53] C. G. Thomas and R. Oppermann. Supporting information consumers by search agents in the world-wide web.

[54] B. Trousse, M. Jaczynski, and R. Kanawati. Using user behavior similarity for recommandation computation : The broadway approach, 1999.

[55] G. Weber, H.-C. Kuhl, and S. Weibelzahl. Developing adaptive internet based courses with the authoring system NetCoach. In P. de Bra and P. Brusilovsky, editors, *Adaptive Hypermedia. Proceedings of the Third Workshop on Adaptive Hypermedia, AH2001, held at the Eighth International Conference on User Modeling, UM2001*, Berlin, 2001. Springer.

[56] C. Wolf. iweaver: towards 'learning style'-based e-learning in computer science education. In *Proceedings of the fifth Australasian conference on Computing education*, pages 273–279. Australian Computer Society, Inc., 2003.

[57] S. Yamada and F. Murase. Intelligent user interface for a web search engine by organizing page information agents. In *The International Conference on Intelligent User Interfaces (IUI-2002), pp.230-231, San Francisco, USA*, 2002.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 137-141, 2004.

Experiments in Web Page Classification for Semantic Web

Asad Satti, Nick Cercone, Vlado Kešelj

Faculty of Computer Science, Dalhousie University
E-mail: {rashid,nick,vlado}@cs.dal.ca

Abstract

We address the problem of web page classification within the framework of automatic annotation for Semantic Web. The performance of several classification algorithms is explored on the Four Universities dataset using page text and link information, with a limited-size feature set. Several well-known classification algorithms are evaluated on the task of web-page classification using the text of web pages. When compared to the methods that use link information, the text-based method shows surprisingly good performance, even with a feature set of limited size.

1 Introduction

The Semantic Web (SW) research area is concerned with transforming information available on the World Wide Web into knowledge by adding semantic structure to it. Embedded knowledge will help achieve efficient information retrieval, web-based question-answering, and intelligent agent-based applications. There are numerous and obvious benefits of having the web content enriched with semantic annotation; however, there are as many difficult challenges that need to be solved in order to pave the way for SW. One of them is defining conventions and ontologies for SW annotation, and another one is involved with the question of who will do the annotation: the web-page owner or an automatic tool. We advocate the use of automatic semantic annotation tools. There are several strong arguments why this is a preferred option: the first one is that the annotation requires significant and hardly justifiable effort on the part of the web-page owner so most of the owners will avoid it, and the second one is that an automatic annotation method will produce more uniform and consistent annotation.

Since the web encompasses many different domains it would be very difficult to try to annotate the whole web at once. We suggest breaking up the annotation of the entire web into the annotation of subsets representing different domains. After some evolution of the separate domain annotations, the problem of annotating the whole web can be tackled more easily. Because of this, the primary task in automatic annotation for SW is classification of web objects, typically pages, into domains. These domains are defined by *ontologies*. Operationally ontology can be described as the conceptual hierarchical organization/classification of concepts (classes). Since the web content and structure for a domain is composed by human beings under the implicit influence of their domain ontology e.g., yahoo directory, we believe that using domain ontology is the right direction. In any particular domain, concepts at the upper levels in the ontology hierarchy are more generic ones and many people agree on that with slightly varying terminology, but lower level concepts might be quite different. It is difficult to ensure that users annotate their web pages without bias.

To first step is to define the domain ontology using domain concepts (classes) and their hierarchical organization, their attributes, relations between the instances of two classes in the ontology, and inferences, if there are any. Semantic annotation of the web pages starts with the classification of web pages into ontology classes followed by extraction of attributes, and extraction of ontological relations between pages. For the World Wide Web and, in general, for any system with a large and growing number of entities classification becomes necessary for better understanding of the system.

2 Related Work

2.1 Classifiers

For our experiments we use the classifiers implemented in the WEKA package [8], which is an open source Java package containing various learning algorithms for classification, clustering, and association.

We apply the following classifiers: IBk, Naïve Bayes, J48, RIPPER, and PART. IBk is an implementation of the k-Nearest Neighbour (k-NN) method and finds the k most similar documents to the test document [1]. It then either assigns the same class to the test document that labels most of these k documents or class with maximum score after weighted count of classes. For our experiment the value of k is one. The Naïve Bayes method is based on the Bayes' rule [1]. J48 is a WEKA version of the well-known C4.5 decision tree algorithm. RIPPER (Repeated Incremental Pruning to Produce Error Reduction) [2] is a propositional rule learner which is an optimized version of IREP [6]. PART [4] is also a rule learner that combined RIPPER and C4.5 algorithms, with pruning confidence threshold being 0.25 and minimum number of instances per leaf is 2.

Rule learning systems often scale relatively poorly with the sample size [2]. RIPPER rule learner was an effort to achieve efficiency for large noisy datasets and competitive generalization performance [2]. The core method REP (reduced error pruning) used for pruning decision trees can also be used for rule set pruning. It partitions the training data into a growing set and pruning set. First, a rule set is formed that overfits the growing set, then rule set is pruned using a pruning operator, which produces the greatest reduction of error on the pruning set from a set of operators. Pruning is terminated when an increase in the error on the pruning set is observed. REP for rules usually improves generalization performance on noisy data but it is computationally expensive. IREP (incremental reduced error pruning) was proposed by [6]. In this method, instead of growing the whole rule set first and then pruning it, each rule is pruned right after it is generated until the point when accuracy of the rule starts decreasing on the pruning set. Then pruned rule is added to the rule set and all positive and negative examples are removed from the training set (i.e., growing and pruning sets). When accuracy of a pruned rule drops below that of the empty rule then that rule is ignored and learned rule set is returned. RIPPER is realized after making three modifications to IREP namely, rule value metric, stopping condition, and rule set optimization in order to closely approximate reduced error pruning [2].

PART combines RIPPER and C4.5, the two rule learning paradigms [4]. Both RIPPER and C4.5 perform global optimization on the set of initially induced rules. Former does so to increase accuracy and later does to reduce the large rule set size. PART induces accurate and compact rule sets without using global optimization. Like RIPPER, it uses a divide-and-conquer strategy for building a rule i.e., it removes instances covered by the rule and continues recursively to create the rules for the remaining instance until no more instances are left [4]. In order to create each rule, a pruned decision tree is built from the current set of instances, then the leaf with the largest coverage is added to the rule set, and the tree is discarded. Algorithm to build pruned decision tree can be found in [4]. The use of pruned trees avoids the over-pruning problem in RIPPER.

2.2 Web Data Mining

The Four Universities dataset [7] consists of 8,282 web pages manually classified in to several classes. The pages are harvested from four universities, and they are kept in separate directories [3]. It was hypothesized in [5] that structural and link information on the web can make the classification of hypertext pages easier and more reliable instead of using just the page text.

3 Method

Since the dataset consists of web pages from four universities, we trained the classifier on three of the universities and tested it on the fourth university subset. This way, we avoid bias in evaluation results due to idiosyncrasies of all pages within one university. The classification evaluation is performed using 4-fold cross-validation [8], i.e., each time the training is performed on a different set of three university sets and the testing is performed on the remaining university set. Additionally, the miscellaneous data set is always used only in training.

A feature set to represent web pages is collected from the web page text after removing the HTML tags. The supporting features, such as link structure analysis, text around in-links and out-links, are not used. The number of features is restricted to the top 100 features, which are selected after ranking them with the information gain feature selection measure.

We test performance of several well-established classification methods. Namely, Naïve Bayes, J48 (pruned), RIPPER, k-Nearest Neighbour (k-NN), and PART have been evaluated to see which one achieves the best performance for the web page classification task.

To observe the class performance, the following standard evaluation measures are used:

- True Positive Rate (TP Rate), or Recall,
- False Positive Rate (FP Rate),
- Precision, and
- F-Measure.

The measures are defined by the following formulas:

$$\begin{aligned}
\text{TP Rate} &= \frac{\text{Correctly classified positives}}{\text{Total positives}} \\
\text{FP Rate} &= \frac{\text{Incorrectly classified negatives}}{\text{Total negatives}} \\
\text{Precision} &= \frac{\text{Correctly classified positives}}{\text{Total predicted positives}} \\
\text{F-Measure} &= \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}
\end{aligned}$$

In addition to this, we use Percent of correctly classified and Percent of incorrectly classified instance to observe overall performance.

4 Evaluation Results

4.1 Dataset

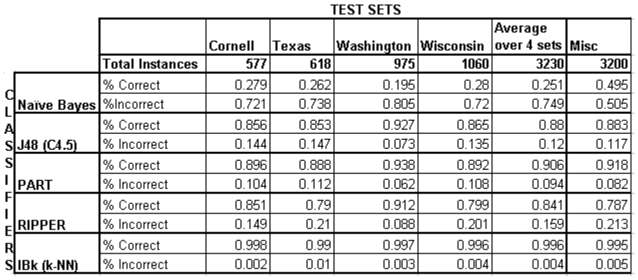
The Four Universities dataset [7] has 8,282 web pages manually classified into student (1641), faculty (1124), staff (137) , department (182), course (930), project (504), and other (3764). Each class of data set contains web pages from 4 universities Cornell (867), Texas (827), Washington (1205), Wisconsin (1263), and miscellaneous (i.e., various other universities) (4120). The web pages for each university and miscellaneous are kept separately [3].

4.2 Classification Experiments

The summary of our experimental results is shown in Figure 1. Rows in the figure correspond to classifiers and columns show the test sets (i.e., held-out university sets). Our objective is to be able to classify the web pages from any new computer science department website using the learned model from the data set. The Miscellaneous (Misc) set is included in the training set to capture the different idiosyncrasies from many universities to improve the prediction accuracy of learners. In the figure, columns corresponding to each university show the percentage of instances correctly/incorrectly classified using corresponding test sets during 4-fold cross-validation and then these results are averaged in the average column. An experiment was performed by training on the 4 universities training set and testing on the Misc set, the results are as good as or better as compared to 4-fold cross-validation This shows that better efficiency can be achieved even without using the Misc set.

These results show that the k-Nearest Neighbour (k-NN) classifier completely outperforms the other classifiers and the percentage of instances correctly classified by k-NN is the highest among others. As in [4], our results show that C4.5 and PART give comparable results followed by RIPPER and Naïve Bayes.

Figure 2 shows the confusion matrix and information retrieval performance of Naïve Bayes. These values are calculated by averaging over the intermediate information retrieval results and confusion matrices, generated during the 4-fold cross-validation. It is interesting to note that one of the larger classes namely 'other' achieved more than 80% precision and another larger class 'student' achieved more than 80% recall. Since most of the instances of 'other' class are incorrectly classified into other classes, this accounts for the lower precision of the other classes. Many of instances of the other classes are incorrectly classified into the 'student,' this caused the lower recall of the other classes.



TEST SETS

			Cornell	Texas	Washington	Wisconsin	Average over 4 sets	Misc
		Total Instances	577	618	975	1060	3230	3200
CLASSIFIERS	Naïve Bayes	% Correct	0.279	0.262	0.195	0.28	0.251	0.495
		%Incorrect	0.721	0.738	0.805	0.72	0.749	0.505
	J48 (C4.5)	% Correct	0.856	0.853	0.927	0.865	0.88	0.883
		% Incorrect	0.144	0.147	0.073	0.135	0.12	0.117
	PART	% Correct	0.896	0.888	0.938	0.892	0.906	0.918
		% Incorrect	0.104	0.112	0.062	0.108	0.094	0.082
	RIPPER	% Correct	0.851	0.79	0.912	0.799	0.841	0.787
		% Incorrect	0.149	0.21	0.088	0.201	0.159	0.213
	IBk (k-NN)	% Correct	0.998	0.99	0.997	0.996	0.996	0.995
		% Incorrect	0.002	0.01	0.003	0.004	0.004	0.005

Figure 1. Percent correctly / incorrectly classified instances by different classifiers

Figure 3 shows the results of C4.5 classifier. Lower precision of class 'department' is due to the class 'other.' Lower precision and recall of the 'staff' class is due to mix-up of its instance with 'student' class.

For the PART rule learner, shown in Figure 4, lower recall of class 'staff' is caused by its mix-up with the 'faculty,' 'student,' and 'other' class. RIPPER classifier also makes similar type of mistakes, as shown in Figure 5. Finally, k-NN results are shown in figure 6 and it gives very promising results.

These results show that on a scale of a web site, such as a university web site, we can rely on some classical, text-based methods for classification for purposes of SW annotation. In particular, the kNN algorithm demonstrates a very high performance.

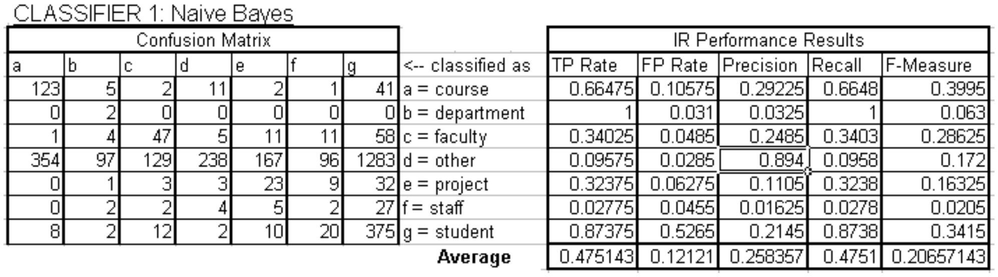
5 Conclusion and Future Work

Web page classification is one of the core elements of our SW annotation system which is based on the automatic extraction of conceptual knowledge from web contents based on the given ontology. We also evaluate classifiers by automatically extracting significant terms from the web pages and using them as a feature vector. Results show that k-NN gives very good classification accuracy.

Our goal is to build ontology-driven information extraction methods based on the combination of pattern matching, machine learning, and natural language processing techniques. We will try to design an adaptive information extraction module which could adapt to the changing nature of the web. Finally annotation of the web will be done with the extracted knowledge, followed by an evaluation.

References

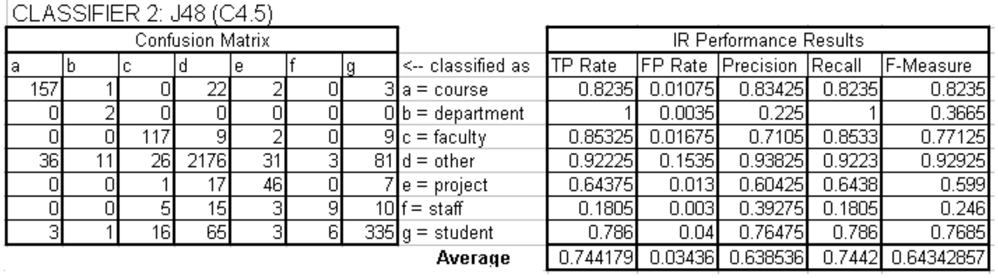
[1] S. Chakrabarti. *Mining the Web Discovering Knowledge from Hypertext Data*. Morgan-Kaufmann Publishers, 2003.



CLASSIFIER 1: Naive Bayes

Confusion Matrix								IR Performance Results				
a	b	c	d	e	f	g	<-- classified as	TP Rate	FP Rate	Precision	Recall	F-Measure
123	5	2	11	2	1	41	a = course	0.66475	0.10575	0.29225	0.6648	0.3995
0	2	0	0	0	0	0	b = department	1	0.031	0.0325	1	0.063
1	4	47	5	11	11	58	c = faculty	0.34025	0.0485	0.2485	0.3403	0.28625
354	97	129	238	167	96	1283	d = other	0.09575	0.0285	0.894	0.0958	0.172
0	1	3	3	23	9	32	e = project	0.32375	0.06275	0.1105	0.3238	0.16325
0	2	2	4	5	2	27	f = staff	0.02775	0.0455	0.01625	0.0278	0.0205
8	2	12	2	10	20	375	g = student	0.87375	0.5265	0.2145	0.8738	0.3415
							Average	0.475143	0.12121	0.258357	0.4751	0.20657143

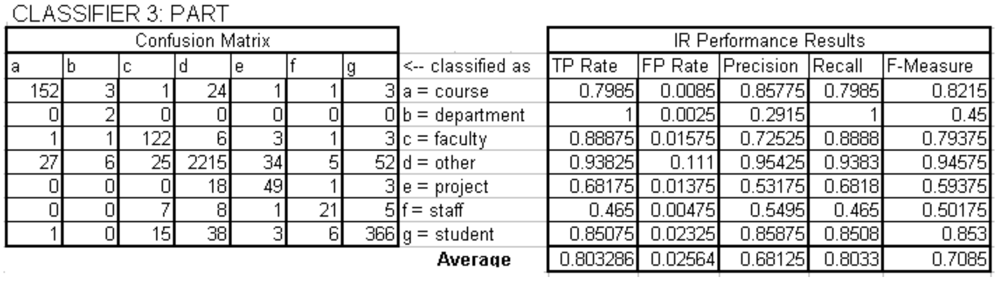
Figure 2. True Positives Rate (TP Rate), False Positives Rate (FP Rate), Precision, and Recall of each class by using Naïve Bayes Classifier



CLASSIFIER 2: J48 (C4.5)

Confusion Matrix								IR Performance Results				
a	b	c	d	e	f	g	<-- classified as	TP Rate	FP Rate	Precision	Recall	F-Measure
157	1	0	22	2	0	3	a = course	0.8235	0.01075	0.83425	0.8235	0.8235
0	2	0	0	0	0	0	b = department	1	0.0035	0.225	1	0.3665
0	0	117	9	2	0	9	c = faculty	0.85325	0.01675	0.7105	0.8533	0.77125
36	11	26	2176	31	3	81	d = other	0.92225	0.1535	0.93825	0.9223	0.92925
0	0	1	17	46	0	7	e = project	0.64375	0.013	0.60425	0.6438	0.599
0	0	5	15	3	9	10	f = staff	0.1805	0.003	0.39275	0.1805	0.246
3	1	16	65	3	6	335	g = student	0.786	0.04	0.76475	0.786	0.7685
							Average	0.744179	0.03436	0.638536	0.7442	0.64342857

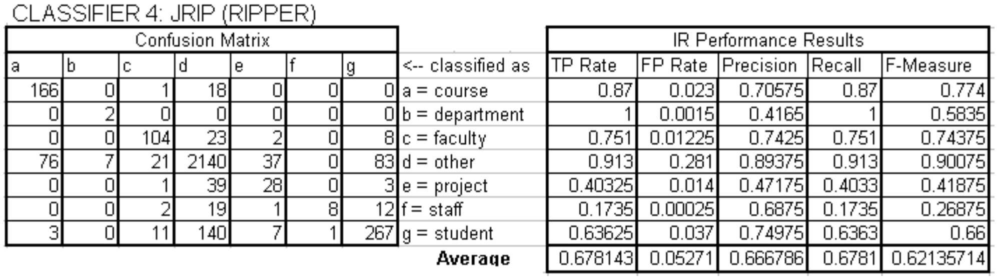
Figure 3. True Positives Rate (TP Rate), False Positives Rate (FP Rate), Precision, and Recall of each class by using C4.5 Classifier



CLASSIFIER 3: PART

Confusion Matrix								IR Performance Results				
a	b	c	d	e	f	g	<-- classified as	TP Rate	FP Rate	Precision	Recall	F-Measure
152	3	1	24	1	1	3	a = course	0.7985	0.0085	0.85775	0.7985	0.8215
0	2	0	0	0	0	0	b = department	1	0.0025	0.2915	1	0.45
1	1	122	6	3	1	3	c = faculty	0.88875	0.01575	0.72525	0.8888	0.79375
27	6	25	2215	34	5	52	d = other	0.93825	0.111	0.95425	0.9383	0.94575
0	0	0	18	49	1	3	e = project	0.68175	0.01375	0.53175	0.6818	0.59375
0	0	7	8	1	21	5	f = staff	0.465	0.00475	0.5495	0.465	0.50175
1	0	15	38	3	6	366	g = student	0.85075	0.02325	0.85875	0.8508	0.853
							Average	0.803286	0.02564	0.68125	0.8033	0.7085

