VISUALISING SEQUENCES OF QUERIES : A NEW TOOL FOR INFORMATION RETRIEVAL

Russell Beale, [1] Rodger J. McNab [2] and Ian H. Witten [2]

[1]School of Computer Science
University of Birmingham
Edgbaston, Birmingham
B15 2TT UK
R.Beale@cs.bham.ac.uk

[2]Department of Computer Science
University of Waikato
Hamilton, New Zealand
{rjmcnab, ihw}@cs.waikato.ac.nz

Abstract

This paper describes a system that uses visualisation to assist a user in dealing with the information returned from a search engine. The user's queries, and the documents they return, are represented by a 3D spatial structure that shows their relationships and provides a way of accessing and exploring the documents retrieved. It is implemented to work with the New Zealand Digital Library, a set of large document collections that is available over the Web. The visualisation scheme is a Java applet that is updated dynamically whenever the user makes a new search, and can be browsed alongside the search engine.

Introduction

Retrieval of information using existing search engines presents challenging problems. Direct, explicit, searching is the dominant method of information retrieval today, and the extraordinary explosion in use of search engines, prompted by the sheer unwieldiness of the World-Wide Web, has vastly increased the number of people that must deal with information retrieval systems as part of their daily lives. Yet search engines, for all their features, are ill-equipped to support the actual processes involved in seeking information, processes that involve not just individual queries but sequences of related queries that are generated when homing in on a piece of desired information.

Many projects have addressed the question of visualising the contents of document collections; Card (1996) contains a survey. For example, the "galaxy" visualisation method plots each document as a point in space whose coordinate axes are determined using multidimensional scaling of the document similarity matrix; the space is then projected onto a plane. Kohonen's "self-organising map" is an unsupervised learning algorithm for analysing and visualising high-dimensional data which can also be applied to document spaces (Kohonen, 1996). When document collections contain some structure such as explicit links, other techniques can be used, for example hyperbolic coordinate systems (Munzner and Burchard, 1994) and navigational aids like HotSauce and the Navigational View Builder (Mukherjea and Foley, 1995). When even more structure is present, visualisers such as the Butterfly (Mackinlay et al, 1995) become applicable. All these visualisation schemes expedite the browsing of a document collection by departing radically from the model of information retrieval by textual query.

This paper describes a system that uses a self-organising visualisation to assist a user in dealing with the information returned from a search engine. The system is context-dependent in that it provides a visualisation of the results of particular searches. Moreover, it is unique in that it explicitly supports sequences of related queries, which is by far the most prevalent way in which search engines are used in practice. The information is displayed in a three-dimensional abstract visualisation that lets the user examine relationships between the sets of documents returned by different queries, identify and revisit documents that have been scanned before, and look at documents that have been returned but not yet examined. The key to this novel approach is the use of textual retrieval to select a (possibly quite large) set of potentially relevant documents out of a huge library, and to focus on the relationships between these documents through visualisation techniques. These two activities can proceed in tandem: the visualisation builds up in real time as queries are issued.

The structure of the paper is as follows. The next section describes HyperSpace, a generic tool for the production of visualisations that are self-organising in that they do not require any extra information to be added for the purpose of display. Construction of the visualisation scheme is greatly facilitated by the fact that we have available a search engine, a large database of documents, and a Web interface, that has been developed as part of the New Zealand Digital Library project (Witten *et al.*, 1996); we briefly describe this next because it provides an essential test-bed for the new visualisation scheme. Following that, we describe the visualisation scheme itself, and illustrate its operation pictorially. The implementation is not yet in a state where we can conduct a formal evaluation of the scheme, but we end with some remarks about its usage.

