Chapter 10

User Interfaces and Visualization

by Marti A. Hearst

To appear in *Modern Information Retrieval*, edited by Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Addison-Wesley Publishing Company. Do not redistribute or cite.

10.1 Introduction

This chapter discusses user interfaces for communication between human information seekers and information retrieval systems. Information seeking is an imprecise process. When users approach an information access system they often have only a fuzzy understanding of how they can achieve their goals. Thus the user interface should aid in the understanding and expression of information needs. It should also help users formulate their queries, select among available information sources, understand search results, and keep track of the progress of their search.

The human-computer interface is less well-understood than other aspects of information retrieval, in part because humans are more complex than computer systems, and their motivations and behaviors are more difficult to measure and characterize. The area is also undergoing rapid change, and so the discussion in this chapter will emphasize recent developments rather than established wisdom.

The chapter will first outline the human side of the information seeking process and then focus on the aspects of this process that can best be supported by the user interface. Discussion will encompass current practice and technology, recently proposed innovative ideas, and suggestions for future areas of development.

Section 10.2 outlines design principles for human-computer interaction and introduces notions related to information visualization. Section 10.3 describes information seeking models, past and present. The next four sections describe user interface support for starting the search process, for query specification, for viewing retrieval results in context, and for interactive relevance feedback. The last major section, Section 10.8, describes user interface techniques to support the information access process as a whole, and Section 10.9 concludes the chapter.

10.2 Human-Computer Interaction

What makes an effective human-computer interface? Ben Shneiderman, an expert in the field, writes [168][p. 10]:

> Well designed, effective computer systems generate positive feelings of success, competence, mastery, and clarity in the user community. When an interactive system is well-designed, the interface almost disappears, enabling users to concentrate on their work, exploration, or pleasure.

As steps towards achieving these goals, Shneiderman lists principles for design of user interfaces. Those which are particularly important for information access include (slightly restated): provide informative feedback, permit easy reversal of actions, support an internal locus of control, reduce working memory load, and provide alternative interfaces for novice and expert users. Each of these principles should be instantiated differently depending on the particular interface application. Below we outline the impact of those principles that are of special interest to information access systems.

10.2.1 Design Principles

Offer informative feedback. This principle is especially important for information access interfaces. In this chapter we will see current ideas about how to provide users with feedback about the relationship between their query specification and documents retrieved, about relationships among retrieved documents, and about relationships between retrieved documents and metadata describing collections. If the user has control of how and when feedback is provided, then the system provides an **internal locus of control**.

Reduce working memory load. Information access is an iterative process, the goals of which shift and change as information is encountered. One key way information access interfaces can help with memory load is to provide mechanisms for keeping track of choices made during the search process, allowing users to return to temporarily abandoned strategies, jump from one strategy to the next, and retain information and context across search sessions. Another memory-aiding device is to provide browsable information that is relevant to the current stage of the information access process. This includes suggestions of related terms or metadata, and search starting points including lists of sources and topic lists.

Provide alternative interfaces for novice and expert users. An important tradeoff in all user interfaces design is that of simplicity versus power. Simple interfaces are easier to learn, at the expense of less flexibility and sometimes less efficient use. Powerful interfaces allow a knowledgeable user to do more and have more control over the operation of the interface, but can be time-consuming to learn and impose a memory burden on people who use the system only intermittently. A common solution is to use a "scaffolding" technique [159]. The novice user is presented with a simple interface that can be learned quickly and that provides the basic functionality of the application, but is restricted in power and flexibility. Alternative interfaces are offered for more experienced users, giving them more control, more options, and more features, or potentially even entirely different interaction models. Good user interface design provides intuitive bridges between the simple and the advanced interfaces.

Information access interfaces must contend with special kinds of simplicity/power tradeoffs. One such tradeoff is the amount of information shown about the workings of the search system itself. Users who are new to a system or to a particular collection may not know enough about the system or the domain associated with the collection to make choices among complex features. They may not know how best to weight terms, or in the case of relevance feedback, not know what the effects of reweighting terms would be. On the other hand, users that have worked with a system and gotten a feeling for a topic are likely to be able to choose among suggested terms to add to their query in an informed manner. Determining how much information to show the user of the system is a major design choice in information access interfaces.

10.2.2 The Role of Visualization

The tools of computer interface design are familiar to most computer users today: windows, menus, icons, dialog boxes, and so on. These make use of bit-mapped display and computer graphics to provide a more accessible interface than command-line based TTY displays. A less familiar but growing area is that of **information visualization**, which attempts to provide visual depictions of very large information spaces.

Humans are highly attuned to images and visual information [178, 100, 108]. Pictures and graphics can be captivating and appealing, especially if well-designed. A visual representation can communicate some kinds of information much more rapidly and effectively than any other method. Consider the difference between a written description of a person's face and a photograph of it, or the difference between a table of numbers containing a correlation and a scatter plot showing the same information.

The growing prevalence of fast graphics processors and high resolution color monitors is increasing interest information visualization. Scientific visualization, a rapidly advancing branch of this field, maps physical phenomena onto two or three-dimensional representations [92]. An example of scientific visualization is a colorful image of the pattern of peaks and valleys on the ocean floor; this provides a view of physical phenonema for which a photograph cannot (currently) be taken. Instead, the image is constructed from data that represent the underlying phenomena.

