The glass box user model for filtering

Jussi Karlgren, Kristina Höök, Ann Lantz, Jacob Palme, Daniel Pargman

Departments of Computer and Systems Sciences, Computational Linguistics, and Psychology, Stockholm University

Swedish Institute of Computer Science
Box 1263, S-164 28 Kista, Stockholm
Sweden
intfilter@sics.se

February 1994

Abstract

The first requirement on an interactive system in a domain such as information filtering is to be an interface to knowledge, rather than just a knowledgeable interface. We borrow the computation instruction metaphor of a system as “a black box in a glass box” as a means to conceptualize the problem of giving a user control over the actions of an interactive system. The application domain we work in is that of information filtering. In the “black box”, we hide complex knowledge of the domain objects such as facts and assumptions about text genre identification, while the “glass box”, which is what the user sees, only shows the neat top level knowledge of the domain conceptual categories such as e.g. categorization rules.

Introduction

Filtering software is recognized as one way of helping a user to manage large information flows - a problem that is becoming increasingly relevant in the emerging information society. The design of such filters involves two problems: one is finding good criteria and methods for filtering, the other is designing conceptually acceptable interaction methods for the user of the filter. Both of these problems are adressed by the IntFilter project at Stockholm University and some of the findings are presented in this report.

The first problem is harder than might be expected at first. When studying experienced users using the Usenet News conference system, we found that very few clues are in fact given to the user on which to base decisions about whether a message or a newsgroup is interesting or not. Experienced users basically filter on the title of the message, the newsgroup it appears in, and in some cases by author. This is not because these criteria are sufficient: users would filter on other factors were they easily accessible.

The second problem, designing the interaction, became obvious to us when we implemented a prototype system, GHOSTS, in which users can write personal sets of rules to be used as their personal filter[1]. We decided early on that it is important that the

[1] Pargman,1994; Palmgren, 1994.

user have full control over the filter. This is why simple rules seemed appealing, as they are easy to display, conceptually reasonably easy to grasp, and easily editable by the user. The problem we encountered was that the data the rules need to use are very complex. Since our studies indicate that users are disinclined to spend any substantial amount of time learning how to use the system, the rules cannot be too complex. This is why we propose and discuss a “glass box” user model, where certain parts are hidden from the user in “the black box”.

Filtering

New electronic media such as electronic mail and conference systems spread fast as they are adopted by corporations, government agencies, and individuals. They are effective and inexpensive means for fast co-ordination, dissemination, and retrieval of information.

With increased usage, the new media have also shown some disadvantages. The most obvious of the disadvantages is that of information overload or information overflow: simply, the receipt of more information than is desired.[2] An explanation to why electronic media are especially susceptible to information overflow stems from the skewed relationship between the costs of the sender of information, whom the new media benefit to a great extent, and those of the recipient, whose power and initiative have been reuced. Once the sender has created information, the cost in both money and effort of sending it to many rather than to few recipients is very low. When this information is sent to people who have not requested it, or have no need for it, it is they who have to bear the costs of disposing of it. On average, the time to write an electronic message is four minutes, and the time to read it half a minute[3]: ten spurious recipients thus expend more time with the message than did the sender. And adding a dozen or so spurious recipients is not unusual in electronic messaging systems.

From the user’s point of view, it is very likely that the problem is framed as a problem of insufficient quality. A user will notice that although Usenet News is an interesting medium with occasional nuggest of interesting information, most of the material is irrelevant, uninteresting, and of low quality. This, it turned out in our studies, *was* actually more of a problem in the conference system domain than was information overload. With a lowered threshold of publication, the initial filter which exerts a certain quality control at the source by demanding senders to invest greatly in the dissemination of their message is rendered moot.

