

Can readers understand their profiles?

A study of human involvement in reader profiling

Annika Waern and Åsa Rudström
Swedish Institute of Computer Science)
Annika@sics.se Asa@sics.se

Abstract

The aim in information filtering is to provide users with a personalised selection of information, based on a description of their interest profile. In some domains, users will want access to such profiles even if they are system generated.

We have performed a study of the effects of combining automatic profiling with explicit user involvement. Firstly, we wanted to explore if a machine-learned profile would benefit from being based on an initial explicit user profile. Secondly, we tested if profiles that provided better filtering also were better liked by users. Finally, we tested if users could make improvements to machine-learned profiles. We found that the initial setup of a personal profile was effective, and yielded performance improvements even after feedback training. However, the study showed no correlation between users ratings of profiles and their filtering performance, and neither did user modifications to learned profiles improve filtering performance.

1. Introduction

Personalised filtering is a type of service where users often will want to be able to inspect and modify their personal profile. There are at least three reasons for this. Firstly, this is important in domains where users are unwilling to miss out information, and for this reason reluctant to turn to purely automatic filtering. Secondly, hidden filters may raise integrity concerns, in particular on the Internet: what kind of information about you is this web site gathering? Thirdly, hidden filters will have problems coping with temporary user interests. It will help if users can explicitly remove interests that were erroneously inferred or only temporarily added.

Still, setting up manual filters is a tedious and cumbersome task. Also, just as there are users that want to have explicit control, many users are completely uninterested in this. Ideally, filtering systems should be able to combine machine-learning approaches to user profiling, with explicit profile information from users.

Despite this ideal picture, most work on automatic filtering focus on learning from explicit or implicit user

feedback to filtering results, rather than explicit user involvement in setting up and maintaining filters. This is particularly apparent in prevailing approaches to collaborative filtering [1], where the algorithms do not produce individual user profiles at all. Even in content-based filtering, the trend has rather been a move away from human-readable profiles towards more machine-oriented ones, owing mainly to the fact that these have provided better filtering performance in recent studies [2].

In the study presented in this paper, we evaluate an attempt at combining explicit profile information from users with profile modification based on feedback from usage, *in the same format*. The intention behind this choice was that this allows users to get directly involved in the construction of the filter representation. The findings were however largely (but not completely) negative. In the last section, we discuss how these findings influence our work on interface design for filtering systems.

2. Content-based and Collaborative Filtering

Information filtering techniques can be split into two types: content-based and collaborative filtering [3]. In content-based filtering, the information items to be filtered are given a semantic description. User interests are then described using the same semantic format. The semantic descriptions of information items are typically (but not always) manually generated, and not affected by usage. The user profiles are by contrast less stable, as they are either explicitly set by users or inferred from user behaviour, or both.

Collaborative filtering approaches, on the other hand, rely on information about how the information items previously have been used. Typically, collaborative filtering is used in recommender systems, systems that recommend e.g. books, films, records etc., to users based on how other previous users have liked them. A simple recommendation system would just recommend the item with the highest overall rating, but the more advanced algorithms personalise recommendations by weighting recommendations from 'similar' users higher than those from less similar users. The similarity measures in use vary, but the Pierson correlation measurement [1] has

attracted widespread attention. Using this measurement, users are deemed more similar if their previous ratings are similar, than if they differ a lot in their previous ratings. In early recommender systems, users had to provide explicit ratings of profiles, but implicit approaches are also possible.

Both content-based and collaborative filtering have their respective disadvantages. Content-based filtering has the disadvantage of requiring a representation of document content. If this description has to be manually constructed, there is the risk that people will not bother to provide a description, or that the descriptions provided will be poor. An important research issue for information retrieval is the task of providing good document descriptions by automatic means, in particular, if these are to be understood by a human reader [4]. On the other hand, content-based filtering has the advantage of low bootstrap requirements: the system can do something useful as soon as semantic profiles have been supplied for readers and documents. Collaborative filtering, on the other hand, requires no explicit representation, but does require bootstrapping. The problem is most serious when new information enters the system. New information cannot be recommended to people unless some users already has rated it. This makes collaborative filtering almost useless in some domains, such as news filtering. Some systems have tried to overcome this problem by introducing 'rating bots' [5], agents that rate new information based on simple criteria such as length, number of spelling errors, etc. These reasons have made many authors turn towards mixed content-based and collaborative filtering methods. An early attempt was made by Balbanović and Shoham [3], where an algorithm is presented that introduces intermediate 'search' profiles that reflect collective usage.

