A Visual Interface Design for Search Results from Multiple Search Engines

Sheng-Shun Wang, Cheng-Zen Yang, and Ing-Xiang Chen

Department of Computer Science and Engineering

Yuan Ze University

Chungli 320, Taiwan, ROC

{tonia,czyang,sean}@syslab.cse.yzu.edu.tw

Abstract

Because of the explosively growing amount of information on the World Wide Web, it is a time-consuming task to find the demanded information. Although many search engines exist to help users search for the information, most of them reveal limited information, because they show the search results in list-based interfaces, and users need to check each entry to have a clear view of the search results. In this paper, a visual interface is proposed to display the search results collected from multiple back-end search engines and to help users discover important information. A prototype has been implemented, and it visualizes the search results from six popular search engines. Two kinds of information are further visualized: hyperlink reference relationships and whether some pages are of a same site. In the aspect of information exhibition, the visual interface can help users reduce the time of seeking the relevant results. It can be applied to help users search for information efficiently.

Keywords: information visualization, interface design, Web search engine, metaphor

1. Introduction

Because the amount of information on the Web is huge and growing rapidly, it is impractical for users to browse every Web site and search for the needed information. The Web search engine has thus become a helpful tool to search for the demanded information. However, different search engines have their own searching characteristics, and as a result, some Web pages may be indexed by a certain search engine but missed by others. The ranking could be also very different. In [1], it is pointed out that no Web search engines can index all existing Web pages. To increase the coverage of indexing Web pages, metasearch engines, such as Dogpile [13], Ithaki [14], Ixquick [15], Mamma [16], and ProFusion [17] have been developed for Web searching.

Although many metasearch engines have been developed and proved their superior coverage of Web pages [10], they have the same drawback as what exists in the general list-based search engines [1, 12]. In [1], Amento et al. point out that the list-based interfaces take users much time on clicking and browsing until they are satisfied with the results. In [12], Yang et al. point out that the list-based interface displays the result only in linear scoring format and ignores the linkage relationships among the search results. They also point out that because some Web pages from the same site may be ranked in different result pages, the list-based interface thus prevents users from finding these relevant pages.

On account of the shortcomings of the list-based interface, research efforts have been initiated to improve the visual representation of the search results such as Hyperspace [2], VR-VIBE [3], WebQuery [4], Sparkler [5], Butterfly [6], Envision [9], Natto [11], and H3 viewer [12]. However, in the past research, some ignore the linkage relationships among the results [3, 5, 6, 8, 9], and some focus on the thread relationships [2, 4, 11] but incur the visual clutter problem.

Because synthesizing the search results from multiple search engines provides a higher coverage and facilitates information searching, an interface displaying search results of multiple search engines will help users find more relevant information. This motivates us to design such a visual interface. In the visual interface design, the hyperlink relevance among the commonly retrieved Web pages can be displayed to help users find relevant information. Visual metaphors are used to improve the limitations of the past research.

The rest of the paper is organized as follows. Section 2 overviews previous research on visualizing search results of multiple information sources. Section 3 describes the visualization interface design. Section 4 illustrates the operation of the visual interface prototype with a query example. Section 5 concludes the paper.

2. Visual Interfaces for Multiple Information Sources

Many visualization interfaces have been developed to help users find the needed information. Only few of them focus on visualizing the search results from multiple sources. Sparkler [5] and Butterfly [6] are two examples.

In Butterfly, the visual interface displays articles of multiple databases across the Internet. It uses a metaphor of a butterfly to visualize the search results as entries in two wings of the butterfly. One wing is for listing references and another for citers. Search results from the same source are piled as a pyramid and are in the same. It shows both the citers and references from multiple information sources. However, the number of result entries in a screen is limited, and the interface is quite complicated for user interaction.

In Sparkler, the search results from different search engines are visualized as dots in different colors. The result entries with higher ranks are placed near the center, and hundreds of dots can be displayed in a screen at the same time. Although the results from multiple search engines can be distinguished from their different colors, an overlapping problem cannot be avoided when the number of search results is large. Moreover, Sparkler does not visualize the linkage relevance among the search results. Users cannot find the cross-reference information from Sparkler.

3. Visualization Interface Design

In our previous study, it shows that when users search for information from the Internet, the hyperlinks within the returned pages can help them find other relevant Web pages [12]. However, the coverage of a single search engine is limited. Therefore, it is desired to design a front-end visualization interface that can visualize the search results from multiple search engines. The design concepts of the visualization interface and related algorithms are described as follows.

