The Proceedings of the ISCA 15th International Conference on **COMPUTER APPLICATIONS IN INDUSTRY AND ENGINEERING,** held at the Clarion Hotel Bay View, San Diego, California, USA, November 7-9, 2002.

SPONSOR: The International Society for Computers and Their Applications - ISCA

* **All published papers have been peer re**
** **This publication is abstracted and index**

ISCA, 975 Walnut Street, Suie 132, Cary, NC 27615 · Ph: (919) 467-5559 · Fax: (919) 467-3430
Isca@ipass.net http://www.isca-hq.org

INTERNATIONAL SOCIETY FOR COMPUTERS AND THEIR APPLICATIONS

15th International Conference on Computer Applications in Industry and Engineering (CAINE-2002)

November 7-9, 2002
Clarion Hotel Bay View, San Diego, California USA

TECHNICAL PAPER INDEX

COMPUTER ARCHITECTURE / VLSI I

INTELLIGENT SYSTEMS I

INTERNET AND WEB-BASED SYSTEMS / APPLICATIONS I

once a successful admin login is identified. Admin ID is used as a session variable for access control.

Page M20.1: Pending Approvals
This page shows three tables of new requests (new group, new user and new experiment request) that are seeking for approvals. Action buttons are provided for administrator to make decision and email the decision. A detailed "sample view" with option buttons/check boxes is provided in new experiment section for administrator to update every sample's stages. All samples fall into one of following five categories: not received; pending; requested; in process; completed. Logically, this page is one of the most complex ones to implement in the system.

Page M21, M22 and M23: User Management, Group Management, Request Management
These pages are provided to add and modify user, group and experiment request information.

Page M24: File Management
The main function of this page is to upload result files of a specific request from a PC to web server. Uploaded files are stored in a specific directory according to User ID and Group ID. If upload is successful, email will be sent to corresponding user and PI to notify the availability of result files. We only store the file path in the database and allow users to download the file from web server directly with the given file path. This implementation approach is used due to the large size of result file of microarray experiments.

Page M25: Gene Chip Inventory
Page M26: Summary Report

4. CONCLUSION

We developed a web-based microarray experiment management system for UCD Cancer Research Center to save time, cost, and reduce management errors. We used a combination of PHP, MySQL, Apache and Linux open source software development tools [1, 7, 9, 10] to develop the first known open source Microarray experiment management system -- an unique contribution to open source in Bioinformatics. Open source, open data, open service movement [2] in bioinformatics will continue to foster greater cooperative bioinformatics services in the future. In this project, we benefit not only from open source tools such as Linux, Apache, PHP, and MySQL, but also from open source development products like pSlash [6] and LOCal [3]. We firmly believe that every contribution back to the open source community will encourage more contribution to flourish.

In addition to solving a practical problem in microarray experiment management, we also gained valuable experience in working with biologists and some insight on Bioinformatics education program development. In keeping up with the rapid growth of genomic data, biologists and computer scientists are finding themselves working more side by side these days. For this kind of projects to work, productive communication between computer and life scientists is becoming more important than ever. From system requirements to prototype design, we had many face-to-face meetings and email discussions with UCD collaborators. We are convinced that one of the important components in any Bioinformatics program should be learning through real world projects and solving a practical problem at the same time –just like what we did in this project.

Our future work in collaboration with UCD Cancer Research Center includes design and develop more web based bioinformatics systems such as tissue sample database, clinic trials database, and data integration of available data to keep track of cancer survivors and assist doctors in profiling hospital patients. All these efforts will work towards more productive genomic research and better cancer medical diagnosis and treatment.

REFERENCES

[1] Apache, http://www.apache.org.
[2] Ewan Birney, "Open Source Bioinformatics", a keynote presentation at O'Reilly Bioinformatics Technology Conference, January, 2002, Tucson, USA, www.conference.oreilly.com/biocon.
[3] Kevin W. Eliceiri, Vincent P. Chu, and John G. White, "LOCal: A Flexible Web-Based Microscope Reservation System", BioTechniques, Vol. 31, No. 4, October 2001, page 760-765.
[4] Cynthia Gibas, Per Jambeck, "Developing Bioinformatics Computer Skills", O'REILLY, 2001.
[5] GNU General Public License, http://www.gnu.org/copyleft/gpl.html
[6] D. Leung, pSlash: A Web Content Management System, Master degree thesis, Department of Computer Science, California State University, Sacramento, August 2001, www.pslash.com.
[7] Linux, http://www.linux.org.
[8] Microarry http://genome-www.stanford.edu/microarray.
[9] MySQL, http://www.mysql.com.
[10] PHP scripting language, http://www.php.net.
[11] O'REILLY Open Source Convention, www.conferences.oreilly.com/oscon.
[12] Luke Welling, Laura Thomson, "PHP and MySQL Web Development", SAMS, 2001.