HyperSpace: a visualisation tool

HyperSpace is a general-purpose visualisation tool based on Narcissus (Drew *et al.*, 1995) which uses a representation of 3D space to generate images of data. There are two forms of basic representation within the space: nodes and links. Nodes are spherical objects; links join nodes. Each of these basic types has a defined behaviour, which allows the structures produced to organise themselves into a steady minimum energy state. This self-organisation occurs within the virtual 3D space, the nodes and links shuffling around until they reach a steady state. This produces a consistent visual representation for similar structural models. The physics within the space can be complex, but essentially nodes tend to repel each other, which spreads out the visualisation, whilst links act as springs pulling things together. HyperSpace has been successfully used to visualise a range of systems, particularly the World-Wide Web (Wood *et al.*, 1995).

As any type of data can be visualised within the system, decisions have to be made as to which attributes of the data one wishes to inspect. This usually determines what in the raw data will be nodes and what will be links, though careful choices have to be made as to how subsidiary attributes such as node mass and link strength are mapped, as poor choices make the resulting structures semantically meaningless. Sensible choices ensure that a coherent and comprehensible representation will be achieved. For this to happen, the mapping has to reflect the notion that spatial proximity in the evolved structure corresponds to a notion of closeness within the raw data set. For example, in producing visualisations of the Web, URLs were mapped to nodes whilst hypertext links from a page were mapped to links. The resultant visualisations drew highly-linked pages close together, so that clusters of nodes all represented pages that had similar content. We could have chosen to organise the system by the temporal order in which pages were visited, or by geographical location, but the clustering of similar concepts was felt to be a more useful approach.

The original version of HyperSpace is written in C++; for ease of integration with the NZDL we have re-implemented it in Java.

The NZ Digital Library project

The New Zealand Digital Library (NZDL) is a publicly-accessible system that provides full-text indexes to several substantial collections of information. The most prominent is the Computer Science Technical Report collection, containing nearly 40,000 technical reports collected from over 300 sites worldwide—over a million pages of information, culled from 34 Gbyte of PostScript files, along with 25,000 figures extracted from the reports. The NZDL contains several other smaller collections, such as the 500-volume Gutenberg collection of English literature, a collection comprising all Internet FAQ lists, and so on.

Users locate particular documents in the Library by full-text retrieval. All words in the collections are indexed (over 400 million of them in the case of the Computer Science Technical Reports). Search terms can be combined with the logical operators “and”, “or” and “not” to create a Boolean query. Alternatively a general query can be issued and the top few documents—the ones matching most closely according—are returned. Ranking is done using a vector-space model of word frequencies, and taking the cosine of the angle between the vectors representing the query and the document; a standard technique in information retrieval (Salton and McGill, 1983). The full-text index is provided by MG, a system for compressing and indexing large collections of documents (Witten *et al*., 1994).

For either type of query, response is very quick, and the plain text of each matching document, along with facsimile images of the first couple of pages, is instantly made available for browsing. In addition, a pointer allows the original formatted document to be downloaded from its home site.