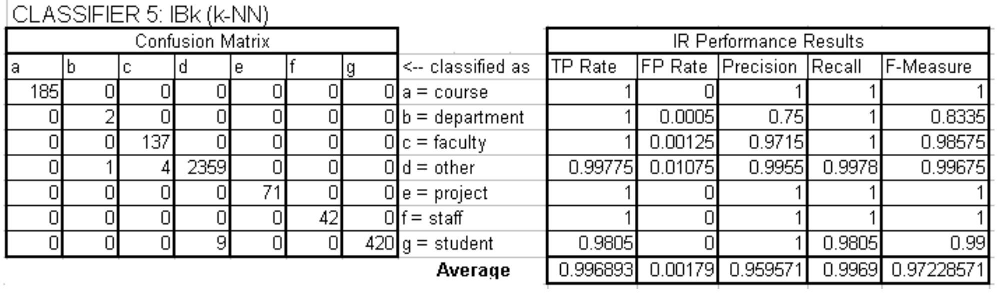
Figure 4. True Positives Rate (TP Rate), False Positives Rate (FP Rate), Precision, and Recall of each class by using PART Classifier



CLASSIFIER 4: JRIP (RIPPER)

Confusion Matrix								IR Performance Results				
a	b	c	d	e	f	g	<-- classified as	TP Rate	FP Rate	Precision	Recall	F-Measure
166	0	1	18	0	0	0	a = course	0.87	0.023	0.70575	0.87	0.774
0	2	0	0	0	0	0	b = department	1	0.0015	0.4165	1	0.5835
0	0	104	23	2	0	8	c = faculty	0.751	0.01225	0.7425	0.751	0.74375
76	7	21	2140	37	0	83	d = other	0.913	0.281	0.89375	0.913	0.90075
0	0	1	39	28	0	3	e = project	0.40325	0.014	0.47175	0.4033	0.41875
0	0	2	19	1	8	12	f = staff	0.1735	0.00025	0.6875	0.1735	0.26875
3	0	11	140	7	1	267	g = student	0.63625	0.037	0.74975	0.6363	0.66
							Average	0.678143	0.05271	0.666786	0.6781	0.62135714

Figure 5. True Positives Rate (TP Rate), False Positives Rate (FP Rate), Precision, and Recall of each class by using RIPPER Classifier



CLASSIFIER 5: IBk (k-NN)

Confusion Matrix								IR Performance Results				
a	b	c	d	e	f	g	<-- classified as	TP Rate	FP Rate	Precision	Recall	F-Measure
185	0	0	0	0	0	0	a = course	1	0	1	1	1
0	2	0	0	0	0	0	b = department	1	0.0005	0.75	1	0.8335
0	0	137	0	0	0	0	c = faculty	1	0.00125	0.9715	1	0.98575
0	1	4	2359	0	0	0	d = other	0.99775	0.01075	0.9955	0.9978	0.99675
0	0	0	0	71	0	0	e = project	1	0	1	1	1
0	0	0	0	0	42	0	f = staff	1	0	1	1	1
0	0	0	9	0	0	420	g = student	0.9805	0	1	0.9805	0.99
							Average	0.996893	0.00179	0.959571	0.9969	0.97228571

Figure 6. True Positives Rate (TP Rate), False Positives Rate (FP Rate), Precision, and Recall of each class by using k-NN Classifier

[2] W. Cohen. Fast effective rule induction. In *Machine Learning: Proceedings of the Twelfth International Conference (ML95)*, 1995.

[3] M. DiPasquo, D. Freitag, D. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to construct knowledge base from the world wide web. *Artificial Intelligence*, 118:69–113, June 2000.

[4] E. Frank and I. Witten. Generating accurate rule sets without global optimization. In *Machine Learning: Proceedings of the Fifteenth International Conference*, San Francisco, CA, 1998. Morgan Kaufmann Publishers.

[5] J. Furnkranz. Using links for classifying web-pages. Technical Report OEFAI-TR-98-29, Austrian Research Institute for Artificial Intelligence, Wien, Austria, 1998.

[6] J. Furnkranz and G. Widmer. Incremental reduced error pruning. In *Machine Learning: Proceedings of the Eleventh Annual Conference*. Morgan Kaufmann Publishers, 1994.

[7] A. McCallum. The 4 universities data set. Retrieved Dec 3, 2003, *http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/*.

[8] I. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 1999.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 142-148, 2004.

MICE-*f*: Financial Reviews Analysis using Category Model

Saravadee Sae Tan[1], Gan Keng Hoon[1], Tang Enya Kong[1], Chan Huah Yong[2]
[1]*Computer Aided Translation Unit (UTMK)* [2]*Grid Computing Lab*
School of Computer Sciences
Universiti Sains Malaysia
11800 Penang Malaysia
{saratan|khgan|enyakong|hychan}@cs.usm.my

Abstract

In the domain of financial, financial news, articles, reports about financial reviews are helpful and important information which give investors or financial analyst an indication to help decision making in financial matters. However, due to the volume of this information and the diversity of financial topics, it is difficult for a human to track and interpret each of them in a consistent manner.

Based on this motivation, we propose that this information can be classified into categories, in which the categories are based on a particular objective and goal as required by a user. In each category, the financial information is further classified based on a status indicator, to reflect the positive, neutral or negative status in term of financial outlook stated in the information. Thus, user can directly focus on interested financial topic, and get an indication about the financial outlook of that topic.

For classification purpose, we propose to combine linguistic technique and statistical technique to select features to represent the categories and status indicators.

In this work, we present the architecture of MICE-f that will crawl financial information from multiple financial sources and classify them based on the defined categories and status indicators.

1. Introduction

In the domain of financial, large quantities of articles, news, reports about financial reviews are generated daily. With the technology of Internet, this information can be easily retrieved using online sources such as Reuters, Bloomberg, CNN Financial Network etc. This information contains a wealth of financial knowledge that can be used to help decision making in financial matters. Human interpretation (e.g. financial analyst) is needed to analyze and transform the textual information into useful knowledge, in order to get a summary or conclusion about the positive or negative status of financial outlook reflected by the financial reviews. However, due to the tremendous amounts of information, it is impossible for a user to go through every single piece of information from various sources every day.

Although financial information had been classified by various financial sources into categories such as economy, politic, market etc, we have different users with different needs and objectives when accessing this information. Therefore, we would like to orientate the retrieval and classification of financial information based on the interest of each user.

In this paper, we propose a classification method that is able to classify the financial information based on categories defined by user. Intuitively speaking, the user has the flexibility to define the concepts in which each category represents. For this reason, he may foresee the information or content being classified in each category thus can directly focus on information he is seeking. In each category, the information is further classified based on three status indicators i.e. positive, neutral and negative to reflect the financial outlook status of the information.

As an extension to our previous work in text classification for general web information [6], we propose a system, MICE-*f* that is orientated to the financial domain. In MICE-*f*, we

i) Justify our Category Model (CM) to represent categories related to financial domain. The Category Model consists of two levels. The first level is to organize financial information into pre-defined categories. The second level is to classify the financial information based on our status indicators, whether a piece of financial information reflect positive, neutral or negative financial outlook with respect to the category.
ii) Enhance our feature selection technique [7] by including linguistic analysis. Due to differences of financial text (more precise and specific) compared to general web text, we have identified two types of features to be selected for our classification purpose,

i.e. *concept features* which represent the categories and *descriptive features* which represent status indicators.

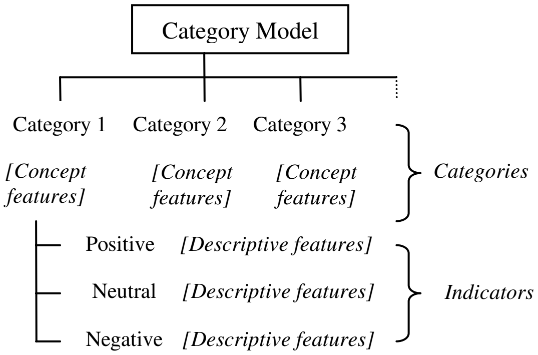


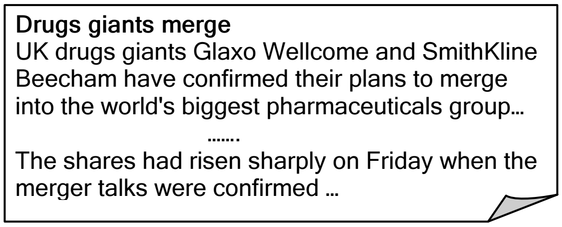
Figure 1: The category model for MICE-*f*.

In MICE-*f*, we adapt the Category Model in Figure 1 as the basis to classify financial information. MICE-*f* allows crawling of desired financial information such as news, reports, commentaries, articles from identified financial sources, classifying them based on user-defined categories as well as the status indicators. The overall idea is to assist users to effectively focus and attend to financial information based on categories relevant to their needs rather than navigating through a pool of overwhelming financial information.

2. The Methodology

2.1 The Nature of Financial Text

Financial texts (e.g. financial news, financial reports, commentaries etc) are usually more compact and straight forward compared to other natural language text like stories, web pages, electronic mail etc. Most financial texts have specific and objective contents. We can easily recognize a particular event (e.g. company takeover, company merge, management succession, election etc) from the text as well as infer whether the information reflects positive or negative status in term of financial outlook [10]. For example,



Drugs giants merge
UK drugs giants Glaxo Wellcome and SmithKline Beecham have confirmed their plans to merge into the world's biggest pharmaceuticals group…

…….

The shares had risen sharply on Friday when the merger talks were confirmed …

From the financial text, we can easily recognize that the news is about company merge and can infer that this news reflect positive status of financial outlook.

Based on the nature of financial text, we propose a classification model which allow user to directly focus on a particular financial topic, such as "company merge", and followed by indication on whether the text is good or bad.

2.2 The Concept of Category Model

Our Category Model has two levels (as in Figure 2). The first level consists of a set of categories defined by user. The second level consists of three indicators which are positive, neutral and negative.

For example, at the first level, user can define a category called 'Company Activities' that represent all events related to company changes, such as company takeover, company merge, management succession, recruitment, and etc. In a more specific scenario, user can define categories "Election", "Governance Transparency" that reflect "Political" issues in a country. The categories should help to focus on certain financial analysis objective.

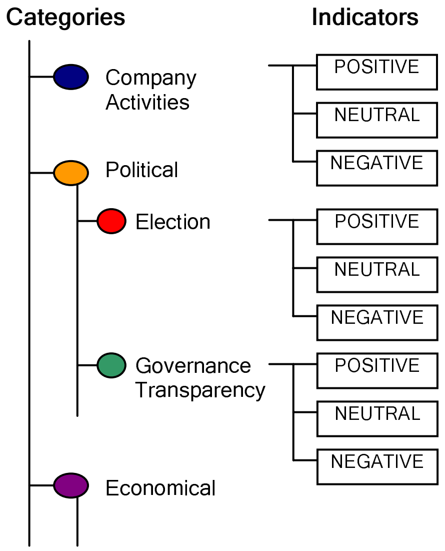


Figure 2: An example set of categories.

At the second level, there are three status indicators reflecting the financial outlook of the information for the defined category. Considering the nature of financial information, the following three status indicators are considered to be pertinent:

i) Positive – information which show good evidence of financial outlook.
ii) Neutral – information which did not mention anything about financial well-being or the influence to financial outlook is unclear.

iii) Negative information which show bad evidence of financial outlook.

In the Category Model, the concept of a category or a status indicator is reflected by a set of characteristic keywords (also known as features). In this work, we identified two types of features to be selected for our model.

i) *concept feature* that can express the concepts and contents of a category.
ii) *descriptive feature* to reflect positive, negative or neutral financial outlook of the information in a status indicator.

2.3 Training the Category Model

The selection of features into the Category Model is the crucial part in our work, as the features selected directly represent the meaning and concept of the defined categories and their status indicators. For each *category-indicator* pair, a number of corresponding financial texts have to be prepared for training purpose (refer to Figure 3). These financial texts should reasonably reflect the concept of the category and financial outlook status they belong.

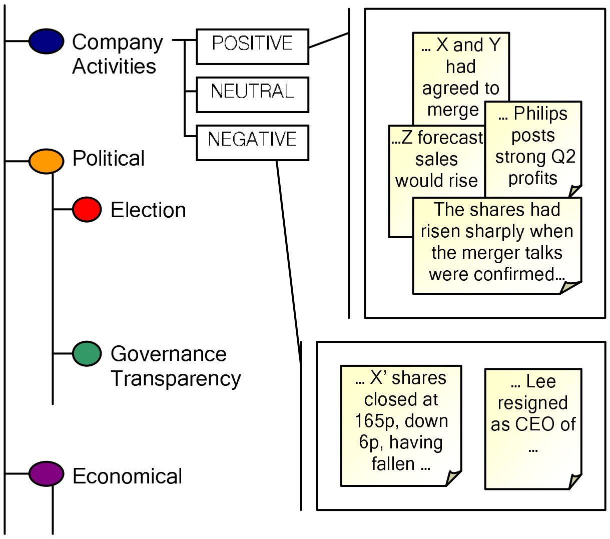

Company Activities
POSITIVE
NEUTRAL
NEGATIVE
Political
Election
Governance Transparency
Economical
… X and Y had agreed to merge
… Philips posts strong Q2 profits
…Z forecast sales would rise
The shares had risen sharply when the merger talks were confirmed…
… X' shares closed at 165p, down 6p, having fallen …
… Lee resigned as CEO of …

Figure 3: An example financial training texts for Company Activities .

Two main tasks in training a Category Model are:

i) Identify and generate candidate features from the training texts. Candidate features are potential keywords to be selected as *concept features* and *descriptive features*.
ii) From the candidate features, select an optimum set of keywords as *concept features* and *descriptive features*.

2.3.1 Text Parsing

In this paper, we propose to use Partial Parsing technique to analyze a financial text in order to extract relevant information to be considered as candidate features. Information such as subject of a sentence, object of a sentence, noun phrase and etc can be easily identified by analyzing the syntactic structure of a sentence.

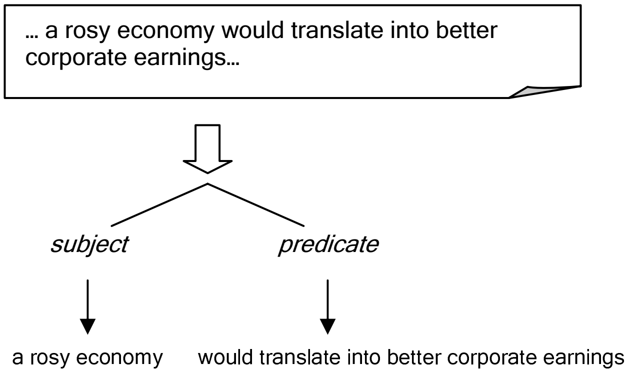
Partial Parsing is a linguistic technique to analyze syntactic structure of natural language texts. Partial Parsing perform partial analysis of the syntactic structures in a text. There are several possible levels of partial parsing: from identification of base noun phrases, to identification of chunks and to identification of clauses [2] [5] [9].

Clause Identification is a method in Partial Parsing to identify clauses in a text. A clause is a sequence of words in a sentence that contains a subject and a predicate [2]. Since the financial text selected for training purpose reflect the concepts of its category, we can make *assumptions* that:

i) The subject of a clause in a financial text is assumed to be related to the concepts or contents of the category the financial text belongs. Thus, the subject can be further analyzed to extract relevant word/phrase to be considered as a candidate for our *concept features*.
ii) The predicate of a clause may describe the action of the subject or may contain information about the influence of an event mention in the subject. This action or influence may have a direct relation with the financial outlook of the event. Thus it can be further analyzed to extract relevant word/phrase to be considered as a candidate for our *descriptive features*.

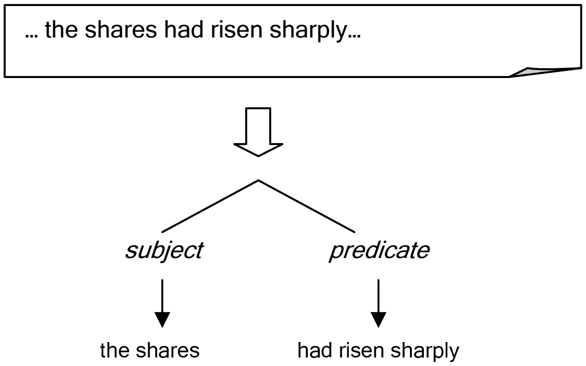
Here are examples of clauses obtained from financial news.

Example 1:


… a rosy economy would translate into better corporate earnings…
subject
predicate
a rosy economy
would translate into better corporate earnings

In this example clause, the event is *rosy economy* and this event has caused a *better corporate earnings*. Thus, *'rosy economy'* can be considered as a candidate for *concept feature* and *'better corporate earnings'* can be considered as a candidate for *descriptive feature*.

Example 2



In this clause, the subject *'shares'* can be considered as a candidate for *concept feature* and the behavior of the subject, *'risen sharply'* can be considered as a candidate for *descriptive feature*.

The methodology of Text Parsing is not finalized and further research will be carried out on it.

2.3.2 Feature Selection

In our methodology, Feature Selection technique is applied to select the *concept features* and *descriptive features* from the candidate features in order to compose the Category Model.

Feature Selection is a process that chooses a subset of features from the original set of features so that the dimensionality of feature space is optimally reduced according to a certain criterion. This tends to produce classification models that are simpler, clearer and computationally less expensive [4].

Various approaches of feature selection have been developed for dimensionality reduction in a classification task. Basically, these methods can be broadly divided into 2 main approaches, (i) Feature Selection in Machine Learning, and (ii) Feature Selection in Text Learning. Feature Selection in Machine Learning traverse a feature space and evaluate every candidate feature subset in order to find the best subset. These methods are less practical when the number of features is large. On the other hand, Feature Selection in Text Learning evaluates every feature independently, in which a scoring criterion is used to measure the goodness of a feature. All features are sorted in a list and a predefined number of best features are selected. However, the number of features to be selected is a main experimental issue in these methods [4][8].

Feature selection approach adopted in this paper is from the author's previous work. It combines the idea from both methods of feature selection in machine learning and text learning [7] [8]. All features are sorted in a list using a feature weighting function. An optimum set of features is selected by finding a cut-off point in the list using a consistency measure.

Feature Weighting

Statistical information of a feature, i.e. *frequency distribution of the feature across categories,* is used to indicate the importance of a feature in term of the discriminating power between categories. This comes from two major concerns. A feature is considered as representative if it appears many times *within* a text. On the other hand, a feature is regarded as not informative if it appears too many times *among* texts [1]. In our weighting function, these two aspects are taken as the basis in weighting a feature [7]:

i) ***Feature Frequency*** of a feature denotes the frequency occurrences of the feature in a category. The rational behind is that the ability of a feature in discriminating categories depends on how frequent it occurs in a category as against the other categories

ii) ***Document Frequency*** of a feature denotes the number of documents/texts in a category in which the feature occurs at least once. The main idea is that features that occur in more documents in a category against other categories are more discriminative than features that occur in many documents in many categories.

Every candidate feature is assigned a score using the Feature Weighting function. The score can reflect the significance of a feature in term of the discriminating power between categories. All candidate features are sorted from the most significant to the least significant, and top-*N* features are selected to compose the Category Model.