Visualization of inherently abstract information is more difficult, and visualization of textually represented information is especially challenging. Language is our main means of communicating abstract ideas for which there is no obvious physical manifestation. What does a picture look like that describes negotiations over a trade agreement in which one party demands concessions on environmental policies while the other requires help in strengthening its currency?

Despite the difficulties, researchers are attempting to represent aspects of the information access process using information visualization techniques. Some of these will be described later in this chapter. Aside from using **icons** and **color highlighting**, the main information visualization techniques include **brushing and linking** [54, 181], **panning and zooming** [16], **focus-plus-context** [112], **magic lenses** [19], and the use of **animation** to retain context and help make occluded information visible [157, 29]. These techniques support dynamic, interactive use. Interactivity seems to be an especially important property for visualizing abstract information, although it has not played as large a role within scientific visualization.

Brushing and linking refers to the connecting of two or more views of the same data, such that a change to the representation in one view effects the representation in the other views as well. For example, say a histogram is shown in one window representing the years of publication for a set of documents, and another window depicts this document set as a list of titles in order of relevance. An example of brushing and linking is a mechanism by which the user's assigning the color red to a bar of the histogram (brushing it with red) causes the titles of the documents that were published in the corresponding year to also be colored red.

Panning and zooming refers to the actions of a movie camera that can scan sideways across a scene (panning) or move in for a closeup or back away to get a wider view (zooming). For example, text clustering can be used to show a top-level view of the main themes in a document collection, and zooming can be used to move "closer," showing individual documents represented as icons, and then move in closer still to see the text associated with a single document.

When zooming is used, the more detail that is visible about a particular item, the less can be seen about the surrounding items. Focus-plus-context is used to partly alleviate this effect. The idea is to make one portion of the view – the focus of attention – larger, while simultaneously shrinking the surrounding objects. The farther an object is from the focus of attention, the smaller it is made to appear, like the effect seen in a fisheye camera lens (also in some door peepholes).

Magic lenses are directly manipulable transparent windows that, when overlayed on some other data type, cause a transformation to be applied to the underlying data, thus changing its appearance. The most straightforward application of magic lenses is for drawing tasks, and it is especially useful if used as a two-handed interface. For example, the left hand can be used to position a color lens over a drawing of an object. The right hand is used to mouse-click on the lens, thus causing the appearance of the underlying object to be transformed to the color specified by the lens.

Additionally, there are a large number of graphical methods for depicting trees and hierarchies, some of which make use of animation to show nodes that would otherwise be occluded (hidden from view by other nodes) [62, 81, 89, 106, 157].

It is often useful to combine these techniques into an interface layout consisting of an **overview plus details** [64, 149]. An overview, such as a table-of-contents of a large manual, is shown in one window. A mouse-click on the title of the chapter causes the text of the chapter itself to appear in another window, in a linking action. Panning and zooming or focus-plus-context can be used to change the view of the contents within the overview window.

10.2.3 Evaluating Interactive Systems

From the viewpoint of user interface design, people have widely differing abilities, preferences, and predilections. Important differences for information access interfaces include relative spatial ability and memory, reasoning abilities, verbal aptitude, and (potentially) personality differences [49, 168]. Age and cultural differences can contribute to acceptance or rejection of interface techniques [129]. An interface innovation can be useful and pleasing for some users, and foreign and cumbersome for others. Thus software design should allow for flexibility in interaction style, and new features should not be expected to be equally helpful for all users.

An important aspect of human-computer interaction is the methodology for evaluation of user interface techniques. Precision and recall measures have been widely used for comparing the ranking results of non-interactive systems, but are less appropriate for assessing interactive systems [105]. This results in part because of a strong topic-system interactions; the kind of question being asked can have a strong effect on the performance of the system. Also, the standard measures of precision and recall evaluate the system at high recall levels; in the TREC tasks systems are compared to see how well they return the top 1000 documents. In many interactives settings, users require only a few relevant documents.

For evaluating highly interactive information access systems, useful metrics beyond precision and recall include: time required to learn the system, time required to achieve goals on benchmark tasks, error rates, and retention of the use of the interface over time. Throughout this chapter, empirical results of user studies are presented whenever they are available.

Empirical data involving human users is time-consuming to gather and difficult to draw conclusions from. This is due in part to variation in users' characteristics and motivations, and in part to the broad scope of information access activities. Formal psychological studies usually only uncover narrow conclusions about restricted contexts. For example, interaction moves such as the length of time it takes for a user to select an item from a fixed menu under various conditions have been characterized empirically [28], but variations in interaction behavior for complex tasks like information access are difficult to account for accurately. Nielsen [138] advocates a more informal evaluation approach (called heuristic evaluation) in which user interface affordances are assessed in terms of more general properties and without concern about statistically significant results.

10.3 The Information Access Process

A person engaged in an information seeking process has one or more **goals** in mind and uses a search system as a tool to help achieve those goals. Goals requiring information access can range quite widely, from finding a plumber to keeping informed about a business competitor, from writing a publishable scholarly article to investigating an allegation of fraud.

Information access **tasks** are used to achieve these goals. These tasks span the spectrum from asking specific questions to exhaustively researching a topic. Other tasks fall between these two extremes. A study of business analysts [140] found three main kinds of information seeking tasks: monitoring a well-known topic over time (such as researching competitors' activities each quarter), following a plan or stereotyped series of searches to achieve a particular goal (such as keeping up to date on good business practices), and exploring a topic in an undirected fashion (as when getting to know an unfamiliar industry). Although the goals differ, there is a common core revolving around the information seeking component, which is our focus here.