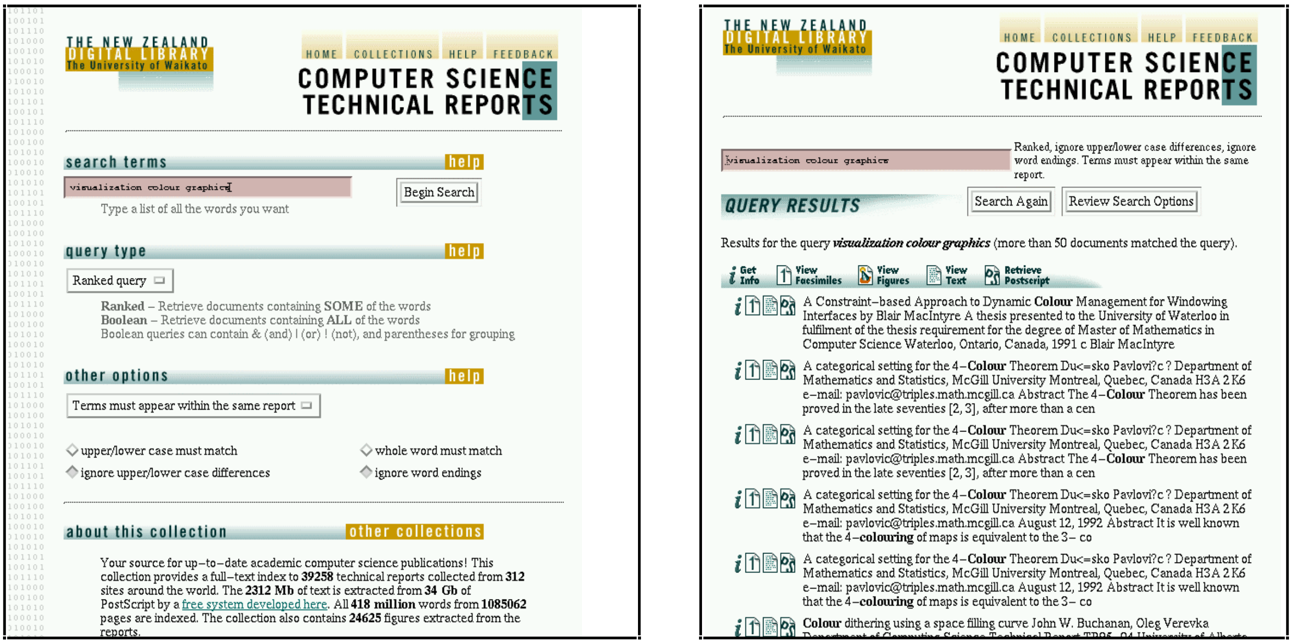
The NZDL software makes it relatively easy to integrate the visualiser into the interactive query process. If we attempted to visualise the results of a third-party information service like AltaVista, it would not be so easy to gather the necessary information at the time that the query was processed. Our current implementation simply identifies the user and appends the queries, and the documents returned by them, to a file which can then be read by the visualisation system. This file is cleared automatically when the query session is over, though we are investigating retaining the history across multiple sessions as well.

Figure 1 shows a query being made on the Computer Science Technical Report collection. On the left is the query page, into which the user has typed the search terms *visualisation colour graphics*. A ranked query has been selected, and the user has chosen to seek these terms in the same reports (rather than in the same paragraph, or the same page, or on the front page of reports). On the right is the query results page that is returned to the user, which contains entries for the first ten matching reports (only six of which are visible in the Figure). For each one, the first few words of the report are shown, along with icons that give access to information about the report (i.e. where it came from), facsimile images of the first page or two, images of the figures that have been extracted from the report, the full text of the report, and the PostScript file—which, unlike the other information, will be downloaded from the original site from which this report was harvested.

Visualising sequences of queries

The NZDL has an HTML-based interface that provides general access to the database across the Web. The HyperSpace system is integrated into this transparently as an applet that opens up a separate window for itself in which the visualisation operates. Once launched, this is independent of the browser.

The browser allows the user to pose queries to the full-text index, and each query returns a



visualization colour graphics

Type a list of all the words you want

Ranked query

Ranked – Retrieve documents containing **SOME** of the words
Boolean – Retrieve documents containing **ALL** of the words
Boolean queries can contain & (and) | (or) ! (not), and parentheses for grouping

Terms must appear within the same report

◇ upper/lower case must match ◇ whole word must match
◆ ignore upper/lower case differences ◆ ignore word endings

Your source for up–to–date academic computer science publications! This collection provides a full–text index to **39258** technical reports collected from **312** sites around the world. The **2312 Mb** of text is extracted from **34 Gb** of PostScript by a free system developed here. All **418 million** words from **1085062** pages are indexed. The collection also contains **24625** figures extracted from the reports.

Search interface

visualization colour graphics

Ranked, ignore upper/lower case differences, ignore word endings. Terms must appear within the same report.

Results for the query ***visualization colour graphics*** (more than 50 documents matched the query).

