Published in Proceedings CHI '95, Denver, CO. May 1995.

Pointing The Way: Active Collaborative Filtering

David Maltz
Dept. of Computer Science
Carnegie-Mellon University
Pittsburgh, PA 15213-3890
dmaltz@cs.cmu.edu

Kate Ehrlich
Lotus Development Corporation
One Rogers St
Cambridge MA 02142
Kate_Ehrlich@crd.lotus.com

ABSTRACT

Collaborative filtering is based on the premise that people looking for information should be able to make use of what others have already found and evaluated. Current collaborative filtering systems provide tools for readers to filter documents based on aggregated ratings over a changing group of readers. Motivated by the results of a study of information sharing, we describe a different type of collaborative filtering system in which people who find interesting documents actively send "pointers" to those documents to their colleagues. A "pointer" contains a hypertext link to the source document as well as contextual information to help the recipient determine the interest and relevance of the document prior to accessing it. Preliminary data suggest that people are using the system in anticipated and unanticipated ways, as well as creating information "digests".

Keywords: Collaborative filtering, information retrieval, hypertext, World Wide Web, Lotus Notes.

INTRODUCTION

Connectivity, networking, and the National Information Infrastructure have become the buzzwords of the day. Underlying the excitement of these times is the promise that each of us will soon have unlimited access to a computer that will cheaply bring us information from sources around the world. We will be in direct contact with all the world's repositories of information -- no matter how small or large -- and in direct contact with the experts and people who create those repositories. Just like the upswell of creation and learning that followed the development of the printing press and the widespread access to information it created, we anticipate a new surge of knowledge to enrich our lives.

Unfortunately, our situation parallels that of the printing press in more ways than one. As the first libraries were built and books became available to larger groups of people, a new problem arose. Once buildings could be filled with more piles of books than any human could possibly read in a lifetime, how could people find books on the topics they wanted? The solution to that problem evolved into an entire field called Library Science. A solution to the problem of finding useful information on a global network promises to be no easier.

Just to put some numbers on the size of problem, it is informative to look at several of the information systems available to users. The World Wide Web allows nearly anyone with a machine on the Internet to create hypertext multimedia documents and link them to other documents at other sites [1]. The Web, started in 1990, had grown to more than 4600 servers in October 1994, and over 10,000 servers in December 1994. Together, these servers make over 1.05 million different documents available for retrieval [11]. Usenet Net News is also growing exponentially. Estimates show that in May 1993 there were over 2.6 million users of Net News at some 87 thousand sites throughout the world. These users generated over 26 thousand new articles a day, amounting to 57 Mbytes of data [13]. Just over a year later, some estimates put the number of users at almost 10 million. The problem gets worse when one considers all the information that organizations are putting on-line in file folders, databases and Lotus Notes amongst other repositories. Not only is the amount of information available to many people far in excess of what can be retrieved through informal browsing methods, but only a very small fraction of the available information will be accurate, up to date and relevant.

Building on the notion of collaborative filtering which originated with the Tapestry project at Xerox PARC, this paper describes a system that supports the informal practice of sharing pointers to interesting documents and sources of information with colleagues. We begin by reviewing existing information filtering systems.

INFORMATION FILTERING

If the problem is that people are swamped by too much information [12, 16], the solution seems to lie in developing better tools to filter the information so that only interesting, relevant information gets through to the user. Many present filtering systems are based on building a user profile. These systems attempt to extract patterns from the observed behavior of the user to predict

which items would be selected or rejected. Examples of this form of filtering include LyricTime [9] which compiles a profile of musical preferences, an agent that filters NetNews [15] and INFOSCOPE [4] an interesting hybrid news reader that frames the filtering problem as one of restructuring the information space on a user by user basis to place all the articles relevant to a user in a few known and accessible locations.

However, these systems all suffer from a "cold-start" problem. New users start off with nothing in their profile and must train a profile from scratch. Even with a starter profile, there is still a training period before the profile accurately reflects the user's preferences. During the training period the system can't effectively filter for the user. A better system would allow new users some type of access to the experiences of current users to create a better initial profile. The second drawback is that the user's searches can become circumscribed by the profile. The profile only selects articles similar to the ones the user has already read, and new areas that might be of interest can be missed completely. If the user tries to explore new content areas a new profile must be created that is customized to that subject matter and the user again faces the "cold start" problem.