Consistency Measure

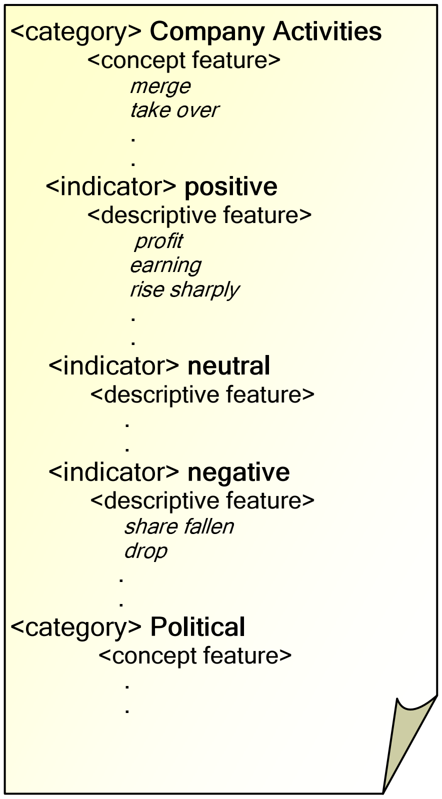
The size of Category Model is a main concern in the processing speed of our classification algorithm. Thus, it is important to control the set of features selected, (the value of *N*) in order to compose an optimum Category Model. We expect that the selected set of features are informative enough to represent the concepts of the categories, neither too few to miss the semantics or too many to burden the processing speed.

In our approach, a selected *feature subset* is evaluated by *class separability* measure. The feature subset is considered 'optimal' when it maximizes the class separability within a corpus (a collection of training texts). Consistency measure is a conservative way of achieving class separability. It does not attempt to maximize the class separability but tries to retain the power of class separability defined by the original set of features. The idea is to find the smallest set of features that can distinguish the user defined categories as well as the full set of the candidate features [3].

2.3.3 Category Model

Concept feature and *descriptive feature* selected by Text Parsing and Feature Selection are represented in our Category Model. Every category has a set of *concept feature* to differentiate it from other categories. Similarly, every status indicator of the category is represented by a set of *descriptive feature*.

A simple and frequently used representation is the feature vector representation. In our representation, each category or status indicator is characterized by a Boolean vector. All vectors are embedded in a *feature space* where each dimension corresponds to a feature (*concept feature* or *descriptive feature*). In a Boolean vector, each feature has a Boolean value that indicates whether the feature appears or not. The Category Model representation is visualized in Figure 4.



<category> **Company Activities**
    <concept feature>
        merge
        take over
        .
        .
  <indicator> **positive**
    <descriptive feature>
        profit
        earning
        rise sharply
        .
        .
  <indicator> **neutral**
    <descriptive feature>
        .
        .
  <indicator> **negative**
    <descriptive feature>
        share fallen
        drop
        .
        .
<category> **Political**
    <concept feature>
        .
        .

Figure 4: An example of Category Model

3. The Architecture of MICE-*f*

The architectural design of MICE-*f* consists of three major components, i.e. Category Model Generator, Information Crawler and Information Classifier. Upon receiving a request from user, MICE-*f* submits the query to user specified information sources, retrieves the financial information, and processes the information by classifying them into appropriate categories as well as indicating the financial outlook status of the information.

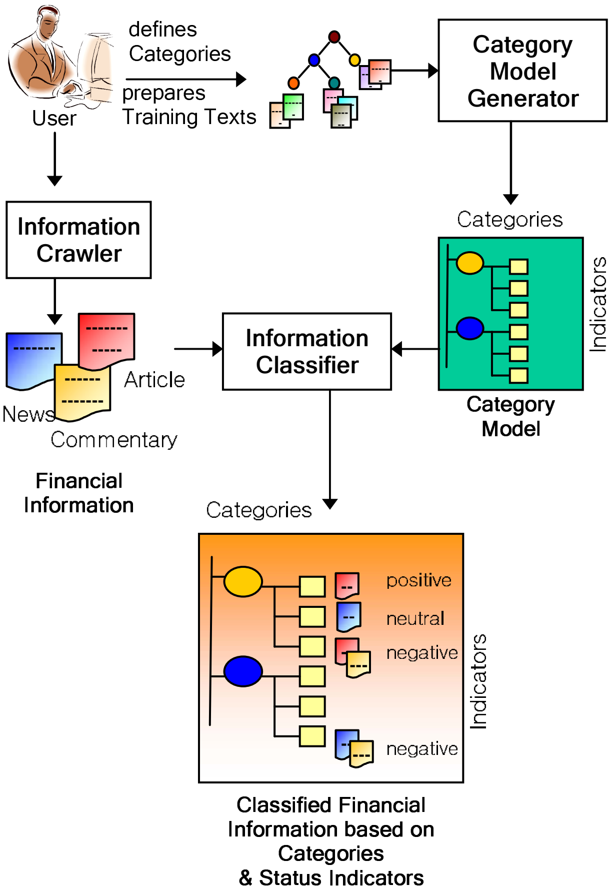


Figure 5: The Architecture of MICE-*f*.

.

3.1 Category Model Generator

The role of Category Model Generator is to learn the concept and characteristics of categories defined by user, and represent them in a Category Model. First, the user has to define a set of categories. The categories should be "well-separated" so that their intersection and overlapping is minimized. Each defined category will be associated to three types of status indicators, i.e. positive, neutral and negative. For each *category-indicator* pair, a set of financial training texts need to be prepared. The Text Parser (refer to section 2.3.1) will use linguistic technique to analyze the syntactic structure of sentences in the financial texts and extract relevant word or phrase to be considered as candidates of *concept features* and *descriptive features*. From these candidate features, the Feature Selector (refer to section 2.3.2) then uses statistic technique to measure the significance of each candidate feature in term of discriminating power between categories. Finally an optimal set of *concept features* and *descriptive features* are selected to compose the Category Model.

The detail process flow of Category Model Generator is shown in Figure 6.

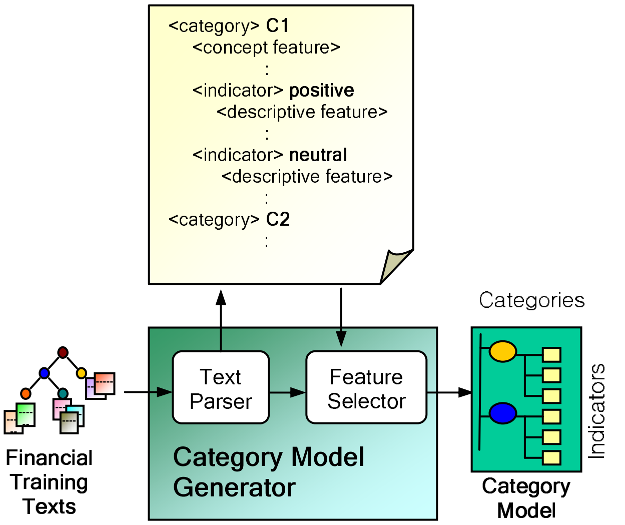


Figure 6: The process of Category Model Generator.

3.2 Information Crawler

There are endless sources for financial information on the World Wide Web. Common sources for financial information are like financial news portal (Google Business News, Yahoo Financial News), company's web sites, and news sites (Reuters, CNN Financial Network, theStar Business, Bloomberg). In the Information Crawler, we can crawl and extract required financial information from multiple sites simultaneously. As the financial information needed by users are varies, this component allows user to specify their required financial sites, company sites or news sites.

When requested by user, the Query Formulator will formulate queries based on the format of the selected financial sources. The Query Dispatcher then sends the queries to these sources simultaneously. From the raw financial texts gathered from these financial sources, Information Extractor will then process and extract useful financial information from the raw financial texts. The process flow for Information Crawler is shown in Figure 7.

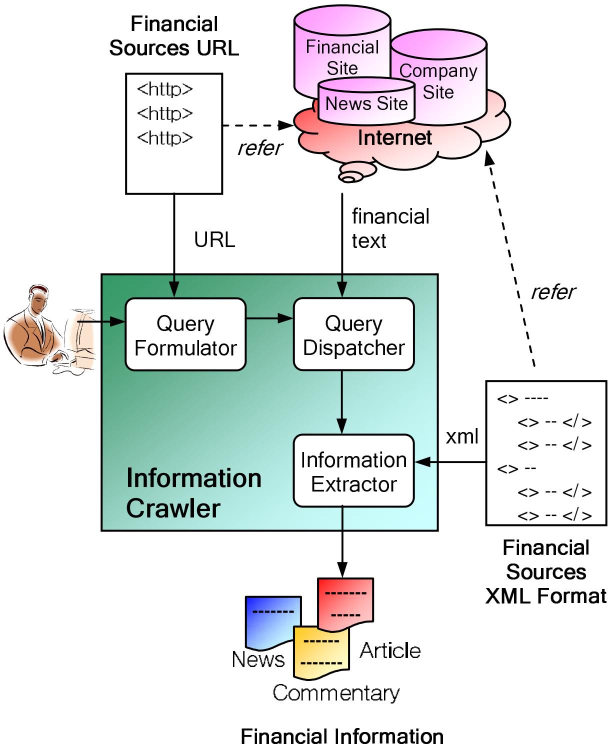


Figure 7: The process of Information Crawler.

3.3 Information Classifier

For each piece of financial information, Information Classifier will analyze its content in order to classify the information into appropriate categories and also infer the status of financial outlook stated in the content.

Text Parser (refer to section 2.3.1) will analyze the syntactic structure of the content and retrieve *concept features* and *descriptive features* found in the content. The *concept features* and *descriptive features* are represented by a vector model with respect to the Category Model vector space (refer to section 2.3.3).

The Similarity Calculator first compares the vector representations for concept features between the financial information and each category in the Category Model. The financial information is assigned to the most similar category. Next, the Similarity Calculator will measure the similarity of the vector representation for descriptive features between the financial information and each status indicator. An appropriate financial status is assigned to the financial information.

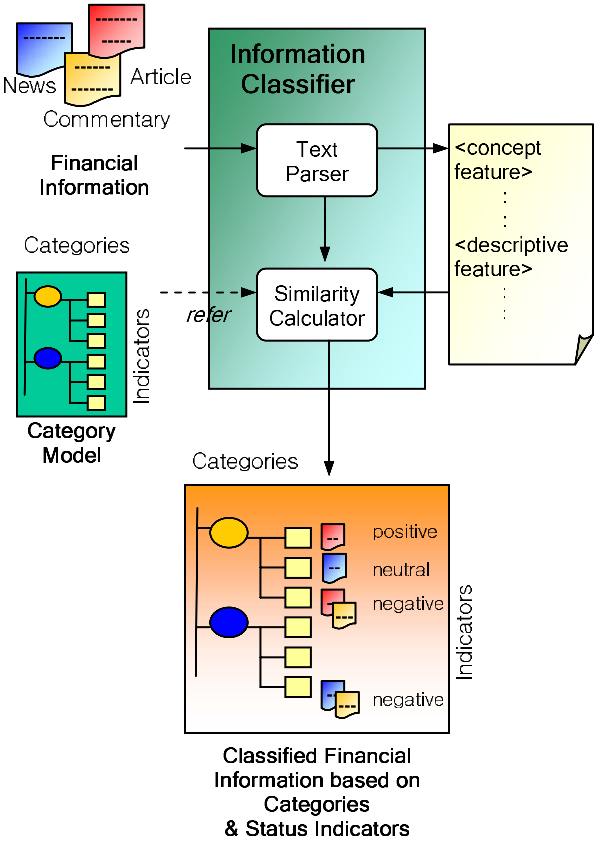


Figure 8: The process of Information Classifier.

4. Conclusion

The increasing number of financial information on the Internet demands a personalized and specialized service to effectively gather and manage the information. We propose an architecture MICE-*f* to crawl financial information from various sources, classify them based on user defined categories and infer the financial outlook status reflect by the information.

Our classification approach combines linguistic technique and statistical technique to select features to represent categories and status indicators. The linguistic technique is still an on-going research and we plan to look into other techniques of parsing in order to extract more specific and accurate information from a text to be considered as *concept feature* and *descriptive feature*.

5. References

[1] C. Liu, "A Survey: Automatic Text Categorization". *CS412 Report*, University of Illinois at Urbana-Champaign, 2004.

[2] E.F. Tjong Kim Sang and H. Dejean, "Introduction to the CoNLL-2001 shared task: Clause identification", In W. Daelemans, and R.Zajac, editors, *Proceedings of CoNLL-2001*, Toulouse, France, 2001, pp53-37.

[3] H. Liu, H. Dash and H. Motoda, "Consistency Based Feature Selection", *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2000),* 2000*,* pp98-109

[4] M. Sahami and D. Koller, "Toward Optimal Feature Selection", *Proceedings of the 13th International Conference on Machine Learning (ICML 1996)*, San Fransicisco CA, Morgan Kaufmann, pp284-292.

[5] S.P. Abney, "Parsing by chunks", In R.C. Berwick, S.P. Abney, and C. Tenny, editors, *Principle-based parsing: Computation and Psycholinguistics*, Kluwer, Dordrecht, 1991, pp257-278.

[6] S.S. Tan, K.H. Gan, E.K. Tang, S.L. Cheong, S.L. Chan and W.Y. Foo "MICE: Aggregating and Classifying Meta Search Results into Self-Customized Categories", *Proceedings of Web Intelligence (WI 2004)*, Beijing, 2004, accepted as demo-track paper.

[7] S.S. Tan, K.H. Gan, H.Y. Chan, E.K. Tang and S.L. Cheong, "Mapping Search Results into Self-Customized Category Model", *Proceedings of International Conference on Intelligent Information Processing (ICIIP 2004),* Kluwer Academic Publisher, Beijing, October 2004, accepted to appear.

[8] S.S. Tan, "*Topic Hierarchy Annotation using Feature Selection Technique*", MSc Thesis, School of Computer Sciences, Universiti Sains Malaysia, 2002.

[9] X. Carreras, L. Màrquez, V. Punyakanok and D. Roth, "Learning and Inference for Clause Identification", *Proceedings of the 13th European Conference on Machine Learning (ECML 2002)*, Helsinki, Finland, 2002, pp35-47.

[10] Y. Seo, J.A. Giampapa and K. Sycara, *"Text Classification for Intelligent Portfolio Management"*, Technical Report, CMU-RI-TR-02-14, Robotics Institute, Carnegie Mellon University, May 2002.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 149-154, 2004.

Design of the Merchant Reputation System: A Web-based Purchase Decision Support System

Ming Wang, California State University, Los Angeles
ming.wang@calstatela.edu

ABSTRACT

The focus of the paper is to investigate the design of current merchant reputation systems on shopping agent websites. The study explores their roles and impacts in customer decision-making process, identifies their rating methodology and rating reliability. The findings will help industrial practitioners for their design of merchant reputation systems, and will benefit online shoppers and e-merchants for their utilization of merchant reputation systems.

1. Introduction

A merchant reputation system works as a recommendation system on a shopping agent website. The system provides customers with the opportunity to compare electronic service quality (E-SQ) and to choose a merchant they feel comfortable shopping with. A recent Internet search shows that more and more merchant reputation systems have become available on shopping agent websites. BestWebBuys.com, BizRate.com, NexTag.com, PriceGrabber.com and Shopping.com are five very popular merchant reputation systems involved with a large number of online customers and merchants.

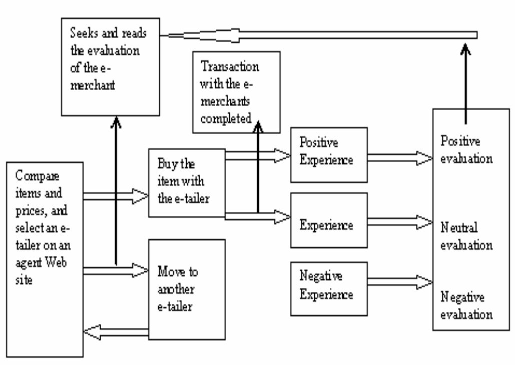
Despite its undeniable importance and widespread adoption, very little research has been published on the standardization of merchant reputation systems so far. With the increasing number of merchant reputation systems coming upon shopping agent websites, different designs of different reputation systems may generate rating discrepancy for the same merchants which might mislead the online shoppers. Customers may find ratings on the same merchant not consistent across different merchant reputation systems. Based on their research, Wang and Christopher [15] indicated that customer ratings on the same set of merchants are neither consistent within the same reputation system nor across different reputation systems. Rating discrepancy between different reputation systems may attribute to different evaluation approaches and different categorized rating criteria utilized in different reputation systems. To meet the challenge, the study of design standards of merchant reputation systems needs to be conducted.

This paper aims to investigate the current merchant reputation systems and to identify the rating methodology and design standard. The intent is to improve the design of reputation systems on agent websites and rating reliability, thereby helping customers to find their most appropriate merchants and reducing the risks inherent in interactions with strangers on the Internet. The paper explores the roles and impacts of the merchant reputation system, provides an overview of five most popular merchant reputation systems, identifies their evaluation methodology and rating reliability, and proposes suggestions for the design of merchant reputation systems in the future development.

2. Background

Prior to the reputation system, the ratings of merchant service quality were answered through e-mail and survey forms popped out on the merchant website. The recent development of agent technology makes it possible for the merchant reputation system to collect customer feedback on multiple merchants from the Internet. Since one shopping agent website has access to hundreds and thousands of merchants' stores, the reputation system is able to collect the customers' feedback on multiple merchants. Customers who have finished shopping with merchants listed on the agent websites are invited to write text comments, and evaluate their merchants on the reputation system. In response, the online reputation system will display these evaluations including both text comments and ratings to the public. The prospective customers can possibly use these evaluations in their purchase decision-making process. Figure 1 illustrates how a merchant reputation system works on a shopping agent website.

Figure 1 Role of a Reputation System


Seeks and reads the evaluation of the e-merchant
Transaction with the e-merchants completed
Compare items and prices, and select an e-tailer on an agent Web site
Buy the item with the e-tailer
Move to another e-tailer
Positive Experience
Experience
Negative Experience
Positive evaluation
Neutral evaluation
Negative evaluation

The role of the merchant reputation system is to provide information about the past behavior of merchants to online shoppers. The purpose of establishing the reputation system is to build trust by using comments from previous customers as a valuable asset to other customers. Trust is an essential concept that an e-business should attend to since trust has been considered a building block that strengthens relationships between customers and merchants [13]. In the B2C e-commerce environment, trust is more difficult to establish and even more critical for success than in traditional business. The retailers down the block will likely be there the next day, but the merchant that exists in cyberspace is often not real in the customer's eyes [6]. The customers' lack of inherent trust in "strangers" in the e-stores is logical and to be expected. If an e-tail store wants to do business, it has to prove its trustworthiness by satisfying customers for many years as it grows [12].

The rating of the merchant reputation system is important because of its great impact on online shopper behavior and public opinion formation of merchants. As Resnick, et. al [9] defined, a reputation system collects, distributes, and aggregates feedback about participants' past behavior. Though few customers of ratings know one another, these systems help people decide who to trust, and encourage trust behavior.

The merchant reputation has significant impact on customers' trust and on their intentions towards adopting e-services [10]. Taylor [14] found that consumers tend to regard information obtained by "word of mouth" as more objective and possibly more accurate. A satisfied customer will tell three people about his or her experience, but a dissatisfied customer will complain to thirty people. Therefore, consumer comments can be a powerful influence on the purchasing decisions of others [8]. While word of mouth always disseminates reputations informally, Internet can now vastly accelerate and add structure to the process, gathering information about past behavior swiftly and systematically, and distributing it to a broad audience. The reputation system is best known so far as a technology for building trust and encouraging trust behavior Dellarocas [3], and encourages trust worthiness in e-commerce transactions by using past behavior as a public available predictor of likely future behavior.

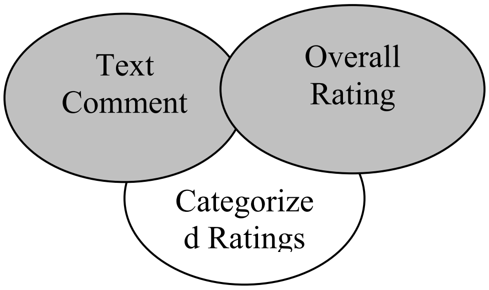
Recent research shows that price and promotion are no longer the main draws for customers to make a decision on a purchase. More and more sophisticated online customers would rather pay higher prices to merchants who provide high quality e-service [12]. Thus, the customer e-satisfaction rating is an important measure of e-service quality. Conventional marketing research has also illustrated the relationship of customer service variables with customer outcomes such as customer satisfaction [1, 4, 18]. The most experienced and successful merchants are beginning to realize that key determinants of success or failure are not merely web presence or low price, but instead center on e-service quality [17]. Study of reputation systems is crucial for both customers and merchants. Customers need to have reliable merchant ratings from merchant reputation systems before they make a purchase decision, and merchants need to have a reliable resource to get the customers' feedback so that they can adjust their marketing strategies and improve their service quality.

