Browsing around a digital library

Ian H. Witten

Department of Computer Science
University of Waikato
Hamilton
NEW ZEALAND
ihw@cs.waikato.ac.nz

Abstract. What will it be like to work in the digital library of the future? We begin by browsing around an experimental digital library of the present, glancing at some collections and showing how they are organized. Then we look to the future. Although present digital libraries are quite like conventional libraries, we argue that future ones will feel qualitatively different. Readers—and writers—will work "inside" the library using a kind of context-directed browsing. This will be supported by structures derived from automatic analysis of the contents of the library—not just the catalog, or abstracts, but the *full text* of the books and journals—using new techniques of text mining.

Introduction

Over sixty years ago, science fiction writer H.G. Wells was promoting the concept of a "world brain" based on a permanent world encyclopedia which "would be the mental background of every intelligent [person] in the world. It would be alive and growing and changing continually under revision, extension and replacement from the original thinkers in the world everywhere. … even journalists would deign to use it" (Wells, 1937). Eight years later, Vannevar Bush, the highest-ranking scientific administrator in the U.S. war effort, invited us to "consider a future device for individual use, which is a sort of mechanized private file and library … a device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility" (Bush, 1945). Fifteen years later, J.C.R. Licklider, head of the U.S. Department of Defense's Information Processing Techniques Office, envisioned that human brains and computing machines would be coupled together very tightly, and imagined this to be supported a "network of 'thinking centers' that will incorporate the functions of present-day libraries together with anticipated advances in information storage and retrieval" (Licklider, 1960). Today, we are accustomed to hearing similar pronouncements from the U.S. President.

Digital libraries, conceived by visionary thinkers and fertilized with resources by today's politicians, are undergoing a protracted labor and birth. Libraries are society's repositories for knowledge, and digital libraries are of the utmost strategic importance

in a knowledge-based economy. Not surprisingly, many countries have initiated large-scale digital library projects. Three years ago the DL-I initiative was set up in the U.S. (and is now entering a second phase); in the U.K. the Elib program was set up at about the same time; other countries in Europe and the Pacific Rim have followed suit. Digital libraries will likely figure amongst the most important and influential institutions of the 21st Century.

But what is a digital library? A simple working definition is

> *a focused collection of digital objects, including text, video, and audio, along with methods for access and retrieval, and for selection, organization, and maintenance of the collection.*

This definition deliberately gives equal weight to user (access and retrieval) and librarian (selection, organization and maintenance). Other definitions in the literature, emanating mostly from technologists, omit—or at best downplay—the librarian's role, which is unfortunate because it is the selection, organization, and maintenance that will distinguish digital libraries from the anarchic mess that we call the World Wide Web. However, digital libraries tend to blur the distinction between these heretofore very different kinds of user—because the ease of augmenting, editing, annotating and re-organizing electronic collections means that they will support the development of new knowledge *in situ*.

What's it like to work in a digital library? Will it feel like a conventional library, but more computerized, more networked, more international, more all-encompassing, more convenient? I believe the answer is no: it will feel qualitatively different. Not only will it be with you on your desktop (or at the beach, or in the plane), but information workers will work "inside" the library in a way that is quite unlike how they operate at present. It's not just that knowledge and reference services will be fully portable, operating round the world, around the clock, throughout the year, freeing library patrons from geographic and temporal constraints—important and liberating as these are. It's that when new knowledge is created it will be fully contextualized and both sited within and cited by existing literature right from its conception.

In this paper, we browse around a digital library, looking at tools and techniques under development. "Browse" is used in a dual sense. We begin by browsing a particular collection, and then look briefly at some others. Then we examine the digital library's ability to support novel browsing techniques. These situate browsing within the reader's current context and unobtrusively guide them in ways that are relevant to what they are doing, giving this feeling of working "inside" the library that I am trying to convey. Context-direct browsing is supported by structures derived from automatic analysis of the library's contents—not just the catalog, or abstracts, but the *full text* of the documents—using techniques that are being called "text mining." Of course, other ways of finding information are important too—user searching, librarian recommendations, automatic notification, group collaboration—but here we focus on browsing.