COLLABORATIVE FILTERING

Most present filtering systems fail to capitalize on a key resource made available by on-line information systems, namely the knowledge and wisdom accumulated as different people find and access documents and form opinions of them. By giving users access to others' prior experience with an information source, we can create a collaborative information filter.

Collaborative filtering systems work by including people in the filtering system, and we can expect people to be better at evaluating documents than a computed function. Current automatic filtering systems attempt to find articles of interest to their user, often using some scoring function to evaluate features of the documents and returning the documents with the highest scores. People can effortlessly evaluate features of a document that are important to other people, but would be difficult to detect automatically. Examples of such features are the writing style and "readability" of a document, or the clarity and forcefulness of an argument the document contains. Imagine the difficulty an automatic filtering system would have figuring out which of two cake recipes is "easier to follow."

Another motivation for collaborative filtering comes from comparing the rich environment of real objects to the much poorer one in which computer users operate. When a user reads a computer file he usually has no way of telling whether he is the first person to ever read it, or if he is looking at the most commonly used reference on the system. Collaborative filtering works in part by associating with computer documents the history of their use. For instance, Hill et al make the observation [7] that the objects we use in everyday life accumulate wear and tear as a normal part of their use: pages in books become wrinkled, bindings creased, and margins smudged with fingerprints. The objects with more wear are the more commonly used ones, and further the wear acts as an index to relevant information inside the object. An example is the way reference books open to commonly used pages when dropped on a desk. Giving searchers access to this usage history lets them take advantage of the type of subtle hints that we commonly use when making read/don't read decisions in the real world.

Tapestry

The concept of collaborative filtering originated with the Information Tapestry project at Xerox PARC [5]. Among its other features, Tapestry was the first system to support collaborative filtering in that it allows its users to annotate the documents they read. Other Tapestry users can then retrieve documents to read based not only on the content of the documents themselves, but also on what other users have said about them. Tapestry provides free text annotations as well as explicit "likeit" and "hateit" annotations so users can pass judgments on the value of the documents which they read.

In its current incarnation, Tapestry suffers from two distinct problems. The first problem is the size of its user base. Because Tapestry is based on a commercial database system it can not be given away freely. Further, Tapestry was not designed for use by large numbers of people at distributed sites. Both these factors combine to limit the pool of potential Tapestry users to researchers at Xerox PARC. Based on anecdotal evidence, this pool does not seem large enough to support a critical mass of users. The vast majority of documents go unannotated, so there is little collaborative information to use when filtering. The second problem with Tapestry is the means by which users enter filters into Tapestry. One common interface to Tapestry requires users to specify requests for information in the form of queries in an SQL-like language. Writing such a query requires the user to have a firm sense of what types of articles he wants to read, which is a hindrance to exploration of new areas and makes it hard to browse the available information for serendipitous hits.

Collaborative Filtering Of Usenet Net News

A further collaborative filtering system for Usenet Net News was created to scale up to a critical mass of users by branching out to more sites, and providing those users with a simpler method for accessing articles [10]. This system modified two common Net News readers to provide collaborative filtering functions: voting and

display of articles based on their "popularity". Aside from these extra functions, the news readers appear and handle as they did before. Users of the modified news readers were encouraged to cast votes for or against the articles they read. These votes are used by the system to create a net-wide collective opinion on the usefulness of each article. The news reader clients can then use this information to help future readers of the newsgroup find articles of interest. Users are able to associate their names with their votes or not as they chose.

A positive result of the work on this system was finding that people really will vote. Although no reward or incentive of any kind was offered to the users of one of the modified news readers, 24 of the approximately 40 users of the newsreader voted for articles during the course of one three month study period. However, this number was insufficient to reach critical mass. That is, because of the large number of different documents, this system was dependent on lots of people reading and voting on the same documents. Users reported that when they didn't see any votes in the groups they read, they thought the system must be broken or not working so they didn't bother to vote further because their votes would be lost and wasted. If they knew something useful was happening to their votes, they would have kept voting.

Grouplens

A similar system for NetNews, GroupLens, combines collaboration with user-profiles [14]. Communities of users rank the articles they read on a numerical scale. The system then finds correlations between the ratings different users have given the articles. As viewed by a user called Jane, the goal of the system is to identify a peer group of users whose interests are similar to Jane's, and then to use their opinions of new articles to predict whether Jane will like the articles. Like other filters based on a user-profile, GroupLens suffers from the cold-start problem.