Visualization of Web Retrieval Results

Philip Johnson, Beomjin Kim, Lwin Moe, David Erbach
Computer Science Department
Indiana University - Purdue University
Fort Wayne, IN, 46805, U.S.A.
johnpr01@holmes.ipfw.edu

Abstract

Information visualization can be an effective solution to support users' search tasks to obtain desired insight from an expansive information workspace. This research has developed a visualization methodology in which Web retrieval results are transformed into intuitive visual representations. This visualization system shows distribution pattern of search terms, structural overview of the Web pages, and the characteristics of the Web resources simultaneously. Through this visual expression, users can instantly recognize related resources from a collection of possibilities, but can also pinpoint specific segments of a resource instead of having to scan through all of it. An experimental study has been conducted to compare this visualization method to traditional commercial search engines, and subjects have shown a high degree of satisfaction using the visual approach in presenting the search results.

1 INTRODUCTION

The Internet search engine has served as a major directory to assist users' navigation in finding desired information from an unbounded digital workspace. In response to users' search requests, the majority of commercial search engines provide index pages. The chief characteristics of each Web page in the index are described by a simple summary produced either manually or by text generation. The manual summarization approach relies on the synopsis that is provided either by the authors of Web pages or readers at the institution providing the search engine. The automatic summarization approach works by extracting focused segments of the web page, using language generation, or applying artificial intelligence [1]. The summary-based approach cannot fully deliver the contents of a Web page through the short description and it also ignores the structure of the contents. Especially when the Web document is lengthy, the summary usually concentrates on the main topic, while ignoring a subtopic that could include the users' interests. In addition, this approach has limitations in providing distributional information of the search terms, which is an important component in the search process.

Furthermore, the retrieved index page commonly includes hyperlinks to expired Web pages and unreachable Web sites that are provided to users as a search result without any notice. The exclusion of such information is a deficiency in current commercial search engines [2,3].

Visualization is an effective way to present search results in an intuitive format for easy understanding of the information via perceptual cognition. Studies have shown that an appropriate visualization technique can increase users' search speeds and accuracies in judging the relevancy of documents [4]. WebBook, developed by Card et al. (1996), is a 3D virtual interface tool which allows users to keep compressed, multiple Web pages but also allow users to examine specific portions of the Web pages [5]. This technique, although very useful, still does not address the limitations in showing detailed search terms distributions, and users are still required to read or scan a large amount text before they may find the information they are searching for.

This research has developed a Search Retrieval Visualization system (SRV), which transforms the results returned by a search engine into an iconic representation expressing the distributional pattern of search terms and a structural overview of the Web pages simultaneously for intuitive location of relevant information. The SRV also provides users information about the accessibility of relevant Web sites and characteristics of the Web resources. A pioneer study has been conducted to compare the efficiency of the SRV system over a commercial summary-based search engine (SB). The results of the experiment show that the participants using the SRV system perform better, while maintaining the accuracy, than the participants using the SB system with respect to the relevancy of the web contents to the search terms.

2 METHODOLOGY

Three groups of procedures in the SRV system transform the search results to an abstracted visual representation. The collection procedure retrieves and classifies the resources related to the search terms that are located in the dispersed Web servers. The HTML parser extracts the text information in the Web page and a series of filtering processes alters the text into a simpler form to speed up the analysis of search terms. Finally, a visualization algorithm transforms the distributional pattern of search terms into a visual abstraction by using three attributes: color, intensity, and size.

2.1 Collection

With the given search terms, a meta-query is generated, which is sent to a commercial search engine through a HTTP connection. The search engine generates an index page that includes URLs to the relevant resources on the Internet. The main goal of this research is not ranking the relevant information, but presenting the search results to users in a more intuitive fashion. Thus, the SRV system uses the ranked order from the existing search engine. The HTML parser extracts the URLs to the related resources in the retrieved index page. Multiple socket connections are set up to each of the Web servers pointed to by the URLs.

With the rapid change of online information, the hyperlinks of the index page commonly point to expired Web pages or unreachable Web servers. These communication errors are detected from the information in HTTP header and is recorded and later reported to users through a distinctive iconic representation. Unlike the existing search results, by signaling communication errors in advance, users can avoid visiting broken links. This procedure also recognizes hyperlinks to non-HTML resources, for example: Adobe files, Microsoft Office™ formats, and Macromedia Flash™ files. The hyperlinks to the non-HTML format are mapped to exclusive icons and displayed on the SRV window.

