Alternative Implementation Techniques for Web Text Visualization

Omar Alonso
Oracle
500 Oracle Parkway
Redwood Shores, CA 94065 - USA
omar.alonso@oracle.com

Ricardo Baeza-Yates
CWR/DCC – Universidad de Chile
Av. Blanco Encalada 2120
Santiago - Chile
rbaeza@dcc.uchile.cl

ABSTRACT

We present an approach for building text visualizations that avoids using plug-ins or clients based on languages like Java. Instead we propose to make the search engine application more aware of the visualization process and use the web browser standard features to do the rendering work. We demonstrate the ideas with a text visualization metaphor implementation that is part of a search engine.

1.INTRODUCTION

Information visualization techniques can help users navigate through large sets of answers as well as selecting the appropriate web page or document [4]. Some of those visualization metaphors have been around for a number of years but they are not getting massive adoption.

As the web matures and people are more familiar with search engines seems like everybody is an expert with the top-10 hit user interface (although on average users see on average less than 20 results). However jumping into the next level of advanced user interfaces like visualization is still a challenge.

We have identified two main problems. First, there is an adoption issue from users. A lot of the visualization metaphors are sophisticated and they expect an advanced user or knowledge worker to take full advantage of it. This is also expected. The same happened when users moved from text only to GUI-based user interfaces. Second, there is technology available to implement the metaphors but not all the components are available in all platforms or browsers. Plug-ins solves part of the problem but you still have to manage all the different configurations. In this poster we will concentrate in the second problem: how to provide inexpensive visualizations using standard features available in most web browsers.

First we briefly describe our approach, followed by the interface and the implementation, ending with some preliminary conclusions.

2.OUR APPROACH

We want to focus our solution in two items that we consider very important. The first one is to provide visualizations that enforce comparison among the retrieved documents instead of just nice pictures of them. The second item is to present an implementation that performs well and it can be deployed using existing technology.

In terms of implementation our approach is to avoid using plug-ins or clients based on a language like Java to implement visualization metaphors. The idea is to make the server side (the search engine in this case) produce more data for visualization and for the browser to actually work as a client with standard features like CSS (Cascading Style Sheets).

Of course there are still features support issues with specific browsers but at least it is possible to standardize on versions or most popular ones. The important part is that the rendering time is faster and hence performance and scalability improves.

3.WEB TEXT VISUALIZATION

Usually the work in the area of information visualization is more concentrated in the representation of well-known structures (like trees, graphs, etc.) or relationships between different attributes. We would like to emphasize the visualization on the text itself (a web page or document) after a search was issued.

Our visualization is based on the bookpile model presented in [3] and then implemented in [1] using Java and focused on documents, not text. The new interface is a two-view model where the left side presents structure and the right content and operations on it. The user enters a query and the system displays the search results in a strip or thumbnail view on the left side. A click on an item of the structure opens the actual content.

The strip view is based on [5] where a line maps to a text object (in this case a line too). The thumbnail view shows the content per token in a very small font with each search term highlighted with a different color followings the ideas used in [6]. The user can move the mouse over the document and text in bigger font is presented. The views also include metadata information like ranking score, categories and so on. It's important to notice some of Tufte's small multiples principles in action [8]. For example it is easy to compare which documents have categories, its hits distribution (numerical and spatial) in the document, size, etc.

The user can click on the strip or thumbnail and the search engine will show the document in its original format on the content view. Figure 1 shows the strip view and figure 2 presents the thumbnail view.