One problem with many of these collaborative filtering systems is that they require a critical mass of users to vote or leave their mark for any aggregate score to be meaningful. Until systems such as the NetNews collaborative filter are well in use, there are uncertain rewards for any user who participates. The lack of a clear reward system can be a major barrier in the acceptance of a groupware application [6]. Some systems also suffer from usability problems due to excessive overhead in either registering a vote, accessing the result of other people's votes, or creating a profile.

"ACTIVE" COLLABORATIVE FILTERING

These three examples of collaborative filtering are well suited to situations where there are a lot of documents in a single database such as NetNews. In these cases, there is a benefit to aggregating the votes of the many readers who access the database. We call this "in-place" or "passive" filtering because there is no direct connection between a person casting a vote and the readers who come later and filter documents based on these aggregated votes.

Another approach to collaborative filtering, and one that we have adopted here, builds on the common practice where people tell their friends or colleagues of interesting documents. We call this "active" collaborative filtering because there is an intent on the part of the person who finds and evaluates a document to share that knowledge with particular people. For example, as part of the World Wide Web system users collect "hotlists" which are effectively lists of hypertext links to the interesting World Wide Web pages that they have found. Several simple systems have been developed to help users format and distribute these hotlists to others, thereby spreading information about good documents on the Web. Another example in the World Wide Web context is Simon [8]. Users of this system create "subject spaces" which are effectively lists of hypertext links to the interesting World Wide Web pages that have been found, and comments on those documents. Individual people can use subject spaces to keep track of their own explorations, but they can also send their subject spaces to a group Simon server. However there is no provision in this system for sending hypertext links to particular individuals.

We see "active" collaborative filtering being useful for distributed information sources such as the World Wide Web, on-line information services, and Lotus Notes databases where users may need help simply finding the source. In "passive" collaborative filtering the system works better the higher the convergence of votes on the same set of documents. In contrast, the benefit of "active" collaborative filtering increases with the divergence of documents that are found.

DESCRIPTION OF THE SYSTEM

The design of the system was informed by the results of a recent study of information retrieval behavior in a customer support group [2]. Contrary to the expectation that support people use on-line or printed documentation to help answer customer's questions, Ehrlich and Cash report that the support people rely on each other to diagnose and solve customer problems, as well as find and interpret formal and informal documentation. Moreover, there was one person in the group who was especially skilled at finding relevant information and applying it to the problem at hand. He was remarkable for his breadth of knowledge of the domain as well as of useful sources of information, general problem-solving, and, communication skills. By virtue of these skills he assumed the role (though not the title) of an "information mediator". That is, he was available as a resource to other

FIGURE 1: *A sample pointer as seen by a recipient*

analysts in the group to help them think through especially troublesome customer problems.

Based on this study we wanted to build a system that supported some level of collaboration and information sharing amongst colleagues. More importantly, we wanted the system to provide tools that would let the "information mediators" in a workgroup easily distribute references and commentary of documents they find. Moreover, we believe that these mediators are important not only for selecting relevant documents but also for selecting reliable sources especially where there are a large number of distributed sources. Because systems such as the World Wide Web or Lotus Notes have few restrictions on authoring and no formal review process, there will be a lot of variability in the accuracy, relevance and completeness of information. Ensuring the quality of information passed on to customers is well known to librarians for whom source validation is an important part of their job [3].

The system described in this paper was developed around three key ideas that we felt were missing from existing informal methods of sharing references.

1. Package contextual information with hypertext links. Existing methods for sharing references to on-line documents are often limited to just the hypertext link, perhaps with a few comments. Yet additional contextual information about the name or location of the source, the date of the document as well as knowledge of the sender's selection biases can be used to judge the relevance of a document prior to reading it.

2. *Ease of use*. Informal systems are often awkward to use. For instance the ability to add annotations to the hypertext link may not be readily available. Not only should these features be easily accessible but the system in general should return value to the senders and recipients early on to encourage usage. It is through continued and broad adoption that an active filtering system demonstrates its usefulness.

3. *Flexible*. The main drawback of many of the informal systems is that they work well for a particular situation but are not flexible enough for general use. We intend to explore, for instance, different methods of distributing the package of hypertext link, comments and context to probe the potential extendibility of our system.