The HTML parser converts the HTML document into a strictly text document. To present the distributional pattern of search terms in association with the structure of the Web page, the text information is partitioned into multiple text blocks. The segment markers - tags in HTML script - are utilized for the text partitioning. The headings and itemized sentences in the Web page are also clustered together to form a text block. Based on the layout of the text blocks, the original Web page is also segmented as a text-with-html segment so that it can be presentable to users in its original form whenever the user finds it necessary.

Some authors provide an index page, instead of providing detailed information relevant to the search terms, which includes various hyperlinks to other resources. This anchor page is a useful directory that allows users further navigation to find desired information by following the offered hyperlinks. The users need another investigation of the Web page to locate the related links, because, commonly, the anchor page supplies links to non-related information. Providing an outline of the anchor page will be beneficial in users' search procedures. The SRV system investigates the anchor pages in advance, and presents the anchor page in a more informative fashion. When a Web page is found to have an excessive number of hyperlinks, compared to text not directly related to the hyperlink, then the web page is categorized as an anchor page. Whenever this occurs, the user is informed about the distribution of related and non-related links in the page. Through this outline, the users can make their decision about the worthiness of visiting the anchor page in advance.

2.2 Parsing

A series of processes are applied to the segmented text block that transforms the text into a basic form. The tokenization process breaks the text block into a set of words from which various stop-words, that are not used to facilitate the content of the document's connotation, are removed. The stemming procedure is applied to the remaining tokens, which conflates groups of words into a stem word. These filtering procedures reduce the complexity of the text information, which will be beneficial to the visualization process that requires a significant amount of processing time.

2.3 Visualization

A color-coding algorithm has been applied to visualize the search terms' frequencies and relationships, and the dimensional attributes deliver the structural information of the Web page to users [6]. To generate color-codes that show the distributional pattern of search terms', a primary color (RGB color model) is assigned to a subset (S) of search terms. For each text block, B_i, the frequencies of the subset, $(F^S)_i$, are computed where search terms in the S are processed as an OR relationship.

$$(F^S)_i = (\sum_{j=1}^{n} t_j^S)_i$$

where t_j^S is the frequency of a search term in the i^{th} text block B_i, S is one a subset of search terms, $S \rightarrow \{R, G, B\}$.

The ratio $(R^S)_i = (F^S)_i / F_i$, where F_i is the number of words in a text block, is mapped to an intensity scale of sixteen different intensities to determine the intensity of the subset's color. The higher ratio, $(R^S)_i$, maps to the brighter intensity of the corresponding color and the $(R^S)_i$ greater than a predefined threshold value is mapped to the brightest hardware intensity of the color. The iteration of frequency-to-intensity mapping for each S to B_i defines the three intensities of the primary colors, I_i^R, I_i^G and I_i^B, where ^{R}, ^{G} and ^{B} represent the primary colors, red, green, and blue. By blending I_i^R, I_i^G and I_i^B together, a color code Ω_i, will represent the search terms distribution in B_i. Through the frequency-to-intensity mapping, a text block that has higher concentration of the search terms' occurrences will be projected as a brighter color and the color code also shows the search terms' distributive tendency within B_i. The Ω_i will be painted onto an icon whose width is determined by $f(F_i) = F_i / \lambda$ where λ is a scaling factor between the number of words and pixels. Concatenation of icons for every B_i forms a stripe of icons that correspond to a visual representation of a Web page.

The SRV system provides interactive browsing tools to assist users in further navigation of the Web page. When a user requires a closer investigation of a segment of Web page before making a decision, by pressing the right mouse button, the corresponding web page segment is displayed at the bottom of the interface.

3 EXPERIMENT

To investigate the effectiveness of the SRV method, the application's performance was compared to a conventional summary-based search engine. In this experiment, the swiftness and correctness of finding relevant information amongst the retrieved Web pages were compared between the two applications.

3.1 Experimental Environments

Ten participants who were native English speakers, had no difficulties using the computers, no problems in color perception, and experience using some type of Internet search application, participated in this pioneer experiment.