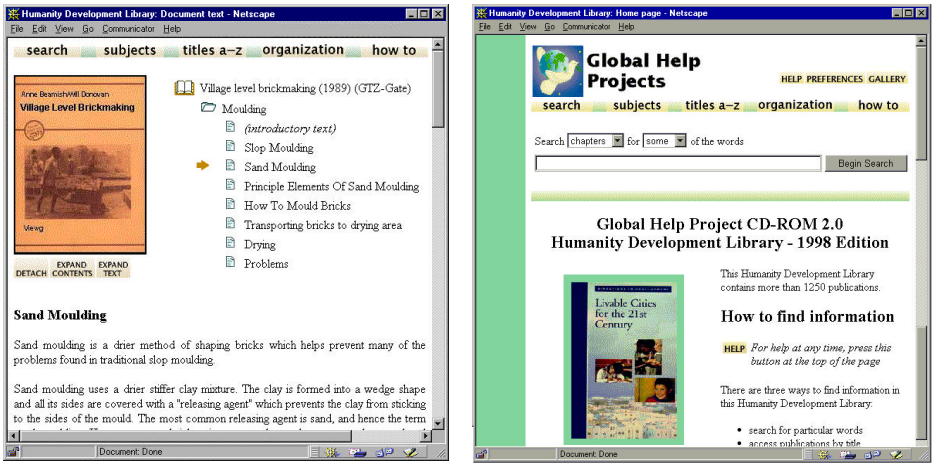


Figure 1 (a) *Village Level Brickmaking* (b) HDL home page

The Humanity Development Library

Figure 1a shows a book in the *Humanity Development Library* (HDL), a collection of humanitarian information put together by the Global Help Project to address the needs of workers in developing countries (http://www.nzdl.org/hdl). This book might have been reached by a directed full-text search, or by browsing one of a number of access structures, or by clicking on one of a gallery of images. On opening the book, which is entitled *Village Level Brickmaking*, a picture of its cover appears at the top, beside a hierarchical table of contents. In the figure, the reader has drilled down into a chapter on *moulding* and a subsection on *sand moulding*, whose text appears below. Readers can expand the table of contents to what is in the section or even the whole book; and expand the text likewise (which is very useful for printing). The ever-present picture of the book's cover gives a feeling of physical presence and a constant reminder of the context.

Readers can browse the collection in several different ways, as determined by the editor who created it. Figure 1b shows the home page, at the top of which (underneath the logo) is a bar of buttons that open up different access mechanisms. A subject hierarchy provides a tree-structured classification scheme for the books. Book titles appear in an alphabetical index. A separate list gives participating organizations and the material that they contributed. A "how-to" list of helpful hints, created by the collection's editor, allows books to be accessed from problem-oriented key phrases. However a book is reached, it appears in the standard form illustrated in Figure 1a, along with the cover picture to give a sense of presence. The different access mechanisms help solve the librarian's dilemma of where to place a book on the shelves (Mann, 1993): each one appears on many different virtual shelves, shelves

that are organized in different ways.

Full-text search of titles and entire documents provide important additional access mechanisms. The search engine that we use, MG (Witten *et al.*, 1994), supports searching over the full text of the document—not merely a document surrogate as in conventional digital library retrieval systems. User feedback from an earlier version of this collection indicated that Boolean searching was more confusing than helpful for the targeted users. Previous research suggests that difficulties with Boolean syntax and semantics are common, and transaction log analysis of several library retrieval systems indicates that by far the most popular Boolean operator is AND; the others are rarely used. For all these reasons, the interface default for this collection is ranked queries. However, to enable users to construct high-precision AND searches where necessary, selecting "search … for *all* the words" in the query dialog produces the syntax-free equivalent of an AND query.

Just as libraries display new acquisitions or special collections in the foyer to pique the reader's interest, this library's home page (Figure 1b) highlights a particular book that changes every few seconds, it can be opened by clicking on the image. This simple display is extraordinarily compelling. And just as libraries may display a special book in a glass case, open at a different page each day, a "gallery" screen shows an ever-changing mosaic of images from pages of the books, remarkably informative images that, when clicked, open the book to that page. There is also a scrolling "Times Square" display of randomly selected phrases that, when clicked, take you to the appropriate book. The possibilities are endless.