10.3.1 Models of Interaction

Most accounts of the information access process assume an interaction cycle consisting of query specification, receipt and examination of retrieval results, and then either stopping or reformulating the query and repeating the process until a perfect result set is found [162, 169]. In more detail, the

Figure 10.1: Models of information access processes.

standard process can be described according to the following sequence of steps (see Figure 10.1):

1. Start with an information need
2. Select a system and collections to search on
3. Formulate a query
4. Send the query to the system
5. Receive the results in the form of information items
6. Scan, evaluate, and interpret the results
7. Either stop or
8. Reformulate the query and go to step 4.

This simple interaction model used by web search engines is the only model that most information seekers see today. This model does not take into account the fact that many users dislike being confronted with a long disorganized list of retrieval results that do not directly address their information needs. It also contains an underlying assumption that the users' information need is static and the information seeking process is one of successively refining a query until it retrieves all and only those documents relevant to the original information need.

One way users learn during the search process is by scanning information, such as reading the titles of result sets of documents, reading retrieved documents themselves, viewing lists of topics related to their query terms, and navigating within hyperlinked web sites. The recent advent of hyperlinks as a pivotal part of the information seeking process makes it no longer feasible to ignore the role of scanning and navigation within the search process itself. In particular, today a near-miss is much more acceptable than it was with bibliographic search, since an information seeker using the web can navigate from a near miss in the hopes that a useful page will be a few links away.

The standard model also downplays the interaction that takes place when the user scans terms suggested as a result of relevance feedback, scans thesaurus structures, or views thematic overviews of document collections. It de-emphasizes the role of source selection, which is increasingly important now that, for the first time, tens of thousands of information collections are immediately reachable for millions of people.

Thus while useful for describing the basics of information access systems, this simple interaction model is being challenged on many fronts [14, 140, 22, 82, 40]. Bates [14] proposes the "berry-picking" model of information seeking, which has two main points. The first is that, as a result of reading and learning from the information encountered throughout the search process, the users' information needs, and consequently their queries, continually shift. Information encountered at one point in a search may lead in a new, unanticipated direction. The original goal may become partly fulfilled, thus lowering the priority of one goal in favor of another. This is posed in contrast to the assumption of "standard" information retrieval that the users' information need remains the same throughout the search process. The second point is that users' information needs are not satisfied by a single, final retrieved set of documents, but rather by a series of selections and bits of information found along the way. This is in contrast to the assumption that the main goal of the search process is to hone down the set of retrieved documents into a perfect match of the original information need.

The berry-picking model is supported by a number of observational studies [55, 22], including that of O'Day and Jeffries [140]. They found that the information seeking process consisted of a series of interconnected but diverse searches on one problem-based theme. They also found that search results for a goal tended to trigger new goals, and hence search in new directions, but that the context of the problem and the previous searches was carried from one stage of search to the next. They also found that the main value of the search resided in the accumulated learning and acquisition of information that occurred during the search process, rather than in the final results set.

Thus, a user interface for information access should allow users to reassess their goals and adjust their search strategy accordingly. A related situation occurs when users encounter a "trigger" that causes them to pursue a different strategy temporarily, perhaps to return to the current unfinished activity at a later time. An implication of these observations is that the user interface should support search strategies by making it easy to follow trails with unanticipated results. This can be accomplished in part by supplying ways to record the progress of the current strategy and to store, find, and re-load intermediate results, and supporting pursuit of multiple strategies simultaneously.

The user interface should also support methods for monitoring the status of the current strategy in relation to the users' current task and high-level goals. One way to cast the activity of monitoring the progress of a search strategy relative to a goal or subgoal is in terms of a cost/benefit analysis, or an analysis of diminishing returns [161]. This kind of analysis assumes that at any point in the search process, the user is pursuing the strategy that has the highest expected utility. If, as a consequence of some local tactical choices, another strategy presents itself as being of higher utility than the current one, the current one is (temporarily or permanently) abandoned in favor of the new strategy.

There are a number of theories and frameworks that contrast **browsing, querying, navigating,** and **scanning** along several dimensions [18, 34, 123, 183]. Here we assume that users scan information structure, be it titles, thesaurus terms, hyperlinks, category labels, or the results of clustering, and then either select a displayed item for some purpose (to read in detail, to use as input to a query, to navigate to a new page of information) or formulate a query (either by recalling potential words or by selecting categories or suggested terms that have been scanned). In both cases, a new set of information is then made viewable for scanning. Queries tend to produce new, ad hoc collections of information that have not been gathered together before, whereas selection retrieves information that has already been composed or organized. Navigation refers to following a chain of links, switching from one view to another, toward some goal, in a sequence of scan and select operations. Browsing refers to the casual, mainly undirected exploration of information structures, and is usually done in tandem with selection, although queries can also be used to create subcollections to browse through. An important aspect of the interaction process is that the output of one action should be easily used as the input to the next.

