Lighthouse: Showing the Way to Relevant Information

Anton Leuski and James Allan
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003 USA

E-mail: leuski,allan@cs.umass.edu

Abstract

Lighthouse is an on-line interface for a Web-based information retrieval system. It accepts queries from a user, collects the retrieved documents from the search engine, organizes and presents them to the user. The system integrates two known presentations of the retrieved results – the ranked list and clustering visualization – in a novel and effective way. It accepts the user's input and adjusts the document visualization accordingly. We give a brief overview of the system.

H.3.3 Information Search and Retrieval – Relevance feedback. H.3.5 Online Information Services – Web-based services; H.5.2 User Interfaces – Graphical user interfaces, Screen design;

1. Introduction

Locating interesting information on the World Wide Web is the main task of on-line search engines. Such an engine accepts a query from a user and responds with a list of documents or web pages that are considered to be relevant to the query. The pages are ranked by their likelihood of being relevant to the user's request: the highest ranked document is the most similar to the query, the second is slightly less similar, and so on. The majority of today's Web search engines (Google, Infoseek, etc.) follow this scenario, usually representing a document in the list as a title and a short paragraph description (snippet) extracted from the text of the page. The evaluation methods for this approach are well-developed and it has been well studied under multiple circumstances [3].

The ordering of documents in the ranked list is simple and intuitive. The user is expected to follow the list while examining the retrieved documents. In practice, browsing the ranked list is rather tedious and often unproductive. Anecdotal evidence show that users quite often stop and do not venture beyond the first screen of results or the top ten retrieved documents.

Numerous studies suggest that document clustering (topic-based grouping of similar documents) is a better way of organizing the retrieval results. The use of clustering is based on the Cluster Hypothesis of Information Retrieval: "closely associated documents tend to be relevant to the same requests" [10, p.45]. An overview of the related work on clustering and document visualization can be found in the extended version of this paper [5].

We describe Lighthouse [8], an interface system for a typical web search engine that tightly integrates the ranked list with a clustering visualization. The visualization presents the documents as spheres floating in space and positions them in proportion to their inter-document similarity [2]. If two documents are very similar to each other, the corresponding spheres will be closely located, whereas the spheres that are positioned far apart indicate a very different page content. Thus the visualization provides additional and very important information about the content of the retrieved set: while the ranked list shows how similar the documents are to the original query, the clustering visualization highlights how the documents relate to each other.

A simple corollary of the Cluster Hypothesis is that if we find one relevant document, some of the relevant documents are likely to be similar to it. With our clustering visualization it literally means that relevant documents tend to be in the neighborhood of the other relevant documents. Locating interesting information should be as easy as examining the spheres that are close to the sphere of a known relevant document. We have designed a foraging algorithm that selects documents for examination based solely on their proximity information and confirmed that assumption experimentally [7, 6]. The algorithm is significantly more effective in locating relevant documents than the original ranked list (measured by average precision) and it is comparable to the

interactive relevance feedback approach [1].

Our past research [7, 6] dealt only with analysis of the clustering visualization and no actual system was built. The Lighthouse system described here has grown out of that study.

2. System Overview

Figure 1 shows two screen shots of the system. All examples of using the system in this paper refer to that figure. We ran the query "Samuel Adams" on the Infoseek search engine (`www.infoseek.com`). The top fifty documents retrieved are presented as the ranked list of titles and fifty spheres corresponding to each page.

The ranked list is broken into two columns with 25 documents each on the left and on the right side of the screen with the clustering visualization in the middle. The list flows starting from top left corner down and again from the top right corner to the bottom of the window. The pages are ranked by the search engine in the order they are presumed to be relevant to the query. The rank number precedes each title in the list.

The clustering visualization, or the configuration of fifty spheres, is positioned between the two columns of titles. This organization makes the user focus on the visualization as the central part of the system. The spheres appear to be floating in space in front of the ranked list. We believe that such an approach allows us to preserve some precious screen space and at the same time it stresses the integration of the ranked list and the visualization.