A Constraint–based Approach to Dynamic **Colour** Management for Windowing Interfaces by Blair MacIntyre A thesis presented to the University of Waterloo in fulfilment of the thesis requirement for the degree of Master of Mathematics in Computer Science Waterloo, Ontario, Canada, 1991 c Blair MacIntyre

A categorical setting for the 4–**Colour** Theorem Du<=sko Pavlovi?c ? Department of Mathematics and Statistics, McGill University Montreal, Quebec, Canada H3A 2K6 e–mail: pavlovic@triples.math.mcgill.ca Abstract The 4–**Colour** Theorem has been proved in the late seventies [2, 3], after more than a cen

A categorical setting for the 4–**Colour** Theorem Du<=sko Pavlovi?c ? Department of Mathematics and Statistics, McGill University Montreal, Quebec, Canada H3A 2K6 e–mail: pavlovic@triples.math.mcgill.ca Abstract The 4–**Colour** Theorem has been proved in the late seventies [2, 3], after more than a cen

A categorical setting for the 4–**Colour** Theorem Du<=sko Pavlovi?c ? Department of Mathematics and Statistics, McGill University Montreal, Quebec, Canada H3A 2K6 e–mail: pavlovic@triples.math.mcgill.ca August 12, 1992 Abstract It is well known that the 4–**colouring** of maps is equivalent to the 3– co

A categorical setting for the 4–**Colour** Theorem Du<=sko Pavlovi?c ? Department of Mathematics and Statistics, McGill University Montreal, Quebec, Canada H3A 2K6 e–mail: pavlovic@triples.math.mcgill.ca August 12, 1992 Abstract It is well known that the 4–**colouring** of maps is equivalent to the 3– co

Colour dithering using a space filling curve John W. Buchanan, Oleg Verevka

Query results

Figure 1 Searching with the NZDL

maximum of fifty documents that satisfy it. If the query is a "ranked" (rather than a boolean) one, these are the fifty documents judged most relevant to the query according to the cosine rule mentioned above. The serial numbers of the documents returned are stored in the user's log file on the server. Each new query adds another set of documents to the log file. This data is parsed and mapped into an appropriate representation for the HyperSpace system.

The documents returned are designated as HyperSpace nodes. Each query made by the user is also mapped to a node. Links are drawn between each document and the query that returned it. This structure dynamically updates itself as the user makes a series of queries.

Completely separate independent queries produce a series of "dandelion heads"—unconnected clusters of nodes, each one centred on the query that generated it. More interesting patterns appear when the queries are related, because if the same document is identified by different queries it becomes linked to more than one node. A whole series of queries on one topic will produce a more complex pattern comprising a densely connected mass of nodes in which the relationship between different queries can be discerned in terms of the degree of overlap (and hence commonality) of the documents they generate.

We are experimenting with different ways of mapping some of the information in the space of retrieved documents into visual aspects of the display. For example, the size of each node indicates the size of the document that it represents. This has the effect of pushing larger documents farther out. The relevance of a document to a query governs the strength of the link joining them, so that more relevant documents move closer to the query, and documents that are very relevant to two or more queries have the effect of pulling these queries together. Whether or not a document has been viewed in this query session determines the colour of the node representing it: thus it is very easy to see which documents have been examined so far.

Whether these effects provide a convenient and easily-grasped display for users is an experimental question. For example, increasing its relevance brings a document closer to a query, whereas increasing its size pushes it further away: this can be rationalised if one regards longer documents as inherently less useful than short ones that are just as relevant.

Using the visualisation scheme

The system is designed to be interactive: the browsing view is not merely a representation of the document space structured according to the query, but is an aid to navigating that space. Any node in the visualisation can be selected, causing the document that it refers to be returned in the browser window. As the user zooms in towards a particular node or set of nodes, some information associated with each node appears as a label on the display, allowing the user to identify documents precisely. The user can therefore view any particular document, and then continue with their search, perhaps refining it, perhaps moving off in a new direction, confident that they can return to previously-visited documents with ease. By colour-coding the nodes according to whether they have been viewed or not, the user can immediately see the effects of both browsing and searching. If a node corresponds to a query, the text of the query appears when the user approaches that node.