3. Rating Methodology

The study is conducted on five most popular merchant reputation systems: BestWebBuys.com, BizRate.com, NexTag.com, PriceGrabber.com and Shopping.com. The results are drawn from examination and investigation of these five reputation systems on the Internet. Three online rating methodologies: text comments, category rating, and overall rating, are identified based on the study of the five online merchant reputation systems. The discussion and illustration these three rating approaches are presented in this section.

The result shows that three evaluation approaches are applied to the five selected merchant reputation systems. Figure 2 shows the combination of three approaches on reputation systems. Text comment and overall rating are in two solid color ovals because they occur in all the reputation systems. The categorized rating is in a transparent oval because it only occurs on some reputation systems.

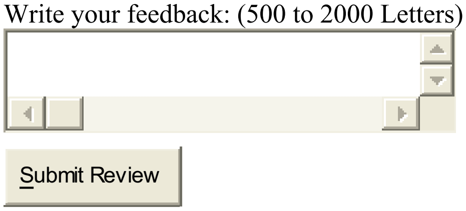
Figure 2 Three Rating Approaches



3.1 Text Comment

The text comment approach provides an opportunity for customers to rate the store according to their own personal experience. This approach allows customers to write their own feedback in 500 to 2000 characters on the e-tail store where they have shopped. The written feedback may include descriptive comments and constructive suggestions that will help the merchant to improve their service and buyers to decide whether or not to do business with this merchant. A sample of a text comment form is illustrated in Figure 3.

Figure 3 Text Comment Form


Write your feedback: (500 to 2000 Letters)

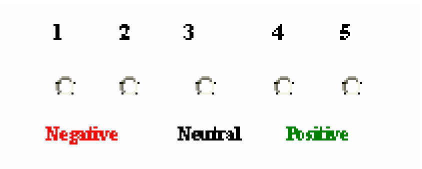
The result shows that all the five merchant reputation systems have adopted the text comment approach to collect customers' feedback demonstrating its important role in the reputation system. As Schellhese et al. [11] indicates in their business marketing research, no one can observe directly how satisfied a customer is in a certain situation. One can, however, ask this person to verbalize the psychological processes the experience.

3.2. Overall Ratings

Overall ratings use an ordinal rating system with a scale of 1 to N where N is the best rating. The overall rating approach occurs in the same format for the majority of the selected reputation system which reflects the customer overall feedback to the merchants. Figure 4 illustrates an example of the overall rating form on the reputation system.

Figure 4 Overall Rating Form

On a scale of 1 to 5, with 5 stars being the best.
Choose the circle below for your rating of this merchant.

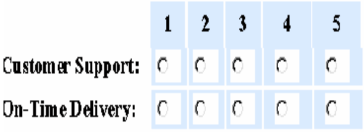


The result shows that four of five merchant reputation systems invite customers to provide direct input for overall ratings except BestWebBuy.com. Instead of collecting overall rating input from its customers directly, the BestWebBuys.com reputation system calculates the overall rating on a merchant based on the customer ratings on the customer-support and on-time delivery issues posted on the system.

3.3 Categorized Ratings

Categorized rating known as a prompted online questionnaire asks customers to rate a number of issues that affect e-satisfaction using a scale of 1 to N where N is the best rating. Figure 5 illustrates the categorized rating form of a reputation system with two most common criteria: customer support and on-time delivery.

Figure 5 Categorized Ratings



	1	2	3	4	5
Customer Support:	○	○	○	○	○
On-Time Delivery:	○	○	○	○	○

The result of the study shows that the categorized rating criteria utilized on the four different systems are very different from one system to another in terms of the number of rating criteria and the content of rating criteria. For example, the Shopping.com invites customer to rate merchants for only three criteria. Whereas, BizRate invites customers to rate merchants for eleven criteria for: six for the pre-order mode and five for post-order mode.

There is similarity shown in the categorized rating criteria on all the 3 merchant reputation systems that have the categorized ratings – they all include on-time delivery and customer support as rating criteria. On-time delivery and customer support. Table 1 illustrates that two common rating criteria occurs on the following three reputation systems.

Table 1 Common Categorized Rating Criteria on Reputation Systems

	BestWeb Buys	BizRate	Shopping
Custome r support	* * * * *	☻ ☻ ☺ ☹	+ + + + +
On-time Delivery	* * * * *	☻ ☻ ☺ ☹	+ + + + +

3.4 Summary

Table 2 shows that combination of text comment and overall rating is used in each of the five reputation systems, but the categorized rating only occurs in three out of five reputation systems. As indicated in Table 2, a text comment approach is used on all the five reputation systems. A categorized rating approach is used in three out of five reputation systems. An overall rating approach is used on five reputation systems, which will be the focus of the next phase of research. In summary, the following three types of rating methodologies have been identified based on the investigation of five reputation systems though they could be in different kind of combination for each reputation system.

Table 2 Three Rating Approaches

	Text Comm ent	Categorized Rating	Overall Rating
BestWebBuys	Yes	* * * * *	* * * * *
BizRate	Yes	☻ ☻ ☺ ☹	☻ ☻ ☺ ☹
NexTag	Yes	N/A	* * * * *
PriceGrabber	Yes	N/A	* * * * *
Shopping	Yes	+ + + + +	+ + + + +

The result also shows that four out of five reputation systems using a rating scale of 1 to 5 where 5 is the best rating. The BizRate.com only has outstanding, good, satisfactory and poor with neutral omitted. It is suggested that BizRate.com may consider neutral ratings not significant enough in the reviews for readers.

Three out of five reputation systems have periodic overall ratings. The periodic ratings will certainly help readers view the progress of merchants' performance in e-service quality. The length of period is different ranging from a past week, a past month, to the past three months and all-time reviews on PriceGrabber.com, BizRate.com and Shopping.com systems.

4. Rating Reliability

The research on the rating reliability of reputation systems can be traced back as early as 1998. Friedman and Resnick [5] pointed out risks related to the reputation system, indicated the ease with which online community participants can change their identity, and concluded that the assignment of lowest possible reputation value to new comers was an effective mechanism for discouraging participants to misbehave and, subsequently, to change their identity. Kollock [7] stated that online rating systems had emerged as an important risk management mechanism in the e-commerce community based on his study of eBay case. Dellarocas [2] identified several scenarios ("ballot stuffing", "bad-mouthing", positive seller discrimination, negative seller discrimination and unfair ratings "flooding") in which buyers and sellers can attempt to "rig" an online rating to their advantage resulting in biased ratings, which don't accurately reflect the expected service quality of a given merchant. Some important management mechanisms have been developed.

So far, most merchant reputation systems have developed measures to counteract the above potential threats in various ways. The study shows that most systems have created registration systems to restrict online reputation writers only to its member customers. Being member customers, they have to log on to their accounts before they evaluate their merchants. Four out of five reputation systems require to log on a registered account before rating is performed with the excerption of NexTag.com.

All the five systems claim that they keep the right to be able to detect and eliminate fraudulent ratings by using a combination of sophisticated mathematical algorithms and large numbers of reviews from a variety of sources that were checked for consistency. This is a technical issue that remains within each company for its confidentiality.

A newly developed mechanism occurs on the GraberPrice.com system is to display the overall rating on a merchant with the purchase transaction number and item description. By clicking the displayed transaction number, readers can see the transaction details including the merchandise name, and its description, price, and date of purchase. This makes the merchant evaluation more meaningful and convincible to readers. The problem is that the rater must have a transaction with the merchant who has registered with the Storefront Company before he/she contributes a rating. Thus, the mechanism has not been popularly used even on the PriceGrabber.com system.

5. Concluding Remarks

The significance of this paper is to provide industrial practitioners, customers, and merchants with a valuable evaluation of current design standards of merchant reputation systems on shopping agent websites. This study makes contributions to the design of merchant reputation systems in three areas: 1) identification of rating methodology, 2) examination of categorized rating criteria, and 3) study of countermeasures for rating reliability.

Three rating approaches: text comment, overall rating and categorized ratings are identified through investigation of the five merchant reputation systems. Each reputation system utilizes the combination of at least two, possibly three approaches because each of the three types of approaches has its pros and cons. Text comment accurately describes real personal purchasing experience, but might not incorporate similar perceptions among individual customers. Overall ratings describe customers' general impression of the store using ordinal numbers, but it misses specific details. Categorized ratings describe feedback on some specific issues of e-satisfaction with ordinal numbers, but these issues may not cover all of the customers' concerns and do not provide reasons to support these categorized ratings. To overcome the above limitations of each of the approaches, the combination of two or three reputation approaches is utilized for different reputation systems.

On-time delivery and customer support are the two major factors identified by Wang & Huarng [16] in their content analysis of text comment of reputation systems. The result of the study further confirms their research result by identifying these two issues as common criteria utilized in categorized ratings on reputation systems. Although the categorized rating criteria on reputation systems are different from one system to another in terms of the number of rating criteria and the content of rating criteria, on-time delivery and customer support criteria are always shared by all the reputation systems.

Logging on to the account before rating has become a very popular mechanism. To create an account to rate a merchant, the rater needs to register with a valid e-mail account at least. Most reputation systems require raters to log onto their account before they write their evaluations.

The study shows that the categorized rating criteria utilized on the different systems are different from one to another in terms of the number of rating criteria and the content of rating criteria. Categorized ratings will affect overall ratings. Different categorized ratings on different systems may generate different overall ratings for the same merchant across different reputation systems. Rating discrepancies may occur on different rating systems. One shopper may be exposed to multiple online merchant reputation systems and be confused by different ratings on the same merchants across different reputation systems.

Rating consistency is crucial for both customers and merchants. A solution to the problem of rating discrepancy across different reputation systems is to develop a new agent website that displays the merchant ratings posted on different merchant systems. It is possible to classify e-tailer stores into three categories: the best stores that received high ranks from all the shopping agents and the worst stores that received low ranks from all the shopping agents. The remaining stores with controversial ratings will also be displayed as a comparison at a glance reference resource for customers. With this, customers can have confidence to shop at highly ranked stores and avoid low ranked stores.

REFERENCES

[1] R.N. Bolton, and J. H. Drew, “A Longitudinal Analysis of the Impact of Server Changes on Customer Attitudes”, *Journal of Marketing*, 1991, 55:1, 1-9.

[2] C. Dellarocas, “Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior”, *Proceedings of the 2nd ACM Conference on Electronic Commerce*, Minneapolis, MN, October 17-20, 2000.

[3] C. Dellarocas, “The Digitization of Word-of-mouth: Promise and Challenges of Online Feedback Mechanisms”, *MIT Sloan Working Paper No, 4296-03*, March 2003, Cambridge, MA.

[4] E. H. Fram, and D.B. Grady, “Internet Buyers: Will the Surfers Become Buyers?”, *Direct Marketing*, 58:10, 1995, 63-65.

[5] E. J. Friedman and P. Resnick, “The Social Cost of Cheap Pseudonyms”, *Telecommunications Policy Research Conference*, Washington DC, October 1998.

[6] M. Head, & K. Hassanein, “Trust in E-commerce: Evaluating the Impact of Third Party Seals”, *Quarterly Journal of Electronic Commerce*, 3(3), 2002, 307-325.

[7] P. Kollock, The Production of Trust in Online Markets. *In Advances in Group Processes* (Vol. 16), JAI Press, Greenwich, CT, 1999.

[8] R. McGaughey, and K. Mason, “The Internet as a Marketing Tool,” *Journal of Marketing Theory and Practice*, 6(3), 1998, 1-11.

[9] P. Resnick, R. Zeckhauser, E. Friedman, and K. Kuwabara, “Reputation systems”, *Communications of the ACM*, 43(12), 2000, 45-48.

[10] K. Ruyter. M. Wetzels, M. Kleijnen, “Customer Adoption of E-service: an Experimental Study”, *International Journal of Service Industry Management*, Vol. 12/ No. 2, 2001, 184-207.

[11] R. Schellhese, P. Hardock, and M. Ohlwein, “Customer Satisfaction in Business to Business Marketing: The Case of Retail Organizations and their Suppliers”, *The Journal of Business & Industrial Marketing*, 15 (2/3), 2000, 106-122.

[12] G. P. Schneider, and J. T. Perry, Electronic Commerce. Course Technology. 3rd Edition. 2002.

[13] T. Siebel, & P. Hous, Cyber Rules, Currency/Doubleday, New York, NY, 1999.

[14] J. Taylor, “The Role of Risk in Consumer Behavior”, *Journal of Marketing*, 38, 1974, 389 - 398.

[15] M. Wang, and D. Christopher, “The Consistency of Customer Online Ratings of E-tailers on Shopping Agent Websites”, Research of Business Review, 1(1), 2003, 14-20.

[16] M. Wang, and S. A. Huang, “An Empirical Study of Internet Store Customer Post-shopping Satisfaction”, *Issues in Information Systems*, Vol. III, 2002, 632-638.

[17] V. Zeithaml, “Service Excellence in Electronic Channels”, *Managing Service Quality*, 12(3), 2002, 135-139.

[18] V. Zeithaml, L. Berry, and A. Parasuraman, “The Behavioral Consequences of Service Quality”, *Journal of Marketing*, 60 (April), 1996, 31-46.

Enhancing Peer-to-Peer Network with RDF and Web

Yanwu Wang
Dept. of Computer Science
Xi'an Jiaotong University
Xi'an , China, 710049
blunt_hust@yahoo.com.cn

Huanzhao Wang
Dept. of Computer Science
Xi'an Jiaotong University
Xi'an, China, 710049
hzhwang@mail.xjtu.edu.cn

Abstract

As a new computing paradigm, peer-to-peer (P2P) has gained great popularity. However resource sharing, resource management and resource localization meets great challenge due to the distribute nature of P2P network. In this paper we introduce the using of Semantic Web recommendation RDF in P2P network. Through annotating resources with RDF metadata, the distributed resources link to each other in a semantic way. Thus application like resource discovery and searching, semantic routing and collaborative activities can become more efficient. For this purpose, we propose Web-based P2P metadata Network Infrastructure (WMNI) to support RDF metadata management, exchange and query. Another feature of WMNI is that it supports accessing P2P resources simply via a web browser i.e. User need not to install P2P application and bootstrap to join the P2P network. This feature is really important to capability- constraint devices like PDA and cell phone.

1. Introduction

P2P computing has proved to be a great success in distributed computing and instant messenger services. Applications such as Napster [1], Gnutella [2], Freenet [3] have been developed to share content over internet. In contrast to the conventional centralized way of managing resource, P2P network organizes these distributed resources in a decentralized manner. Furthermore the resources shared by peers are heterogonous. That means the resources are different in types (digital resource, education resource, medical resource), size etc. However most existing P2P applications characterize resource only with a few keywords. Keywords searching is of low preciseness and has "too many or none" problem: too few keywords lead to many searching results and too many keywords lead to no searching results. Another problem that P2P application faces is efficient resource query routing. Distributed Hash Table (DHT) method used in CAN [4], Chord [5] resolve keywords to location where resources are located. DHT maps each shared contents to string of number using hash function. Mapping resources into non-sense number induces the loss of relation between resources. More disadvantage of DHT is discussed in [6].

RDF [7] metadata itself is not something new, however in P2P network it really is to our best knowledge. Using rich metadata to describe the various heterogonous resources is a reasonable solution. But it incurs interoperability problem. For example someone would like to comment the writer of a book with "author", while another people would like to use "creator" instead. What's more the meaning of "author" in different context is also different. W3C consortium's recommendation Resource

Description Framework (RDF) is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information [7]. We adopt RDF/RDFS to describe resources and relation between resources, thus contents shared by peers are semantically related, e.g. Distributed resources of the same type are structured into a tree or list which can be efficiently navigated by user. The reasoning ability of RDF also enables the content-based search possible and improves the resource discovery ability of peer. Through the exchange of RDF metadata between different peer, resources are easily located. Resource query can route in a semantic way since each peer "know" which peer holds the requested resource.

In order to access resource in the P2P network, user has to install the P2P application in advance, starting the bootstrap to join the P2P network. However popular devices like PDA, cell phone have limited CPU cycles, finite memory and narrow link bandwidth. Our solution is to utilize web to present these resource to user. As shown in Figure 1, any device with a web browser can obtain desired resource via a P2P network Access Peer (AP). The expense of Web approach is protocol convention.

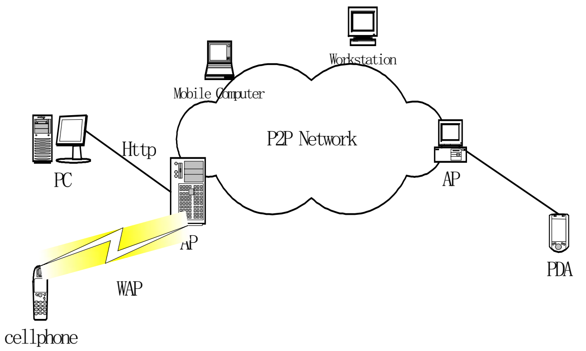


Figure 1. Access resource in P2P via web

For the above considerations, we propose our Web-based Metadata P2P Network Infrastructure (WMNI) upon JXTA. WMNI deals with RDF metadata management generated by different peer, metadata query and metadata exchange between peers. JXTA [8] is an open source computing platform designed for P2P computing, initiated by Sun Microsystems. Its goal is to develop basic building blocks and services to enable innovative applications for peer groups.

The rest of the paper is organized as follows. Section 2 gives a brief introduction to JXTA. In section 3 we present our proposed framework WMNI and elaborate it. In section 4 we have a detailed discussion of accessing P2P resource via web. Section 5 gives example applications built on WMNI. Section 6 is related works and Section 7 is conclusion.

2. JXTA architecture

Almost every P2P application introduces a different protocol, replicating already done work and causes unnecessary incompatibilities. JXTA provides protocols [8] that standardize the core P2P functionality, which includes peer discovery, self-organization into peer groups, advertisement and discovery of resources, communication and monitoring. Figure 2 shows JXTA's software architecture.

JXTA Core Layer deals with peer establishment, communication management such as routing, and other low-level plumbing. JXTA Services Layer is a service layer that deals with higher-level concepts, such as indexing, searching, and file sharing. These services, which make heavy use of the plumbing features provided by the core, are useful by themselves but also are commonly included as components in an overall P2P system. In JXTA all components such as Peers, Peer Groups, Pipes, peer services, Peer Group Services, are represented with corresponding advertisement. An advertisement is an XML document denoting these components. Our framework is built on the JXTA Services layer. WMNI makes use of JXTA core to provide basic service like peer organization and communication, but the methods that we proposed is not limited to JXTA.

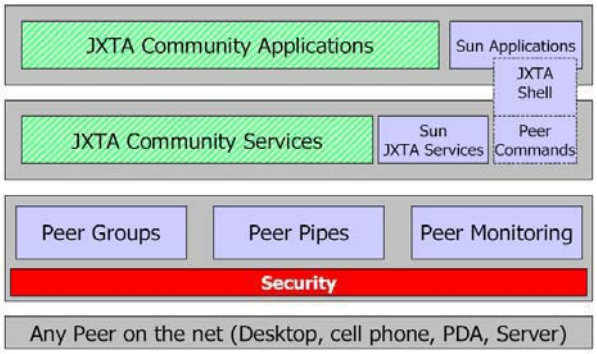


Figure 2. Software architecture of JXTA

3. WMNI architecture

The WMNI architecture aims to provide an underlying RDF metadata infrastructure to facilitate the development of various P2P applications. Our basic idea is to utilize RDF metadata to annotate resource, making distributed resources link to each other in a semantic way. WMNI consists of the following components: information space to store RDF metadata, management and exchange of metadata between peers, interfaces to query and reason on metadata. Figure 3 shows our proposed architecture WMNI.