Each sphere in the visualization is linked to the corresponding document title in the ranked list so clicking on the sphere will select the title and vice versa. Selecting a document puts a black outline around the corresponding title and sphere – e.g., the documents ranked 12 and 24 in Figure 1. The user can examine the clustering structure and place it in the best viewing angle by rotating, zooming, and sliding the whole structure while dragging the mouse pointer. (Only the spheres can be manipulated in this fashion – the ranked list remains in place.)

If the user points to a document title or a sphere with the mouse pointer while keeping a control key pressed, a small window similar to a comics balloon pops up showing the document description. The content of that window is the description paragraph (or snippet) returned by the search engine for the document. In addition a line connects the sphere and the title. This design preserves screen space and keeps the snippet readily available to the user by a gesture with a mouse. The line literally links the two document representations – the title and the sphere – together. A double-click on the document title (or sphere) opens the document in the web browser.

2.1. Multiple Dimensions

The same set of spheres can appear as either a 3-dimensional (Figure 1, top) or 2-dimensional (Figure 1, bottom) structure. The user can switch the dimensionality on the fly by selecting the button in the toolbar at the top of the window. We achieve the effect of depth in the visualization by using perspective projection of the spheres – the remote spheres appear smaller than their front counterparts – together with the fog effect – the color of the remote spheres is closer to the background color than the color of the front spheres.

The similarity relationship among documents is rather complex and cannot be exactly reproduced by the clustering visualization (it is calculated in the several hundred dimensional "term-space"). An additional dimension provides an extra degree of freedom, which in turn results in a more accurate representation of document relationships. Thus, a 3-dimensional picture has to be more accurate and therefore more effective for the navigation than a 2-dimensional one. This assumption was confirmed in a previous study, when our foraging algorithm proved to be more effective in 3D than in 2D [7]. We have also observed that the differences in effectiveness between foraging for relevant documents using proximity information in the original "term-space" and in 2- or 3-dimensional visualization space are small, suggesting that the visualization is indeed an accurate representation of the document configuration (accurate enough for the retrieval purposes).

However, our user studies of the visualization showed that people prefer the 2-dimensional presentation over the 3-dimensional one for a similar foraging task. This observation confirms a well-known fact that given a flat image, the users apply a significant cognitive effort to recreate a 3-dimensional structure in their minds [9]. The best results also require physical actions – it is much easier for the user to recognize and understand the proximity relationship among the spheres in the picture while slowly rotating the structure with the mouse pointer. We have shown that these difficulties may eliminate all the advantages of the greater accuracy of the 3-dimensional visualization [7].

Because people differ in their ability to visualize spatial structures, we give the user the freedom to choose the dimensionality of the presentation he or she is more comfortable with. From our own experience we found the ability to switch the dimensionality very rewarding: a 2-dimensional picture provides a great overview of the whole document set, but when a more precise analysis is required – e.g., when it is necessary to establish if two or more documents as close as they appear to be – the accuracy of the 3D picture can be more helpful. In this case we select the documents in question and switch the dimensionality to examine them. Sometimes this action reveals that spheres separated in 3D