If a conference system is used and controlled centrally by an organisation or used by a small group of people, problems such as these can better be addressed through social conventions[4], or by reorganising and formalising the use of the conference system through moderators or closed newsgroups, so that a certain competence or level of experience is required of its members. On the other hand, if the system has no central control and a large number of participants, as is the case with Usenet News, filter systems are a viable alternative, where filter systems are to be understood as any software tool that helps the user to structure, control, and select information from an information

---

[2] See for instance Robert M.Losee Jr., 1989; Sproull and Kiesler, 1991.

[3] Palme, 1981.

[4] Hiltz and Turoff, 1985; Sproull and Kiesler, 1991: pp.138-139.

flow in a manageable way. This paper describes the work of the IntFilter project.[5] Our work is directed towards conferencing systems, not at mail systems.[6] Although similar in many ways, conferencing systems involve some additional aspects: the information flow is likely to be less relevant on average than a mail flow, and there will be the additional problem of selecting conference groups (newsgroups in Usenet News). Large conferencing systems also are used as information bases by users, so that the filtering task also involves more information retrieval than does the typical mail organisation task.

The design of filtering software is not trivial. As outlined above, there are two ends to the problem, or two (not unrelated) research problems: one is finding good criteria and methods for filtering and the other is designing a habitable interaction between the filter and the user. This paper will address the latter as its main point, but we will first, in some detail, give an overview of our principles for the former.

Criteria and Methods for Filtering

Finding good criteria and methods for filtering is non-trivial. We have performed a series of interview studies with experienced users of Usenet News. Usenet News can be accessed with any of a large number of interface systems ("newsreaders") and most or all of them provide the user with extremely few clues on which to base decisions about whether a message or newsgroup is interesting or not.

Naturally, the first step of filtering in this type of hierarchically ordered system is to decide what newsgroups to subscribe to. This information - a user model of sorts - is saved in a file (the .newsrc file) which the newsreader accesses at every sessions.

Usenet News involves several million users[7], carries many thousands of different newsgroups and grows very fast (10-20 % per month) . Each newsgroup can have tens or even hundreds of thousands of subscribers[8]. The average traffic in Usenet News is 30-35 000 messages or 55-75 megabytes of data every day[9]. During the course of a week, several dozen new newsgroups can be created of which most belong to the alt, or "alternative" hierarchy, where 46%[10] of all Usenet News traffic by volume is transmitted.

```
Newsgroup:  comp.ai.nat-lang                 Articles:  123  of  4258/11
```

---

[5] The IntFilter project is an on-going project at the departments of Computer and Systems Sciences, Computational Linguistics, and Psychology of Stockholm University, in cooperation with SICS (Swedish Insitute of Computer Science). It is funded by NUTEK, the Swedish Board for Technical Development.

[6] As have others, e.g. Malone et al (1987).

[7] 3 million users according to Hauben (1993). 4,2 million users in November 1993 as estimated by the DEC Network Systems Laboratory after sample measures.

[8] November 1993 statistics as measured by the DEC Network Systems Laboratory, to be found in the newsgroup `news.lists`.

[9] November 1993 statistics by UUNET communications (the lower figure) and the DEC Network Systems Laboratory (the higher figure), from the newsgroup `news.lists`.