3. Social Filtering

In our research on edited information filtering [6], we have focussed on a less discussed advantage of content-based approaches. The advantage we see is that the explicit representation of content and profiles allow for a combination of manual and automatic filtering. In many domains, readers are both willing and able to do set up explicit profiles: email filtering is a perfect example of this. Moreover, the reader is not the only human involved in a high-quality filter service. The information overflow on the WWW has spurred the creation of numerous information brokering services, where skilled professionals gather, rate, organize, and resend to the appropriate reader groups. These services provide added value precisely from the human involvement in the filtering process. There exist numerous simpler but very important editor roles, such as the devoted individuals maintaining link lists, moderators of news groups and

mailing lists, and the numerous volunteered recommendations of 'good web pages' among friends. With a common term, we can name these approaches 'social filtering', as they are closely related to social information navigation [7]. The crucial feature of a social information navigation system is that it allows information to be interpreted from a social context, clearly displaying hints about things like who wrote it, who read it, who recommended it to whom.

The study reported in this paper is set in this context. Our overall aim is to develop methods for social information navigation, which allows both for automatic construction of collaborative filters, and still allow readers and editors to get involved in the filter construction process. In the reported study, we have studied how readers react to computer-learned profiles, and if they are able to improve the filtering performance by making changes to the profiles. In related studies, we are also investigating editor involvement, how editors can affect filtering performance by similar involvement in modifying document descriptions.

Current approaches to collaborative filtering are not entirely appropriate for social filtering, as they do not provide enough means for user inspection and control. This paper reports on an attempt to construct a filtering algorithm that is more appropriate for this purpose.

4. Scope and Purpose of Study

In the study reported in this paper we simulated a filtering service, that allowed readers to set up and make modifications to their filtering profiles, but that also used self-adaptive techniques for profile maintenance. The domain of application was calls for scientific conferences and journals directed towards researchers. In a parallel project [6], we have implemented such a system. We will have reason to get back to this in the last section of the paper.

The study reported here was however done entirely offline without using the implemented system. The study involved twenty-five subjects acting as readers for the simulated service, and five subjects acting as service editors. Eighty-four documents (conference and journal calls) were used in the study.

The researchers acting as editors were responsible for the selection of calls. Each of them supplied a set of calls representative for their own field of research. They were also asked to annotate the conference and journal calls with appropriate keywords. We did not for indexing agreement between editors, as these were not from the exact same field. Our aim was to obtain a collection of calls that covered a large part of the computer science field, so that the reader subjects all would find at least some interesting material in the collection.

Note that in this study, we chose not to use any automatic or semi-automatic ways for keyword generation. Originally, we wanted to provide editors with an initial suggestion based in a TF/IDF analysis of significant words in the documents. However, our informal initial tries showed that editors found very few terms that meant anything to them in these lists. This forced us to use an entirely manual method for term selection, where editors were free to select any terms that they found highly significant for the documents. We encouraged editors to not only use topical terms, but also annotations such as names of people in the program committee, location of the conference, etc.

To achieve some commonality between the annotations from different editors, the lists were edited in two ways. Firstly, very long terms (combinations of more than two words) were cut into two-word combinations (as well as kept in their original form). Secondly, slight variations in form between editors were harmonised. The harmonisation concerned spelling, choice of the singular / plural form, spelled-out abbreviations versus abbreviations, and similar variations.