10.3.2 Non-Search Parts of the Information Access Process

The O'Day and Jeffries study [140] found that information seeking is only one part of the full work process their subjects were engaged in. In between searching sessions many different kinds of work was done with the retrieved information, including reading and annotating [142] and analysis. O'Day and Jeffries examined the analysis steps in more detail, finding that 80% of this work fell into six main types: finding trends, making comparisons, aggregation, identifying a critical subset, assessing, and interpreting. The remainder consisted of cross-referencing, summarizing, finding evocative visualizations for reports, and miscellaneous activities. The Sensemaking work of Russell et al. [161] also discusses information work as a process in which information retrieval plays only a small part. They observe that most of the effort made in Sensemaking is in the synthesis of a good representation, or way of thinking about, the problem at hand. They describe the process of developing a way to formulate and crystallize the important concepts for a given task.

From these observations it is convenient to divide the entire information access process into two main components: search/retrieval, and analysis/synthesis of results. User interfaces should allow both kinds of activity to be tightly interwoven. However, analysis/synthesis are activities that can be done independently of information seeking, and for our purposes it is useful to make a distinction between the two types of activities.

10.3.3 Earlier Interface Studies

The bulk of the literature on studies of human-computer information seeking behavior concerns information intermediaries using online systems consisting of bibliographic records (e.g., [124, 164, 21]), sometimes with costs assessed per time unit. Unfortunately, many of the assumptions behind those studies do not reflect the conditions of modern information access [68, 46]. The differences include the following:

- The text being searched now is often full text rather than bibliographic citations, and thus in many cases questions can be answered directly. Because users have access to full text, rather than document surrogates, it is more likely that simple queries will find relevant answers directly as part of the search process.
- Most studies were performed on Boolean systems whereas modern systems use statistical ranking (which is more effective when abstracts and full text are available than when only titles and citations are available).
- Much of modern searching is done by end users, many new to online searching, rather than professional intermediaries, which were the focus of many of the earlier studies.
- Tens of thousands of sources are now available online on networked information systems, and many are tightly coupled via hyperlinks, as opposed to being stored in separate collections owned by separate services. Earlier studies generally used systems in which moving from one collection to another required prior knowledge of the collections and considerable time and effort to switch. A near miss is much more useful in this hyperlinked environment than in earlier systems, since hyperlinks allow users to navigate from the near miss directly to the source containing information of interest. In a card catalog environments, where documents

are represented as isolated units, a near miss consists of finding a book in the general area of interest and then going to the bookshelf in the library to look for related books, or obtaining copies of many issues of a journal and scanning for related articles.

- Finally, most users have access to bit-mapped displays allowing for direct manipulation, or at least form fillin. Most earlier studies and bibliographic systems were implemented on TTY displays, which require command-line based syntax and do a poor job of retaining context.

Despite these significant differences, some general information seeking strategies have been identified that seem to transfer across systems. Additionally, although modern systems have remedied many of the problems of earlier online public access catalogs, they also introduce new problems of their own.

10.4 Starting Points

Search interfaces must provide users with good ways to get started. An empty screen or a blank entry form does not provide clues to help a user decide how to start the search process. Users usually do not begin by creating a long, detailed expression of their information need. Studies show that users tend to start out with very short queries, inspect the results, and then modify those queries in an incremental feedback cycle [7]. The initial query can be seen as a kind of "testing the water" to see what kinds of results are returned and get an idea of how to reformulate the query [183, 14]. Thus one task of an information access interface is to help users select the sources and collections to search on.

For example, there are many different information sources associated with cancer, and there are many different kinds of information a user would like to know about cancer. Guiding the user to the right set of starting points can help with the initial problem formulation. Traditional bibliographic search assumes that the user begins by looking through a list of names of sources and choosing which collections to search on, while web search engines obliterate the distinctions between sources and plunge the user into the middle of a web site with little information about the relationship of the search hit to the rest of the collection. In neither case is the interface to the available sources particularly helpful.

In this section we will discuss four main ways to get started in search: **lists, overviews, examples,** and **automated source selection**.

10.4.1 Lists of Collections

Typical online systems such as Lexis-Nexis require users to begin any inquiry with a scan through a long list of source names and guess which ones will be of interest. Usually little information beyond the name of the collection is provided online for these sources (see Figure 10.2). If the user is not satisfied with the results on one collection, they must reissue the query on another collection.

Frequent searchers eventually learn a set of sources that are useful for their domains of interest, either through experience, formal training, or recommendations from friends and colleagues. Often-used sources can be stored on a "favorites" list, also known as a bookmark list or a hotlist on the

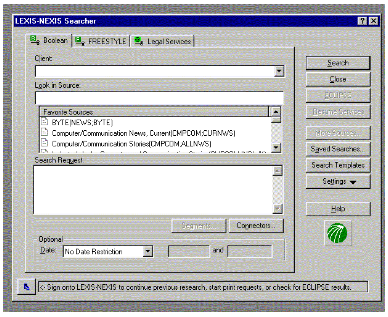

Figure 10.2: Lexis-Nexis source selection screen.

web. Recent research explores the maintenance of a personalized information profile for users or workgroups, based on the kinds of information they've used in the past [60].

However, when a user wants to search outside their domains of expertise, a list of familiar sources is not sufficient. Professional searchers such as librarians learn through experience and years of training which sources are appropriate for various information needs. The restricted nature of traditional interfaces to information collections discourages exploration and discovery of new useful sources. However, recently researchers have devised a number of mechanisms to help users understand the contents of collections as a way of getting started in their search.

