Beyond Search

The Information Access Research Group at Apple

Daniel E. Rose

Introduction

By the end of the 1980s, the people who accessed information for a living were finally beginning to recognize the benefits of relevance-ranked full-text information retrieval (IR). As more and more information became available online, and needed to be accessed by an increasingly broad audience, systems like WAIS offered new levels of power and simplicity to searching. Users neither had to tag the data nor learn complex Boolean queries to search it. Matches could be as exact or as "fuzzy" as desired, and the system would indicate how good a match each result was. Users could ask for "more documents like these," taking advantage of the ease of recognition and avoiding the difficulty of recall. Ironically, all of these concepts were developed by the IR research community in the 1960s, yet were not widely used in commercial systems until the late 1980s. (Even today, most of the major Internet search services only take advantage of some of these attributes.)

As a result of the growth of online data, many companies began setting up groups to conduct research in the burgeoning field of information access. Generally, these groups fell into two camps. One of these focused almost exclusively on the domain of traditional information retrieval – the search engine – and reported results at the annual ACM SIGIR conferences. A common focus of this community was investigating various techniques that would enable the system to overcome the "vocabulary problem" – the fact that people use different words to describe the same thing, and in particular, end users use different words than the authors whose works they are searching. The goal was to improve precision (the proportion of retrieved documents that are relevant) and recall (the proportion of relevant documents retrieved). These two measures of effectiveness were typically obtained by running the system in batch on a set of predefined queries and predefined correct responses. If users were considered at all, it was primarily in the design of better interfaces for searching.

The second camp traditionally ignored the work of the information retrieval community, and in some cases, was actually unaware of it. Instead, researchers in this second group focused on new ways of interacting with the data once it was available, and reported their results at CHI conferences and in publications such as this one. Typically, this work took the form of various kinds of information browsers that allowed users to view document collections from a variety of perspectives. Often these tools neither used nor provided any access to the content of the document.

In the Advanced Technology Group at Apple Computer, our perspective was to take an integrated view of the techniques for interacting with information. We believed that it was meaningless to study text indexing without thinking about a user's particular task context, which might include many issues beyond search. Conversely, we felt that a visualization tool that neglected to take into account the contents of documents was failing to tap into the power of the computer. Thus, during the eight years of its existence, the Information Access Research Group chose to explore both spaces, and particularly areas where the two overlapped.

Over time, we began to think of the indexing and search technology as a way to provide what we called *content awareness* – the ability of a computer system to take actions based on the text content of the documents it is manipulating. Content awareness could provide not only information search, but a

META-1019
IPR2025-00430

rich variety of capabilities for helping the user interact with and manage information. This offered the possibility of profound changes to the user experience of nearly any computing environment. This paper is an attempt to convey some of the possibilities we explored.

It would be impossible to cover every project conducted in the history of the group; both space and confidentiality constraints prevent it. Instead, this paper will highlight a sampling of the research; the interested reader is encouraged to follow the references for more details.

Guides: Expressing Point of View

One of the first projects that demonstrated the benefits of combining text indexing technology with a novel interaction style was the Guides system [Oren, Salomon, & Kreitman, 1990; Laurel, Oren, & Don, 1990]. Designed for children studying the migration of European settlers in the American West, the Guides interface offered not only texts but also video stories from three different points of view: a Native American, a Frontiersman, and a Pioneer Woman. Students learned that historical events were subjective and that documents could be used to tell different stories. As the user browsed the texts and images in the collection (drawn from an Encyclopedia), the three "guides" would each pay more or less attention depending on their respective interests. A guide would appear to be resting or sleeping if he or she was uninterested in the current topic. In contrast, when a guide had something to contribute about the current topic, the screen would show the guide looking excited and raising his or her hand. (This is shown in Figure 1.) If the user clicked on the picture of a guide, that character would tell a video story. The guides also made suggestions (in the form of items in a pulldown menu below their pictures) for articles or images that shared each one's perspective on the issue. There was also a "system guide" who made suggestions about items related to the one currently being examined by the user. The system also provided a way to make custom guides that would represent the interest of particular children, or a particular lesson.