4.1. Reader involvement

The reader subjects were asked to do three things. The first reader task concerned rating the documents. We asked the readers to envision a service that would send them journal and conference information by email. To achieve a rating, we asked them to sort the entire set of eighty-four documents into two heaps: those that they would have liked to see arriving in their email in-box, and those that they would have liked to not see in their inbox. Our subjects turned out to have very different preferences in this process: some readers wanted a very narrow selection of documents (down to three documents), whereas others wanted to see more than half (up to 50) of the documents. The difference was due partially to the fact that some of the subjects found little material in the collection that was relevant for their own research. But there was also a very large difference in the breadth of interest: some people were interested in obtaining a very wide range of calls just for information, whereas others would prefer a very narrow selection targeting their exact current research topic.

The second task concerned setting up a personal profile. This was done based on a list of available topic terms, but readers could also freely add terms. The list of terms that we provided was compiled from editor annotations: we included all terms that were used to annotate more than one document. In total it consisted of 282 terms. Readers were asked to mark the terms that fit their own interests. They were also allowed to add additional terms to describe interests that they thought were missing from the lists. The reader that added most

terms on his own added 21 terms, on top of the ones he selected from the precompiled list. This setup worked well. Readers were able to select terms that they found fitting their interests, and when they found a term that they did not recognize, they commonly (and rightly) inferred that it was peripheral to their interests. In general, readers were rather satisfied with their initial profiles, and preferred them to the machine-generated ones (see section 7).

When the initial data had been collected, it was used for training and testing computer-learned filters. Users were then asked to provide feedback for two different machine-learned profiles. We asked them both to rate them, and to make arbitrary changes to them. This final test was done by email, approximately three months after the first session. Out of the twenty-five subjects involved in the study, twenty-two responded to this last email request, and provided complete or almost complete feedback to the generated filters.

4.2. The algorithm

In this study, we were keen to test a collaborative filtering approach as well as a more classical approach. This proposed a problem to us, as the most widespread algorithms for collaborative filtering do not generate any individual user profiles.

For this reason, we decided to use a non-standard filtering algorithm. Since the algorithm was non-standard, we need to describe it at least superficially.

As in many similar approaches (see e.g. [8]), the generated profile consisted of a set of features with associated values. The features are in our case a list of terms, and the values can be both positive and negative, indicating positive or negative interest.

The collaborative algorithm updated *both* reader and document profiles on feedback. Assume that a feature F has value V1 in the document profile and V2 in the reader profile. Then each of the values are changed with a delta value

$$\Delta_{F} = (1 - P(\text{Feedback})) * |V_1 - V_2|$$

where P(Feedback) is an estimate of how likely the user was to provide this particular feedback, given the current user and document profiles.

The delta value was then used to update the values for feature F in both profiles. Conceptually, if the feedback was seen as negative, the values were to be pushed further away from each other, and if it was positive, they were drawn closer to each other. This was achieved by associating each possible feedback action with a K value, that could either be positive or negative. The feature values were then adjusted as follows.

$$W_{iF}(T+1) = W_{iF}(T) + K * \Delta_{F} * (\sim W_{F}(T) - W_{iF}(T))$$

where $\sim W(T)$ denotes the average value of the feature at time T.

In this particular domain, users only provided two types of feedback: the feedback that they wanted to see a document, or that they did not want to see a document. During an initial tuning session we selected K values for each of these. The goals of this tuning process was not to optimize precision and recall, but to retrieve roughly the correct ratio of retrieved documents (18%).

Whenever two profiles change, they also ‘inherit’ terms from each other, so that eventually most documents have values associated to most of the terms used in reader or document profiles. Most of the terms have values close to zero. This is clearly not optimal from a computational perspective, but it was feasible for this kind of simulation study.

The method for updating filters that we used was inspired by that suggested by Wafsi [8]. We adjusted the size of the change according to how probable the system thought that the reader’s action was. In our setting, we simply used a distance measurement as the basis for estimating this probability: the more similar the vectors were, the more probable should the save action be. We tested a number of different distance measurements for this purpose and found no large differences in the outcome in filtering performance. We ended up using the sum of absolute differences in feature values, but we could just as well have used the dot product or a cosine distance measurement.