3.1 Visual Metaphors

To display the visual search results from multiple search engines, the result entries and the linkage relationships are the two subjects to be visualized in our design. In visualization interfaces such as [3, 5], the result entries are generally visualized as dots in same size. Those dots with higher ranks are placed near the apex of the pyramid or near the center. However, when hundreds or thousands of result entries are displayed, an overlapping problem cannot be avoided.

In [4, 8, 11], the linkage relationships are displayed as the connected lines with the same color. However, when many linked lines are displayed, users may get confused in linkage chaos.

The drawbacks existing in previous visualization interfaces hurt the visualization benefits. To avoid the visual clutter problem, our visual interface uses metaphors with careful considerations of object placement. Because the same result entries may be returned from multiple search engines, a metaphor can represent the number of the occurrences of the same URL. Therefore, a building metaphor is used in our design.

In Figure 1, the search results from the same search engine are grouped in a scroll panel, where each rectangular building represents a result entry. The buildings arrayed from left to right represent the decreasing ranks from top to bottom in the search results. The height of each building represents the number of the same URLs being returned by the multiple search engines. For example, a one-floor building indicates that the URL is merely returned by one search engine. A two-floor building means that the URL appears in two search engines, and so forth.
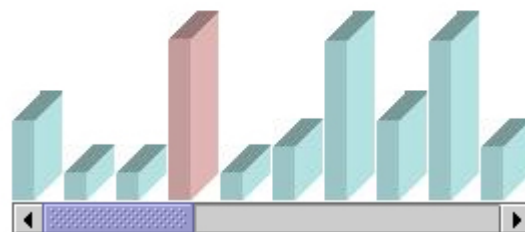

Figure 1. The building metaphor of search results from a search engine.

Besides, if a result entry has any lines connected to other entries, its corresponsive building is designed in the contrast color. The connected lines here are designed in the

concept of virtual road metaphors connecting to different buildings in the visualization interface, where the connected entries are either from the same Web site or have hyperlink relationships. In Figure 2, the entry in the contrast color is connected to the relevant entry in another search engine, and the lines from different start node are painted in different colors to reduce the visual clutter problem.
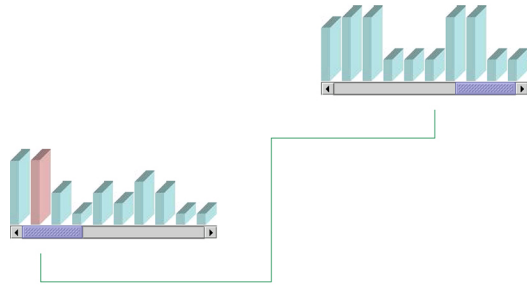

Figure 2. The two relevant buildings are connected with a road.

3.2 The Layout of Road Metaphor

To connect the lines among the relevant entries from multiple search engines, the layout will be different in accordance with the positions of the connected buildings. The methods of drawing the connected roads are classified into two types.
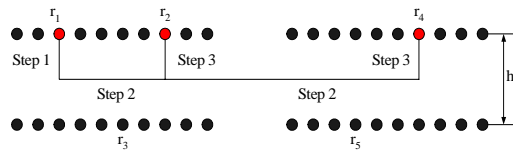

Figure 3. Three steps to connect relevant search results located in the same layer.

In the first type, when the buildings to be connected are in the same horizontal layer, it needs three steps that are shown below to complete the road connection. An example is illustrated in Figure 3 where r_1 is the start node and r_2 and r_4 are the destination nodes, and they are located at the same layer. The positions of the start node and the destination node are expressed as (S_x, S_y) and (D_x, D_y) respectively, and h is the height of one layer. Here are the drawing steps.

Step 1. Draw a vertical line connecting (S_x, S_y) and $(S_x, S_y + h/2)$.

Step 2. Draw a horizontal line connecting $(S_x, S_y + h/2)$ and $(D_x, S_y + h/2)$.

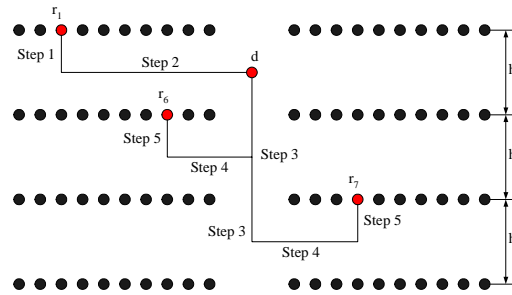Step 3. Finally, draw a vertical line connecting $(D_x, S_y + h/2)$ and (D_x, D_y).