This is a focused collection of 1250 books—miniscule by library standards, but nevertheless surprisingly comprehensive within the targeted domain. It contains 53,000 chapters, 62 million words, and 32,000 pictures. Although the text occupies 390 MB, it compresses to 102 MB and the two indexes—for titles and chapters respectively—compress to less than 80 MB. The images (mostly in PNG format) occupy 290 MB. Associated files bring the total size of the collection to 505 MB. Even if there were twice as much text, and the same images, it would still fit comfortably on a CD-ROM, along with all the necessary software. A single digital videodisk would hold a collection twenty times the size—still small by library standards, but immense for a fully portable collection.

An experimental testbed: The New Zealand Digital Library

The HDL is just one of the twenty or so collections produced by the New Zealand Digital Library (NZDL) project. Operational for several years now, this project aims to develop the underlying infrastructure for digital libraries and provide example collections that demonstrate how it can be used. Most of the collections are publicly accessible over the Web. The library is international: there are interfaces in English, Maori, French, German, and Arabic, and collections have been produced in all these languages. Digital libraries are particularly empowering for the disabled, and there is a

text-only version of the interface intended for visually impaired users.

The editors of the HDL have gone to great lengths to provide a rich set of access structures. However, this is a demanding, labor-intensive task, and most collections are not so well organized. The basic access tool in the NZDL is full-text searching, which is available for all collections and is provided completely automatically. Some collections allow, in addition, traditional catalog searching based on author, title, and keywords, and full-text search within abstracts. Our experience is that while the user interface is considerably enhanced when traditional library cataloging information is available, it is prohibitively expensive to create formal cataloging information for many electronically-gathered collections. With appropriate indexes, full-text retrieval can be used to approximate the services provided by a formal catalog.

Collections

The core of any library is the collections it contains. A few examples will illustrate the variety and scope of the services provided.

The collection of *Computer Science Technical Reports* contains 46,000 reports—1.3 million pages, half a billion words—extracted automatically from 34 GB of raw PostScript. There is no bibliographic or "metadata" information: we have only the contents of the reports (and the names of the FTP sites from which they were gathered). Many are Ph.D. theses which would otherwise be effectively lost except to a small local community: full-text search reaches right inside the documents and makes them accessible to anyone looking for information on that topic.

Figure 2 A query to the *Computer Science Technical Reports* (German interface)

Many different languages are represented in this collection. Using the German-language interface option, Figure 2 shows the page received in response to a query for the word *Boolesche*; this returns several German documents. Accents are supported by and are included in searches when specified: the incorrect display of umlauted characters in Figure 2 is due to deficiencies in the PostScript extraction process. The raw, unpolished, form of Figure 2 compared with Figure 1 reflects the difference between a carefully edited set of documents, including hand-prepared classification