Figure 2 shows the visualisation produced for the three-term query illustrated in Figure 1, *visualisation colour graphics*, issued as a ranked query with stemming and case-folding in place (the default). Fifty documents are returned and are shown spread around the central node, pulled in by relevance and pushed out by size. Although the display looks a bit messy, the user can navigate around the space to get a feel for how the data is configured. The system allows rotation in 3D about an arbitrary, user-selectable point, and supports zooming in and out at will. Moreover, the nodes are coloured on the display, the query being shown in a different colour from the documents, and documents which have been viewed during this session in a different colour again. It should be noted that the 3D effect is more apparent on the screen, where the user can rotate and zoom the structure in real time.

Figure 3 shows the effect of making a second query, for the three terms *3D surface graphics*: the display is automatically updated as soon as the query is made. The new query is on the left. When the user zooms in, the labels identifying the queries become apparent, but are omitted from the wider view to avoid screen clutter. It is apparent that there are two documents in common between the two queries, and the user may well wish to browse them at this stage by clicking on the spheres that represent them. Performing this action causes the applet to send the document URL to the browser for display.

Suppose instead the user issues a third query, this time for the single term *agents*. It is clear from the display shown in Figure 4 (in which the new query is shown on the left) that the top fifty documents returned for this query have no overlap with those returned by the other queries. The other two queries have retained their structure but drifted away from this most recent query.

Finally, Figure 5 shows the result of a fourth query being added to the sequence, for *collaborative agent visualisation*. Because this relates strongly to both the *agents* query and the *visualisation colour graphics* one, it has the effect of connecting up the document sets, and the queries automatically fall into the order (from left to right) *agents*, *collaborative agent visualisation*, *visualisation colour graphics*, and *3D surface graphics*. It is clear that almost all the documents returned for the final query are related to either the *agents* query or to *visualisation colour graphics*; there are only three that are not. However, none of these documents are related to *3D surface graphics*.

Conclusions

This system provides an interactive visualisation of a set of documents that match a potentially complex set of queries, and highlights clearly the relationships between them. One of the major areas under investigation at present is the detailed mapping between document information and the visualisation space: not all users agree that smaller documents are inherently more useful, for example, and so perhaps documents should not be pushed away from a query simply by virtue of their size. Additionally, whilst proximity to the query should suggest that a document closely matches that query, many users report that they first focus on the larger nodes that occupy less cluttered regions of space. Numerous options exist: node size could be kept constant, with density altered according to document size. Alternatively, we could make more use of colour coding within the display. Once the final development work is completed, a detailed regimen of user testing will be carried out to test these ideas.

In initial trials with users, qualitative reactions have been very encouraging. The