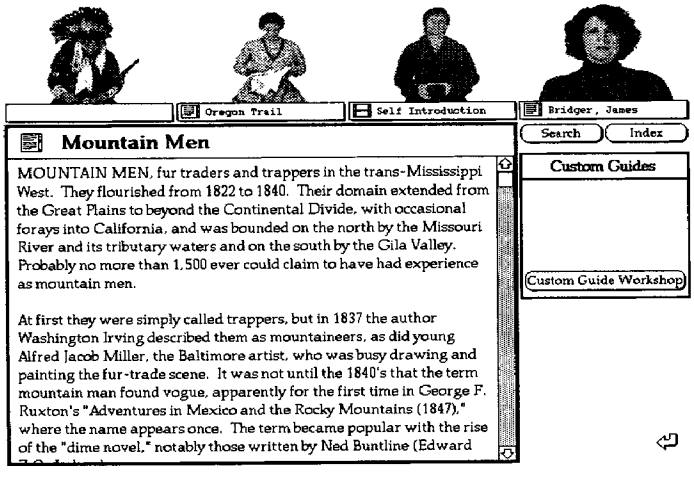


Figure 1: Guides

The Guides system also provided one of the most compelling examples of agents, without ever using that term. Unlike many so-called "agent-based" systems in which the agent is little more than smiling face on a saved query, Guides used its multiple characters not only to provide a more engaging form of interaction for users, but also to change the information users received according to each guide's point of view.

What made this type of interaction possible was the content representation that underlay it. Each guide, as well as each document, was internally expressed as a vector of content components (which were in turn obtained by principal components analysis of the term-document space). Thus figuring out how "alert" a guide was required only a simple computation based on the distance of the guide's vector with a continually updated vector corresponding to the most recent documents being examined.

Piles: Casual Organization of Documents

One of the foremost examples of our approach to information access was a joint project conducted with the Human Interface Group (HIG) that looked at a fundamentally new way of organizing documents in file system interfaces. Early advertisements for the Macintosh had stressed that the system was easy to learn and use because of the metaphorical similarity of its desktop interface with the familiar work environment of the user. A photograph illustrating the point showed an overhead view of a pristine desk with a folder or two, a document being worked on, some file drawers, and a nearby wastebasket. As nearly anyone with a desk can attest to, this view is grossly oversimplified. In HIG user studies, it became clear not only that people organize their documents in a more casual fashion than a hierarchy of files, folders, and file cabinets, but that in many cases they prefer more casual schemes for organizing collections, namely, piles. Piles have many desirable properties not found in hierarchical filing schemes, such as the ability to reveal at a glance approximately how many items they contained and (by viewing distinctive edges) which items they were, and the ability to access any document in the pile in constant time.

The result of the study was a design for a new file system interface based on the notion of piles as first-class objects [Mander, Salomon, and Wong, 1992]. One of the findings of the study was that many managers and other knowledge workers had assistants who helped manage piles, for example, by sorting new items into appropriate piles. In some cases a manager would ask his or her assistant to take a large unwieldy pile and reorganize it, dividing it into "subpiles" each of which had some topical coherence. The HIG team incorporated these ideas into the interaction design, only it was the system, in addition to the human, which was to provide assistance in organizing the documents.

But how could this vision of file organization, demonstrated so effectively in HIG's Director animation, actually be implemented? This was the task that occupied the Information Access group. Using the technology of information retrieval, we were able to create content representations that represented each document and each pile in the same information space.

Thus as early as 1991 we had a working prototype which could automatically maintain piles of actual documents, supporting both direct manipulation and automatic sorting [Rose et al., 1993]. The most compelling capability of this working prototype was its ability to perform the assistant's "subpiling" task: automatically subdividing a pile into new piles without any prior knowledge of what topics were discussed or how many piles were needed. (A before-and-after example of this process is shown in Figure 2; the system has automatically suggested names for the new subpiles based on their topical content.) This was accomplished by using clustering techniques on the underlying file representations. Although information retrieval researchers had experimented with clustering years earlier as a potential way to improve indexing, the Information Access system was, to our knowledge, the first use of clustering to alter and enhance the user's interaction with a collection of documents.