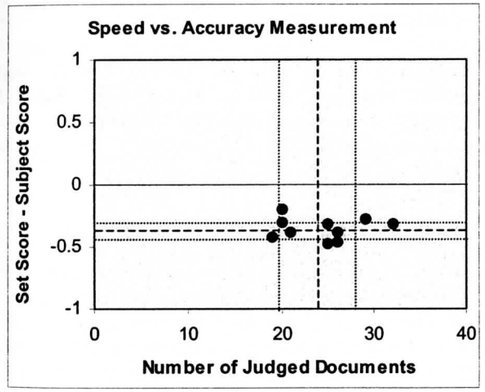

Figure 1: Experimental result for SRV

The experimental data set was constructed based on the search results from the Yahoo search engine. Randomly selected queries were provided to Yahoo and the first fifty Web pages, which were self-contained resources, were used for the experiment. For a more accurate evaluation of the visualizing approach as a search-supporting tool, the datasets were stored locally so that the network traffic required to perform the experiment was not a variance factor in the experiment.

A randomly selected query was given to a participant who was asked to judge as many Web pages as possible in a ten-minute period. Using two different experimental workspaces, each participant evaluates the Web pages with the SB system for the first workspace and the SRV system for the second workspace.

3.2 Experimental Results

The following shows the experimental results comparing multiple users' speed and accuracy in judging the relevancy of Web pages in both systems. The chief variance factor in this experiment was the decision-making tendency of the participants. For instance, some participants felt that they had to investigate significantly more portions of a Web page than others did before they could make a decision about the relevancy of the Web page.

The mean number of Web pages analyzed using the SRV system was 24.30 ± 4.27 articles (mean ± standard deviation), while the mean number of Web pages analyzed using the SB system was 11.70 ± 2.67. The mean accuracy of the SRV system was -0.35 ± 0.09, while the mean accuracy of the SB system was 0.24 ± 0.13 (Figures 1 and 2).

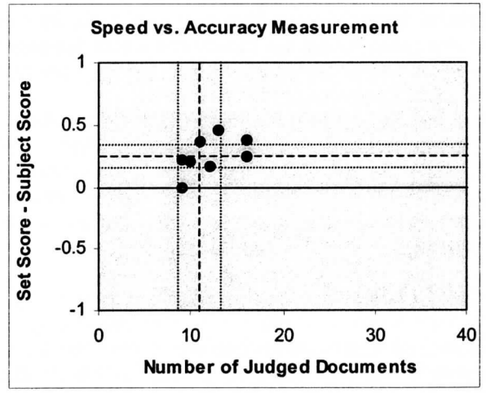

Figure 2: Experimental result for SB

4 DISCUSSION AND FUTURE RESEARCH

The effectiveness of the SRV system as a search-supporting tool was also demonstrated by a post-experimental survey. The questionnaire asked the participants' opinions about using the SRV system and compared the SRV system with the SB system. For the question regarding their satisfaction using the SRV system for finding information, participants were much more satisfied using the SRV system than the SB system. For

other questions regarding the efficiency of the SRV system as a search supporting tool, most of the participants strongly agreed that: 1) their ability to efficiently find the relevant information was greatly improved with the SRV system, 2) they could easily compare multiple Web pages at a glance using the SRV system, and 3) in the future, they would prefer to use the SRV system instead of a traditional SB system.

The pioneer study, which was conducted with a limited number of experiment subjects, has shown the possibility of using the SRV system as a search-supporting tool. For more non-biased evaluation of the SRV system, the SRV system needs more usability testing with a larger number of subjects and performance comparisons to similar methods that represent Web resources through visualization.

To provide pre-analyzed Web pages information to users, the current SRV system can retrieve unnecessary resources. This excessive resource retrieval can be addressed by changing the way search engines index web pages. When the search engine maintains and provides the pre-indexed color codes, the overloaded resource request will be resolved. Meanwhile, the partitioning of the Web page used in the SRV system can contribute to decrease the network traffic. When the users need further investigation of the Web page, instead of downloading an entire page, the system would request the specific segment from the Web server.

Exploitation of anchor pages as an extended information resource is another area needing additional study. Presenting the linkage structure of the anchor page where relevant links are visualized in an intuitive manner will be a useful extension to the SRV system.

Lastly, a current limitation of the SRV system is the inability to visualize non-html formats. With the widespread use applications like Adobe, Word, Power Point, and Flash, the ability to visualize the information contained in these sources can extend the usefulness of the SRV system.

5 CONCLUSIONS

Our preliminary experiment shows that the SRV system is a viable method to present search results efficiently to users. With the aid of visualization, the SRV transforms Web resources into compact, but informative iconic representations. Pre-constructed visual abstraction for a Web page allows users to swiftly analyze various attributes of the page, which assists in intuitive judgment pertaining to the relevancy of a Web page to given search terms while physically viewing a minimal portion of the underlying contents. Moreover, by displaying multiple Web resources simultaneously, the SRV system can exploit the capacity of human perception to scan, compare, and recognize multiple resources. In summary, the SRV can be an effective tool in the search process that will eventually support users' decision-making processes, and save time and effort by alleviating the need to reference any of the irrelevant information.