Figure 4. An example to show how to connect relevant search results at the different layers.

If the buildings to be connected are located at different layers, five steps are taken to connect the roads. In Figure 4, r_1 is the start node, r_6 and r_7 are the destination nodes, and d is the dummy node (d_x, d_y) whose ordinate is to connect the nodes at other layers.

Step 1. Draw a vertical line connecting (S_x, S_y) and $(S_x, S_y + h/2)$.

Step 2. Draw a horizontal line connecting $(S_x, S_y + h/2)$ and (d_x, d_y).

Step 3. Draw a vertical line connecting (d_x, d_y) and $(d_x, D_y + h/2)$.

Step 4. Draw a horizontal line connecting $(d_x, D_y + h/2)$ and $(D_x, D_y + h/2)$.

Step 5. Draw a vertical line connecting $(D_x, D_y + h/2)$ and (D_x, D_y).

The visual clutter problem in a visualization interface usually happens when the number of dots or lines becomes large. In our visual interface design, when users click many nodes to reveal the relevant search results, more than one line will be displayed in a block and these lines will be evenly placed. In Figure 5, the first line L_1 is located in the height of $h/2$. Other six lines L_2 to L_7 are continuously added at $h/4, 3h/4, h/8, 5h/8, 3h/8,$ and $7h/8$.
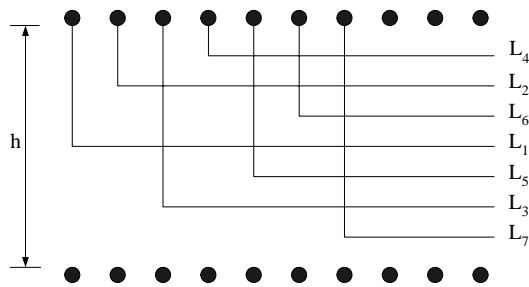
Figure 5. The layout of the lines added sequentially from L1 to L7.
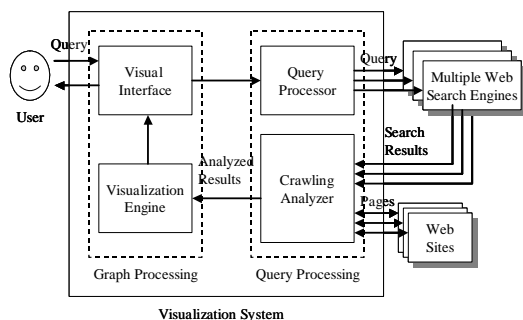
3.3 System Architecture Overview

Figure 6. An overview of the system architecture.

In Figure 6, when the visual interface receives an input query and passes it to the query processor, the query processor then requests the search results from the back-end multiple search engines. After the search results are retrieved and parsed by the crawling analyzer, it starts to collect the Web pages from the origin servers and then analyzes them. The analyzed line relevance and the information of duplicate entries are passed to the visualization engine for graph drawing. Finally, the visualized results are displayed via the visual interface. The system architecture consists of two main modules: the query processing module and the graph processing module. Their functionalities are respectively described below.

3.3.1 The Query Processing Module

In the query processing module, the query processor is to request the search results from the multiple back-end Web search engines. After the search results are returned to the crawling analyzer, it parses out the URL entries, retrieves the content from the origin servers, and analyzes the relationships among different result entries. If the search results exist linkage relationships or they are from the same Web site, the crawling analyzer will pass the analyzed structures among the relevant entries to the visualization engine.

3.3.2 The Graph Processing Module

In the graph processing module, the task of the visualization engine is to deal with the graph drawing, which consists of two parts, the virtual buildings and roads. The rectangular buildings with different heights represent the different number of duplicate URLs in the search results. The roads connected to the buildings represent the relevance among the result entries. Both the buildings and the roads are generated after the crawling analyzer passes the analyzed results to the visualization engine in the client side.

The architecture design of the query processing module in the server side and the graph processing module in the client side has the following two advantages.

First, it reduces the amount of network usage at the client side. Because the query processing module in the server has completed most of the data analysis, it returns the simplified structure to the user rather than all the raw Web pages. In the viewpoint of the client side, returning the simplified structure requires less network bandwidth than directly accessing the raw Web pages.