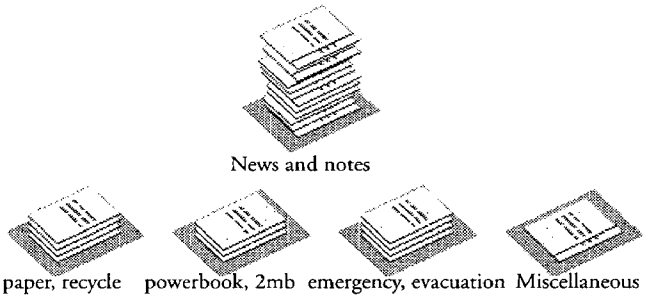


Figure 2: Piles

We would often demonstrate the system by asking a user for a collection of personal files, have the system organize them, and then let him or her judge the utility of the resulting piles; occasionally the system would discover useful groupings of which the user had been unaware.

Information Rendezvous

The Information Rendezvous project was aimed at the problem of asynchronous group communication: how to get information from those who have it to those who need it. This problem, which is evident in nearly every organization, is made more difficult when producers of information aren't sure who the interested consumers are, and when consumers don't know where to find information they might be interested in, or even that such information exists. The goal of an information rendezvous system is to mediate between producers and consumers of information.

To accomplish this task, we specified three principles under which the system would operate: First, there should be no addressing burden on information producers and no locating burden on information consumers. Second, the value of the information must be balanced with the value of the users' time. Third, the knowledge and intelligence of the user community is an essential resource in identifying useful information.

While we have discussed the motivation for these principles elsewhere [Rose & Bornstein, forthcoming], we will briefly examine how they were achieved in our prototypes.

The addressing/locating burden is evident everywhere in traditional systems used for asynchronous group communication. Senders of e-mail need to know the addresses of their intended recipients, or rely on group addresses that invariably include people uninterested in the message. Information producers using bulletin boards like Usenet need to know where to post their information; while consumers need to know to look in the same place and in a close enough window in time. Consumers of information posted on the Web or other shared databases often need search tools to find what they're looking for, and are unaware of when new information of interest becomes available.

To address all these problems, the information rendezvous approach was to eliminate the notions of addressing and locating altogether. Producers of information simply add it to a global "soup," while consumers sample the portions of the soup of most interest to them. This brings us to the next principle: managing users' time. Since many more items are potentially interesting than a user is likely to have time for, the system orders all the messages according to predicted relevance to that individual user. A user with just a few minutes of time may choose to read only the first few messages, while someone with more time to spend may examine many more.

How does the system determine which items might be most relevant to each user? We created an architecture with multiple predictors, each of which assessed the value of each message to each user according to a different criterion. One such criterion was the content similarity of this message to messages which the user has found useful in the past. Another, relying on the third principle above, used a correlation of user preferences, so that items that one user found useful would be recommended to other users who had similar tastes.

In our earlier prototype, called "MessageWorld" [Rose, Bornstein, and Tiene, 1994], the predictors were separate, and were used to prioritize different types of messages. As early as 1993, long before the term "collaborative filtering" had been coined, our prototype was recommending movies to researchers in ATG based on the correlation of their tastes. A later prototype, which was being tested by the Apple Library for use in current awareness, combined evidence from the content and correlation predictors.

V-Twin

Throughout its history, the Information Access group was committed to the notion of content-awareness as provided by the statistical indexing techniques of information retrieval. We experimented with a variety of techniques, from singular value decomposition of term-document matrices to spreading activation neural-network approaches. For some experiments, such as an investigation of the effectiveness of part-of-speech tagging for queries, we used commercial search engines licensed from other vendors. Most of the time, however, we found that to accomplish the specific task at hand (such as

implementing the "pile" idea) we needed more power and flexibility than the commercial systems afforded. Thus we would find ourselves writing yet another text indexer, a task which we performed several times (in different programming languages and development environments) during the years of the group's existence.