[10] January 1994 statistics by UUNET communications, from the newsgroup `news.lists`.

```
a.Manuela Boros    14  robust parsing
b.miles@minster   116  >robust parsing
c.Carole Hafner    24  >>robust parsing
d.Carole Hafner    24  >>robust parsing
e,Shona Douglas    45  Position in NLP at Edinburgh
f,Graeme Hirst     92  Research positions on Computational Linguistics
g,rshain@mitre.org 12  Autmatic Generated Abstracts
h,J.J. Paijmans    15  >Autmatic Generated Abstracts
i,Francesc Ribas   20  Request for spanish corpora
j,Ben B. Nystuen   13  Request for CD-ROM Corpora
k,Tony Davis       31  information retrieval systems
l,Ted Dunning      18  >request for ASCII-Dictionarys
m,Raj Dash         22  >>request for ASCII-Dictionarys
n,Gabriel.Climesc 282  CFP: ConTI'94, Timisoara, Romania
o,Ted Pedersen     24  unification vs. constraint based grammar
p,Alex Franz       16  >unification vs. constraint based grammar
q,Henry Thompson   29  >unification vs. constraint based grammar
r,rshain@mitre.org 12  Representing Mandarin in ASCII
s,kfc@gnv.ifas     16  >>Representing Mandarin in ASCII
t,M Covington       8  >Representing Mandarin in ASCII
u,Mark Van Haren   16  >>>Representing Mandarin in ASCII
v,Shawn O'Donnell  15  >Representing Mandarin in ASCII
w,Brian Wilkins    33  -
x,kfc@gnv.ifas     23  >Representing Mandarin in ASCII
y,Carlos McEvilly  44  >Representing Mandarin in ASCII
z,Stefan Svenberg  19  multi-lingual generation
0,Tim Dugan         4  wanted:public NLP software written in C/C++

-- 18:44 -- SELECT -- help:? -----Top 28%-----<level 2>--
```

Figure 1: A typical research-oriented newsgroup, and a typical newsreader interface.

How Users Do Today

In our initial interviews we found that users used Usenet News for various reasons, and consequently, used various methods to select interesting information. The most common reason for using News is for recreation or to keep informed about a ceratain area of interest. Some users used Usenet News to solve work-related problems, to look up specific information or to keep up with colleges in a field of research or development, but for most users, reading Usenet News is a fringe activity rather than a component of the job. For most users, Usenet News was their first experience with a large constant flow of information. This novelty of the media in itself render old methods that have been used successfully with other media (radio, television, newspapers etc.) of little value.

In a study by Wendy Mackay, e-mail users were among other categories descibed as *prioritisers* and *archivers*.[11] Prioritisers did not have time to read all the mail they received. They instead selected what was most important to them and they needed help to identify important messages, or messages that pertained to certain classes of actions. Archivers, tried to read and archive all their mail. Archivers needed help to categorize messages in one of several categories. Prioritisers and archivers have different strategies to cope with a large information flow, and subsequently have different needs for tools to aid them in coping. What strategy gets chosen depends partly on personality, but also on past experiences, the situation in itself (large or small information flow) and on the tools that are used (some actions are easier to do with certain tools than with other).

[11] Mackay, 1988.

It seems many users do not learn to use News from anybody, but have learned to use News by themselves. A number of different newsreaders are used, but most users had not tried or heard very much about newsreaders other than the one they use, and the one they use is often the first one they tried. It often seemed to be a pure coincidence that a particular newsreader was used.

One of our studies was a small pilot think-aloud study, where our aim was to investigate how experienced users of the Usenet News select information – how they decide what newsgroups to subscribe to, and how they go about to pick out submissions to read from newsgroups of interest for their work.[12] We found that experienced users mostly filter on information in the header: mainly keywords they see in the title, and partly through browsing through the text, where keywords and general layout are used for selection. In an unfamiliar newsgroup, the selection process takes longer, apparently since the title keywords are unfamiliar. In this pilot interview, none of the subjects complained about overflow *per se*, but instead stressed the low quality of contributions.

The information given to the recipient is impoverished as compared to the non-textual surface information implicit in the appearance of messages arriving through the paper mail. Envelope form, quality, size, thickness and other such factors that are used for initial filtering are completely lacking in the electronic mailbox and the incoming flow can be much larger. One aim of the project is to identify information, including, but not limited to, what is found in the header fields, that can be used to filter on.

We also asked subjects what their attitudes would be to assist the system by providing feedback about their reading habits or preferences for public use. This seemed to be an acceptable way to work, but the subjects stressed the need to retain control over what information the system would have access to.

In addition, our conclusion was that, from the viewpoint of the user, we cannot separate the filtering functionality from the problem of providing the user with an overview and information seeking tools for the entire system. We need not only a filter, but an interface which provides structure to the entire information flow. Below, we will outline the design of GHOSTS, our prototype system.