Note that by using this method for feature updates, both document and user profiles are changed. This is the reason why this algorithm can be considered to be collaborative. Eventually (after quite extensive usage), the document profiles will reflect the collective usage of documents, rather than the original editor’s model of the document. Furthermore, through interaction with this type of document profiles, reader profiles will also start to converge, so that readers who tend to like similar documents tend to build up similar profiles. The approach is very similar to that taken by Balbanović and Shoham [3] in their mixed approach to content-based and collaborative filtering. During the brief experiment reported in this paper, this kind of behaviour did not have time to surface. Some of the document profiles showed a bit of interesting changes, that however fall outside the scope of this paper.

We used two different versions of the basic algorithm in the study. The first version started out by using both the document and reader profiles. It then modified both profiles depending on reader feedback, as described above. The other algorithm started out from using only the document profiles, and built up the user profiles from these. In this version, only the reader profiles were changed on feedback; document profiles were kept stable. This was a compromise. In this setting, the user profiles built up very slowly and unevenly, and there was a risk that the influence from user profiles on document profiles

would be very unevenly distributed. Note that due to this compromise, the algorithm is no longer collaborative: reader profiles are only affected by their own ratings and not by the ratings of others. The visible effects of this compromise in terms of user profile content should however be very small. Note that there was very little visible collaborative effects already in the first setting, even though the initial user profiles were used in that setting.

In the further discussions, we will refer to the profiles generated in the first setting as ‘system-modified’, and in the second setting as ‘system-generated’.

5. Filtering Improves when Users Supply an Initial Profile

We now turn to our first question. Does it improve filtering to use initial reader profiles as a basis for filtering? This turned out to be true in our study.

5.1. Experiment setup

This test was done in a ‘train and test’ setup. First, the filters were trained using a randomly selected portion of all information about readers’ ratings of documents. Then, the resulting filters predictive power was tested on the rest of the readers’ ratings of documents. We used three setups: one that simulated a new system, one that simulated a stable system with a new reader, and finally one that simulated a stable system. In the full test, we also included a comparison with more standard algorithms, to test that our filtering algorithm did perform reasonably well. Reporting on the full effects of this study would go outside the scope of this paper, but we found that our non-standard algorithm in general did no worse than more traditional filtering methods. It performed comparably but not better than e.g. the GroupLens filtering algorithm [3]. Note that we did not optimize the algorithm for precision and recall, only for retrieval rate. (It should also be said that our algorithm was computationally rather inefficient.)

Of all of the algorithm tests we performed, three test setups are relevant for the topic of this paper.

- The first test is that where we used the initial reader and document profiles directly for filtering, without any additional training. The test included four test runs in this setup, re-sampled from the data base of all reader ratings. We will refer to this case as the ‘original profiles’ case.
- The second test is that where both the initial reader and document profiles were used, but in addition half of the reader data was used to modify the profiles. The other half was used for testing. The test included eight test runs in this setup, resampled from the

database of reader ratings. We will refer to this case as the ‘modified profiles’ case.

- The third test was done in the same manner as the second, but the initial reader profiles were not used. We will refer to this case as the ‘generated profiles’ case.

Note that we did not use ‘hold-out-one’ testing (training on all but one observation) in the simulation of a stable system. Since we had the complete set of reader ratings for all documents as our database, ‘hold out one’-testing would be highly unrealistic. In practice, it would simulate a situation where readers were asked to rate all documents except one before filtering would start, in our case 83 documents. Our ‘stable system’ test simulated a situation where readers on the average had to rate 42 documents in order to train the system, which still is a bit optimistic.

5.2. Experiment results

	Average Precision	Average Recall
Original profiles	0,21	0,7
Modified profiles	0,35	0,77
Generated profiles	0,31	0,62

Figure 1. Filtering performance of the algorithms (Figures significantly lower than those obtained in the modified profiles setting are shown in bold.)