Second, it reduces the data processing time. Because the relevance of search results is completely analyzed by crawling analyzer before visualization, users can directly view the processed results without requesting data from the server again. Therefore, the time cost of data processing is greatly reduced.

4. The Visual Interface Prototype

4.1 The System Environment

In the current interface prototype, six popular Web search engines are adopted as the back-end search engines. They are About [18], Excite [19], Fast (Lycos) [20], Google [21], Inktomi [22], and Teoma (Ask Jeeves) [23]. These search engines have their own searching rules. The prototype is developed with JDK1.4.1 and Red Hat Linux 7.3.
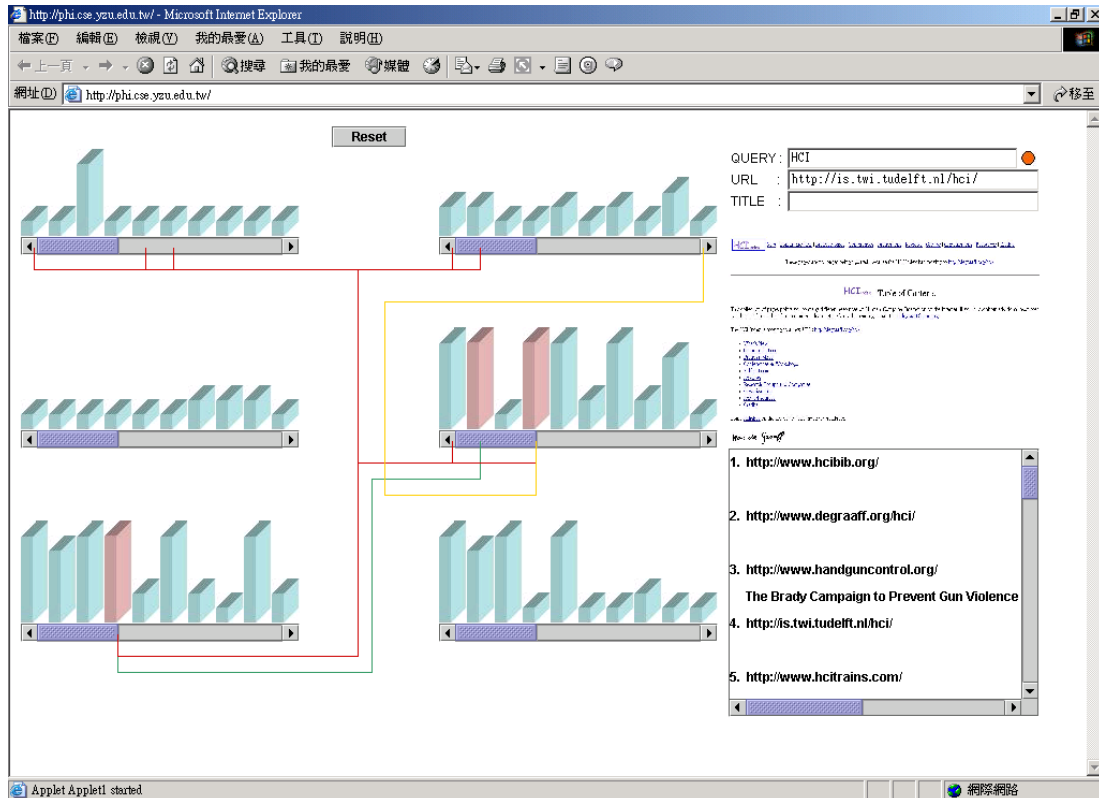
4.2 The Interface Prototype

Figure 7. Visualization of the search results of querying "HCI".

In Figure 7, the graphical result of querying "HCI" from multiple search engines is presented. The search results from the six search engines are visualized as different buildings. When the mouse passes by a building, the label of its title information is displayed to help users navigate the search results. After the building being clicked, it turns into the contrast color and shows the roads connected to the relevant buildings; meanwhile, the URL and title of the clicked building are shown in the right upper blanks.

When different buildings are clicked to show the road connections among different buildings, the road metaphors are displayed in different colors to avoid the visual clutter problem. If users think that the interlaced lines confuse them, clicking the Reset button on the screen top will clean all the lines and resume their navigation.

5. Conclusions

Because of the explosively growing amount of information on the World Wide Web, finding information of the user needs becomes a difficult problem. Search engines can help users retrieve information of interest from a large amount of Web pages. However, different search engines have their own searching characteristics, and as a result, some Web pages may be indexed by a certain search engine but missed by others. Besides, the ranking policies could be also very different.