3.1 RDF and data storage

Relations between resources are expressed with RDF statements, so back-end databases are employed to support this feature. The graphic representation of RDF is very similar to ER diagram, so RDF metadata can be easily stored in relation database. Another alternative method is to store these statements in XML database since RDF data is serialized with XML syntax. Various tools are already available for handing RDF-based metadata. Jena [9] is an open source project hold by HP. It provides java-programming interface for reading, writing, querying RDF in XML. Redland [10] is a library that provides a high-level interface for RDF allowing the RDF graph to be parsed from XML, stored, queried and manipulated.

Shared resources can be stored either in database or file system according to the characteristic of content. Relation Database is suitable for storing highly structured data that are generated dynamically such as report forms. File systems can be used to store video files, web document.

To associate each resource with its corresponding annotation (RDF statements), an additional Object Oriented Database storage is required to record the mapping between resources and RDF statements. Each shared contents is assigned with a Unique Resource Identifier (URI). This identifier can be acquired through hash digest, for example SHA1, on the shared content.

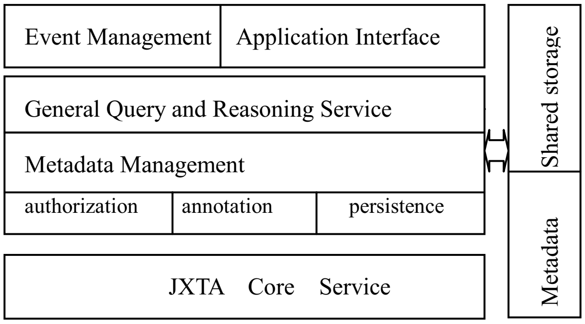


Figure 3 Architecture of WMNI

3.2 Metadata management

Metadata management includes annotation, persistence and authorization operation on metadata. When a user decides to share a resource (video file, web document etc) within the P2P network, several steps are taken in the following order. First the sharing contents are annotated according to some RDF schemas, and then the annotation is saved in metadata storage. To announce the existing of content to other JXTA peers, the information about the metadata is encapsulated in an advertisement, and then propagates to other peers. Here we have two problems: what granularity of metadata information to send and which peers to receive the information. Granularity means the abstract levels of information to be published, e.g. in schema level (DC[11], LOM[12]) or in entries level (DC, DC:Creator, DC:language, LOM). The second question concerns about clustering of information. Sending the information to all peers blindly may incur unnecessary traffic. In [13] various cluster strategies are covered based on the consideration of efficient query message routing.

Annotation service: To make the distributed resource in P2P network semantically link to each other. We propose to comment resource with RDF metadata according to RDF Schemas. In Semantic Web context, RDF Schemas are vocabulary of domain. The design of domain specific vocabulary is so-called ontology engineering. It is really a sophisticated process and only can be done by domain expert. Several standard RDF Schemas have been developed in some domain. For example Dublin Core (DC) is designed for category and retrieve learning material, IEEE-LOM/IMS (LOM) is designed for exchanging educational resources.

Persistence and monitoring: persistence includes add, remove, update of metadata in RDF database. Monitoring keeps track of modification of RDF metadata and triggers modification events. The interfaces provided by database system are primitive. These interfaces should be published as JXTA service so that others peers can consume them. As we have mentioned early in this section, metadata changes in one peer should be publish to others peers. So each persistence operation will be monitored and trigger an event. Other peers are notified with the event in a JXTA message. We introduce two modes to dispatch the message to other peers: *active mode* and *lazy mode*. In active mode, when a persistence event occurs, a message is sent to remote peers immediately. In lazy mode, only when the peer receives a query, would the peers send such a message. Different model can be adopted in different application environment. The active model is suitable for collaborative application like blackboard writing/reading, while lazy mode suitable for common content sharing application.

Replication, leasing, and authorization: The objective of metadata replication is to facilitate resource localization, while the objective of data replication is to achieve reliable storage in case of network failure. Here we mainly deal with replication of RDF metadata. The redundancy of metadata can utilize the JXTA replication service. Leasing is lifetime management of replicas. To avoid RDF storage from being flooded with metadata, outdated information should be removed according to a certain strategy. Some simple strategies like LRU (least recently used) or LFU (least frequent used) can be adopted. Authorization deals with security issue of RDF metadata. Metadata authorization prevents malicious user from fabricating metadata.

3.3 Query and reasoning

The common query and reasoning service is designed to shield the difference between underlying metadata query language, presenting user an easy to use interface. As discussed before, the resource description (i.e. metadata) in our system can be stored in XML database or traditional relation database. Since the metadata is expressed with XML, some XML query languages like XPath, XQuery can be used to query metadata. Also some SQL like languages, for example RQL [14], have been developed to query RDF metadata in database. These query languages have different syntax. Thus we need a general-purpose query language that acts as bridge between them. RDF-QEL [15], an XML-based Query Exchange Language, provides the syntax for an overall standard query interface across heterogeneous peer for any kind of RDF metadata. The General query service is application interfaces for this uniform language. Peer who wants to perform some query task can simply register query request to peers that provide query service. When query service is invoked, the query message is distributed to P2P network transparently. The requesting peer is notified with advertisement when the query process is finished.

4.4 Application interface

The peers in our system are highly heterogeneous peers. That means they may apply different service implementation .The service implementation is service provider of application interface. This idea is very similar to JNDI. The service interfaces are highly abstract, independent of its implementation. Applications developed in one peer can easily migrate to another peer after only modifying the service binding statement. With these application interfaces powerful and efficient application can be built.

3.5 Event management

The P2P network is dynamic and communication often takes long delay, asynchronous communication is preferred. Event subscribing and notification model satisfies this requirement. As discussed before metadata persistence operation will trigger events. The “interested peers” are notified with the changes of metadata information. In addition, the metadata query service also works in asynchronous mode. When a user issue a query, the query request is submitted to P2P network, registering with a callback function to handle the notification. Event management is especially important for those collaborative P2P applications.

4. Accessing resource via Web

4.1 Notion definition

The following is notions that are used later:

Foreigner: User who consumes the P2P network resource without having to install P2P application in advance. In this paper the foreigner acquires resources through a web browser.

Service Peer (SP): peer that provides resources consumed by foreigner.

Access Peer (AP): peer that mediates the communication between foreigner and SP.

4.2 web-base solution

The web-base approach enables those capability-constraint devices, i.e. foreigner, to access resources in P2P network. Peers communicate with each other using specific platform protocols, which are not compatible with current Internet protocols such as Http, RMI, and IIOP. Foreigners cannot interact with peers if they don’t participate the P2P network in advance. One solution is to re-implement P2P service as web service. JXTA SOAP binding [16] is designed to allow SOAP communication over the JXTA P2P network. [17] discusses how to expose existing P2P service as web service. The JXTA-RMI project [18] allows to develop applications with familiar Remote Method Procedure. Through running a java applet in the client’s web browser, foreigners can communicate with peer directly. But [16], [18] are ongoing, not mature and restricted to JXTA platform only. Exposing JXTA service as web service seems attractive and promising, however, the protocol conversion needs much technique work due to considerable protocol incompatibility. [19] built a web system over its SIONET P2P system. It employed a P2P application control in every peer to eliminate protocol difference.

In contrast to [19], our solution is a light-weight one. Figure 4 illustrates the suggested solution. Both AP and SP contain a Web server, running a Java servlet engine. The foreigner connects to an AP querying for some resources. The AP sends the query to P2P network through query interface provided by WMNI and locates the SP. Then AP relays the communication between foreigner and SP. The Http packet is load of JXTA pipe. First AP encapsulates the request http packet sent from foreigner into JXTA message, and then AP transmits the JXTA message to SP in JXTA pipe. SP parses the http message from JXTA message and sends web service page to foreigner in a reverse direction. The service page may

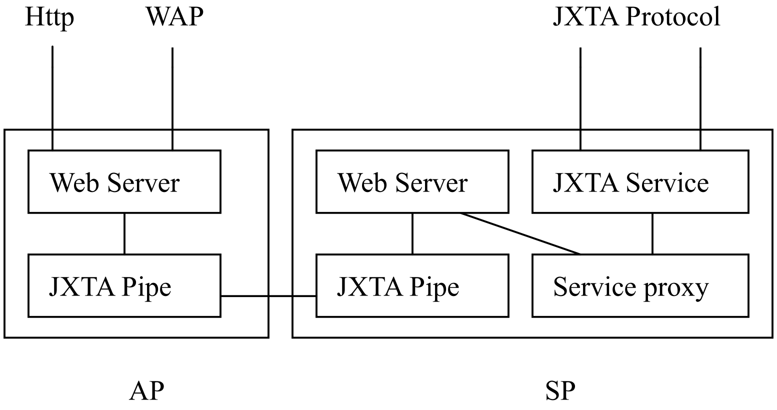


Figure 4. Skeleton of Web approach

contain JXTA Service proxy (proxy may reside in Java component e.g. Applet). A proxy acts as broker between Web client and JXTA service: foreigner invokes the interfaces supplied by proxy to access shared resource, in turn the proxy calls the JXTA service.

4.3 Functionality of proxy

The design of service proxy should satisfy at least the following requirement: 1) The proxy provides interfaces that can be invoked through a Java component. Each proxy is corresponding to a JXTA service.2) message wrapping. Input information submitted by foreigner through web page is a series of Java objects (String, Integer etc). Proxy should serialize these Java objects in form of JXTA message, and vice versa. 3) Proxy should also be responsible for mediation between socket and JXTA pipe. Peers in JXTA network communicate with each other using pipes. Furthermore, most communication in P2P network is asynchronous due to relative long delay. The result from SP can be send back to foreigners in a *push-like* notification. Event subscribing and notifying module in WMNI supports this feature.

5. Application

To further illustrate our proposed WMNI, some potential applications that can be built on WMNI are discussed.

5.1 Collaborative activity

Information exchange and sharing is common in collaborative activities. One user interacts with another user to acquire some information. The user may extract some part of information item that he is interested in from message he received. After processing, some new information is added and forwarded to other collaborative participants. In this context, two kinds of interactions exist: interaction between users and interaction between user and information. It is easy to describe the interaction with RDF statements. The activity participant and information item are mapped to *resource* in RDF statement; interactions are mapped into *predicate*. Through the annotation of interaction, the information can be reused. We believe that WMNI can facilitate the development of such collaborative application. Firstly, activity participants communicate and interact with each other in a P2P manner. While in traditional C/S collaborative application, the server is responsible for session management and need to keep track the state of participant. So the server often becomes complicated and is not easy to be maintained. Secondly, annotating interaction can solve the evolvement and interoperability problems of collaborative application.

5.2 Metadata clustering and semantic routing

P2P network is a kind of application overlay. The peers are not aware of the physical network structure, so message routing in P2P network cannot apply the mechanism used by Internet. Routing strategies like flooding, random walk are not scalable. In [6] disadvantages of DHT are also discussed. Based on our metadata infrastructure, we can set up semantic routing table for message routing. Table 1 shows a sample schema level routing table.

Table 1. Sample schema level routing table

Schema	Destination Super Peer
Dublin Core	Super Peer A
Mp3	Super Peer B
Dublin Core	Super peer B
….	…

In WMNI, all resources are annotated according to RDF schemas. Schema is a collection of domain vocabularies. Initially the peers are divided in small groups (the network diameter of group is several hops). The routing table is held in a *super peer*. The notion of *ordinary peer* and *super peer* is corresponding to host and router in Internet respectively. First the metadata information in a group is clustered to super peer in different level (schema level, property level etc.), then the super peer exchange with each other to setting up the semantic routing table. In three situations the routing table need to be updated: peer join, peer departure and metadata information modification. The event management module in WMNI supports "event registry and notification " and can fulfill this requirement.

6. Related work

In [20] Xin implemented a metadata search layer as an enhancement of JXTA content manager service. They used Dublin core Schema to annotate shared contents. Instead of storing the metadata in separately RDF storage as we have proposed, they stored annotation together with resource advertisement. Edutella [8] aims to build an metadata infrastructure to connect peers in P2P network based on exchange of RDF metadata. It concentrates on providing a common data model and Edutella query exchange language.

7. Conclusion

In this paper, we introduce the use of RDF metadata in P2P network to link distributed resources in a semantic way. To make capability-constraint devices access P2P network in a convenient way, we propose to present the resources with Web approach. The proposed WMNI provides an underlying metadata infrastructure that support the development of efficient P2P application. WMNI also enables capability-constraint devices(PDA, cell phone) to consume the resources without installing the P2P application in advance. Although WMNI is based on JXTA, but it is not limited to JXTA. To get concrete understanding the advantage of WMNI, two applications are discussed. 1)collaborative application: we use RDF statements to annotate the interaction between activities participant. 2)metadata clustering and semantic routing: Peers exchange metadata with each other to built up semantic routing table for resource query.

References

[1] Napster LLC. http://www.napster.com

[2] The Gnutella Project , http://www.gnutella.com

[3] Freenet Home Page, http://freenet.sourceforge.com/.

[4] S. Ratnasamy, R. Francis, M.Handley, R. Krap, J. Padye, and S.Shenker. A scalable content-addressable network. In ACM SIGCOMM, 2001.

[5] I. Stocia, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In ACM SIGCOMM, 2001.

[6] Zhou J., Vijay Dialani, De Roure D. and Hall W, A Distance Based Semantic Search Algorithm for Peer-to-Peer Open Hypermedia Systems, Proceedings of the Fourth International Conference on Parallel and Distributed Computing Applications and Technologies, Pages:7-11 Aug. 2003

[7] Resource Description Framework, W3C consortium http://www.w3.org/TR/1999/PR-rdf-syntax-19990105

[8] The JXTA Project ,www.jxta.org

[9] Jena: A Semantic Web Framework for Java. http://jena.sourceforge.net/index.html

[10] Redland RDF Application Framework . http://www.redland.opensource.ac.uk/

[11] Dublin Core Metadata Initiative, "Dublin Core Metadata Element Set, Version 1.1", http://dublincore.org/usage/terms/dc/current-elements/, October 2002.

[12] IEEE Learning Technology Standard Committee Working Group 12, "Final 1484.12.1 LOM draft standard", http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf, September 2002

[13] Alexander L¨oser1, Wolf Siberski, MartinWolpers, and Wolfgang Nejdl. Information Integration in Schema-based Peer-To-Peer Networks. Lecture Notes in Computer Science, Springer-Verlag Heidelberg, Volume 2681 / 2003, 258 - 272 ,January 2003

[14] G. Karvounarakis, S. Alexaki, V. Christophides, D. Plexousakis and Michel Scholl. RQL, A Declarative Query Language for RDF. The Eleventh International World Wide Web Conference (WWW'02), Honolulu, Hawaii, USA, May 7-11, 2002

[15] Wolfgang Nejdl, Boris Wolf, Changtao Qu, Stefan Decker, Michael Sintek, Ambjorn Naeve, Mikael Nilsson, Mathias Palmer and Tore Risch. EDUTELLA: A P2P Networking Infrastructure Based on RDF. WWW2002, May 7–11, 2002, Honolulu, Hawaii, USA.ACM 1581134495/ 02/0005.

[16] JXTA SOAP Project , http://soap.jxta.org/servlets/ProjectHome

[17] Changtao Qu and wolfgang Nejdl. Interacting the Edutella/JXTA Peer-to-Peer Network with Web Services. International Symposium on Applications and the Internet, IEEE,Proceedings, 2004

[18] jxta-rmi , http://jxta-rmi.jxta.org/servlets/ProjectHome

[19] Shoji Kimura,Koji Sakai, Akira Kurokawa, and Hiroshi Sunaga. Implementation and Evaluation of the P2P Web System, The 9th Asia-Pacific Conference on Communications, APCC 2003 , IEEE ,Volume: 1 ,21-24

[20] Xin Xiang, Yuanchun Shi and Ling Guo. Rich Metadata Searches Using the JXTA Content Manager Service. http://media.cs.tsinghua.edu.cn/~xiangx/paper/cms.pdf

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 163-170, 2004.

FIM-MetaIndexer: a Meta-Search Engine Purpose-Built for the French Civil Service and Statistical Classification of the Interrogated Search Engines.

Katarzyna Wegrzyn-Wolska
ESIGETEL, Ecole Supérieure d'Informatique et Génie de Télécommunication
77-200 Avon-Fontainebleau, France
katarzyna.wegrzyn@esigetel.fr

Abstract

Searching for specific information on the Web is becoming more and more difficult. To facilitate this task, it is necessary to use specialized tools. Our objective is to build a searching system for the French Civil Service. We describe here our work done within the framework of French gouvernmental contract by FIM[1] during the PhD thesis in Ecole des Mines de Paris (ESNMP). We survey the problems related to Web information retrieval and particularly meta-search techniques. We describe the implementation of a system to search for administrative documents. The goal is to retrieve the documents corresponding to a question simultaneously submitted to several Civil Service Web servers. We present our study, choice of methodology and the implementation of our meta-system. We describe some evaluation results of the experiments, performed to evaluate the relevance of the answers received from the search engines.

1. Introduction

This paper describes a Meta-Search system developed for searching the documents produced by the French Civil Service[2]. The main purpose of this system is to obtain the information directly from its source location, from the government website.

First, we describe the general problems of information retrieval on the Web, then we describe the FIM-MetaIndexer system in terms of architecture, information flow and configuration possibilities. We describe also some of our methods of analysis and classification of the different Search-Engines implemented on the government sites. The conclusion summarizes our results and provides suggestions for future improvements.

2. Searching for information on the Web

2.1. General problems of information retrieval on the Web

The Web is an enormous source of useful information. Surfing the Web can be a great pleasure and adventure, but searching for specific information can often be very difficult.

Most of the information retrieval problems are due to [3][7][12] the size of the Web, the quantity and quality of documents [2], reliability of links [5], index database updating, indexing of dynamic pages, the heterogeneity of the document formats and resources, problems related to multiple languages, etc.

The most popular searching methods are: subject catalogues, Search Engines, Meta-Search tools, on-line database agents, and other tools like robots (spiders and crawlers) and monitoring agents [8][4].

The Web-based information retrieval and searching techniques are necessary, the new systems like a WSS[3] and WIRSS[4] [15][16] are steel under development, and the new concepts are introduced.

2.2. Meta-Search Engines

The Meta-Search Engines [6][9][10] are tools, which transmit an individual client's query simultaneously to several different Search Engines (Figure 1).

Meta-Search Engines do not have their own database of Web pages (corpus of Web documents) but can only access the documents from the result pages given by search engines which have been queried.

[1] Fonds Interministériel de Modernisation de l'Administration Française

[2] FIM-MetaIndexer system developed for the *Fonds Interministeriel de Modernisation de l'Administration Française.*

[3] WSS Web-based Support Systems

[4] WIRSS Web-based Information Retrieval Support Systems

There are two kinds of Meta-Search Engines implementations either on the server or on the client.

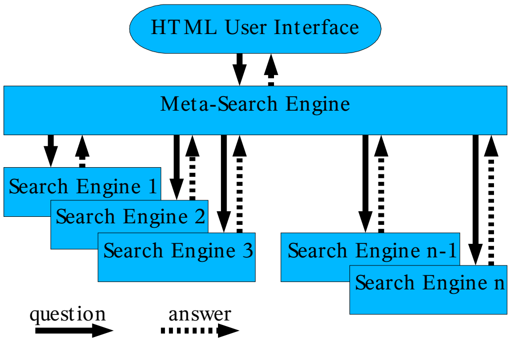


Figure 1. A general Meta-Search structure

2.2.1. Advantages and disadvantages of Meta-Search Engines. Some of the most significant advantages [11] are: simultaneous Search Engines interrogation, uniform request form, homogeneous response presentation, and the possibility of including additional functions such as: link verification, double page elimination, answer relevance evaluation etc.

The most significant disadvantages are: the growth of response time, the risk of elimination of some relevant answers, copyright problems, etc.