The results from the test runs are shown in figure 1. The test shows that precision is significantly better (at the 5% level) for the modified reader profiles than for both the original profiles (ANOVA, PSLD $P < 0,0001$) and the generated profiles ($P = 0,038$). Furthermore, the generated profiles have significantly better precision than the original profiles ($P = 0,001$). The generated profiles have significantly worse recall than the modified profiles ($P = 0,003$), but the other differences in recall are not significant.

These results show that the readers’ initial profiles did help in filtering, as compared to the case when the profiles were learned from document annotations alone. Furthermore, both of the learned filters outperformed filtering on the original profiles alone. Note that this holds even after quite extensive training: after all, we simulated a situation where users had to rate more than forty documents. This shows that at least initially, it did help to combine system and user intelligence.

6. Reader Feedback Study

We now know that readers were able to help the filtering algorithm by setting up initial profiles for themselves. The crucial question now is if this was still possible, when the system had learned (or modified) a profile based on user feedback. Given that a system has generated a profile, can readers inspect this profile and rate it, or make improvements to it?

6.1. Experiment setup

In the second part of the study, we aimed to test both readers’ ability to rate profiles, and their ability to make useful changes to them. We used the results from the previous study to select one particular test run that generated good filtering results, and had readers give feedback on the profiles generated in that test run. The test run that was selected was one where the modified profiles achieved a precision of 0,43 and recall of 0,84 on the test data. The generated profiles from the same test run were quite not as good filters, and obtained a precision of 0,33 and recall of 0,6.

These were the profiles shown to users. Unfortunately, the complete learned profiles could not be shown to readers (they contained over four hundred terms). Instead, the profiles were cut down to contain only the forty most significant terms with their corresponding weights. We selected thirty terms with positive influence on the reader’s interest (positive weights), and ten with negative influence on the reader’s interest (negative weights). When these truncated profiles were used for filtering, the filtering results were not as good as for the full profiles. In particular, recall was lower with the truncated profiles than with full profiles (the exact figures are shown in figure 3). All results presented in this section reflect filtering performance of the truncated profiles and their respective modifications, rather than on the full profiles.

The subjects were asked to do four things:

- They rated each of the profiles on a scale 1 – 5, five being highest.
- They rated the relative quality of the trained profile, compared to the profile they originally set up.
- They were asked to make any changes to the profiles they wished.
- Finally, they rated the quality of the profile *after* these changes.

Of the twenty-five subjects that participated in the first study, twenty-two provided partial or full answers to the feedback questions. One subject did not rate or make changes to the generated profile. Five subjects missed giving one of the eight requested ratings.

7. Readers Cannot Recognise Good Profiles

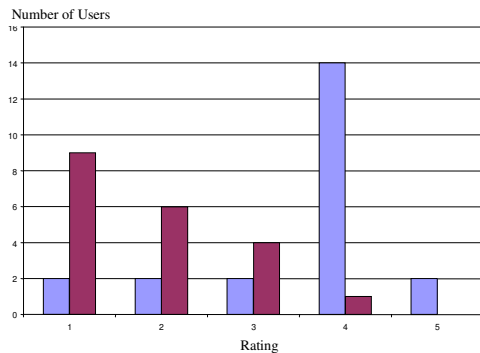


Figure 2. Rating of the computer-learned profiles Staples to the left represent the ratings of the computer-generated profiles, and staples to the right the modified profiles.

Figure 2 shows how readers judged the two profiles. Readers liked the modified profiles significantly better than the ones that were generated from document keywords alone. The large difference in rating here is largely dependent on the fact that readers were allowed to compare the two machine-learned profiles with their own original profile. Since there was a larger resemblance between the modified profile and the original profile, readers had a strong positive bias towards this profile. What's more interesting is the fact that about half of the readers actually thought that the modified profile was an improvement over their initial profile.

This result might seem encouraging, as the modified profiles also provided better filtering results. However, further analysis shows that this most likely is a coincidence. We have reason to be suspicious already from looking at the relative ratings of the original profiles and the modified profiles. We know (see figure 3) that both of the machine-learned profiles performed better (at least in precision) than the readers' original profiles. But readers still tended to like them less than the original profile (the overall average rating was -0.8 with an average of zero for the computer-modified profiles).