What We Get From Users

As a generalised problem, filtering is a question of categorization, or evaluation, depending on the slant of one’s method. In the case of Usenet News, the problem is *text* categorization. The crudest description is that texts are categorized into categories “interesting” and “uninteresting”, or sorted along a scale from “interesting” to “uninteresting”. We aim to evaluate texts along several parameters, or after membership in several categories: what they are for a specific domain is an empirical question. Every domain has its own patterns of variation, and we are investigating Usenet News texts for factors of this sort. We need to have users categorize and evaluate texts in order to provide us with information about relevance on typical variation of texts along parameters.

After we have a sense of what the relevant parameters are, we need to acquire observable and measurable dimensions of variation for the texts, and tractable algorithms to process

[12] The study is reported in an internal working paper of the IntFilter project by Ann Lantz (1994).

them. Using these algorithms and formulae, we can make assumptions about whether a text is interesting or not, or if it belongs to a certain category or not. This step does not involve users, other than for validation: this is a statistical problem.[13]

Topical Filtering

The first step in any message filtering task is to try to recognize the subject matter of the message at hand. For this there are standard information retrieval metrics.[14] In addition, in conference systems, there is a broad categorization by topic in that messages usually occur in conferences or newsgroups, and in discussion threads within them. This is compounded by the information found in the "Subject:" and possibly, the "Keywords:" fields. This is what our prototype system makes use of now. But this is not enough. There are numerous parameters of variation that a user could make use of, if they only were displayed in a comprehensible manner. We will outline some of the ones we are investigating below.

Beyond topic

The *dialogue structure* wherein messages are placed is useful for categorizing messages. A message which is a comment on a comment on a comment will probably not be as pertinent to the topic alluded to in the subject line as the first original message was. Also, a selection of messages from within a conversation, where messages refer to each other, will not be as useful as either the entire conversation or none of it. Similarly, a message which is *cross-posted* to a number of groups usually has low relevance for any one of the addressed groups.

Due to new advances in efficient surface syntactic processing we are able to "tag" large amounts of text very rapidly, marking up words according to grammatical category and identifying syntactic constituents.[15] We will be able to use these data for several purposes: for regular information retrieval with keyword and phrase identification, as well as for *genre identification*.[16] What the relevant genres are in the medium of Usenet News conferences is a research question, and we are currently pursuing a text study with newsgroup material, using parameters such as proportion of nouns, average sentence length, average heaviness of noun phrases and other well and less well established textual data.

Newsgroup messages, in addition to typical general textual data such as the ones mentioned above, have *medium specific textual characteristics*. Length of the message, proportion of quoted material, length of signature field, and other characteristics may all be useful for categorization. Other users' explicit or implicit *recommendations* may be useful as well, after the earlier and more robust data have been used to make the larger distinctions. Data extracted from other users' profiles is being analyzed for informational relevance.[17]

---

[13] Karlgren and Cutting, 1994.

[14] See for instance the textbook by Salton and McGill, (1983).

[15] Koskenniemi, 1983; Karlsson, 1990; Tapanainen and Voutilainen, 1993.

[16] Biber, 1990; Karlgren and Cutting, 1994.

[17] Karlgren, 1991; Goldberg et al, 1992.

Not only messages, but the newsgroups themselves can be partitioned in several typical *newsgroup types*, by data on the flow and type of messages transmitted.

Whatever data we choose to use, we will have a problem trying to explain to the user what the system does. All the data sources are on a low level of granularity – whereas the problem of the user is not. This is a question of interacting with the knowledge in the filter.

Interaction with a Filter

Coming back to the second problem outlined in the introduction, designing the interaction well, is common to all projects that have worked in the domain.[18] Naturally, the interaction with a filter is subject to all the usual caveats of interface design: situation-, user-, task-, and domain-dependence, and in this case, the underlying domain and task is of special interest to us.