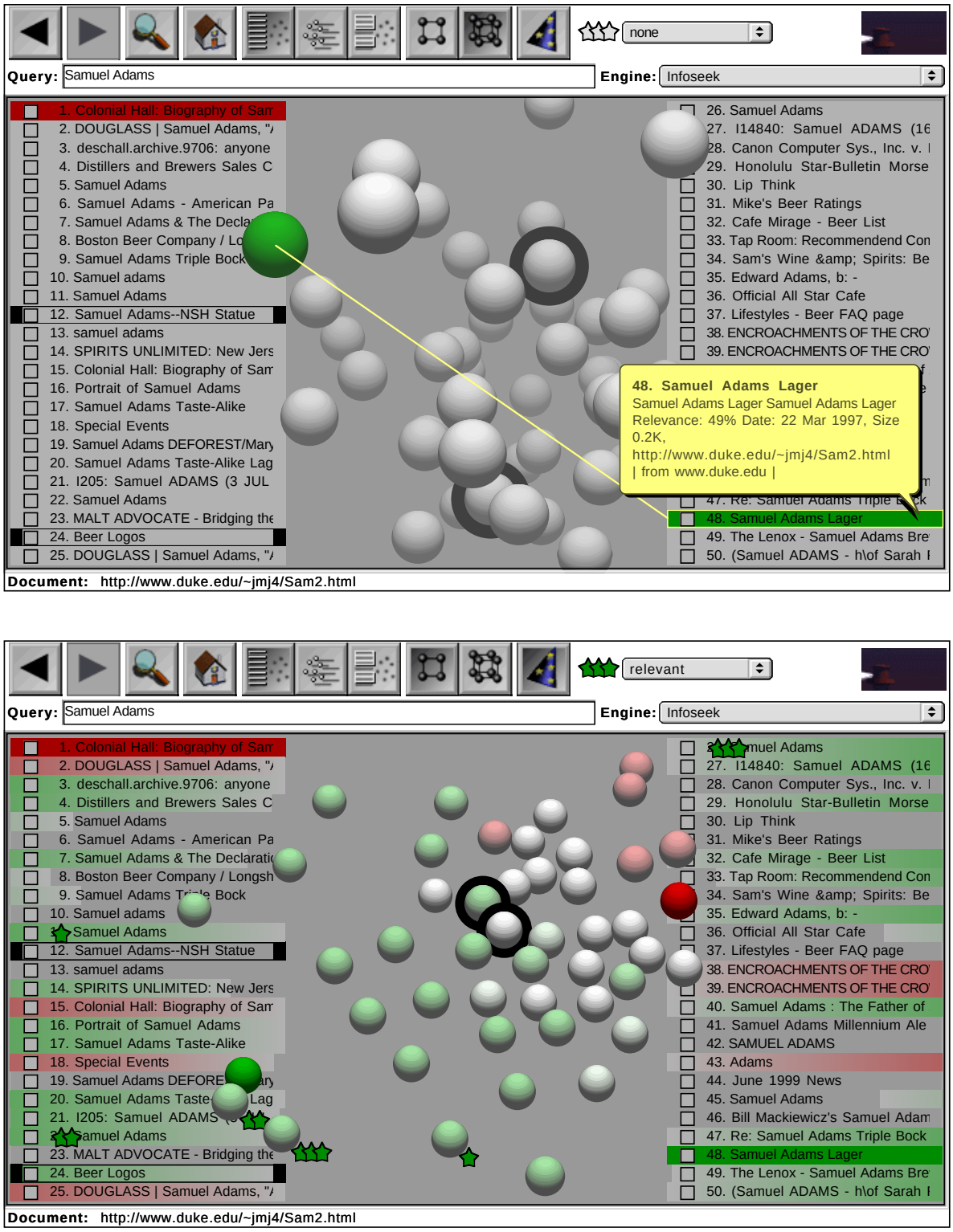

Figure 1. Screen shots of the Lighthouse system. The top fifty documents retrieved by the Infoseek search engine for the "Samuel Adams" query. Both three-dimensional (top) and two-dimensional (bottom) pictures are shown.

appear clumped in 2D. For example, both screen shots show the same configuration of documents. Consider the two selected documents, whose spheres (with black outlines) appear closely placed in the central part of the 2-dimensional picture (Figure 1, bottom). The same two document spheres in 3 dimensions are separated by an additional document sphere (Figure 1, top). A brief examination of titles reveals that these documents (ranked 12 and 24 in the list) discuss unrelated topics.