Finally, we believed the time was right to create, once and for all, an information access toolkit that would not only support all the kinds of investigations we were conducting, but could be a way to provide content-awareness as a standard feature of the Macintosh. The project was code-named V-Twin; it was to be more than just a search engine. Designed primarily by Doug Cutting, V-Twin was designed from the start to be modular, flexible, and portable. Nearly every aspect of the system was accessible in the API and could be specialized by a developer through subclassing. For example, V-Twin made no assumptions about what a "document" was – it could be a file on the file system, an e-mail message in an archive, or an arbitrary range of bytes. Nor was the system biased toward any natural language; the core modules were independent not only of language but of character-encoding. V-Twin's storage system was exposed as well as its indexing and search components. In fact, this combination of storage-and-indexing endeared it to customers such as Apple's Cyberdog group, which used it both for saving and searching messages in its e-mail client.

Although our aim in creating V-Twin was to enable the new kinds of interaction that could arise from content awareness, we were mindful of the need to satisfy the expectations of those who would use the system for basic searching tasks. In the information retrieval research community, a primary forum for evaluating effectiveness of a search engine is the TREC ad hoc task. TREC, the annual Text REtrieval Conference sponsored by the National Institute of Standards and Technology, has a variety of tests for searching, routing, and filtering documents. The "ad hoc" task measures searching: The system is given 50 "topics" (queries) and must find all relevant items from a heterogeneous collection of hundreds of thousands of documents. To demonstrate the effectiveness of V-Twin at traditional searching, we decided to participate in TREC. We were hoping to show that despite its relatively small resource requirements, V-Twin could perform sufficiently well to show its suitability for large search tasks. Instead, V-Twin proved to be one of the most highly rated systems at the conference. Furthermore, while most of the other groups used multiprocessor workstations, requiring hundreds of megabytes of RAM, and several gigabytes of disk to store their indexes, V-Twin exceeded their performance running on a 1995-model Macintosh in 35 MB of RAM and 462 MB of disk – less than a 23% overhead [Rose & Stevens, 1997].

Today, V-Twin is a commercial technology that can be licensed from Apple (under the name "Apple Information Access Toolkit") for use in any Macintosh application.

Ranking for Short Queries

As described above, information retrieval systems have traditionally been evaluated by measuring their performance on a set of predefined queries over a known collection. However, the majority of these queries are very long – from sentence- to paragraph-length. In contrast, we found that from 87 to 94% of the queries we obtained from actual users of interactive search systems were three words or less. The importance of this discrepancy became clear when we tested one of our first V-Twin applications.

The program, eventually known as "Apple e.g." (EG for short), was an indexing and search tool for Mac OS-based web servers. Still used by hundreds of universities, businesses, and other organizations, EG offered many users their first exposure to a visual display of the relative relevance of each search result.

However, EG users were puzzled by certain search results, and suggested that the system might not be functioning correctly. When we investigated this phenomenon, we discovered that the problem arose when there was a mismatch between the system's ordering of results and the users' expectations for short queries. For example, if a query consisted of two terms, users seemed to expect that all documents containing both terms should be ranked higher than documents containing only one term. In fact, this was not the case; a document which used just one more highly weighted term many times could easily get a higher score than one which had just one occurrence each of both terms. We concluded that for short queries, our system should improve the scores of documents with more words in common between the query and the document (this is often known as the "coordination level.") Our challenge was to do this in such a way that would neither require any additional effort on the part of users, nor cause any degradation in performance according to traditional measures.

The technique we developed, known as SQR (for short query ranking), adjusted the relevance scores from a traditional search engine based on a combination of the query length and coordination level [Rose & Cutting, 1996]. With SQR, we solved the mismatch between users' expectations and system behavior. But what about performance? When we tested the SQR-enhanced version of V-Twin, we were hoping to see little or no degradation. Instead, we found – at least for some test collections – an actual increase in precision. Since then, the SQR algorithm has been incorporated into V-Twin.

Document Summarization

Once the earliest versions of V-Twin were available for use, we quickly began building prototype applications that explored its capabilities. We were able to easily create a variety of content-aware applications, from simple searching tools to systems that recreated the automatic organization features of the old "piles" prototype on a much larger scale. These included tools for searching desktop files, e-mail, and web sites; programs to automatically sort, filter, and organize files; and many more. One of the first of these applications, created early in 1995, was an interactive summarization tool. The program allowed a user to take any text document and interactively (and instantaneously) view a condensed version in which sentences deemed less important to the content were removed. Simply by moving a slider, the user could "shrink" the document to a desired level of detail. This might be a one-page executive summary or even a one-sentence characterization of the document. This is

shown in Figure 3. Once it was shown in public, the summarization application became a compelling demonstration of V-Twin's versatility. In fact, many people mistakenly believed that the summarizer was V-Twin.