The task is to give the recipient control. Today, the domain objects, the messages, do not give enough information for this to be easy: all the user gets is the information in the header fields. As outlined in the previous section, we have a grab bag of tentatively useful methods to extract information from the texts and the discourse itself, most of which on their own will not be particularly useful, but in combination with each other may yield reliable results. The problem is that the level of abstraction is too low to be presented to the user.

Our prototype system, GHOSTS, is rule based. GHOSTS is built on top of a freely available and widely spread newsreader: **nn**. GHOSTS is designed to interact with the user only through its interfaces to the system knowledge: any actions GHOSTS takes are performed through the **nn** interface. GHOSTS does not require that the messages or the text material be more structured than the newsreader itself requires. In GHOSTS, the user writes rules that customize the interaction. At present, the rules can address data on standard restricted text retrieval metrics like word occurrence in specified fields or in the text. In order to improve the filtering precision, and to be able to extend it to tentative filtering on quality as well as genre and style, we realize we need a better interface to the data that the system has access to. This has led us to borrow the metaphor of *the black box in a glass box*.

User Models - A Black Box in a Glass Box

The rule set and the subscription list (the .newsrc file) together form a user model. We stress the fact that the model must be editable and comprehensible, to give the user explicit power and integrity over the interaction. The knowledge must be separable from the system itself, and the users must feel that the user model is entirely under their control. The drawback is, of course, that a user-model based system will demand some sophistication of the user. This is all right, as long as the user's conceptual model as regards the domain can stay simple. The problem is that the rules are to use complex data.

The solution for us has been to design GHOSTS so that the complexities of text analysis can be hidden from the user. We call this, borrowing a metaphor which has been used in

---

[18] Malone et al, 1987; Maes and Kozierok, 1993.

Figure 2: The rule editor is the main interface between user and system knowledge.

discussing the teaching of computation, *a black box in a glass box*.[19] By the black box we mean the interface backdrop, behind which we hide unpalatably complex and confusing facts and assumptions about text genre identification, for instance, from the user.

The glass box is where we present the categorization rules, and let the user make use of them. With an intractable and dynamic domain but with a the user community that has a set of relatively stable conceptual categories, we can thus make use of the categories without binding ourselves for specific algorithms. The algorithms in themselves can then be improved along the way. An example of what can be hidden in the black box are metrics such as what the number of agentless passives is in a message (see Figure 3). That metric, weighed together with other similar ones – number of past-participial