Pointers

The basic concept in our system is the "pointer". Modeled after what people do informally, a pointer contains a) a hypertext link to the document of interest; b) contextual information - title and date of document, name of source database, name of sender, and c) optional comments by the sender. A typical pointer is shown in Figure 1.

We implemented our active collaborative filter inside the Lotus Notes environment. Notes is a commercial product that provides a client-server platform for developing and deploying groupware applications. At the user level, Notes consists of documents which are stored in databases on server machines. These documents can be thought of as raw records of field-data pairs. Databases are typically organized around a topic, and the documents they contain can either be created by users inside of Notes, or gatewayed into Notes from external information sources such as Net News, World Wide Web, or clipping services. In addition to documents, each database has a set of forms and views. Database forms can be thought of as document templates containing fields that are filled in during composition time. For instance, the form used to generate the pointer in Figure 1 has a field for each item. Each database has one or more views which are ways of grouping and displaying the list of documents in the database. For instance, documents could be listed in order of document author, document date or category name. The latter being one or more keywords defined by the user to describe a document.

Typical applications in Notes are developed by creating forms, views and databases to support a new type of information, and then adding macros which specify, automate, or regulate the actions that can performed on

the information. Notes allows these macros to be embedded directly into the document as buttons whose macros operate on the document's data. Notes' data management infrastructure also provides the equivalent of a hypertext link called a doclink. Doclinks present themselves on the screen as small page icons; clicking on one of these icons opens the referenced document as a new window on the workspace.

Pointers are implemented in Notes as a special form. These forms are accessed from a button we added to the Lotus Notes SmartIcon (TM) bar. Pressing the button while a document is selected will open one of these forms and fill in the document title, creation date and database name of the currently selected document using heuristic rules. The user can add comments of any length. Since all documents in Lotus Notes are semi-structured by the forms used to create them, the information extraction heuristics are, in general, very effective.

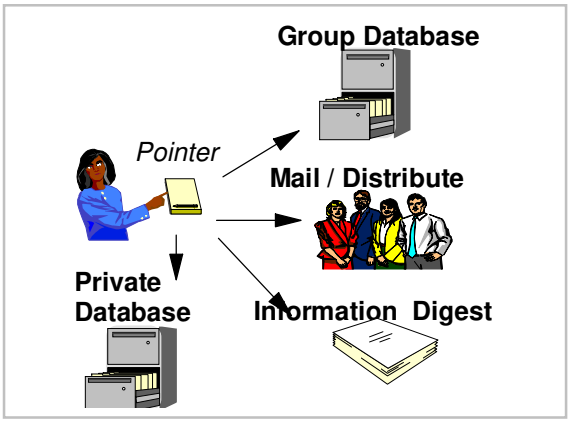

FIGURE 2: *Different ways of distributing a pointer.*

In addition to information about the pointed to document, the pointer itself includes buttons which help the sender to distribute the pointer to selected people and in a variety of ways --- this distribution process is described below. In order to give pointers some functionality which could not be easily implemented from inside of Notes, our collaborative filtering system also involves a small server process which runs outside of Notes. The buttons on a pointer form communicate with this process which then implements the pointer distribution methods, making it easy to experiment with different distribution methods by altering the server.

Distribution Of Pointers

Once a sender has created the pointer, he/she can save it in a private or public database, email it to one or more people or distribution lists, or save it in a special form we call an "Information Digest". These distribution methods are depicted in Figure 2.

Private or group database. At the simplest level, users can "bookmark" favorite documents by saving a pointer in a private database. We think of these private databases as a scrapbook for saving pointers to interesting documents especially those we might not have time to read immediately. Users can add keywords to their pointers to help organize them. The user can view their list of pointers in various ways including by date of source document, name of database or keyword category. Although not yet implemented, the system allows pointers to be saved in a public database where anyone with the appropriate access rights can add or read pointers.

Email and distribution lists. Pointers can be sent to others using email. Figure 1 shows how a pointer might look when it is opened. Whether from a database or email (which is only another database in Notes), the user opens the referenced document by clicking on the doclink icon. It should be noted that because a pointer is a regular Notes document, anyone running Notes can receive a pointer and follow the hypertext link to the referenced document; they don't need to be running the collaborative filter. Moreover pointers can be forwarded just like any other mail message.