2.2. User's Feedback

Our user experiments showed that spatial proximity is an intuitive and well-recognized (by the users) metaphor for similarity between objects. We observed that the users' search strategy tends to follow the model incorporated into our algorithmic approach [7]. The users were significantly more successful with the visualization than they would be by following the ranked list. However, we also observed that the users are likely to make mistakes while deciding on the proximity between two groups of spheres and their foraging performance was somewhat below that of the algorithm. We believe the system can successfully assist users in their browsing of the document set. If a user is willing to supply Lighthouse with his or her interest evaluation of examined documents, the system will suggest the next document to look at.

The user's interest or the relevance assessment of the document is expressed by clicking on the checkbox attached to each document title. One click marks the document as non-relevant, the corresponding title and sphere are highlighted in red. A second click marks the document as relevant and both the sphere and the title show up in green. Another click removes the mark from the document.[1]

Given the ranking information obtained from the search engine and the relevance judgments collected from the user, Lighthouse estimates the expected relevance values [4] for the unjudged web documents and provides two different tools to convey that information to the user. Both tools operate in suggestion mode – they point the user to supposedly interesting material without forcing their choices on him. Both tools can be switched on and off using the controls in the toolbar at the top of the window.

Shade Wizard The first tool, the *Shade Wizard*, indicates the estimated relevance for all unjudged documents by means of color and shape. Specifically, if the system estimates the document is relevant, it highlights the corresponding sphere and title using some shade of green. The intensity of the shading is proportional to the strength of the system's belief in its estimate – the more likely the document is relevant, the brighter the color. The same is true for estimated non-relevant documents – the more likely the document is non-relevant, the brighter the red shade of the corresponding object on the screen. The same color shade is used to highlight the document title backgrounds. Additionally, the length of that highlighted background is proportional to the strength of the system's belief in its estimate. The highlighted backgrounds in the left column are aligned on the left side and the highlighted backgrounds in the right column are aligned on the right side. Note that a white sphere and a very short highlighting for the document title reflects that the system's estimate of that document relevance is almost exactly between "relevant" and "non-relevant" – i.e., even odds that the document is relevant. The unjudged document titles are further separated from the judged documents by using a gradient fill for painting their background.

Consider the example on the Figure 1. We judge relevant all the documents that mention the beer brand "Samuel Adams". The top ranked document is about Samuel Adams the Patriot and we marked it as non-relevant. The bright red sphere corresponding to that document is located on the top right part of the picture. The Wizard immediately pointed us to the document whose sphere is on the bottom left part of the picture. The corresponding document is ranked 48, it is about Samuel Adams Lager and we judged it relevant. Now one quick look tells us that the documents about the beer probably occupy the bottom and left of the picture while the documents about the American patriot take the top right part of the visualization. We can see how the colored shading propagates from the known relevant documents to the known non-relevant documents creating an impression of two lights – one green and one red – shining through the structure. This visual effect gave the name to the system.

Star Wizard Our experience suggests that it can be very difficult to exactly discriminate between several documents with similar relevance estimations – when the documents are painted with what looks like the same shade of green and even the title backgrounds are of the same length – e.g., documents ranked 26 and 27 on the screen shot. We introduce the second tool that we call the *Star Wizard*. It is controlled by the popup button in the window toolbar. It elaborates on the same information used by the Shade Wizard and indicates the three documents with the highest estimate of relevance. The highest ranked document is marked with three stars (document ranked 26 on the screenshot), the next one with two (ranked 22), and the third one is marked with one star (ranked 11). The stars are placed both by the corresponding document sphere and at the start of document title.

[1] The selection of colors reflects a common idea in the western world of green as equivalent to "go" and red as a synonym of "stop". The colors can be easily changed to reflect any other scheme using the preference commands.

While the Shade Wizard provides a global overview of how the relevance estimations are distributed in the document set, the Star Wizard points the user directly to the most likely relevant documents.