Proceedings of the First Latin American Web Congress (LA-WEB 2003)
0-7695-2058-8/03 $17.00 © 2003 IEEE

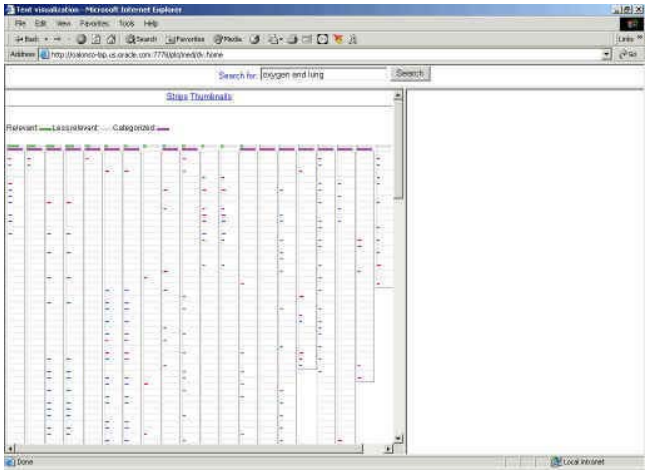

Figure 1. Strip view.

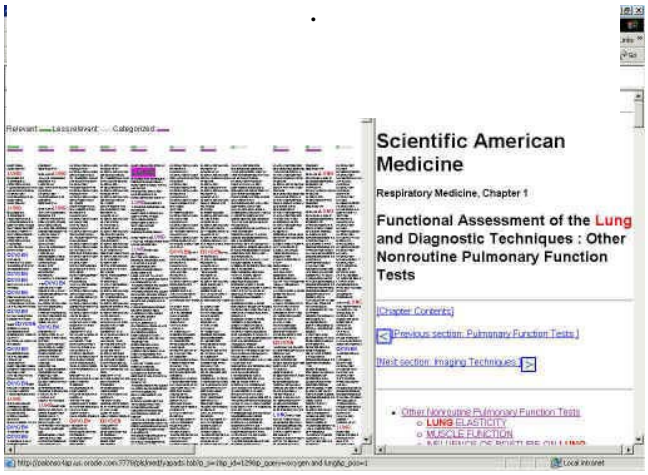

Figure 2. Thumbnails view.

4

T se

s at

allows access to document services and other index information that is useful for rendering the documents [6].

The search engine application generates search results in HTML with a CSS associated for each view. The browse provides all the interaction in the views.

This architecture is also extensible in case you want to add more sophisticated visualizations that do require plug-ins or special viewers. This can be enhanced with special XML protocols targeted to this type of applications [2].

5.CONCLUSIONS

We provide an approach for building inexpensive information visualization metaphors with server side programs that generates different CSSs. We also provide a text visualization technique that allows user to see all the documents in one screen, compare, and eventually select one of them. We understand that our approach is limited compared to more sophisticated graphical techniques but requires less components to implement, development effort and its availability is immediate.

We plan to continue working on this area enhancing the visualization metaphors and improving the performance for large data sets.

6.REFERENCES

[1]O. Alonso and R. Baeza-Yates, "A Model for Visualizing Large Answers in WWW", In *XVIII Int. Conf. of the Chilean CS Society*, Antofagasta, Chile, November 1998, IEEE CS Press, 2-7.

[2]O. Alonso and R. Baeza-Yates, "A Model and a Software Architecture for Search Results Visualization on the WWW". In *Proceedings of the 7th Symposium on String Processing and Information Retrieval (SPIRE'2000)*, La Coruña, Spain, September 2000, IEEE CS Press, 8-16.

[3]R. Baeza-Yates. "Visualizing large answers in text databases". In *Int. Workshop on Advanced User Interfaces (AVI'96)*, Gubbio, Italy, ACM Press, May 1996, 101-107.

[4]S. Card, J. Mackinklay, and B. Shneiderman *Readings in Information Visualization*. Morgan Kaufmann, San Francisco, CA (1999).

[5]S. Eick, J. Steffen, and E. Summer. "Seesoft-A Tool For Visualizing Line Oriented Software Statistics" *IEEE TSE* 18(11), (1992).

[6]K. Kaugars "ntegrated Multi Scale Text Retrieval Visualization". *CHI'98 Summary*, Los Angeles, CA

[7]Oracle Corp. *Oracle Text Reference*, Redwood Shores, CA (2002).

[8]E. Tufte *Envisioning Information*, Graphics Press, Cheshire, CT (1990).