10.4.2 Overviews

Faced with a large set of text collections, how can a user choose which to begin with? One approach is to study an overview of the contents of the collections. An overview can show the topic domains represented within the collections, to help users select or eliminate sources from consideration. An overview can help users get started, directing them into general neighborhoods, after which they can navigate using more detailed descriptions. Shneiderman [167] advocates an interaction model in which the user begins with an overview of the information to be worked with, then pans and zooms to find areas of potential interest, and then view details. The process is repeated as often as necessary.

Three types of overviews are discussed in this subsection. The first is display and navigation of large topical **category hierarchies** associated the documents of a collection. The second is automatically derived overviews, usually created by unsupervised **clustering techniques** on the text of documents, that attempt to extract overall characterizing themes from collections. The third type of overview is that created by applying a variant of **cocitation analysis** on connections or links between different entities within a collection. Other kinds of overviews are possible, for example, showing a bookshelves or piles of books graphically [158, 8].

Category or Directory Overviews

There exist today many large online text collections to which category labels have been assigned. Traditional online bibliographic systems have for decades assigned subject headings to books and other documents [173]. MEDLINE, a large collection of biomedical articles, has associated with it Medical Subject Headings (MeSH) consisting of approximately 18,000 categories [115]. The Association for Computing Machinery (ACM) has developed a hierarchy of approximately 1200 category (keyword) labels.[1] Yahoo[2], one of the most popular search sites on the World Wide Web, organizes web pages into a hierarchy consisting of thousands of category labels.

The popularity of Yahoo and other web directories suggests that hierarchically structured categories are useful starting points for users seeking information on the web. This popularity may reflect a preference to begin at a logical starting point, such as the home page for a set of information, or it may reflect a desire to avoid having to guess which words will retrieve the desired information. (It may also reflect the fact that directory services attempt to cull out low-quality web sites.)

The meanings of category labels differ somewhat among collections. Most are designed to help organize the documents and to aid in query specification. Unfortunately, users of online bibliographic catalogs rarely use the available subject headings [68, 46]. Hancock-Beaulieu and Drabenstott and Weller, among others, put much of the blame on poor (command line-based) user interfaces which provide little aid for selecting subject labels and require users to scroll through long alphabetic lists. Even with graphical web interfaces, finding the appropriate place within a category hierarchy can be a time-consuming task, and once a collection has been found using such a representation, an alternative means is required for searching within the site itself.

Most interfaces that depict category hierarchies graphically do so by associating a document directly with the node of the category hierarchy to which it has been assigned. For example, clicking on a category link in Yahoo brings up a list of documents that have been assigned that category label. Conceptually, the document is stored within the category label. When navigating the results of a search in Yahoo, the user must look through a list of category labels and guess which one is most likely to contain references to the topic of interest. A wrong path requires backing up and trying again, and remembering which pages contain which information. If the desired information is deep in the hierarchy, or not available at all, this can be a time-consuming and frustrating process. Because documents are conceptually stored "inside" categories, users cannot create queries based on combinations of categories using this interface.

It is difficult to design a good interface to integrate category selection into query specification, in part because display of category hierarchies takes up large amounts of screen space. For example, Internet Grateful Med[3] is a web-based service that allows an integration of search with display and

[1] http://www.acm.org/class/1991/cr91.html
[2] http://www.yahoo.com
[3] http://igm.nlm.nih.gov:80/

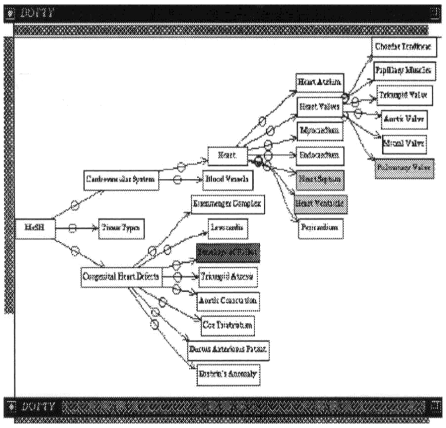

Figure 10.3: The MeshBrowse interface for viewing category labels hierarchically (from Korn and Shneiderman 1995)

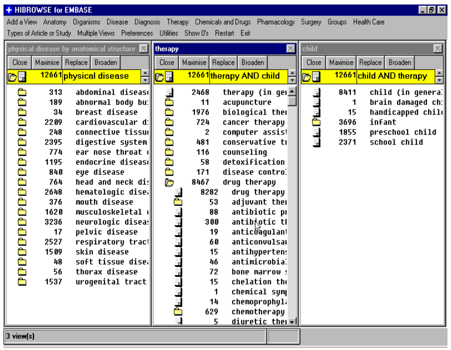

Figure 10.4: The HiBrowse interface for viewing category labels hierarchically and according to facets (from Pollitt 1997).

selection of MeSH category labels. After the user types in the name of a potential category label, a long list of choices is shown in a page. To see more information about a given label, the user selects a link (e.g., Radiation Injuries). The causes the context of the query to disappear because a new web page appears showing the ancestors of the term and its immediate descendants. If the user attempts to see the siblings of the parent term (Wounds and Injuries) then a new page appears that changes the context again. Radiation Injuries appears as one of many siblings and its children can no long be seen. To go back to the query, the illustration of the category hierarchy disappears.

The MeSHBROWSE system [99] allows users to interactively browse a subset of semantically associated links in the MeSH hierarchy. From a given starting point, clicking on a category causes the associated categories to be displayed in a 2D tree representation. Thus only the relevant subset of the hierarchy is shown at one time, making browsing of this very large hierarchy a more tractable endeavor. The interface has the space limitations inherent in a 2D hierarchy display and does not provide mechanisms for search over an underlying document collection. See Figure 10.3.