3. Implementation

We have implemented the Lighthouse system following the client-server model. The client accepts the query and transmits it to the server. The server forwards the query to the search engine, collects the results as a list of URLs and descriptions in HTML format, parses these results, collects the corresponding web pages, parses and indexes the text of each page. For each page it then creates a weighted vector of terms that represent that page, computes the distances between those vectors, generates the configurations for both 2- and 3-dimensional visualizations, and returns this data to the client. The server is written in Perl and C. It takes 0.5 sec to parse and index the documents, and another 0.5 sec to generate the spatial configuration on a computer with 600MHz Alpha CPU. The total time of a retrieval session is generally between 50 and 100 seconds, where most of the time is spend accessing the search engine and downloading the web pages. The efficiency also depends on the current network congestion. The client side is written in Java (language version 1.1) and handles all the interaction between the system and the user including the necessary computations for the wizard tools. It can be installed and run locally as an application or it can be downloaded on the fly and run in a web-browser as an applet. The system is located at our web site [8]. Note that our server is setup to process only one query at a time to avoid overloading the machine.

4. Conclusions

We have described Lighthouse, an interface system for an on-line search engine that integrates the traditional ranked list with the clustering visualization. Lighthouse displays documents as spheres floating in 2- or 3-dimensional visualization space positioned in proportion to the inter-document similarity. The system accepts user relevance judgments and estimates the relevance values for the remainder of the retrieved set. Lighthouse includes two wizard tools that present these relevance estimations to the user using color, shape, and symbolic markings, directing the user towards the most likely relevant documents.

The design choices incorporated into Lighthouse are motivated by an intensive off- and on-line evaluation of the clustering visualization [7]. That study suggests that Lighthouse can be a very effective tool for helping the user to locate interesting information among the documents returned by an information retrieval system. Our experience with the system implementation described in this paper illustrates that Lighthouse is fast and can be deployed in the web-based on-line settings.

Acknowledgments

The authors thank Victor Lavrenko for the help in implementing the document parsing and indexing parts of the Lighthouse server.

This material is based on work supported in part by the National Science Foundation, Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623, SPAWARSYSCEN-SD grant number N66001-99-1-8912, and also supported in part by United States Patent and Trademark Office and Defense Advanced Research Projects Agency/ITO under ARPA order number D468, issued by ESC/AXS contract number F19628-95-C-0235. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsors.

References

[1] J. Allan. Incremental relevance feedback for information filtering. In *Proceedings of ACM SIGIR*, pages 270–278, 1996.

[2] M. Chalmers and P. Chitson. Bead: Explorations in information visualization. In *Proceedings of ACM SIGIR*, pages 330–337, June 1992.

[3] D. Harman and E. Voorhees, editors. *The Fifth Text REtrieval Conference (TREC-5)*. NIST, 1997.

[4] A. Leuski. Relevance and reinforcement in interactive browsing. Technical Report IR-208, Department of Computer Science, University of Massachusetts, Amherst, 2000.

[5] A. Leuski and J. Allan. Details of Lighthouse. Technical Report IR-212, Department of Computer Science, University of Massachusetts, Amherst, 2000.

[6] A. Leuski and J. Allan. Evaluating a visual navigation system for a digital library. *International Journal on Digital Libraries*, 2000. Forthcoming.

[7] A. Leuski and J. Allan. Improving interactive retrieval by combining ranked lists and clustering. In *Proceedings of RIAO'2000*, pages 665–681, April 2000.

[8] Lighthouse. `http://toowoomba.cs.umass.edu/~leouski/lighthouse/`.

[9] M. M. Sebrechts, J. V. Cugini, S. J. Laskowski, J. Vasilakis, and M. S. Miller. Visualization of search results: a comparative evaluation of text, 2d, and 3d interfaces. In *Proceedings of ACM SIGIR*, pages 3–10, 1999.

[10] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979. Second edition.