Pointers can also be sent to a distribution list of people who have previously registered themselves as interested in receiving pointers from the evaluating user, similar to a subscription service. This means of distribution gives our system a method of filtering akin to Tapestry's notion of filtering for messages based on who has annotated them. In Tapestry, users can create filters of the form "show me the messages that Joe Blow has annotated." In our system, users can register themselves as being interested in receiving pointers that Joe Blow decides to distribute. Joe himself does not have to worry about who has registered their interest as the system will automatically lookup the registered users and mail out copies of the pointer when Joe presses the Distribute button. We considered calling the Distribute function automatically when any pointer is created, but decided that failed to give the user enough control over who sees their pointers, an especially important concern since we want the system to be useful to people for keeping pointers for their own use.

Information Digests. We have also begun exploring an advanced use in which multiple pointers are saved directly into a pre-designed document containing a combination of original text and pointers. Newsletters, World Wide Web home pages, company profiles, summary reports or even a table of contents, are all examples of this kind of document. A portion of a typical digest is depicted in Figure 3. It consists of a banner title, a subject and several sections. Users can easily add a pointer to any section of a defined digest. The system automatically extracts the information from the pointer and formats it into the digest.

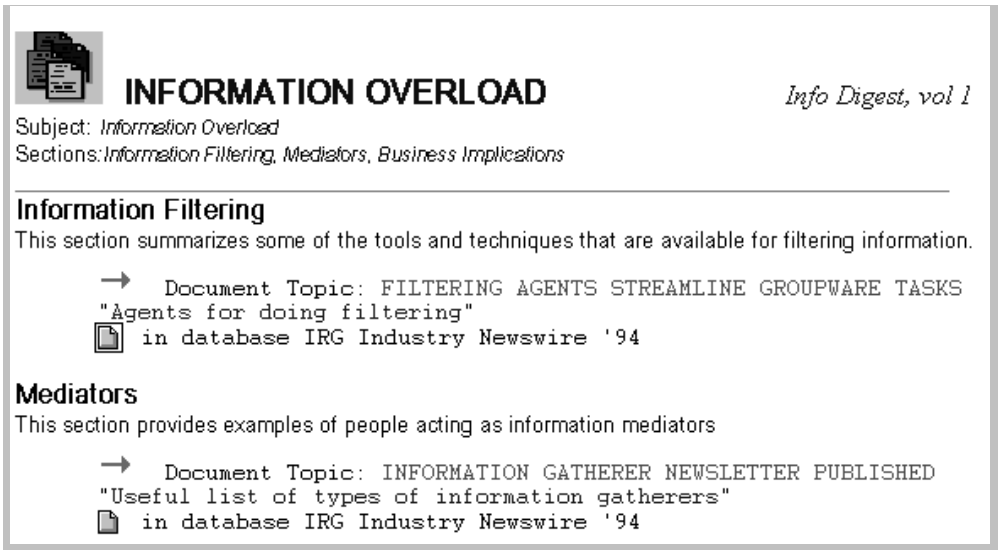

FIGURE 3: *A sample Information Digest containing pointers formatted by our system and narration text added by the author. The pointers are indicated by arrows while the added text is in Helvetica font.*

As we found in [2], there are people who excel at pulling together bits of information from many places. On Usenet Net News, these people appear as the ones who pull together FAQ's (Frequently Asked Questions) for newsgroups. One form of an information digest might be an on-line newspaper or magazine written by people skilled in selecting, editing, annotating and layout.

Passive filtering. For completeness, our prototype also contains methods to support passive collaborative filtering in addition to active filtering. If a user comes across an interesting document but doesn't know of anyone in particular the document should be brought to the attention of, she can invoke the passive filtering system on the document. The system will then annotate the document in-place so that any future readers of the database can use the annotation to help filter for useful documents. The passive filtering system works by marking the annotated document as having been read by the annotator, and creates a response note to the document which contains any comments the annotator has on the document. Our initial goal was to allow users to mark up the annotated document directly, but this turned out not to be feasible in the current version of Notes.

An active collaborative filtering system similar to ours could be created in the World Wide Web environment quite simply by better integrating Web browsers with email readers. Hypertext links would come almost for free, while information digests could be represented as new Web pages. Private and group databases of pointers might be hard to implement, as no Web clients we know of currently support organizing multiple views of data without external database backends.