The HIBROWSE system [150] represents category metadata more efficiently by allowing users to display several different subsets of category metadata simultaneously. The user first selects which attribute type (or facet, as attributes are called in this system) to display, for example, the user may first choose the "physical disease" value for the Disease facet. The categories that appear one level below this are shown along with the number of documents that contain each category. The user can then select other attribute types, such as Therapy and Groups (by age). The number of documents that contain attributes from all three types are shown. If the user now selects a refinement of one of the categories, such as the "child" value for the Groups attribute, then the number of documents that contain all three selected facet types a shown. At the same time, the numbers of documents containing the subcategories found below "physical disease" and "therapy (general)" are updated to reflect this more restricted specification. See Figure 10.4. A problem with the HIBROWSE system is that it requires users to navigate through the category hierarchy, rather than specify queries directly. In other words, query specification is not tightly coupled with display of category metadata. As a solution to some of these problems, the Cat-a-Cone interface [78] will be described in Section 10.8.

Automatically Derived Collection Overviews

Many attempts to display overview information have focused on automatically extracting the most common general themes that occur within the collection. These themes are derived via the use of unsupervised analysis methods, usually variants of document clustering. Clustering organizes documents into groups based on similarity to one another; the centroids of the clusters determine the themes in the collections.

The Scatter/Gather browsing paradigm [43, 42] clusters documents into topically-coherent groups, and presents descriptive textual summaries to the user. The summaries consist of topical terms that characterize each cluster generally, and a set of typical titles that hint at the contents of the cluster. Informed by the summaries, the user may select a subset of clusters that seem to be of most interest, and re-cluster their contents. Thus the user can examine the contents of each subcollection at progressively finer granularity of detail. The reclustering is computed on-the-fly; different themes are produced depending on the documents contained in the subcollection to which clustering is applied. The choice of clustering algorithm influences what clusters are produced, but no one algorithm has been shown to be particularly better than the rest when producing the same number of clusters [186].

A user study [147] showed that the use of Scatter/Gather on a large text collection successfully conveys some of the content and structure of the corpus. However, that study also showed that Scatter/Gather without a search facility was less effective than a standard similarity search for finding relevant documents for a query. That is, subjects allowed only to navigate, not to search over, a hierarchical structure of clusters covering the entire collection were less able to find documents relevant to the supplied query than subjects allowed to write queries and scan through retrieval results.

It is possible to integrate Scatter/Gather with conventional search technology by applying clustering on the results of a query to organize the retrieved documents (see Figure 10.5). An off-line experiment [79] suggests that clustering may be more effective if used in this manner. The study found that documents relevant to the query tend to fall mainly into one or two out of five clusters, if the clusters are generated from the top-ranked documents retrieved in response to the query. The study also showed that precision and recall were higher within the best cluster than within the retrieval results as a whole. The implication is that a user might save time by looking at the contents of the cluster with the highest proportion of relevant documents and at the same time avoiding those clusters with mainly non-relevant documents. Thus clustering of retrieval results may be useful for helping direct users to a subset of the retrieval results that contain a large proportion of the relevant documents.

General themes do seem to arise from document clustering, but the themes are highly dependent on the makeup of the documents within the clusters [79, 77]. The unsupervised nature of clustering can result in a display of topics at varying levels of description. For example, clustering a collection of documents about computer science might result in cluster containing documents about artificial intelligence, computer theory, computer graphics, computer architecture, programming languages, government, and legal issues. The latter two themes are more general than the others, because they are about topics outside the general scope of computer science. Thus clustering can results in the juxtaposition of very different levels of description within a single display.

Scatter/Gather shows a textual representation of document clusters. Researchers have developed several approaches to map documents from their high dimensional representation in document space into a 2D representation in which each document is represented as a small glyph or icon on a map or within an abstract 2D space. The functions for transforming the data into the lower dimensional space differ, but the net effect is that each document is placed at one point in a scatter plot-like representation of the space. Users are meant to detect themes or clusters in the arrangement of the glyphs. Systems employing such graphical displays include BEAD [33], the Galaxy of News [155], and ThemeScapes [188]. The ThemeScapes view imposes a three-dimensional representation on the results of clustering (see Figure 10.6). The layout makes use of "negative space" to help emphasize the areas of concentration where the clusters occur. Other systems , display inter-document similarity hierarchically[117, 3], while still others display retrieved documents in networks based on interdocument similarity [58, 175].

Kohonen's feature map algorithm has been used to create maps that graphically characterize

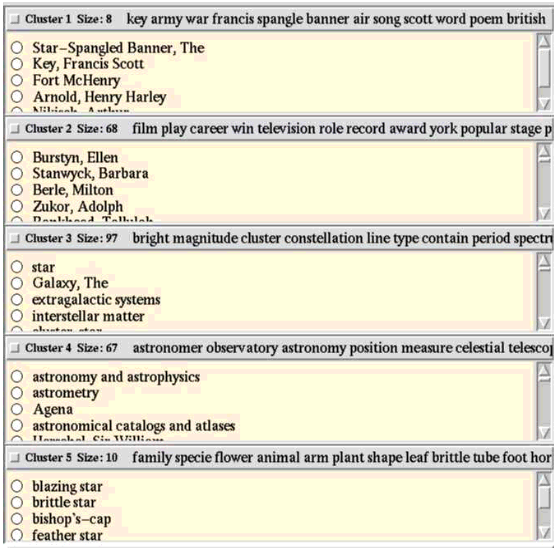

Figure 10.5: Display of Scatter/Gather clustering retrieval results (Cutting et al. 1992).