When studying the individual user responses, we found that the readers' ratings do not correlate with either precision or recall. In both cases, the overall trend is towards a weak negative correlation (no significant correlation was found), hinting that users actually liked worse profiles better than good ones!

The overall conclusion is that in our study, we found no reason to believe that user ratings of profiles had anything to do with their filtering quality.

8. Reader Changes do not Improve Profile Quality

	Before reader changes		After reader changes	
	Precision	Recall	Precision	Recall
Modified profiles	0,43	0,23	0,37	0,22
Generated profiles	0,33	0,29	0,28	0,32

Figure 3. Precision and recall, before and after reader changes.

We will now investigate whether readers were able to make changes to their profiles, that improved filtering performance. In this study, we made use of the feedback data from readers, where they were allowed to make arbitrary changes to the profiles, and turned it more or less directly into machine-readable filter profiles.

In order to achieve this transformation, we had to do a bit of editing. Not all readers had followed the instructions correctly. In the algorithms we used, we limited the values for features to the range between -1 and 1 . Some users entered values outside this scope that had to be scaled down. Secondly, some users rearranged features rather than modifying their values. In this case, we had to select values for the rearranged features to fit their ordering. This process was, if not nontrivial, fairly straightforward and provided very few significant choice situations, but it was definitely a potential source of error. These problems occurred because we used an entirely offline method for the feedback study. The study would have benefited greatly from an online questionnaire.

The edited profiles were then fed back to the filtering system, and used to filter the test portion of the data of the selected test round. The results over all users are shown in figure 3. As is clear from this table, overall results show a performance degradation for both types of profiles. The only improvement found is that for recall, in the case of generated profiles.

To explore this result further, we explored if the number of changes a person made affected the filtering quality of the resulting profile. We found no such relationship, no matter if we looked at all types of changes, or only 'major' changes (the concept of a 'major' change is discussed further below).

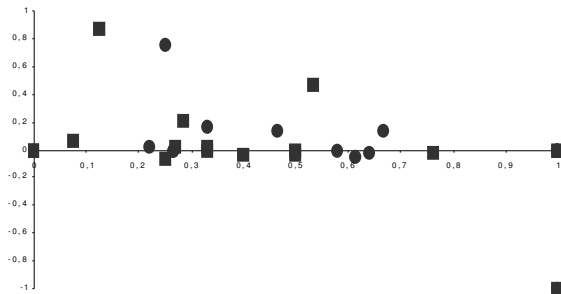


Figure 4. Rate of improvement in precision, versus original filter quality. The data for the modified profiles are shown as square dots, and the data for the generated profiles as circles.

One reason for these poor results was that the machine-generated profiles we showed to readers already performed fairly well. It is hard to improve on a profile that already provides good filtering. If we instead look at the correlation between the original filtering precision, and the precision after changes, we see that readers were better at making changes to poor filters than to good filters. The graph in figure 4 shows how the relative improvement (or degradation) in filtering performance after user changes is related to the filtering quality of the machine-learned profile. The graph shows a weak negative correlation. The correlation between the precision prior to reader involvement, and the change in precision after changes is -0.48 . The corresponding trend line crosses the zero line when the machine-learned precision is at 0.46 . This means that if the quality of the trained filter is lower than that, the user changes are more likely to lead to precision improvement than improvement degradation. The figure is notably high, in particular in comparison to the performance of the initial profiles, which was slightly above 0.2 (see figure 1), and to the chance level, which was on the average 0.18 (varying slightly over test runs).

This potentially positive effect of user changes hold only for precision, however. There is no similar correlation between recall for the machine-learned profiles, and the improvement in recall after user changes.