6 REFERENCES

[1] A. Tombros and M. Sanderson, "Advantages of query biased summaries in information retrieval," Proc. of the Twenty First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2 – 10, 1998.

[2] B. Huberman, P. Pirolli, J. Pitkow, and R. Lukose, "Strong regularities in world wide web surfing," Science, Vol. 280, pp. 95-97, 1998.

[3] M. Kobayashi and K. Takeda, "Information Retrieval on the Web," ACM Computing Surveys, Vol. 32, No. 2, pp. 144-168, 2000.

[4] A. Veerasamy and R. Heikes, "Effectiveness of a graphical display of retrieval results," Proc. of the Twentieth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 236 – 245, 1997.

[5] S. K. Card, G. G. Robertson, and W. York, "The WebBook and the Web Forager: an information workspace for the World-Wide Web," Proc. of the ACM Conference on Human Factors in Computing Systems, pp 111 - 117, 1996.

[6] B. Kim and P.R. Johnson, "Query Fingerprinting: Document Visualization Using Color Coding," Proc. of the International Conference on Information and Knowledge Engineering, pp. 143-149, 2002.

The International Collaborative Environment (ICE)

R. P. Brazile*, K. M. Swigger,
Brian Harrington, Ben Harrington, Xiaobo Peng
***Box 31166**
University of North Texas
Denton, Texas

Abstract

This paper describes the various software modules of the International Collaborative Environment (ICE), a system implemented as a collection of components written in Java. The system allows groups to collaborate, both synchronously and asynchronously, and perform a variety of tasks over the Internet. The components, all of which are shared by users in the same group, include chat, whiteboard, editor, file sharing, help, web browser and desktop sharing tools. Although many of these components are available in commercial software products, this particular system is unique because it is platform independent and has record keeping capabilities. Thus, users (or instructors) can collect and analyze data about how groups use the system and compare that usage with success or failure of performance on different tasks. The current project using ICE involves groups of students at the University of North Texas (UNT) in Denton, Texas collaborating with students at Middle East Technical University (METU) in Ankara, Turkey. A description of the system and an example of how it is used is presented below.

Keywords: Collaborative software, components

Introduction

Although the current version of ICE has been used by groups for over a year, it had an earlier beginning as a system called the Virtual Collaborative University (VCU) [4]. VCU was developed originally using C++ and Microsoft's Netmeeting and was, therefore, restricted to running under only Microsoft operating systems. The decision was made at the beginning of this project to re-write the entire software system using Java. As a result, the latest version of the software, renamed ICE, runs under a variety of different operating systems including Microsoft Windows, Linux, FreeBSD and Solaris.

A user usually starts a collaboration by logging onto the ICE system as a member of a group and uses the components of the system in a shared, synchronous mode. Members of a group may chat with one another, draw or display images using the whiteboard, and edit files through the shared editor. They may also surf the Internet using the shared browser, which allows different members of the group to lead other members to the same site. Groups may share files with one another using the file-sharing component. There is also a desktop sharing tool that allows members of the group to share different applications and watch while various programs are executed. The control of the desktop can be transferred to different members of the group upon their request.

All users must be assigned to a group before they can use ICE. The group naming convention for ICE is modeled on a class/section/group hierarchy, which is a carry over from the previous system where groups consisted of students enrolled in a particular course. Therefore an example of a particular group name is CSCI4350/section001/group01, where the first item refers to the course, the second the section, and the third the group number. Instructors or teachers of the course are allowed to join any group, while all other users are restricted to only their pre-assigned groups.

Each time the ICE system is invoked by a user, the software checks the ftp server for any updates to the system. If there are updates, ICE automatically downloads the software and installs the latest version on the user's machine.

When the user enters their logon name and password, ICE checks the server to validate the logon name and password. If the logon is successful, the server sends a list of current users to the new user. Once this occurs, all contact among members of the group is done peer-to-peer.

The Components of the System

As previously stated, ICE provides support for groups who need to work together on problems in real time. The main ICE interface is designed similar to a web browser (see figure 1), and all the collaborative tools are displayed within this single application. A user accesses the various tools either through the main menu bar, which appears at the top of the page, or by clicking on one of the tool icons displayed on the left-hand side of the application. The interface currently supports a number of different collaborative applications including chat, drawing, file-sharing, browsing, editing, and application sharing. A discussion of each of these applications now follows.

The Chat Tool

The chat interface appears at the top of the application and is the only collaborative tool that is always displayed