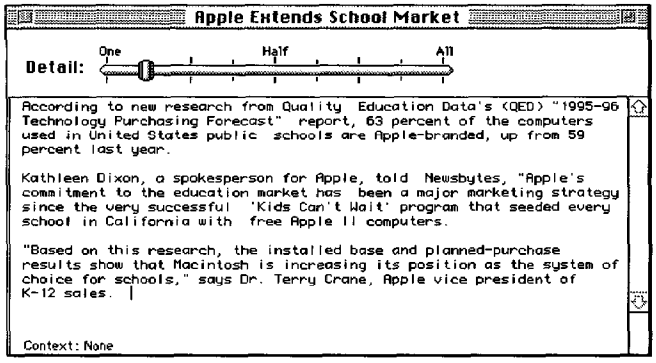


Figure 3: Text Summarization

Conclusions

In September of 1997, Apple closed its Advanced Technology Group, and with it, the Information Access Research Program. Despite this, we believe the work we did and the ideas we came up with will live on, both through those who have stayed at Apple and those who have gone elsewhere, through the products and patents we created for Apple and through the papers and reports we shared with the research community. Content awareness enables many new ways of interacting with large quantities of information. We are only just beginning to tap into its enormous potential.

Acknowledgments

I would like to thank all the people who worked in the Information Access group over the years: Brian Bechtel, Paul Biron, Jeremy Bornstein, David Casseres, Jim Chen, Harry Chesley, Doug Cutting, Abbe Don, Erik Fair, Steve Falkenburg, Andrea Gallagher, Stuart Gill, Sally Grisedale, Michele Haaland, John Hatton, Brenda Laurel, Xia Lin, Nancy Massung, Mike Monan, Tim Oren, Christian Plaunt, Dulce Ponceléon, Charlie Reiman, Srikanth Radhakrishnan, Curt Stevens, Jonathan Steuer, Laura Tognoli, Kevin Tiene, and Dan Walkowski.

References

Mander, R., Salomon, G., and Wong, Y. (1992) "A 'Pile' Metaphor for Supporting Casual Organization of Information," *CHI-92,* pp. 627-634.

Oren, T., Salomon, G., Kreitman, K., and Don, A. (1990) "Guides: Characterizing the Interface." In Brenda Laurel, editor, *The Art of Human-Computer Interface Design,* pp. 367-381. Addison-Wesley, Reading, MA 1990.

Laurel, B., Oren, T., Don, A. (1990). "Issues in Multimedia Interface Design: Media Integration and Interface Agents." *CHI-90,* pp. 133-139.

Rose, D.E., Bornstein, J.J., and Tiene, K., (1995). "MessageWorld: A New Approach to Facilitating Asynchronous Group Communication," *Fourth International Conference on Information and Knowledge Management (CIKM-95)* pp. 266-273.

Rose, D.E. and Bornstein, J.J. (forthcoming). "Information Rendezvous," *Interacting with Computers,* to appear.

Rose, D.E. and Cutting, D.R. (1996). *"Ranking for Usability: Enhanced Retrieval for Short Queries."* Apple Technical Report #163.

Rose, D.E., Mander, R., Oren, T., Ponceléon, D.B., Salomon, G., Wong, Y. (1993). "Content Awareness in a File System Interface: Implementing the "Pile' Metaphor for Organizing Information", *16th International Conference on Research and Development in Information Retrieval (SIGIR-93),* pp. 260-269.

Rose, D.E. and Stevens, C., (1997). "V-Twin: A Lightweight Engine for Interactive Use," Proceedings of the Fifth Text Retrieval Conference (TREC-5).

About the Author

Daniel E. Rose joined the Information Access Research program at Apple Computer in 1991 and managed the group from 1994 until its closure in 1997. He is currently pursuing new opportunities in the field of information and knowledge management.

Author's Address

Daniel E. Rose
6415 Myrtlewood Dr.
Cupertino, CA 95014 USA

email: danrose@acm.org

Tel: +1-408-865-1121