2.2.2. General Meta-Search Engines. The general Meta-Search Engines are neither dedicated nor adapted to information retrieval from the Civil Service Web sites. They consult general Google-like Search Engines, so it is not possible to find a particular Civil Service's documents. These documents are rarely indexed, if at all. It is difficult to retrieve the dynamic pages by autonomous retrieval agents.

2.3. Answer relevance checking

2.3.1. Checking relevance methods. There are two major types of relevance checking methods [14]; textual and topological. The topological method encompasses the textual, whilst taking into account hyperlinks.

2.3.2. Problems. Relevance checking of the answers given by the Search Engines involves a number of difficulties. The most serious problems are: different and nonhomogenized searching methods, heterogeneous relevance evaluation algorithms and different classification methods implemented on the Search Engines queried.

The Meta-Search method involves other problems: real time interrogation and checking slows down search speed (it is necessary to find a compromise between the response quality and the speed of the search system); the unknown total document corpus, which means that it is not possible to use standard retrieval performance checking methods. All the standard methods need fixed limits to the total document corpus. The Web corpus is unlimited and cannot be accessed by the Meta-Search Engine.

3. FIM-MetaIndexer

FIM-MetaIndexer is a Meta-Search system developed to search for documents produced by the French Civil Service[5]. The main purpose of this system is to retrieve the information directly from its source location, the government web site thus giving us the possibility to obtain immediately all of the existing and available information on the web site, as well as irretrievable documents in the search results of a standard Search Engine. These are documents not yet indexed because of the necessary delay time for the indexing task, or the documents named 'Invisible Web", which are not indexed at all by the standard Google-like Search Engine. These documents are dynamically generated as an answer to the query.

3.1. Methods choice and servers choice for queries

3.1.1. Method choice. Our Meta-Search Engine (FIM-MetaIndexer [13]) was installed on a distant accessible from Internet server. The server uses 50 independent agents to interrogate Search Engines simultaneously.

The technology used is based on the HTTP protocol. We used the GET, and the POST methods for page retrieval, and the HEAD method for page existence verification. The system was developed using the Perl programming language.

3.1.2. Choice of Search Engines. Our system interrogates three kinds of Search Engines: standard Google-like Search Engines[6]; more then 30 specialized French Civil Service Search Engines[7] (like the "French Senate" site Search Engine) and the Civil Service portals[8].

[5] FIM-MetaIndexer system developed for the *Fonds Interministeriel de Modernisation de l'Administration Française.*

[6] e.g. : Google (http://www.google.com), AltaVista (http://www.altavista.com), Yahoo.fr (http://www.yahoo.fr)

[7] e.g. : *Assemblée Nationale* (http://www.Assemblee-nationale.fr), *Sénat* (http://www.senat.fr), *Ministère de la Justice* (http://www.justice.gouv.fr), *Ministère de l'Education Nationale* (http://www.education.gouv.fr)

[8] *Adminet-Journal Officiel* (http://www.admi.net/jo), *Service-Publique* (http://www.service-publique.fr)

4. FIM-MetaIndexer architecture

FIM-MetaIndexer is based on modular architecture. There are three major modules (Figure 2):

- **MAD** (*Module d'Analyse des Données*) data input analyzer;
- **MIM** (*Module d'Interrogation des Moteurs*') Search Engine interrogation module;
- **MAR** (*Module d'Analyse des Réponses récupérées*) answer analyzer module.

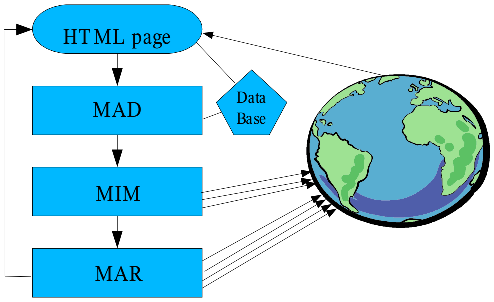


Figure 2. FIM-MetaIndexer architecture

4.1. MAD: Data Input Analyzer

The data input analyzer is the module in charge of comprehension and analysis of client queries and all selected parameters (for example : choice of engine for the interrogation, maximal research time, number of results to be recovered, mode for searching and checking the answers). These parameters are chosen by the user using the HTML based input interface. All necessary data for the interrogation are used to build the general request form which is transmitted to the Search Engine interrogation module (MIM).

4.2. MIM: Search Engine Interrogation Module

This is an agent-based interrogation module which works with selected Search Engines. This module is in charge of three important tasks: translation of client queries and the whole set of parameters into a form which is recognizable by the Search Engines, selection and interrogation of Search Engines, answer recovery and checking that the documents exist.

4.2.1. Query building. There are many different types of Search Engines. They use different search methods, they use and accept only their own specific request format. So, before interrogating the Search Engine, it is necessary to transform the general format of the query prepared by the MAD module into a format which is comprehensible to, and accepted by, each Search Engine. The request building algorithm is presented in Figure 3.

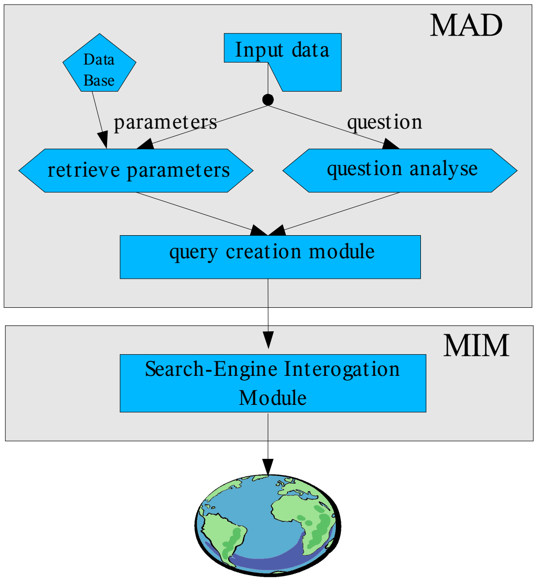
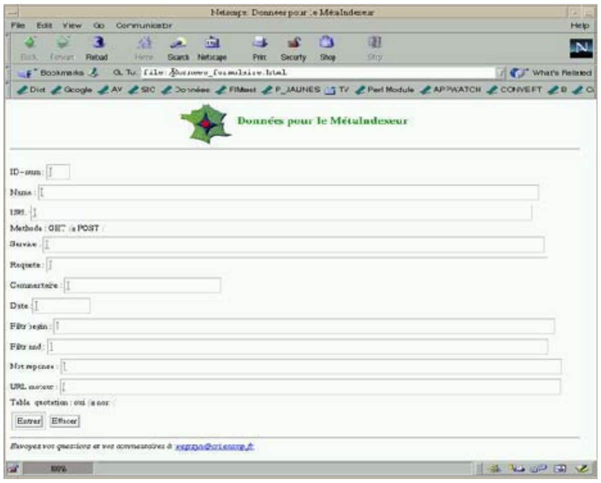


Figure 3. Request construction algorithm

In order to translate from one format to another FIM-MetaIndexer needs to know the configurations data concerning every Search Engine. This data is collected during the integration process of each Search Engine and saved in FIM-MetaIndexer data base.


Netscape: Données pour le MétaIndexeur
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks
Go To: file:/Donnees_formulaire.html
What's Related
Dict Google AV SIC Données FIMaut P_JAUNES TV Perl Module APPWATCH CONVERT
100%

Données pour le MétaIndexeur

ID-num :

Nom :

URL :

Methode : GET ○ POST

Service :

Requete :

Commentaire :

Date :

Filtr begin :

Filtr end :

Nbr reponses :

URL moteur :

Table quotation : oui ○ non

Entrer Effacer

Envoyez vos questions et vos commentaires à [illegible]

Figure 4. Input data form for new Search

Modification of the Search Engine description or insertion of a new Search Engine to the list is possible in two ways : directly in the data bases, or by filling in the special insertion HTML form (Figure 4).

4.2.2. Query translation. Query translation is carried out with special attention to : Boolean expression, accentuated letters, optional parameters and seeking the exact phrase.

The Boolean operation can be expressed in varied forms [1], so it is necessary to modify the expression given by the user according to Boolean functions and Boolean operators appropriate to each Search Engine. For unacceptable functions it is necessary to change Boolean expression according to the possibilities offered by a selected Search Engine. For example, if the Search Engine does not accept an alternative function, each component of the alternative function is sent separately to collect the results.

The treatment of the accentuated letters is also full of pitfalls. It is necessary to take note of and distinguish two kinds of treatment, one for the interrogation and the other for the analysis of the document answers.

In the request different options may be used which are proposed by FIM-MetaIndexer (searching mode, number of results, number of results on one page). This will only work with a Search Engine which uses the same, or very similar parameters, for search configuration.

4.2.3. Recovery and answer checking. FIM-MetaIndexer offers some configuration possibilities for searching methods: with or without checking the document-answers existence. The default option without verification is faster and requires minimal network occupation.

4.3. MAR: Answer analyzer module

This module checks the relevance of the answers according to the selected mode (verification on/off) and presents the results sorted by score. FIM-MetaIndexer sorts the results according to the relevance and prevalence of the same documents. FIM-MetaIndexer eliminates repeated answers. The more often the document is cited, the more relevant it is considered to be. This is referred to as weight of relevance and is increased by a specific coefficient. This coefficient is calculated dynamically for each multiple answer, according to the number of repetitions.

Relevance checking with the "verification on" mode is concerned with existence verification and calculation of the number of repetitions of every significant word used in the question.

In "verification off" mode FIM-MetaIndexer cannot itself evaluate or measure the relevance of the document-answers. It will retain the relevance score supplied by each questioned engine. This method of document sorting is illustrated in Figure 5.

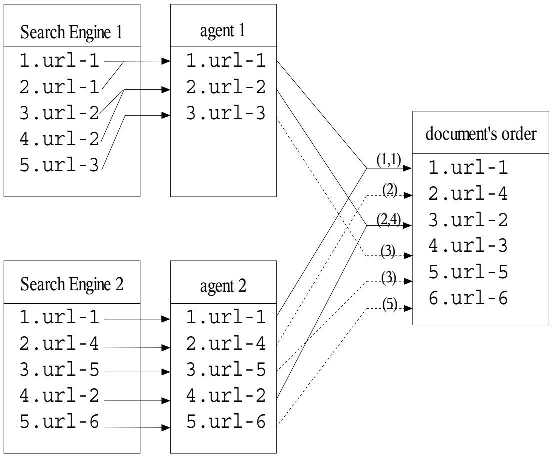
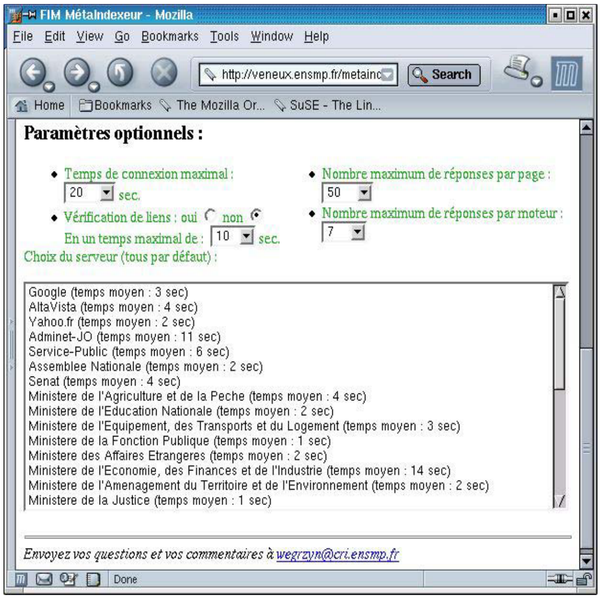


Figure 5. Results sorting algorithm

5. FIM-MetaIndexer user interface

5.1. Input data form

Our Meta-Search Engine gives a user-friendly interface with some configuration options (Figure 6).


FIM MétaIndexeur - Mozilla
File Edit View Go Bookmarks Tools Window Help
http://veneux.ensmp.fr/metainc
Search
Home
Bookmarks
The Mozilla Or...
SuSE - The Lin...
Done

Paramètres optionnels :

- Temps de connexion maximal : 20 sec.
- Vérification de liens : oui ○ non ●
  En un temps maximal de : 10 sec.
- Nombre maximum de réponses par page : 50
- Nombre maximum de réponses par moteur : 7

Choix du serveur (tous par défaut) :

Google (temps moyen : 3 sec)
AltaVista (temps moyen : 4 sec)
Yahoo.fr (temps moyen : 2 sec)
Adminet-JO (temps moyen : 11 sec)
Service-Public (temps moyen : 6 sec)
Assemblee Nationale (temps moyen : 2 sec)
Senat (temps moyen : 4 sec)
Ministere de l'Agriculture et de la Peche (temps moyen : 4 sec)
Ministere de l'Education Nationale (temps moyen : 2 sec)
Ministere de l'Equipement, des Transports et du Logement (temps moyen : 3 sec)
Ministere de la Fonction Publique (temps moyen : 1 sec)
Ministere des Affaires Etrangeres (temps moyen : 2 sec)
Ministere de l'Economie, des Finances et de l'Industrie (temps moyen : 14 sec)
Ministere de l'Amenagement du Territoire et de l'Environnement (temps moyen : 2 sec)
Ministere de la Justice (temps moyen : 1 sec)

Envoyez vos questions et vos commentaires à wegrzyn@cri.ensmp.fr

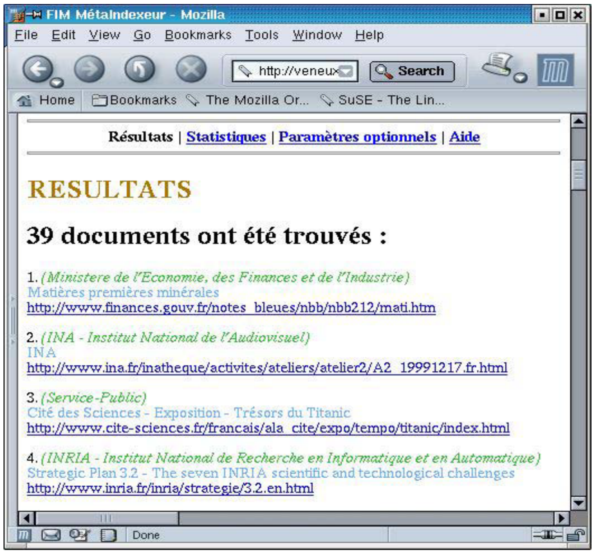
Figure 6. FIM optional configuration parameters page of presentation results

The user has the choice of selecting some optional parameters such as: connection time-out, checking existence of links (on/off), global searching time-out, choice from the list of servers to be interrogated, number of answers and maximum number of answers for each server interrogated.

5.2. The presentation of results

The results' page (Figure 7) presents the list of search results and some statistical estimates such as average: response time, minimal response time, the percentage of right and non relevant responses, and the percentage of interrogations without connection.

For each answer some more details such as: server name, document title and document addresses (URL) are presented.


FIM MétaIndexeur - Mozilla
File Edit View Go Bookmarks Tools Window Help
http://veneux
Search
Home
Bookmarks
The Mozilla Or...
SuSE - The Lin...
Done

Résultats | Statistiques | Paramètres optionnels | Aide

RESULTATS

39 documents ont été trouvés :

1. *(Ministere de l'Economie, des Finances et de l'Industrie)*
Matières premières minérales
http://www.finances.gouv.fr/notes_bleues/nbb/nbb212/mati.htm

2. *(INA - Institut National de l'Audiovisuel)*
INA
http://www.ina.fr/inatheque/activites/ateliers/atelier2/A2_19991217.fr.html

3. *(Service-Public)*
Cité des Sciences - Exposition - Trésors du Titanic
http://www.cite-sciences.fr/francais/ala_cite/expo/tempo/titanic/index.html

4. *(INRIA - Institut National de Recherche en Informatique et en Automatique)*
Strategic Plan 3.2 - The seven INRIA scientific and technological challenges
http://www.inria.fr/inria/strategie/3.2.en.html

Figure 7. FIM page of results

6. Analysis, statistics and classifications of Search Engines

We performed experiments to evaluate the answers received from the selected Search Engines and to classify the profile of the French Civil Service websites.

Our experiment method was based on statistical analyzis of the data obtained form the answers given by the Search Engines queried. All of the experiments were done using the FIM-MetaIndexer.

The next sections describe some statistical evaluation of our results.

6.1. Evaluation of the test sets

Two sets of standard queries were prepared. The first with 1000 queries and the second with 50 queries. The queries were selected from the log file of the Adminet[9] server, the popular French Civil Service Search Engine.

All query sets were carefully chosen to represent the true user question form. Tests proved that selected queries have a similar structure to that of the Adminet's questions, for example: a mean query length of 2.7 words; a mean word length of 7,5 characters. These values are common for the Web log statistics.

Two experiments were carried out with the results saved into the local FIM-MetaIndexer database: "TEST1" - all servers were queried using the first set of queries (1000 queries) and answers (hyper links) were saved locally; "TEST2" – all servers were queried using the second set of queries (50 queries), all the links were followed up locally. The results were then analyzed using statistical methods.

6.2. Response time statistics (TEST1*)*

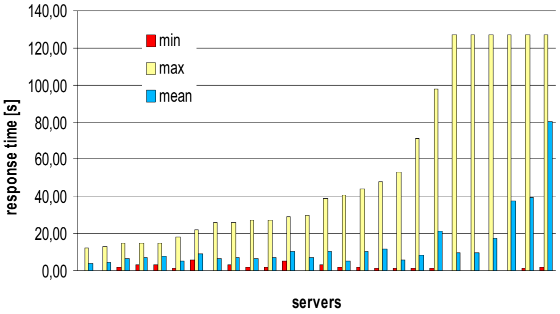


Figure 8. Response time

Different statistics concerning the response time were calculated. We present three of them: response time for each sever (Figure 8), the thematic classification of the Search Engines, and the correlation between the response time and the number of responses.

6.2.1. Thematically classifications. We tried to analyzed the thematic profile of the French Civil Service websites. With the specially prepared, thematically selected questions set, we analyzed the dependance between the theme of the question and the quantity of relevant answers to it. Then we analyzed the thematically correlation between the pairs of selected websites of French Civil Service. Figure 9 presents the graph of thematic dependance of the

[9] Adminet (http://www.adminet.fr)

analyzed Web sites with the different correlation value (correlation ≥ 0.5).

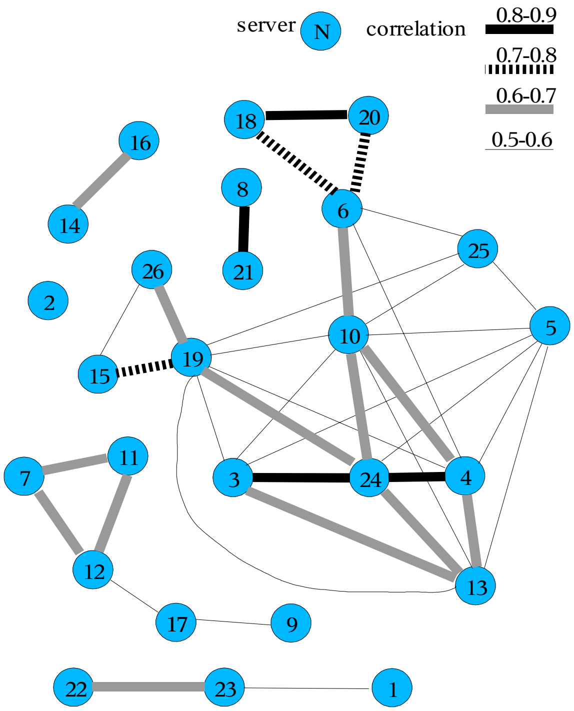


Figure 9. Thematic profile of governmental websites : dependance graph

We observed an large number of the pairs of servers grouped thematically with significant correlation (Figure 10).