Automatically extracting contextual information from Web documents may also pose difficulties. Some contextual information would be easy to obtain, such as the document title. Providing contextual information about the area of Web containing the document would be much harder, since by nature the Web has less structure than Notes databases. Based only on personal experience, it seems like providing both a link to the interesting page and a link to the parent page on which the interesting link was found is a partial solution toward providing recipients with some context on the region of the Web the page was found.

USES OF POINTERS: USER DATA

At the time of writing, the collaborative filter had been distributed to over 50 people at Lotus via email that contained a brief user's guide and the files that made up the system. In some cases the email was sent to an entire workgroup, other times individuals requested it after seeing a colleague use it or getting a demo of it.

Of the people who received the filter, over 50% installed it. It should be noted that people could receive and use pointers sent by others without having to install the filter themselves. The people who didn't install it often did not browse lots of databases themselves and were comfortable relying on others to inform them of interesting or important documents. There is thus an interesting asymmetry between "senders" and "receivers". In a rough survey we found that people in a workgroup of 10 people received 5-10 pointers per week. Approximately 80% of those pointers were sent out by just one person. That person thus serves a similar role to the "information mediator" we identified in [2]. He routinely browsed a few key databases such as the newswire database and sent pointers, including comments, to documents he thought were relevant. People were pleased to receive the pointers and did

indeed follow the hypertext link and read the referenced document if the title or comments on it made the document look interesting. As one person said, "I don't tend to browse those databases. I rely on Joe".

Another person in the group used the collaborative filter as a document management tool. She sent pointers to select people in the group alerting them to a particular document in a group-owned database. She saved pointers for herself which she grouped by date (one of the standard views built into the database) and used the Information Digest as a way of creating links between databases.

Design Goals

We designed the prototype around three key ideas: ease of use, contextual information, flexibility. Our preliminary user data suggests that these ideas were critical to the overall design.

Ease of use. It was so easy to create and send pointers - a single click brought up a partially filled pointer form - that people were encouraged to distribute pointers to their friends. We believe that a lot of this information would not have been shared otherwise and this was confirmed by comments we got from senders. One person said that he would have browsed the database and collected up references anyway but probably just waited till a meeting to pass the references on when there may or may not have been an opportunity to tell people about the documents.

Unlike other collaborative filtering systems, our system does not suffer from cold-start problems, the need to create a user profile or even reaching critical mass. In fact people who are content to just receive documents from others can begin participating without even installing the system. And those users who want to use the collaborative filter to save pointers in their private database can get started as soon as the system is installed.

An interesting difference between our system and those which rely on user profiles is that in our system neither senders nor recipients were constrained to particular topics. This meant that people occasionally received pointers to unusual databases or documents somewhat akin to people who forward mail about strange events, poems or facts that they have received or found while browsing the World Wide Web.

Context. Although sharing hypertext links is a standard feature in Notes, several people commented that they were not comfortable following a link unless they had some information about the document at the other end. Thus, people liked getting the additional contextual information in the pointer. They particularly used the document title, and the sender's comments, in judging whether to read the document.

Flexibility. The system was designed to be flexible with respect to methods of distribution (sending by email, saving in a private database or creating an Information Digest). And indeed, we found that people tended to rely primarily on just one of these methods. That is, people who routinely sent pointers to others often did not save pointers for themselves and vice versa. But more significantly, as we look at how the system is used, we see that the simplicity of it and the lack of formal structures had the unanticipated effect of letting group practices evolve. That is, rather than have a particular person (or role) be designated as the one who selects and sends pointers to others, our system was flexible enough to let such a person emerge.

DISCUSSION

Our collaborative filtering system has been used primarily within small workgroups where people know each other's biases and current interests. In fact, the system manifestly trades on making public the identity of the person sending out the pointer. This acknowledgment contrasts with other collaborative filtering systems that work hard to allow the person(s) who contribute to the filtering process to remain anonymous. We believe that knowing something about the person who selected and commented on a document is critical to evaluating the usefulness of that document.

Having the users who find the information also responsible for sending it to colleagues contradicts a common theory on improving information systems. Namely that information finders should be freed from the task of addressing mail and coming up with recipients [16]. Yet, at least one of our studies showed that often users really do have recipients in mind for the information they discover, and we believe our active collaborative filtering system serves users well by allowing them to easily act on this knowledge. Further, since the user who discovers a piece of information is mostly likely the one who knows how it should be fit in with other information, it makes sense that the information finder should have ways of easily writing down that meta-level knowledge.