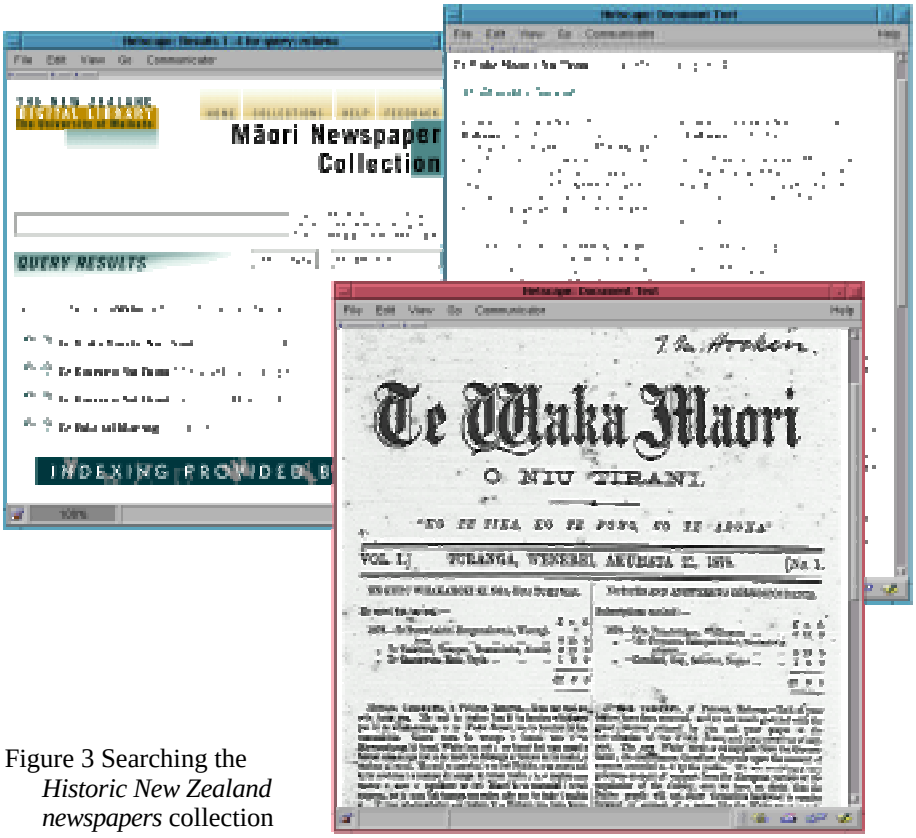


Figure 3 Searching the *Historic New Zealand newspapers* collection

indexes and other metadata, and a collection of information pulled mechanically off the Web and organized without any human intervention at all.

In the *Computer Science Technical Reports*, as (perhaps) befits the different target audience, the query interface is more comprehensive than that of the HDL. Case-folding and stemming can be independently enabled or disabled, and full Boolean query syntax is supported as well as ranked queries. Moreover, searches can be restricted to the first page of reports, which approximates an author/title search in the absence of specific bibliographic details of the documents.

An expressly bilingual collection of *Historic New Zealand Newspapers* contains issues of forty newspapers published between 1842 and 1933 for a Maori audience. Collected on microfiche, these constitute 12,000 page images. Although they represent a significant resource for historians, linguists and social scientists, their riches remain largely untapped because of the difficulty of accessing, searching and browsing material in unindexed microfiche form. Figure 3 shows the parallel English–Maori text retrieved from the newspaper *Te Waka Maori* of August 1878 in response to the query *Rotorua*, a small town in New Zealand. Searching is carried out on electronic text produced using OCR; once the target is identified, the corresponding page image can be displayed.

Text mining: Keyphrase extraction and soft parsing

The HDL illustrates the power of handcrafted information to help support browsing.

But often resources are not available to employ editors to create, manually, different views of the information to support a rich browsing environment, nor even librarians to catalog the collection. And the presence of full text offers even richer possibilities for browsing, by adding structure in the form of hyperlinks or descriptive keyphrases—but only if the requisite information can be extracted from the text. What is the potential for finding relevant information automatically?

Data mining, a burgeoning new technology, is about looking for patterns in data. Likewise, text mining is about looking for patterns in text. More formally, it may be defined as the process of analyzing text to extract information that is useful for particular purposes. Compared with the kind of data stored in databases, text is unstructured, amorphous, and difficult to deal with. Nevertheless, the motivation for trying to extract information from it is compelling.

Keyphrase extraction

Table 1 shows the titles of two research papers, with two sets of keyphrases for each one. In each case, one set gives the keyphrases assigned by its author, and an algorithm that analyzes the paper's text (excluding the author-assigned keyphrases) determined the other. Phrases in common between the two sets are italicized. Which set is which? It is not hard to guess that the keyphrases on the left are the author's, while those on the right are assigned automatically. While many of the automatically-assigned keyphrases are plausible, some are rather strange. Examples are "gauge" and "smooth" for the first paper, and especially "garbage" for the second—while that word my be used repeatedly in a computer science paper, and even displayed prominently in the title, no author is likely to choose it as a keyword for their paper! Although automatically-assigned keyphrases may not reflect exactly what the author might have chosen—and authors, of course, have a whole variety of reasons for selecting particular words and phrases—they are useful for many purposes, in the next section we examine browsing interfaces that use them.

There are two rather different methods for automatically determining keyphrases for papers. Both use machine learning methods, and require a set of documents with keyphrases already assigned for "training." The first is to have a predefined set from which all keyphrases are chosen—in information retrieval, this is known as a "controlled vocabulary." Then the training data provides, for each keyphrase, a set of documents that are associated with it. A new document is compared to all the training documents, and its keyphrases are drawn from those attached to the most similar documents. The second method is to use linguistic and information retrieval techniques to extract phrases from the text of the new document that are likely to be characteristic of it. Here, the training set is used to tune the parameters of the extraction algorithm, not to suggest the actual phrases.