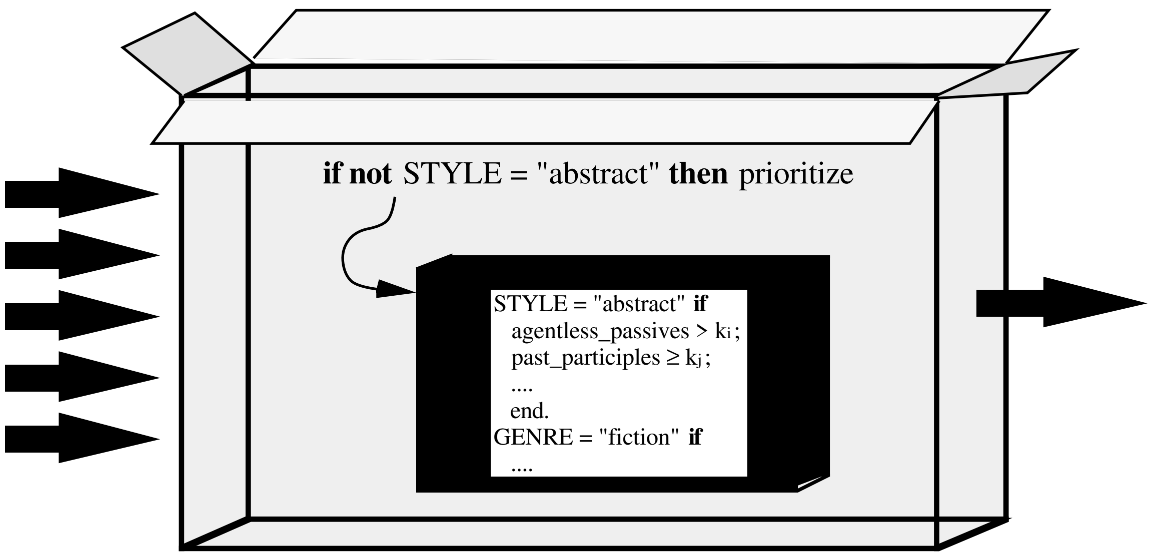

Figure 3: A glass (or openable cardboard) box, with a forbidding black box inside.

clauses, for instance – can in the glass box be represented as a judgement of whether the

[19] Du Boulay, O'Shea, and Monk, 1981.

text is of an abstract or non-abstract style.[20]

It remains yet to be seen which metrics will be most useful on the glass box level. Our assumption is that this will be closely related to how the conference system is used. these measures can be more specific and domain-dependant the more closely coupled its use is to an organizational structure and its purpose. If, on the other hand, the system is used for leisure, more general measures imported from other similar situations can be used.

Intelligence and Adaptivity

Some of the same motivations as for the black box can be advanced (and indeed, have been advanced!) for adaptive interfaces. When the domain is too complex in detail, but understandable in terms of conceptual categories, automatic learning algorithms of different types seem to be useful. The project will extend to investigating the utility of using machine learning mechanisms to learn to weight the multiple knowledge sources available for message analysis automatically, and to learn new categories automatically. However, the first step in the project is simply empowering users. Any intelligent learning mechanisms must result in user editable and verifiable knowledge of the same sort that the users themselves have placed in the model, and any intelligent behaviour must be phased in incrementally, subject to verification and later editing by the user, much as in the system Maes and Kozierok have presented.[21]

We know from other attempts to use adaptive techniques, e.g. in the design of knowledge based systems, that control and visibility are a real problem when a system starts to act on its own.[22] In the worst case, an intelligent interface that was intended to aid users in their tasks, may instead turn into an unpredictable, uncontrollable and obtrusive front-end to the system that the user is trying to use. It is necessary to give users some kind of view of the internal reasoning of an intelligent system. It does not necessarily have to be the complete picture, but again, rather a glass box that the black box resides in.

Interface to Knowledge

Finally, we should point out that even if we have emphasised the design and use of a filter, we know that the ideal filtering system is more than just a filter. As noted above, we realized from interviews with users that it is hard for many users to distinguish conceptually between the processes of filtering information, presenting information and information retrieval. To the users, the ideal filtering system is a tool that also helps them to navigate and search in information space. This suggests that the ideal filter should be an integrated part of a system that also supports these functions.

The complexity of coping with Usenet News has been addressed by the GHOSTS system by explicit and editable knowledge representations. For example, to manage newsgroup

---

[20] Biber, 1990.

[21] Maes and Kozierok, 1993.

[22] Berry and Broadbent, 86; Wærn et al. 93.

Figure 4: The newsgroup browser creates a manageable interaction with the newsgroup hierarchy.

subscription, and to structure the entire information flow in general, GHOSTS includes a newsgroup browser (see Figure 4). The only complexity the user needs to watch out for here is the unmanageable size of the hierarchy, and a simple graphical tool suffices to handle it.

The browser displays the conference hierarchy and allows for certain actions on the conferences. The simple but surprisingly useful graphical interface makes the several thousand meetings of the Usenet News hierarchy manageable and encourages exploration.

Summary

In summary, we have presented a model for filtering based on simple rules, where the data that the rules use are partly hidden by a "black box" in a "glass box". These rules may in the future be learned automatically. Hiding the data is especially necessary in view of the fact that we intend to augment the filter functionality by genre, style, and other elusive parameters of variation, most of which only can be monitored through a large number of relatively sophisticated parameters.