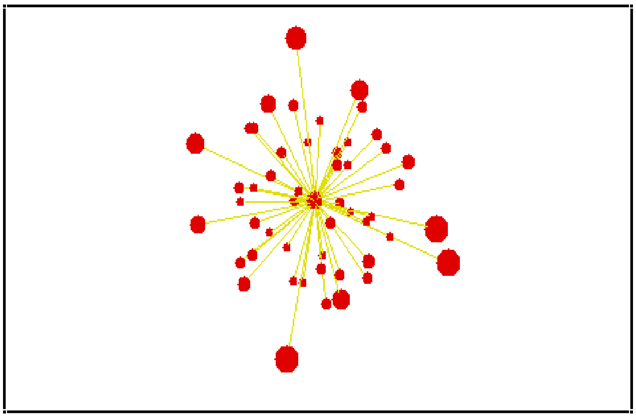


Figure 2 Visualising the result of one query: *visualisation colour graphics*

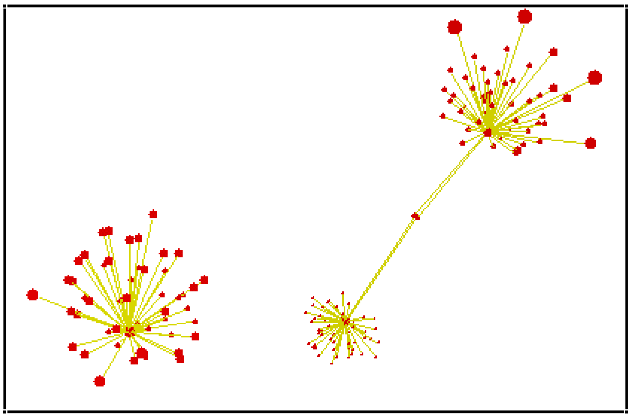


Figure 4 Adding a third, unrelated, query: *agents*

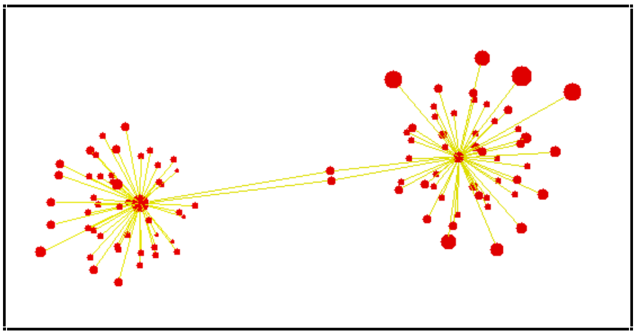


Figure 3 Adding a second query: *3D surface graphics*

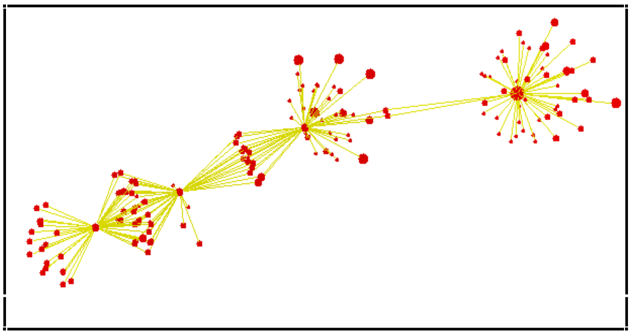


Figure 5 A sequence of four queries

visualisation seems to succeed in providing both a detailed history and good visual feedback relating to the efficacy of the user's search strategy.

References

Card, S.K. (1996) "Visualising retrieved information: a survey." *IEEE Computer Graphics and Applications* pp. 63–67; March.

Drew, N.S., Hendley, R.J., Wood, A.M., and Beale, R. (1995) "Narcissus: Visualising Information." *Proc IEEE Symposium on Information Visualisation* , Atlanta Georgia USA, pp. 90–96; October.

Kohonen, T. (1996) "Websom—Self-organising map for Internet exploration." URL <http://websom.hut.fi/websom/>.

Mackinlay, J.D., Rao, R. and Card, S.K. (1995) "An organic user interface for searching citation links." *Proc CHI 95*, pp. 67–73; May.

Mukherjea, S. and Foley, J.D. (1995) "Visualising the World-Wide Web with the navigational view builder." *Computer Networks and ISDN Systems 27*: 1075–1087.

Munzner, T. and Burchard, B. (1994" "Visualising the structure of the World Wide Web in 3D hyperbolic space." URL <http://www.geom.umn.edu:80/docs/research/webviz/>.

Salton, G. and McGill, M.J. (1983) *Introduction to modern information retrieval*. New York: McGraw Hill.

Witten, I.H., Moffat, A. and Bell, T.C. (1994) *Managing gigabytes: compressing and indexing documents and images*. New York: Van Nostrand Reinhold.

Witten, I.H., Nevill-Manning, C.G. and Cunningham, S.J. (1996) "Digital libraries based on full-text retrieval" *Proc Webnet'96*, San Francisco, pp. 486–495, October.

Wood, A.M., Drew, N.S., Beale, R., and Hendley, R.J. (1995) "HyperSpace: Web Browsing with Visualisation." *Third International World-Wide Web Conference Poster Proceeding*, Darmstadt Germany, pp. 21–25; April.