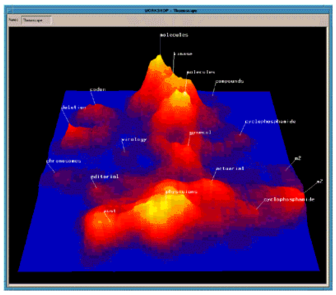

Figure 10.6: A three-dimensional overview based on document clustering (from Wise et al. 1995)

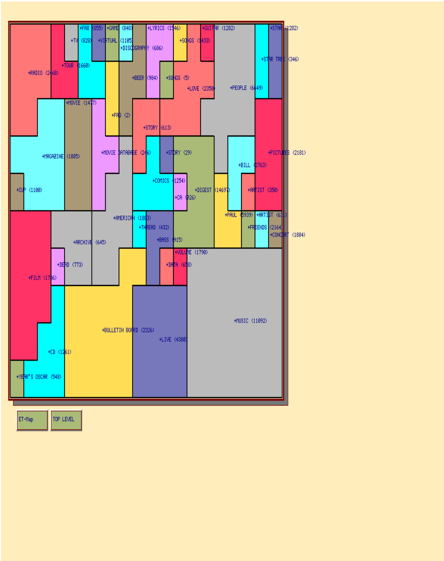

Figure 10.7: A two-dimensional overview created using a Kohonen feature map learning algorithm on web pages having to do with the topic Entertainment (from Chen et al. 1998)

the overall content of a document collection or subcollection [114, 35] (see Figure 10.7). The regions of the 2D map vary in size and shape corresponding to how frequently documents assigned to the corresponding themes occur within the collection. Regions are characterized by single words or phrases, and adjacency of regions is meant to reflect semantic relatedness of the themes within the collection. A cursor moved over a document region causes the titles of the documents most strongly associated with that region to be displayed in a pop-up window. Documents can be associated with more than one region.

Evaluations of Graphical Overviews

Although intuitively appealing, graphical overviews of large document spaces have yet to be shown to be useful and understandable for users. In fact, evaluations that have been conducted so far provide negative evidence as to their usefulness. One study found that for non-expert users the results of clustering were difficult to use, and that graphical depictions (for example, representing clusters with circles and lines connecting documents) were much harder to use than textual representations (for example, showing titles and topical words, as in Scatter/Gather), because documents' contents are difficult to discern without actually reading some text [94].

Another recent study compared the Kohonen feature map overview representation on a browsing task to that of Yahoo [35]. For one of the tasks, subjects were asked to find an "interesting" web page within the entertainment category of Yahoo and of an organization of the same web pages into a Kohonen map layout. The experiment varied whether subjects started in Yahoo or in the graphical map. After completion of the browing task, subjects were asked to attempt to repeat the browse using the other tool. For the subjects that began with the Kohonen map visualization, 11 out of 15 found an interesting page within 10 minutes. Eight of these were able to find the same page using Yahoo. Of the subjects who started with Yahoo, 14 out of 16 were able to find interesting home pages. However, only 2 of the 14 were able to find the page in the graphical map display! This is strong evidence against the navigability of the display and certainly suggests that the simply label view provided by Yahoo is more useful. However, the map display may be more useful if the system is modified to tightly integrate querying with browsing.

The subjects did prefer some aspects of the map representation. In particular, some liked the ease of being able to jump from one area to another without having to back up as is required in Yahoo, and some liked the fact that the maps have varying levels of granularity. The subjects disliked several aspects of the display. The experimenters found that some subjects expressed a desire for a visible hierarchical organization, others wanted an ability to zoom in on a subarea to get more detail, and some users disliked having to look through the entire map to find a theme, desiring an alphabetical ordering instead. Many found the single-term labels to be misleading, in part because they were ambiguous (one region called "BILL" was thought to correspond to a person's name rather than counting money).

The authors concluded that this interface is more appropriate for casual browsing than for search. In general, unsupervised thematic overviews are perhaps most useful for giving users a "gist" of the kinds of information that can be found within the document collection, but generally have not been shown to be useful for use in the information access process.

Cocitation Clustering for Overviews

Citation analysis has long been recognized as a way to show an overview of the contents of a collection [184]. The main idea is to determine "centrally-located" documents based on cocitation patterns. There are different ways to determine citation patterns: one method is to measure how often two articles are cited together by a third. Another alternative is to pair articles that cite the same third article. In both cases the assumption is that the paired articles share some commonalities. After a matrix of cocitations is built, documents are clustered based on the similarity of their cocitation patterns. The resulting clusters are interpreted to indicate dominant themes within the collection. Clustering can focus on the authors of the documents rather than the contents, to attempt to identify central authors within a field. This idea has recently been implemented using web based documents in the Referral Web project [91]. This idea has also been applied to web pages, using web link structure to identify major topical themes among web pages [109, 146]. A similar idea, but computed a different way, is used to explicitly identify pages that act as good starting points for particular topics (called "authority pages" by Kleinberg [95]).