We have emphasised the need to allow the user to inspect and alter the filter to stay in control of its actions.

Finally, we have pointed out that the actual filter must only be one part of a system solution where information is not only filtered but also structured and presented in such a manner that the user is feels in control of the information flow.

References

Douglas Biber. 1989."A Typology of English Texts". *Linguistics* **27**:3-43.

Berry, D. C. and Broadbent, D. E. 1986. "Expert systems and the Man Machine Interface: part two: The user interface". *Expert Systems: The international journal of Knowledge Engineering*, **4**.

Benedict du Boulay, Tim O'Shea, and John Monk. 1981. "The Black Box Inside the Glass Box: Presenting Computing Concepts to Novices", *International Journal of Man-Machine Studies*, **14**:237-249.

David Goldberg, David Nichols, Brian M. Oki and Douglas Terry. 1992. "Using collaborativ filtering to weave an information tapestry". *Communications of the ACM* **35**:12:61-70.

Hauben, R. 1993. "The evolution of Usenet News: the poor man's Arpanet" Speech presented at MACUL (Michigan Association of Computer Users in Learning) in March 1993, transcript posted in the newsgroup `comp.society` on March 28, 1993.

Starr Roxanne Hiltz and Murray Turoff. 1985. "Structuring computer-mediated communication systems to avoid information overload". *Communications of the ACM*, **28**:7:680-689.

Jussi Karlgren. 1991. "An Algebra for Recommendations" SYSLAB working paper. Stockholm: Department of Computer and Systems Sciences, University of Stockholm.

Jussi Karlgren and Douglass Cutting. 1994. “Identifying Text Types Through Simple Metrics”, submission to COLING-94, Kyoto.

Fred Karlsson. 1990. “Constraint Grammar for Parsing Running Text”. In *Proceedings of COLING -90*, Karlgren (ed.), Helsinki.

Kimmo Koskenniemi. 1983. *Two-level Morphology: a General Computational Model for Word-Form Recognition and Production*. Publication no. 11, Dept. General Linguistics, University of Helsinki.

Robyn Kozierok and Pattie Maes. 1993. “A Learning Interface Agent for Scheduling Meetings”. *ACM SIGCHI International Workshop on Intelligent User Interfaces*. Orlando: ACM.

Ann Lantz. 1994. “How do experienced users of Usenet News select their information?”. *IntFilter project document*. available from DSV, Stockholm University, Sweden.

Robert M Losee, Jr. 1989. “Minimizing information overload: the ranking of electronic messages”. *J. of Information Science* **15**:179-189.

Mackay, W.E. 1988. “Diversity in the use of electronic mail: a preliminary inquiry”. *ACM transactions on office information systems*, **6**:4:380-397.

Thomas W. Malone, Kenneth R. Grant, Franklin A. Turbak, Stephen A.Brobst and Michael D. Cohen. 1987. “Intelligent information-sharing systems”. *Communications of the ACM* **30**:5:390-402.

Jacob Palme. 1981. “Experiences with the use of the COM Computerised Conferencing System”, Technical report available via gopher and ftp from dsv.su.se at Dept. Comp & Sys. Sci, Stockholm University, Stockholm.

Olle Palmgren. 1994. *GHOSTS*, Master’s thesis at Uppsala University, Uppsala.

Daniel Pargman. 1994. *Creating a Humane Information Flow*, Master’s thesis at Uppsala University, Uppsala.

Lee Sproull and Sara Kiesler. 1991.*Connections - new ways to work in the networked organisation*. MIT press.

Atro Voutilainen and Pasi Tapanainen. 1993. “Ambiguity Resolution in a Reductionistic Parser", *6th Conference of the EACL,* Utrecht, Holland.

Annika Wærn, Peter Holm, Kristina Höök, and Rune Gustavsson. 1993. “The commonKADS Communication Model”. KADS-II Report, M3/TR/SICS/006/2.0, Swedish Institute of Computer Science.