Based on the informal feedback from users, our system did seem to achieve the goal of providing a simple and hence effective way of sharing knowledge of interesting documents amongst members of a workgroup. Although it has proven useful for that purpose it has some inherent limitations. One limitation is that control over the document selection resides with the sender not with the recipient and puts the recipient in a passive role. This means, for instance, that the recipient cannot use the system to find filtered/reviewed information on a particular subject unless a sender happens to have sent out a pointer to such a document.

On the other hand, the system has the advantage of simplicity and immediacy - no cold-start or critical mass

problems to overcome. It also address a need among new users to learn more about the available information space as a whole. By giving everyone in a workgroup the opportunity to save and share pointers, the burden of finding new and interesting relevant documents becomes a shared exercise. There is intrinsic reward in participating for those people who enjoy browsing around looking for information they can share with others. By providing support for an existing means of information sharing, we leverage the best of both worlds: high quality information filtering for recipients plus easy, immediate sharing tools for the senders.

ACKNOWLEDGMENTS

The first sections of this paper borrow from work done at the Xerox Palo Alto Research Center as part of Maltz's SM thesis for the Massachusetts Institute of Technology. We extend our thanks to David Goldberg of PARC and Karen Sollins of MIT who supervised the thesis work and provided valuable comments on this paper, and to Irene Greif and John Patterson for their insightful reading of a previous draft of the paper.

REFERENCES

1. Danzig, P., Obraczka, K., and Li, S-H. Internet Resource Discovery Services, IEEE Computer, September 1993, pp. 8-22.

2. Ehrlich, K. and Cash, D. Turning Information into Knowledge: Information Finding as a Collaborative Activity. In Proceedings Digital Libraries '94 (College Station, 1994), 119-125.

3. Ehrlich, K. and Cash, D. I am an Information Waitress: Bringing Order to the New Digital Libraries. 1994. Lotus Internal Report.

4. Fischer, G. and Stevens, C. Information Access in Complex, Poorly Structured Information Spaces. In Proc. CHI'91 Human Factors in Computing Systems (New Orleans,1991), Addison-Wesley, pp. 63-70.

5. Goldberg, D., Oki, B., Nichols, D., Terry, D.B. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, December 1992, Vol 35, No12, pp. 61-70.

6. Grudin, J. Why CSCW Applications Fail: problems in the Design and Evaluation of Organizational Interfaces. In Proc. CSCW '88 (Portland, 1988), ACM, pp. 85-93.

7. Hill, W., Hollan, J., Wrobleski, D. and McCandless, T. Edit Wear and Read Wear. In Proc. CHI'92 Human Factors in Computing Systems (Monterey, 1992), Addison-Wesley, pp. 3-9.

8. Johnson, M. Simon Homepage: "Welcome to SIMON," University of London, available via World Wide Web at http://www.elec.qmw.ac.uk/simon/.

9. Loeb, S. Architecting Personalized Delivery of Multimedia Information. Communications of the ACM, December 1992, Vol 35, No 12, pp. 39-48.

10. Maltz, D.A. Distributing Information for Collaborative Filtering on Usenet Net News. MIT Department of EECS MS Thesis (May, 1994). Also available as Tech Report MIT/LCS/TR-603.

11. Mauldin, M., personal communication. see also "The Lycos Home Page:Hunting WWW Information" http://lycos.cs.cmu.edu/.

12. Palme, J. You have 134 unread mail! Do you want to read them now? Proceedings IFIP WG 6.5 Working Conference on Computer-Based Document Services, Nottingham England May 1984, pp 175-184.

13. Reid, B. Usenet Readership Summary Report For May 93, Usenet news.lists, 2 June 1993.

14. Resnick, P., Neophytos, I., Mitesh, S. Bergstrom, P. and Riedl, J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In Proc. of CSCW '94: Conference on Computer Supported Cooperative Work (Chapel Hill, 1994), Addison-Wesley.

15. Sheth, B., and Maes, P. Evolving Agents for Personalized Information Filtering. Proceedings of the Ninth IEEE Conference on Artificial Intelligence for Applications, 1993.

16. Terry, D.B. 7 Steps to a Better Mail System. Proceedings IFIP International Symposium on Message Handling System and Application Layer Communication Protocols, October 1990.