Thus, metasearch engines are designed to increase the coverage of Web pages and to facilitate Web searching. Nevertheless, they have the same drawbacks existing in the general list-based search engines. The list-based interfaces take users much time on clicking and browsing until users are satisfied with the result, and they display the search results in linear scoring format and ignore the linkage relationships among the search results. Besides, if the result entries are from the same site but located in different result pages, users may thus miss them.

In this paper, we propose a visualization interface to provide more reference information to Internet users and to help them find information of their needs among millions of Web sites on the Web. Because the visual metaphor design of buildings and roads is close to human living experiences, users can easily get used to interacting with the interface. The visual objects of different heights, colors and positions

also greatly reduce the visual clutter problem.

Displaying the relevance among the search results of multiple Web search engines in a visualization interface provides users with a visual navigation and enhances finding information of user needs. In the aspect of information exhibition, the graphical representation can help users reduce the time cost of seeking relevant results. However, how to efficiently retrieve a large number of Web pages from the remote servers for later analysis is still a bottleneck. The caching and data prefetching mechanisms are practical methods to improve the retrieval efficiency. These future improvements on data retrieval and human-computer interface in our visual interface design will help users find the needed information more quickly.

References

[1] B. Amento, W. C. Hill, L. G. Terveen, D. Hix, and P. Ju. "An Empirical Evaluation of User Interfaces for Topic Management of Web Sites." In *Proceedings of the SIGCHI' 99 Conference on Human Factors in Computing Systems*, pp. 552–559, May 1999.

[2] R. Beale, R. J. McNab, and I. H. Witten. "Visualizing Sequences of Quries: A New Tool for Information Retrieval." In *Proceedings of 1997 IEEE Conference on Information Visualization*, pp. 57–62, August 1997.

[3] S. D. Benford, D. N. Snowden, C. C. Brown, G. T. Reynard, and R. J. Ingram. "Visualizing and Populating the Web: Collaborative Virtual Environments for Browsing, Searching and Inhabiting Webspace." In *Proceedings of the 8th Joint European Networking Conference,* pp. 1751–1761, May 1997.

[4] J. Carriere and R. Kazman. "WebQuery: Searching and Visualizing the Web through Connectivity." In *Proceedings of the 6th International World Wide Web Conference,* pp. 701–711, April 1997.

[5] S. Havre, E. Hetzler, K. Perrine, E. Jurrus, and N. Miller. "Interactive Visualization of Multiple Query Results." In *Proceedings of the IEEE Symposium on Information Visualization,* pp. 105–112, October 2001.

[6] J. D. Mackinlay, R. Rao, and S. K. Card. "An Organic User Interface for Searching Citation Links." In *Proceedings of the ACM SIGCHI'95 Conference on Human Factors in Computing Systems*, pp. 67–73, May 1995.

[7] A. Mowshowitz and A. Kawaguchi. "Bias on the Web." *Communications of the ACM Vol. 45, No. 9, pp. 56–60, September 2002.*

[8] T. Munzner. "H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space." In *Proceedings of the 1997 IEEE Symposium on Information Visualization,* pp. 2–10, October 1997.

[9] L. T. Nowell, R. K. France, and D. Hix. "Visualizing Search Results with Envision." In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 338–339, August 1996.

[10] E. Selberg and O. Etzioni. "Multi-Service Search and Comparison Using the MetaCrawler." In *Proceedings of the 4th International World Wide Web Conference,* pp. 195–208, October 1995.

[11] H. Shiozawa and Y. Matsushita. "WWW Visualization Giving Meanings to Interactive Manipulations." In *Proceedings of Advances in Human Factors / Ergonomics*, pp. 791–794, August 1997.

[12] H.-C. Yang, M.-C. Tzeng, and C.-Z. Yang. "A Web Interface for Visualizing Web Search Engine Results." In *Proceedings of the 3rd International Conference on Parallel and Distributed Computing, Applications, and Techniques (PDCAT 2002)*, pp. 283–288, 2002.

[13] http://www.dogpile.com/info.dogpl/

[14] http://www.ithaki.net/dir.html

[15] http://ixquick.com/

[16] http://www.mamma.com/

[17] http://www.profusion.com/

[18] http:// www.about.com/

[19] http:// www.excite.com/

[20] http://www.lycos.com/

[21] http://www.google.com/

[22] http:// www.inktomi.com/

[23] http:// www.teoma.com/