The keyphrases in Table 1 were extracted using the second method. The procedure is as follows (Witten *et al*., in preparation; Turney, 1997). First, the input text is

regularized by deleting apostrophes, splitting words at hyphens, and using punctuation and non-words (such as numbers) to split the text into phrases. Then all subphrases of these preliminary phrases are taken to be candidate keyphrases, except ones that begin or end with any word in a rather long "stopword" list of 425 common words. Words are stemmed to remove their endings, and two phrases are considered the same if they contain the same sequence of stems. For each candidate keyphrase, some feature values are computed. Two useful features are the distance from the beginning of the document to the first occurrence of the keyphrase (normalized to lie between 0 and 1), and the TF×IDF, or "term frequency times inverse document frequency," measure for the keyphrase. This latter measure is widely used in information retrieval: it takes the number of times the term—in this case, phrase—appears in the document, and multiplies it by a factor that is small for common terms and large for rare (and hence more important) ones. Whether a phrase is common or not is determined using an auxiliary corpus of similar documents. We have experimented with a large number of other features, but find that these two give good performance in keyphrase extraction.

Once the features of each candidate keyphrase have been determined, they are combined into a single "probability" figure using a standard machine learning model (naïve Bayes). The role of the training data is both to provide material for the global frequency corpus, and to tune the parameters of this model. The resulting probability ranks candidate keyphrases in order of their likelihood of being actual keyphrases. Some post-processing is done on this ranked list (for example, further examination of keyphrases that are subphrases of other keyphrases), and the required number of phrases is taken from the top of the list (or, alternatively, a probability cutoff is used).

As Table 1 illustrates, this scheme works well; we will its application shortly.

Soft parsing

Soft parsing provides a way of automatically locating particular kinds of information in text, again driven by machine learning using training data. Ordinary documents are full of structured information: people's names, phone numbers, fax numbers, street addresses, email addresses, URLs, abstracts, tables of contents, lists of references, tabular data containing stock market information, amongst many others. Most of these items are detectable by special-purpose parsers, and some systems allow them to be specified by explicit grammars and acted on, for example, by "intelligent agents" (Nardi *et al*., 1998). An excellent example of how a digital library can benefit from the automatic identification of references in text is given by the CiteSeer system (Giles *et al.*, 1998), which is an automatic citation indexing tool driven by a robust, *ad hoc*, parser.

Neural multigrid for gauge theories and other disordered systems		*Proof nets, garbage, and computations*	
disordered systems	disordered	*cut-elimination*	cut
gauge fields	gauge	linear logic	*cut elimination*
multigrid	*gauge fields*	*proof nets*	garbage
neural multigrid	interpolation kernels	sharing graphs	*proof net*
neural networks	length scale	typed lambda-calculus	weakening
	multigrid		
	smooth		

Table 1 Different keyword sets for three computer science papers

However, there are many drawbacks to taking a hard-edged approach to parsing for such information. First, it is difficult to decide what to use as tokens. Second, parsing decisions are categorical and irrevocable. Third, it is difficult to generate—and worse still, debug and extend—appropriate grammars, because problem specifications are inherently incremental and evolutionary.

A novel approach, which is currently under development, is to locate tokens in context by considering the input as an interleaved string of information from different sources (Witten *et al*., 1998). Character-based language models provide a convenient and powerful way to recognize lexical structure. Tokens can be compressed using language models derived from different training data, and classified according to which model provides the most economical representation. The Viterbi algorithm, based on dynamic programming, can be used to determine an optimal sequence of models for a given text, and decide exactly where each one should begin and end. This algorithm has been used successfully to correct corrupted text (such as OCR produces) based on language models of the underlying text: our application is similar in that the text is "corrected" by inserting begin and end markers for the different kinds of token, with the essential difference that within each kind of token, a different language model is used appropriate to that token. All language models can be trained from a corpus of marked-up documents. The advantage is that if the specifications change incrementally, all that needs doing is to mark up some new documents and re-train the language models. This provides a convenient technique for both debugging and incremental development.

Happily—and surprisingly—this scheme can be applied hierarchically without any extra technical difficulty. A reference consists of a name, date, title, journal, volume number, issue number, page numbers, along with appropriate intervening characters. These characters are very highly determined: names are separated by ", " or " and ", between the name and date field is " (", etc. Many different forms exist, of course: examples of each must appear in the training data. Training data for the name, date, title, etc., models is very easy to come by: bibliography files are a convenient source. Some fields—such as author and editor—will be impossible to distinguish on the basis of their models alone, but the higher-level reference model will contain well-determined clues (such as the text "edited by") that allows them to be accurately identified.