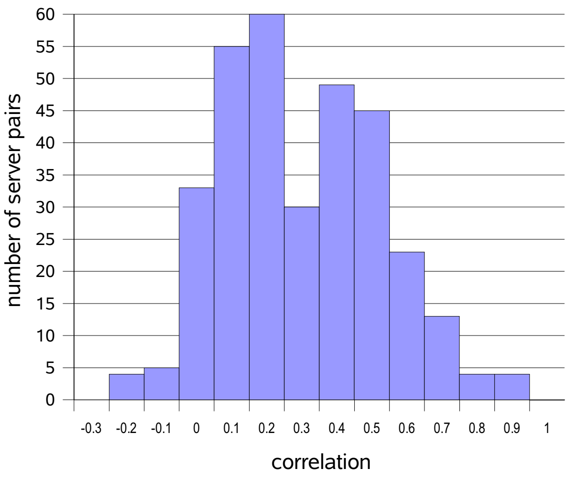


Figure 10. Correlation for the different pair of the websites

6.2.2. Correlation between the response time and the number of responses. The high positive correlation confirms the natural dependance between the measured values. We observed for some servers the high negative correlation. The negative correlation (-0.65;-0.4) was observed on servers using the cache mechanism that explains very fast access for frequently used sets of answers (Figure 11).

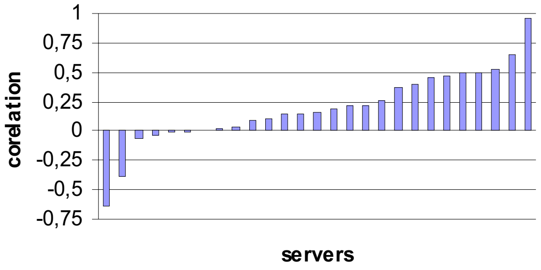


Figure 11. Correlation between the response time and the number of response

6.3. Test of the response-document accessibility (TEST2)

We verified response-document accessibility, size, and relevance. Our relevance-checking algorithm was based on verification of request-word existence in the answer-documents. We checked it in three-separate parts of the documents: in the title; the body and between the "meta" tags.

This test proved that about 93% answers arrived satisfactorily (Returns HTTP Code 200 OK). There was a small percentage of bad answers due to the incorrect question formulation (Returns HTTP from Code 4XX family): 4% Code 404; 3% Code 400 and Code 403 together. About 0.5% answers were wrong due to the Internal Server Error (Code 500).

7. FIM-MetaIndexer Meta-Search system statistics

Internet users used the FIM-MetaIndexer and the log was collected over several months. We used this log to calculate some statistics.

7.1. General statistics

Many answers returned by the questioned Search Engines were not pertinent. Our statistics (Figure 12) show that only 34% of the results returned by the Search Engines interrogated were of interest.

The FIM-MetaIndexer creates its own results-page from these answers. Our analysis shows that usage of the FIM-MetaIndexer decreases "information noise" significantly because it is able to eliminate the majority of the non-relevant answers.

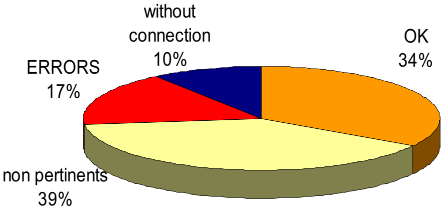


Figure 12. Response classification

7.2. Number of answers in a fixed period of time

These tests concerned the number of answers received during different time periods. The graph presented (Figure 13) shows that the majority of answers were provided within a period of time of under 15 seconds.

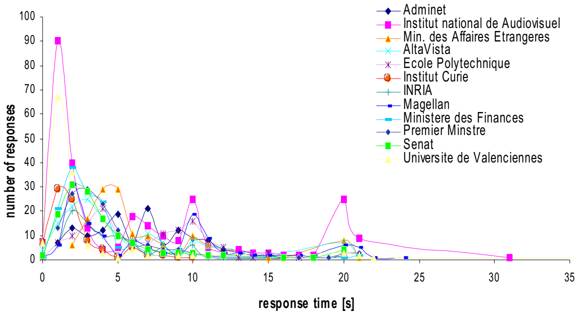


Figure 13. Number of responses received in the different time periods

We observed a significant growth in numbers of answers for the response time equal to 20 seconds. This can be explained by the fact that the default value of the "Search Server wait" time-out of FIM-MetaIndexer is equal to 20 seconds. Users rarely modified this parameter and the majority of responses slower than this time-out were stopped.

7.3. Usage of the FIM-MetaIndexer

The FIM-MetaIndexer is particularly queried during working hours (in France), with a strong increase in peak activity to about 20% in the beginning of the afternoon (Figure 14).

FIM-MetaIndexer usage decreases during the night. This would seems to be normal, as the FIM-MetaIndexer is purpose-built for the French Civil Service, and is therefore mainly used by the French community.

The statistics (Figure 14) show the user activity (the number of requests in percentage) and the response time of the FIM-MetaIndexer according to the time of the day. The FIM-MetaIndexer response time is more or less constant and equal to about 5s.

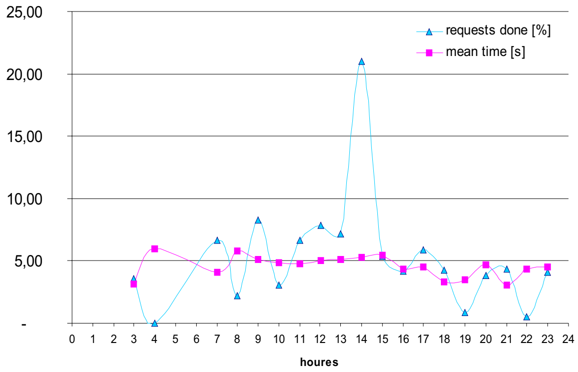


Figure 14. FIM-MetaIndexer response time

8. Conclusion and future work

The FIM-MetaIndexer is a Meta-Search Engine used to search for documents produced by the French Civil Service. Its first version was available in February 1998. Since, FIM-MetaIndexer has been used by various kinds of users and it has been a well-known and effectively used search tool.

Some new functions could be included in the FIM-MetaIndexer like a bi-directional co-operation module for Search Engines and an economic and technological survey module.

9. References

[1] A. H. Alsaffar, J. S. Deogun, V. V. Raghavan, and H. Sever. Concept-based retrieval with minimal term sets. In Z. W. Ras and A. Skowron, editors, *Foundations of Intelligent Systems: Eleventh Int'l Symposium, ISMIS'99 proceedings*, Warsaw, Poland, Jun. 1999, pp. 114--122.

[2] T. Bray,. Measuring the web: *Proceeding of Fifth International World Wide Web Conference,*1999.

[3] L. Chen,K. Sycara,. Webmate, A personal agent for browsing and searching: *Second International Conference on Autonomous Agents*, 1998, pp. 132-139. ACM SIGART, ACM Press.

[4] D. Green, The evolution of web searching: *Online Information Review,* 24(2), 2000, pp. 124-137.

[5] M. Henzinger, A. Heydon, and M. Najork Measuring index quality using random walks on the Web: *Proceeding of the _the International Word Wide Web Conference*, 1999, pp. 213-225.

[6] S. Lawrence, and G. Lee, Inquirus, the Neci meta search engine In: *Computer Networks and ISDN System.,* 1995, v30, pp. 1-7.

[7] S. Lawrence and G. Lee, Searching the world wide web: *Science, 280(5360),* 1998, pp. 38-6.

[8] A. Nicholson, A proposal for categorisation and nomenclature for web search tools: *Journal of Internet Cataloging*, 2000, 2(3/4), pp. 9-28.

[9] A. Sainul, Meta search engines: effective tool for information retrieval: *6th National Convention for Automation of Libraries in Education and Research (CALIBER 99),* Nagpur, India, 1999*,* pp. 362-369.

[10] E. Selberg and O. Etzioni, Multi-service Search and Comparison Using the MetaCrawler: *Proceeding of Fourth World Wide Web Conference*, Boston, 1995, MA.

[11] T. Stanley, Meta-search engines: where are the limits? In: *Proceeding of the Second International Online Information Meeting*, London, 1999, pp. 297-300.

[12] K. Wegrzyn, Etude et réalisation d'un robot pour la recherche d'information sur le Web: *Rapport DEA E/193/CRI*, Ecole des Mines de Paris and Université d'Evry-Val-d'Essone, 1996.

[13] K. Wegrzyn, Etude et réalisation d'un meta-indexeur pour la recherche sur le Web de documents produits par l'administration française: *Thesis A/339/CRI*, Ecole des Mines de Paris, 2001

[14] A. Weiss, The evolution of world wide web search tools: *Proceedings of the Second International Online Information Meeting*, London, 1998, pp. 289-295.

[15] J. T. Yao, Y.Y., Yao, Web-base Support Systems: *Proceedings of the Workshop on Applications, Products and Services of Web-based Support Systems (WSS'03),*Halifax, Canada,Oct 13, 2003, pp. 1-5.

[16] J. T. Yao, Y.Y., Yao, Web-base Information Retrieval Support Systems: building research tools for scientists in the new information age, *Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03)*, Halifax, Canada, Oct 13-17, 2003, pp. 570-573.

J.T. Yao, V.V. Raghvan, G.Y. Wang (Eds.): WSS'04, pp. 171-178, 2004.

Automatically Detecting Boolean Operations Supported by Search Engines, towards Search Engine Query Language Discovery

Zonghuan Wu, Dheerendranath Mundluru, Vijay V. Raghavan
The Center for Advanced Computer Studies
University of Louisiana at Lafayette
Lafayette, LA 70504-4330, USA
{zwu, dnm8925, raghavan}@cacs.louisiana.edu

Abstract

Each Web search engine provides query language through which it can communicate with its users and retrieve corresponding results to user queries. Supporting Boolean operations is a major characteristic of the query language. In this paper, we propose a novel, fully automatic, query probing based approach to identify what Boolean operations that are supported by a search engines and their corresponding syntaxes. Experiments show high effectiveness and efficiency. Along with this, we also provide a Web application called SE-BOSS (Search Engine Boolean Operation Scanning System) for interested users.

1. Introduction

There are hundreds of thousands of **s**earch **e**ngines (SEs) existing on the Web, most of which are *Deep Web* SEs [13] and contain high quality content that are not crawlable by SEs like Google.

MetaSearch Engine (MSE) is a system that provides unified access to several SEs that it knows how to communicate with. When an MSE receives a user query, it dispatches the query to selected SEs; results returned from SEs are then reorganized, merged and displayed to the user by the MSE [9]. The MSE approach provides convenient concurrent access to multiple SEs. More importantly, an MSE built on multiple *Deep Web* SEs provides a platform for users to search on tremendous amount of Web content that are not searchable through crawler based SEs such as Google. The state-of-the-art MSEs, such as Dogpile [1], Mamma [2], Kartoo [3], Profusion [4], Search.com [5], turbo10 [6] and others, are built on top of tens to up to *3,000* SEs.

Each SE provides an interface through which its users can input their queries to retrieve results. In most cases, as a Web information retrieval system, each SE has its query language model with operators and syntaxes (we will call them query patterns in the rest of this paper) through which a user can submit a more complex query than just keywords. The SE can understand queries in these patterns so that corresponding operations will be executed. Using Google as an example, it supports the Boolean operation of disjunction by using the operator "OR" between two keywords. However, due to the heterogeneity of SEs, different SEs may support different operations and/or use different symbols and syntaxes (i.e. different query patterns) to represent same operators. For example, the SE www.scrubtheweb.com supports disjunction by using the operator "|" between two keywords while "OR" is regarded as a stopword.

Knowing the query language model of SEs is important to SE users as well as to MSEs. By understanding SE query language models, users may resolve their confusions like, when they submit a query "Computer Science", whether the search engine explain the query as "Computer *AND* Science" or "Computer *OR* Science" or the phrase "Computer Science". Also, by applying the language model to their queries, users can send complex queries and use SEs more effectively. Similarly, when an MSE has knowledge of the query language model of its underlying SEs, it can effectively translate complex queries, in the MSE query language, into the language that its underlying SE uses and get more accurate results back. Moreover, by applying a customized combination of probe queries, especially by using Boolean operators, that an autonomous search engine supports, an MSE will have the capability to collect special representative information about SEs that would not otherwise be possible. For example, this additional information will be helpful to better determine rank position of documents in the retrieval output, compared to using only term distribution statistics and hyperlink-based popularity characteristics of documents in the retrieval output [17, 18].

However, not many present MSEs have the capabilities of discovering the query language model of its underlying SEs. There are a few MSEs that support complex queries such as Boolean operations and "*PHRASE* Search", either manual approaches or proprietary techniques are used to discover the operation syntaxes of SEs.

Manual and semi-automatic approaches are expensive and not scalable when the number of SEs that an MSE has increases. Toward solving the query language discovery problem, in this paper, we propose a query probing based, robust, highly effective and automatic approach, to detect the basic operations, including Boolean operations such as disjunction (OR), conjunction (AND) and negation (NOT) and a few other common operations that an SE may support, such as *PHRASE, FST and SND* operations, and discover their corresponding syntaxes.

In section 2, we introduce the background knowledge including query language model, query probing and SE connection. After we introduce and discuss prior relevant research on SE query language discovering in section 3, we describe our approach in section 4 and it is validated through experiments that are presented in section 5. Finally, we conclude in section 6.

2. Background

In this section, we briefly introduce basic terminology such as Customized MSE, Query-Probing and SE Connection that are needed for subsequent developments.

Customized MetaSearch Engine. Customized MSE systems such as SELEGO, Bright Planet's DQM and Turbo10 emerged recently [6, 8, 10, 12]. Their users are able to build own MSEs on demand by simply providing the url's of those SEs they wish to include in the MSE and the SE incorporation is automatic and instant. Users are then able to submit their queries to the new MSE right away.

If an MSE is able to detect the query language model of SEs in the process of automatic incorporation, MSE will also be able to handle complex queries such as Boolean operations, Phrase search and so on.

Query Probing. By sending queries with pre-selected terms to an autonomous SE and exploring the SE by analyzing the returned results, query-probing approach has been used to discover SE document language models [16], categorize SEs [15] and discover query language models [11, 14].

Search Engine Connection. When an MSE dispatches a query to an SE, it constructs and sends a query string in the format that the SE understands, and gets returned page from the SE. We call this the process of SE connection. In this paper, we need a program to conduct SE Connection process for the purpose of probing SEs. We use the SE Connection component of SELEGO [10, 12], which is a customized metasearch engine system that we built previously. Through this component, by giving the url of an SE and the query terms, a query string can be properly assembled and sent to the search engine and the corresponding result page can then be obtained.

3. Prior Research

To the best of our knowledge, the approach proposed in [11, 14] by Bergholz and B. Chidlovskii is the closest to our work. It uses query probing along with machine learning algorithms to identify query language features of Web data sources. The approach assumes that (1). SEs can be automatically connected (refer to section 2). (2). On the result page returned by an SE, *the number of returned documents that match the query* is reported by the SE and can be identified and extracted.

Authors defined a few query models such as 'A', 'A B', '"A B"', "+A +B", "A AND B", "A + B" and so on. When queries are submitted with an operator to be classified, by examining the matched numbers of documents in the result set with rules in Boolean algebra such as $|A \vee B| \geq |A|$, $|A \wedge B| \leq |A|$ and a few others, the corresponding operation could be derived. Authors defined features based on the matched document numbers and used a set of *22* predefined probe queries and a number of SEs to train the learner and generate classification rules that distinguish different semantics. A mean accuracy of *86%* for the set of most frequent operators has been reported [14]. However, the approach is not applicable to SEs that do not report the number of retrieved documents on result pages. In our survey (see section 4 for detail) on *182* SEs, it was found that 20% of them were not providing the match numbers. Moreover, it is not trivial to automatically identify and extract this number from result pages and the effectiveness was not discussed.

Among other approaches, one possible way to detect the operators is to analyze the "help page" of an SE, which normally provides information about how to use operators to construct complex queries. This process involves identifying links with captions like, "Help", "Tips" etc. at the SE interface page and, once such a link is found, an analysis of the page is done to find the information. This approach has apparent limitations. First, many SEs do not have such help pages; besides, it is found that the information on the help pages of some SEs are sometimes obsolete, incomplete, or incorrect [11]. Second, it is difficult to automatically locate the help pages effectively.

Another approach is to download and analyze the result documents returned by SEs. The drawback is that it is very time consuming to download documents and then to parse it to detect the presence of query terms. Also, distinguishing the valid document link to download from other links (such as links for service pages, advertisements) is a problem that may affect the effectiveness of the analysis.

To overcome the limitations that above approaches have, we propose an efficient approach based on query probing and link analysis for automatically discovering the query language features of an SE. As reported in section 5, our approach showed an accuracy of over 97% in correctly detecting the operators.

4. Proposed Methodology

Before we get into details, we first define few terms that we use for the purpose of simplifying the description of our approach:

Definitions 4.1: *Impossible Query Term* and *Valid Query Term*:

When a term *t* is submitted to an SE *s* as a query, δ results are returned,

t is an *Impossible Query Term* to *s* when $\delta = 0$.The query is an *impossible query*.

t is a *Valid Query Term* to *s* when $\delta > 0$. The query is a *valid query*.

Definitions 4.2: *Impossible Query Page* and *Valid Query Page*:

The result page (which displays 0 results for the query) that an SE returns for a given impossible query is an *impossible query page*.

The result page (which displays δ ($\delta > 0$) results for the query) that an SE returns for a given valid query is a *valid query page*.

Note that though most SEs return multiple pages of results for a valid query, in this paper, we only need the first result page and we refer it as the result page.

Definitions 4.3 *Static Links* and *Dynamic Links*:

On *n (n>0)* different html pages, a link is a *Static Link* if it appears on each of the *n* pages with the same captions or urls. Otherwise, it is a *Dynamic Link*. An example for static link in SE result pages can be a link with caption like "Home", "Products", "Services", or "Help" etc. that usually points to a fixed url. However, sometimes, the url may contain a unique session id. An example of a dynamic link could be a link to a commercial advertisement displayed which change periodically or with every request. Table 1 gives all scenarios of static and dynamic links.

Table 1. Static and Dynamic Links

URL	Caption	Link Type
Same	Same	Static
Different	Same	Static
Same	Different	Static
Different	Different	Dynamic

Just like the approaches in [11, 14], we assume that the SEs can be programmatically connected. However, unlike their approach, instead of using pre-selected terms for the probe queries, we dynamically generate two special query terms. One is an *Impossible Query Term* and the other is a *Valid Query Term*. They are connected by different operators to formulate a set of probe queries to discover the supported operations of any SE with their corresponding query patterns. Figure 1 illustrates our three-step approach to automatically detect the supported Boolean operations of an SE.

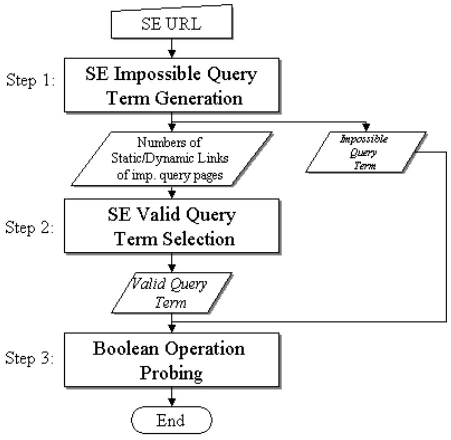

SE URL
Step 1:
SE Impossible Query Term Generation
Numbers of Static/Dynamic Links of imp. query pages
Impossible Query Term
Step 2:
SE Valid Query Term Selection
Valid Query Term
Step 3:
Boolean Operation Probing
End

Figure 1. Steps for query language detection

In the rest of this section, we explain the three steps from Figure 1 in detail.

Step 1: Impossible Query Generation

As specified above, when an impossible query term is submitted, an SE always returns *0* results. In our approach, we simply create terms like 'AnIm2345possibleQuery' that are extremely unlikely to be indexed by any SE in practice. When probed with such a query, a given SE would usually return a page with some statement saying that no results were found for the query. In addition, such a page usually has a few links, which can be either static (appear in all impossible query pages) or dynamic (appear in the specific impossible query page). The numbers of static and dynamic links are used as a feature to identify valid query terms in step 2 and to detect query operations in step 3.