10.4.3 Examples, Dialogues, and Wizards

Another way to help users get started is to start them off with an example of interaction with the system. This technique is also known as **retrieval by reformulation**. An early version of this idea is embodied in the Rabbit system [187] which provides graphical representations of example database queries. A general framework for a query is shown to the user who then modifies it to construct a partially complete description of what they want. The system then shows an example of the kind of information available that matches this partial description. For instance, if a user searching a computer products database indicates an interest in disks, an example item is retrieved with its disk descriptors filled in. The user can use or modify the displayed descriptors, and iterate the procedure.

The idea of retrieval by reformulation has been developed further and extended to the domains of user interface development [134] and software engineering [154]. The Helgon system [56] is a modern variant of this idea applied to bibliographic database information. In Helgon, users begin by navigating a hierarchy of topics from which they select structured examples, according to their interests. If a feature of an example is inappropriately set, the user can modify the feature to indicate how it would appear in the desired information. Unfortunately, in tests with users, the system was found to be problematic. Users had problems with the organization of the hierarchy, and found that finding a useful example by critiquing an existing one to be tedious. This result underscores an unfortunate difficulty with examples and dialogues: that of getting the user to the right starting dialogue or the right example strategy becomes a search problem in itself. (How to index prior examples is studied extensively in the case-based reasoning (CBR) literature [110, 97].)

A more dynamic variation on this theme is the interactive dialogue. Dialogue-based interfaces have been explored since the early days of information retrieval research, in an attempt to mimic the interaction provided by a human search intermediary (e.g., a reference librarian). Oddy did early work in the THOMAS system, which provided a question and answer session within a command-line based interface [141]. More recently, Belkin, et al. have defined quite elaborate dialogue interaction models [18] although these have not been assessed empirically to date.

The DLITE system interface [40] uses an animated focus-plus-context dialogue as a way to acquaint users with standard sequences of operations within the system. Initially an outline of all of the steps of the dialogue is shown as a list. The user can expand the explanation of any individual step by clicking on its description. The user can expand out the entire dialogue to see what questions are coming next, and the collapse it again in order to focus on the current tactic.

A more restricted form of dialogue that has become widely used in commercial products is that of the **wizard**. This tool helps users in time-limited tasks, but does not attempt to overtly teach them about the processes required to complete the tasks. The wizard presents a step-by-step shortcut through the sequence of menu choices (or tactics) that a user would normally perform in order to get a job done, reducing user input to just a few choices with default settings [145]. A recent study [31] found wizards to be useful for goals that require many steps, for users who lack necessary domain knowledge (for example, a restaurant owner installing accounting software), and when steps must be completed in a fixed sequence (for example, a procedure for hiring personnel). Properties of successful wizards included allowing users to re-run a wizard and modify their previous work, show an overview of the functions they contain, and provide lucid descriptions and understandable outcomes for choices. Wizards were found not to be helpful when the interface did not solve a problem effectively (for example, a commercial wizard for setting up a desktop search index requests users to specify how large to make the index, but supplies no information about how to make this decision). Wizards were also found not to be helpful when the goal was to teach the user how to use the interface, and when the wizard was not user-tested. Most likely information access is too variable a process for the use of wizards.

A **guided tour** leads a user through a sequence of navigational choices through hypertext links, presenting the nodes in a logical order for some goal. In a dynamic tour, only relevant nodes are displayed, as opposed to the static case where the author decides what is relevant before the users have even formulated their queries [66]. A recent application is the Walden Paths project which enables teachers to define instructionally useful paths through pages found on the web [63]. This approach has not been commonly used to date for information access but could be a promising direction for acquainting users with search strategies in large hyperlinked systems.

10.4.4 Automated Source Selection

Human-computer interfaces for helping guide users to appropriate sources is a wide open area for research. It requires both eliciting the information need from the user and understanding which needs can be satisfied by which sources. An ambitious approach is to build a model of the source and of the information need of the user and try to determine which fit together best. User modeling systems and intelligent tutoring systems attempt to do this both for general domains [44, 185] and for online help systems [84].

A simpler alternative is to create a representation of the contents of information sources and match this representation against the query specification. This approach is taken by GLOSS, a system which tries to determine in advance the best bibliographic database to send a search request to, based on the terms in the query [176]. The system uses a simple analysis of the combined frequencies of the query words within the individual collections. The SavvySearch system [85] takes this idea a step further, using actions taken by users after a query to decide how to downweight or increase the ranking of a search engine for a particular query.

The flip side to automatically selecting the best source for a query is to automatically send a query to multiple sources and then combine the results from the various systems in some way. Many metasearch engines exist on the web. How to combine the results effectively is an active area of research, sometimes known as collection fusion [12, 177, 86].

10.5 Query Specification

To formulate a query, a user must select collections, metadata descriptions, or information sets against which the query is to be matched, and must specify words, phrases, descriptors, or other kinds of information that can be compared to or matched against the information in the collections. As a result, the system creates a set of documents, metadata, or other information type that match the query specification in some sense and displays the results to the user in some form.

Shneiderman [168] identifies five primary human-computer interaction styles. These are: **command language**, **form fillin**, **menu selection**, **direct manipulation**, and **natural language**.[4] Each technique has been used in query specification interfaces and each has advantages and disadvantages. These are described below in the context of Boolean query specification.

[4] This list omits non-visual modalities, such as audio.