Extending the soft parsing model to extract metadata—such as author and title—from plain text seems straightforward. Title pages have a characteristic structure that is easy to capture. It may be necessary to add features such as the distance from the start of the document (to help distinguish the author's name from other names that appear in the document's body); this appears to be easy to do using a machine learning model trained on example occurrences. Indeed, the algorithm for keyphrase extraction by determining candidate keyphrases and calculating appropriate features, as described in the previous subsection, can be fitted into the same soft parsing framework.

Browsing in the digital library of the future

Now that we have seen how text mining techniques allow some of the structure of text to be automatically elucidated, we demonstrate how they will be used to facilitate browsing in the digital library of the future. Current digital library systems often contain handcrafted indexes and links to provide different entry points into the information, and to link it together into a coherent whole. This can produce high-quality, focused collections—but it is basically unscalable. Excellent new material will, of course, continue to be produced using manual techniques, but it is infeasible to suppose that the mass of existing, archival material will be manually "converted" into high-quality digital collections. The only scalable solution that is used currently for amorphous information collections is the ubiquitous search engine—but browsing is poorly supported by standard search engines. They operate at the wrong level, indexing words whereas people think in terms of topics, and returning individual documents whereas people often seek a more global view.

We look first at automatic link generation through soft parsing, then at two browsing interface that capitalize on the existence of automatically-generated keyphrases. The first of these is a kind of search engine that is specifically designed to support topic browsing of large information collections. The second is a workbench that facilitates skimming, reading, and writing documents "within" a digital library—a qualitatively different experience from working in a library today.

Improved browsing using dynamic link generation

The items identified by soft parsing will be turned into dynamically-evaluated hyperlinks that are bound at click time to searches of appropriate indexes. When document collections are built manually, links are inserted by authors, editors, librarians. But this is not a scalable solution: links quickly go out of date as the collection grows. However, links that are evaluated dynamically are always current, because they perform a search for relevant items every time they are invoked.

References, located by soft parsing, will transport readers directly to the cited work (as does CiteSeer at present, but using hand-coded heuristics to detect references;

Giles *et al*., 1998). Names, identified in the source document by soft parsing, will take the reader to biography entries, or to works written by that person, or to their contact details. Searching appropriate information resources to come up with the relevant facts is the easy part: the hard bit is identifying, in the source text, the character strings that represent names. Locations will take the reader to maps, transport details, geographical gazetteers, and so on. The difference between this scheme and the grand "everything-is-linked" hypertext visions of pioneers like Ted Nelson is that here, nothing is done manually. The source of links is identified using soft parsing techniques, and full-text searching locates the targets. The only manual parts are providing training data for the soft parser, and deciding which collections (and which indexes) are to be consulted for each type of data.

Improved browsing using keyphrase indexes

We have built a new kind of search interface that is explicitly designed to support browsing (Gutwin et al., 1998). Automatically-extracted keyphrases form the basic unit of both indexing and presentation, allowing users to interact with the collection at the level of topics and subjects rather than words and documents. The system displays the topics in the collection, indicates coverage in each area, and shows all ways a query can be extended and still match documents.

The interface is shown in Figure 4. A user initiates a query by typing words or phrases and pressing the "Search" button, just as with other search engines. However, what is returned is not a list of documents, but a list of keyphrases containing the query terms. Since all phrases in the database are extracted from the source documents, every returned phrase represents one or more documents in the collection. Searching on the word *text*, for example, returns a list of phrases including *text editor* (a keyphrase for twelve documents), *text compression* (eleven documents), and *text retrieval* (ten documents) (see Figure 4). The phrase list provides a high-level view of the topics represented in the collection, and indicates, by the number of documents, the coverage of each topic.

Following the initial query, a user may choose to refine the search using one of the phrases in the list, or examine a topic more closely. Since they are derived from the collection itself, any further search with these phrases is guaranteed to produce results—and furthermore, the user knows exactly how many documents to expect. To examine the documents associated with a phrase, the user selects it from the list, and previews of documents for which it is a keyphrase are displayed in the lower panel of the interface. Selecting any preview shows the document's full text.