By probing an SE twice with two different impossible query terms, we can get the number of static links by extracting the links that have common urls or captions for both the queries. The links left on both pages are dynamic links. If the numbers of the dynamic links on both pages are slightly different, we use their average.

Step 2: Valid Query Generation

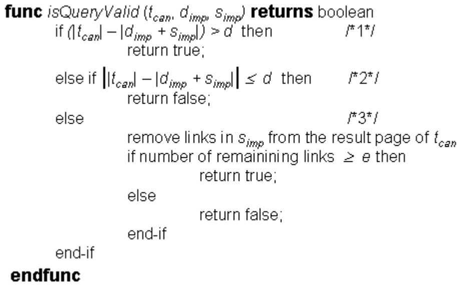
After collecting impossible query term and related information, the next step is to find a valid query term.

For a valid query term, an SE always returns more than *0* results on the returned page (valid query page). Usually, a valid query page has significant difference from an impossible query page. Assuming this is true; we generate a list of candidate valid query terms, submit them to the SE, and evaluate the returned pages by using the total number of links. For some candidate valid query term, if the difference between the total number of unique links in its result page and the numbers that we collected from impossible query pages in step 1 is above certain threshold, which means there is significant difference between the two pages, we select it as a valid query term. Otherwise, we just discard this term. This step can be further divided into two parts: in step 2.1, the candidate valid query terms are generated; in step 2.2, the heuristic logic for selecting a valid query term is provided. The candidate terms are checked by this logic one by one, until a valid query term is found.

2.1. Generating candidate valid query terms. We generate two lists of query terms to be candidate valid query terms. Usually, the SE interface page provides descriptions that relates to the content of the SE. For example, the description may include important terms such as the name of a company and its products/services that may frequently appear on the company's SE interface page as well as in many Web pages that this SE indexes. By querying the SE with such important terms, the SE is likely to return results; so these terms are likely to be valid query terms. Based on this observation, we made an assumption that the more frequent a term appears on the interface, the more likely that it is a valid query term; thus, we create the first candidate term list that contains 10 most frequent terms extracted from the SE's interface page (stop words are removed). However, a few SEs (e.g. http://www.metor.com) have extremely simple interface that do not have enough good terms, so we also collect a set of terms that are very generic to construct the second candidate term list. Our assumption is that these terms are so generic that some of them will always be found in document collections of most, if not all SEs. We collect candidate terms from the homepage of CompletePlanet's [7] SE directory that classify *70,000+* SEs into *42* categories. We used all *54* terms that appear in these categories as the other set of candidate valid query terms.

We start with trying to probe SE by using the terms from the first list; if none of them is identified as a valid query term, we then try the terms in the second list. The following section explains how a valid query term is selected.

2.2. Valid query term selection. Figure 2 shows the rules to check whether a candidate term is indeed a valid query term.



```
func isQueryValid ($t_{can}$, $d_{imp}$, $s_{imp}$) returns boolean
    if ($|t_{can}| - |d_{imp} + s_{imp}|$) > $d$ then            /*1*/
        return true;
    else if $\left||t_{can}| - |d_{imp} + s_{imp}|\right| \leq d$ then     /*2*/
        return false;
    else                                                      /*3*/
        remove links in $s_{imp}$ from the result page of $t_{can}$
        if number of remainining links $\geq e$ then
            return true;
        else
            return false;
        end-if
    end-if
endfunc
```

Figure 2. The heuristic logic for identifying a valid query term

In Figure 2, respectively, d_{imp} and s_{imp} indicate dynamic links and static links on the impossible query page*s* that we got previously (see step 1); $|d_{imp} + s_{imp}|$ is the number of all links on a impossible page. t_{can} is the candidate valid query term. For a given SE, when t_{can} is submitted, the returned result page, called *Candidate Query Page*, has $|t_{can}|$ unique links. Let d be a threshold used to determine whether the candidate query page is significantly different from an impossible query page. At /*1*/, the first if-condition tests whether the total number of links in the candidate query page is significantly greater (difference > d) than the sum of the total number of links in impossible query page (i.e., the sum of dynamic links and static links in the impossible query page) and threshold d. If yes, it implies the page is different from the impossible query page and therefore is a valid query page. Note that d = 7, was found to be good in our experiments. At /*2*/, when the first if-condition is not satisfied, but the difference between the total number of links in the candidate query page and the total number of links in the impossible query page is less than or equal to d (i.e. the number of results retrieved is between 0 and d), then we discard the term because of the insignificant difference between the candidate query page and impossible query page.

Otherwise, at /*3*/, there are two possibilities left: A) The candidate query is a valid query page if it includes 0 or only a subset of all the static links displayed. B) The candidate query page is an *error page* that generally has 0 or very few links unrelated to the user query. This might happen when the query contains characters that may not be recognizable by the SE. We still consider this as a special form of impossible query. In order to differentiate the two cases, from all the links extracted from the candidate query page, we remove all the static links that are found in s_{imp} and then check whether the total number of the remaining links is greater than a threshold *e*. If yes, we regard t_{can} as a valid query term. Otherwise, we discard t_{can} and try with a new candidate term.

Step 3: Query Operation Detection

To conduct the detection, we've surveyed *182* SEs including both general purpose and specialty SEs that we randomly picked from CompletePlanet's [7] website. As shown in table 2, we found that *89%* of these SEs support *AND*, *71%* support OR and 60% support NOT.
Note that a few SEs provide "Advanced Search" interfaces at which a complex form is provided for users to customize queries (eg. Google's Advanced Search interface is located at: http://www.google.com/advanced_search?hl=en)However, in this paper, we only deal with query language models of SE simple interfaces at which there is only one text field for users to input queries.

Table 2. Search engines and their supported Boolean operations

Operation	Number of SEs	Percentage
AND	162	89%
OR	132	71%
NOT	110	60%

In this step, based on the survey, we automatically detect which of the above three operations are supported by an SE and how the query patterns are represented. Also, with slight extension, we also detect *FST* (first query term should appear in all retrieved documents) and *SND* (second query term should appear in all retrieved documents) and *PHRASE* (all retrieved documents should contain all the specified query terms in the given order) operations.
In this step, the valid query term and the impossible query term found in previous steps are used to formulate probe queries to discover the operations supported by a given SE. Based on our survey, we summarized 15 query patterns that are used to generate all the probe queries. Table 3 displays these query patterns and their possible semantics.

Table 3. Possible Operators and Query Patterns

Operations Query Patterns	AND	OR	NOT	FST	SND	PHRASE
t_1 t_2 (default op)	Y	Y		Y	Y	Y
$+t_1$ $+t_2$,	Y					
t_1 ***AND*** t_2, t_1 ***and*** t_2 t_1 ***And*** t_2, $t_1 + t_2$	Y				Y	
t_1 ***OR*** t_2, t_1 ***or*** t_2, $t_1 \mid t_2$, $t_1 \mid\mid t_2$		Y				
t_1 ***AND NOT*** t_2, t_1 ***ANDNOT*** t_2, $t_1 - t_2$, t_1 ***NOT*** t_2			Y			
"t1 t2"						Y

For example, for the conjunction operation, possible patterns are 't_1 t_2', '$+t_1$ $+t_2$', 't_1 **and** t_2', 't_1 **And** t_2', 't_1 **AND** t_2' and '$t_1 + t_2$'. Also for simplification, we omitted the possible semantics when any operator specified can be considered as a stopword or a literal.
In our method, we first detect the operation of the pattern 't_1 t_2', which we call it as the "*default operation*" of an SE. Based on the default operation, which can be *AND, OR, FST SND* or *PHRASE,* we then detect other operations through probing as described in Figure 3.

```
if default operation is OR then
        Detect the support for AND, SND, NOT
else if default operation is AND then
        Detect the support for OR, NOT, SND
else if default operation is FST then
        Detect the support for OR, AND, SND
else if  default operation is SND then
        Detect the support for OR, AND, NOT
end-if
```

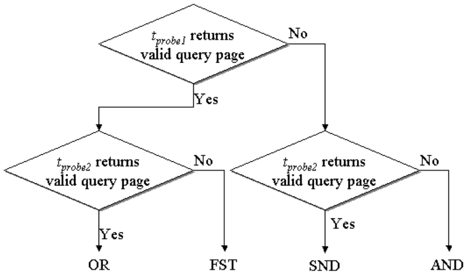
Figure 3. Boolean operation detection for search engines

Please note that the *FST* operation can only appear as a default operation (See Table 3). Also note that one flaw of the query probing using valid query term and impossible query term is that it is not able to detect the support for *NOT* operation when the default operation is *FST*. Another issue that needs to be clarified is that, in this algorithm, we consider *PHRASE* as a special form of *AND.* To further differentiate it from the *AND* operation, several result document samples needed to be taken. We will not discuss it in this paper since we found that, in our survey, it is very rare (One out of *182* SEs) that the default operation of a SE is *PHRASE*.
The rest of the section describes this step in detail.

3.1. Detecting the Default Operation. As it can be seen from table 3, 't_1 t_2' can mean five kinds of operations for different SEs (*AND, OR*, *FST*, *SND*, or *PHRASE*).

Figure 4 shows the rules to detect the default operator of an SE.

As shown in Table 3, other than *default* query pattern, only *AND* query patterns can be semantically equivalent to *SND*. For example, if 't_{vld} **AND** t_{imp}' returns an impossible query page but if 't_{imp} **AND** t_{vld}' returns a valid query page, then we conclude that query pattern t_1 **AND** t_2 is for *SND* operation. Thus, let t_{probe1}, t_{probe2} be the default probe queries obtained by binding t_1 to t_{vld} and t_2 to t_{imp} where t_{probe1} = '$t_{vld}\, t_{imp}$' and t_{probe2} = '$t_{imp}\ t_{vld}$'. If both t_{probe1} and t_{probe2} return valid query pages, then the default operation is *OR*. If at least one result page returned by t_{probe1} and t_{probe2} is a valid query page, the default operation should be either *FST* or *SND*. If both t_{probe1} and t_{probe2} return impossible query pages, then the default operation should be *AND*. The algorithm flowchart is shown in Figure 4. In Figure 4, to find if a result page is valid query page or not, we re-use the function defined in Figure 2.



t_{probe1} returns valid query page
No
Yes
t_{probe2} returns valid query page
No
t_{probe2} returns valid query page
No
Yes
Yes
OR
FST
SND
AND

Figure 4. Default operation detection

3.2. Detecting the support for OR, AND operations. As shown in Figure 3, we detect the support for *OR* operation with its possible patterns only if the default operator is either *AND* or *FST* or *SND* search. Similarly, we detect the support for *AND* operation and its entire different syntaxes only if the default operation is either *OR* or *FST* or *SND*. Different patterns of *OR* queries used are: t_{vld} **OR** t_{imp}, t_{vld} **or** t_{imp}, $t_{vld} \mid t_{imp}$, and $t_{vld} \parallel t_{imp}$. Similarly the different patterns of *AND* queries used are: $+t_{vld}$ $+t_{imp}$, t_{vld} **AND** t_{imp}, t_{vld} **and** t_{imp}, $t_{vld} + t_{imp}$, and t_{vld} **And** t_{imp}. The rules to find the support for the both *OR, AND* operation patterns is similar to the rules shown in Figure 4. The only difference is that we need to apply the query patterns corresponding to AND and OR operations.

3.3. Detecting the support for NOT operation. We form different probe queries to detect the *NOT* operation support based on the supported default operation. For example, if the default operator is *AND*, different patterns of *NOT* queries used are: t_{vld} - t_{imp}, t_{vld} **AND NOT** t_{imp}, t_{vld}**NOT** t_{imp}, and t_{vld} **ANDNOT** t_{imp}. Note that t_{vld} always appears in front of t_{imp}. However, if the default operation is *OR*, different patterns of *NOT* operation used are: t_{imp} $-t_{vld}$, t_{imp} **AND NOT** t_{vld}, t_{imp} **NOT** t_{vld}, and t_{imp} **ANDNOT** t_{vld}. Note the order of appearances of t_{vld} and t_{imp} are reversed. For example, if the default operation is known as to be *AND*, we send probe query 't_{vld} $-t_{imp}$' and if a valid query page is returned, it indicates that *NOT* is supported by pattern t_1 $-t_2$. If the default operation is known to be *OR*, we send probe query 't_{imp} $-t_{vld}$' and if an impossible query page is returned, it implies the *NOT* operation has been executed.

5. Experimental Results

We have tested our algorithm on *128* SEs, which includes both general purpose and specialty SEs from various domains. Most of the specialty SE's have been taken from CompletePlanet's SE directory [7]. We also randomly collected a few SEs from other sources, including SEs incorporated by profusion.com, search.com, turbo10.com. Following are the different domains from which SE's are included in our test collection:

General Purpose SE's:	*21*
Sports/Basketball:	*10*
Business/Small Business:	*22*
Health/Cancer:	*16*
MetaSearch Engines:	*5*
SE's randomly taken from other sources:	*54*
Total:	*128*

Please note that, to avoid bias of experimental results, these SEs are selected independently of the search engines used in our survey.

5.1. Impossible Query Term Generation

Our impossible query term generator uses very simple method to generate dynamic impossible query terms such as 'AnIm2376possibleQuery' in which the value *2376* is randomly generated. We found that the approach is so effective that, in all cases of our experiment, it successfully generates impossible queries.

5.2. Valid Query Term Selection

The success rate for finding a valid query term from the two lists of candidate valid query terms has been over *99.21%.* The only failed case in our experiments is an SE that only returned at most 5 results (which is < *d*) for all the queries on its result page (for most of other SEs, the number is usually *10* or more).

Another important question is how efficiently a valid query term can be generated from the two lists of candidate valid query terms. For this we tracked the generation of valid query terms in our experiment and found that, on average, only *1.132* probe queries were used to select a valid query term. Therefore it indicates that the approach to automatically select a valid query term is not only accurate, but is also very efficient.

5.3. Boolean Operation and Query Pattern Detection

Since we failed to generate valid query term for one of the *128* SEs, we detected the Boolean operations and their corresponding query patterns on the remaining *127* SEs. To validate our results, we manually inspected all SEs and compared the results with programmatically generated results.

Overall, our system showed an accuracy of *97.63%* i.e. out of *127* SE's, *124* SE's were correctly classified. This accuracy is specified based on consideration of all the operation patterns used in the system i.e. if there is at least one operator which was wrongly detected for a particular SE, we considered that the system failed to classify this particular SE as a whole. For the 3 failed SE's, the default operation was wrongly classified. It also results in the failure of detecting other operations since they rely on the detection of default operation.

5.4. Efficiency Analysis of the approach

The number of probe queries used for an SE is the key factor for the time involved in detecting an SE's query language features. As shown in section 5.2, the number of probe queries used in selecting a valid query term is on average only 1.132. 2 probe queries are used to get 2 impossible query pages for detecting the static and dynamic links. 2 probe queries are needed for detecting default operation and 4 probe queries are needed for *NOT* operation detection. In addition, if default operation is *OR*, we need 5 *AND* probe queries whereas we need 4 *OR* queries when default operation is *AND*. Therefore, in total, 13.132 or 14.132 probe queries are needed to detect the query patterns of all three basic Boolean operations (*AND, OR, NOT*), depending on whether the default operation is *AND* or OR. 4 more probe queries are needed if *SND* is needed to be detected. In other cases of default operations (FST and SND), the number of probe queries used is just slightly different from what has been used in the case when default is either AND or OR.

6. Conclusions and Future Work

In this paper, we have proposed a novel approach to detect the basic Boolean operations that an SE supports. This highly effective and efficient approach is based on a series of simple and robust techniques such as impossible query generation, valid query generation and link analysis. By a comprehensive survey, followed by experiments on 128 SE's with a set of most commonly used operators, we achieved a very high overall accuracy of over 97%. With SE Boolean operation detection, MSEs will be capable of dispatching more accurate queries to search engines; it also provides researchers a tool for analyzing search engines in large scale.

We have set up a Web application called SE-BOSS, which detects the various operations and their query patterns supported by a SE given its URL, for experiment/demo purposes at: http://lincstaff2.cacs.louisiana.edu:8080/metasearch/SubmitURL.

Finally, we list a few directions for future work:

1. **Large Scale Test:** Current testing on 128 SE's showed high effectiveness of the algorithm heuristics. However, due to the highly dynamic Web environment, we plan to enlarge the test bed and validate our method.
2. **Improve Valid Query Generation and Valid Query Page Identification:** The four cases of false detection were caused due to the inability of correctly identifying valid query pages as the current approach of using the numbers of dynamic and static links is not able to handle these particular SEs, though the approach is simple and effective in most of the cases. More robust and sophisticated features, such as the structure of the result page, may need to be studied and applied to further improve the effectiveness.
3. **Detecting the support of more complex Boolean operations.** In this paper, we dealt with the simple Boolean operations. It is still an open question to find out how heterogeneous SEs support complex operations such as ((t_1 AND t_2) OR t_3 NOT t_4). We plan to extend our work to address this issue too.
4. **Automatically discover operations of advanced search interfaces**. Our current work deals with only basic search interfaces. However, many SEs have advanced interfaces on which a complex html form is provided, which can be customized to organize complex queries. It would be interesting to be able to automatically discover the query language features of such SE advanced interfaces.

Acknowledgements: This work is supported in part by the IT Initiative of the State of Louisiana to Lafayette.

7. References

[1] Dogpile. http://www.dogpile.com/

[2] Mamma. http://www.mamma.com/

[3] KartOO. http://www.kartoo.com/

[4] Profusion. http://www.profusion.com/

[5] Search.com. http://www.search.com/

[6] Turbo 10. http://www.turbo10.com/.

[7] CompletePlanet, http://www.completeplanet.com.

[8] BrightPlanet, http://www.brightplanet.com/.

[9] W. Meng, C. Yu, K. Liu. Building Efficient and Effective Metasearch Engines. ACM Computing Surveys, Vol. 34, No. 1, March 2002, pp.48-89.

[10] http://www.selego.com, Creating MetaSearch Engines On-Demand.

[11] Bergholz, B. Chidlovskii. Using query probing to identify query language features on the Web. In Proceedings of the SIGIR 2003 Workshop on Distributed Information Retrieval, Toronto, Canada, August 2003.

[12] Zonghuan Wu, Vijay Raghavan, Weiyi Meng, Hai He, Clement Yu, and Chun Du. Creating Customized Metasearch Engines on Demand Using SE-LEGO. In Proceedings of Fourth International Conference on Web-Age Information Management (WAIM'03), Demo paper, pp.503-505, Chengdu, China, August 2003.

[13] Bergman, M. The Deep Web: Surfacing Hidden Value. Journal of Electronic Publishing, 7(1), 2001.

[14] Bergholz, B. Chidlovskii. Learning Query Languages of Web Interfaces. In Proceedings of the 2004 ACM Symposium on Applied Computing: 1114 – 1121.

[15] P. G. Ipeirotis, L. Gravano, and M. Sahami. Probe, count, and classify: Categorizing hidden-web databases. In Proc. ACM SIGMOD Conf., pp. 67-78, Santa Barbara, CA, USA, May 2001.

[16] J. P. Callan, M. Connell, and A. Du. Automatic discovery of language models for text databases. In *Proc.* ACM SIGMOD Conf., pp. 479-490, June 1999.

[17] M. Kim, V. V. Raghavan, and J. S. Deogun. Concept based retrieval using generalized retrieval functions. Fundamenta Informaticae, 47(1-2):119--135, 2001.

[18] A. H. Alsaffar, J. S. Deogun, V. V. Raghavan, and H. Sever. Enhancing concept-based retrieval based on minimal term sets. J. of Intelligent Information Systems, 14(2-3):155--173, 2000.

Table of Contents WSS2003

Index of Authors

Published in Collaboration with

Beijing University of Technology

By

Saint Mary's University

Halifax, Nova Scotia, Canada

Department of Mathematics and Computing Science
Technical Report Number: 2004-05 September, 2004
ISBN: 0-9734039-6-9