Experiments with users show that this interface is superior to a traditional search system for answering particular kinds of questions: evaluating collections ("what's in this collection"), exploring areas ("what subtopics are available in area X"), and general information about queries ("what kind of queries will succeed in area X", "how can I specialize or generalize my query"). However, it is not intended to replace

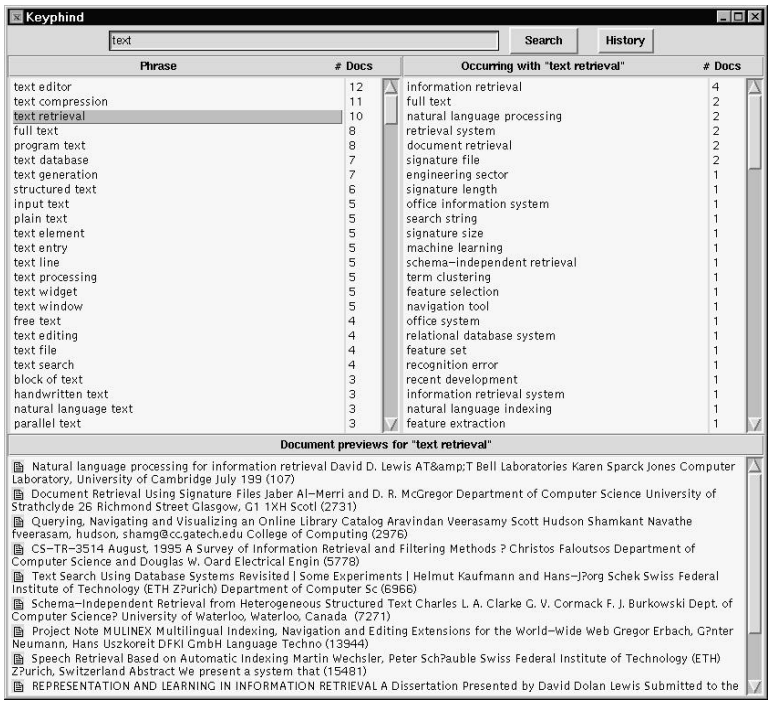


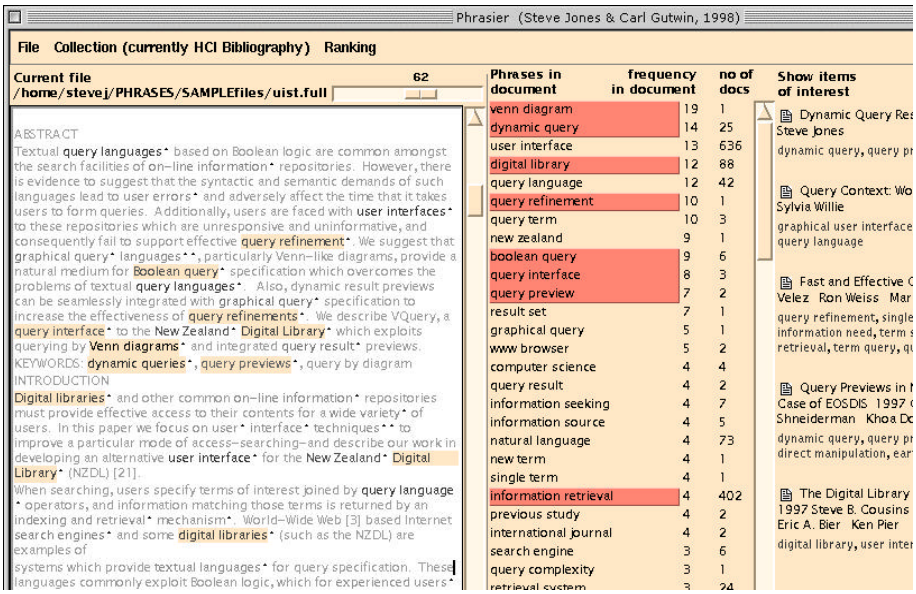
Figure 4 Browsing a keyphrase index to find out about topics involving *text*

conventional search systems for specific queries about specific documents. Note that many of these questions are as relevant to librarians as they are to library users.

Reading and writing in a digital library

A second prototype system shows how phrases can assist with skimming, reading, and writing documents in the digital library (Jones, 1998). It uses the keyphrases extracted from a document collection as link anchors to point to other documents. When reading a document, the keyphrases in it are highlighted. When writing one, phrases are dynamically linked, and highlighted, as you type.

Figure 5 shows the interface. To the left is the document being examined (read or authored); in the center is the keyphrase pane; and to the right is the library access pane. Keyphrases that appear in documents in the collection are highlighted; this facilitates rapid skimming of the content because the darker text points out items that users often highlight manually with a marker pen. Different gray levels reflect the "relevance" of the keyphrase to the document, and the user can control the intensity to match how they skim. Each phrase is hyperlinked, using multiple-destination links, to other documents for which it is a keyphrase (the anchor is the small spot that follows the phrase). The center panel shows all the keyphrases that appear in this document, with their frequency and the number of documents in the library for which



they are keyphrases. Controls are available to sort the list in various different ways. Some of these phrases have been selected by the user, and on the right is a ranked list of items in the library that contain them as keyphrases—ranked according to a special metric designed for use with keyphrases.

With this interface, hurried readers can skim the document by looking at the highlighted phrases. In-depth readers can instantly access other relevant documents (including, perhaps, dictionaries or encyclopaedias). They can select a subset of particularly relevant phrases and instantly have the library searched on that set. Writers can immediately—as they type—gain access to documents that are relevant to what they are writing.

Conclusion

Digital libraries have finally arrived. They are different from the World Wide Web: libraries are focused collections, and it is the act of selection that gives them focus. For many practical reasons (including copyright, and the physical difficulty of digitization), digital libraries will not vie with archival national collections, not in the foreseeable future. Their role is in specialist, targeted collections of information.

Established libraries of printed material have sophisticated and well-developed human and computer-based interfaces to support their use. But they are not well integrated for working with computer tools: a bridging process is required. Information workers can immerse themselves physically in the library, but they cannot take with them their tasks, tools, and desktop workspaces. The digital library will be different: we will work “inside” it in a sense that it totally new.

But even for a focused collection, creating a high-quality digital library is a

highly labor-intensive process. To provide the richness of access and inter-connection that makes a digital library comfortable requires enormous editorial effort. And when the collection changes, maintenance becomes an overriding issue. Fortunately, techniques of text mining are emerging that offer the possibility of automatic identification of semantic items from plain text. Carefully-constructed user interfaces can take advantage of the information that they generate to provide a library experience that is qualitatively different from a physical library—not just in access and convenience, but in terms of the quality of browsing and information accessibility. Future digital libraries will put the right information at your fingertips.

Acknowledgments

Many thanks to members of the New Zealand Digital Library for their work that supports this paper, particularly Rodger McNab, Steve Jones, Carl Gutwin, Stefan Boddie, Sally Jo Cunningham, Mark Apperley, Bill Rogers, Malika Mahoui, David Bainbridge, Te Taka Keegan, and Craig Nevill-Manning. Rob Akscyn, Michel Loots and Harold Thimbleby also made valued contributions.

References

Bush, V. (1947) "As we may think." *The Atlantic Monthly*, Vol. 176, No. 1, pp. 101–108.

Giles, C.L., Bollacker, K. and Lawrence, S. (1998) "CiteSeer: an automatic citation indexing system." *Proc ACM Digital Libraries '98*, pp. 89–98.

Gutwin, C., Paynter, G., Witten, I.H., Nevill-Manning, C., and Frank, E. (1998) "Improving browsing in digital libraries with keyphrase indexing." Technical Report, University of Saskatchewan, Canada.

Jones, S. (1998) "Link as you type:." Working Paper 98/16, University of Waikato, NZ.

Licklider, J.C.R. (1960) "Man-computer symbiosis." *IRE Trans* HFE-1, pp. 4–11.

Mann, T. (1993) *Library research models*. Oxford University Press, NY.

Nardi, B.A., Miller, J.R. and Wright, D.J. (1998) "Collaborative, programmable intelligent agents." *Comm ACM*, Vol. 41, No. 3, pp. 96–104.

Turney, P. (1997) "Extraction of keyphrases from text: Evaluation of four algorithms." National Research Council of Canada Report NRC/ERB-1051.

Wells, H.G. (1938) *World Brain*. Doubleday, NY.

Witten, I.H., Moffat, A., and Bell, T.C. (1994) *Managing Gigabytes*, VNR, NY.

Witten, I.H., Bray, Z., Mahoui, M. and Teahan, W. (1998) "Text mining: a new frontier for lossless compression." Working Paper, University of Waikato, New Zealand.

Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C,.G. and Jones, S. (in preparation) "Practical automatic keyphrase extraction.", University of Waikato, NZ.