9. Other Observations

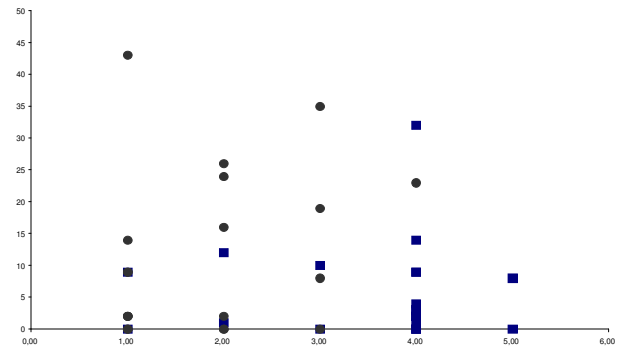


Figure 5. Number of changes users did to profiles, versus profile ratings. The data for the modified profiles are shown as square dots, and the data for the generated profiles as circles.

It is useful to see if the readers' ratings influenced the way they made changes to their profiles. To arrive at this analysis, we distinguished between major and minor changes. When a reader added or deleted a term from a profile, this was considered as a major change. In the same manner, we considered it to be a major change if the reader changed the sign of a term, from positive to negative or from negative to positive. All other changes were seen as minor changes. These occurred when a reader changed the weight of a term (or just rearranged terms).

We found no correlation between the total number of changes to a profile and the reader's rating of it. Most readers made many minor changes to profiles. If we look only at major changes, a trend occurs. The major changes seem to be more common when readers dislike a profile (see figure 5). The trend shows most clearly in the top values for changes. There is no significant difference between means, nor any correlation between the ratings and the number of changes. Note that independent of profile rating, there are always a number of users that make very few changes. As we stated in the introduction, even in we can expect some users to want to be in control over their profiles, not all users will bother about it.

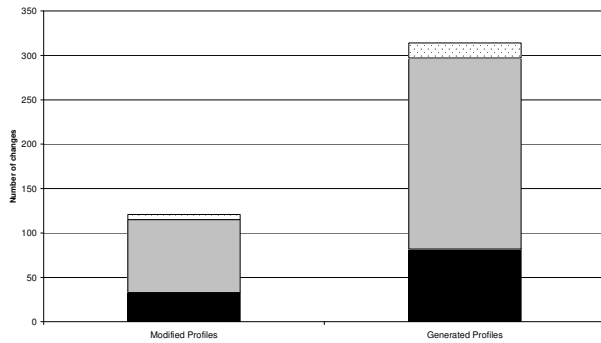


Figure 6. Number of major changes to profiles.

It is also interesting to study what types of major changes that people made. Figure 6 shows the number of major changes for the two profiles, separated into the respective types. The bottom black section of the blocks show the number of added terms. The total number of deleted terms is shown in grey immediately above. At the top of the blocks, there is a small layer corresponding to the number of terms that were changed from positive to negative significance, or vice versa.

We can see that people made many more changes to the generated profiles, than they did to the modified profiles. This is only to be expected, as the latter were in many cases very similar to their original profiles. Note that this difference had little effect on the performance change: we have already mentioned the fact that we found no correlation between the number of changes people did to their profiles and the change in filtering performance.

Readers were much more likely to delete a term that they thought unfitting, than to add a term that they thought was missing from a profile. In total, readers deleted 297 terms and added 89 terms, summed over all readers and both profiles. This might explain why reader changes had little effect on recall. When readers delete a term with positive sign (which most often is the case), this cuts down the number of documents retrieved. This would decrease recall even if the profile had been randomly generated. It is interesting to note that readers were more prone to delete terms than to add new terms, even though they had access to their original profiles. When readers did add terms, they were almost invariably picked from their original profiles.

10. Design for Reader Involvement

The findings from this study provide a good source of information about how filtering interfaces could be designed. We cannot trust readers to improve on filter quality. But under certain circumstances (such as when the system is initiated, or when filtering performance is poor), reader involvement could improve filtering performance.

Furthermore, the improvement may persist even after the filter has been extensively tuned.

One design suggestion that goes half the way was proposed by Michael Pazzani [4], where readers were allowed to see and approve on profiles, but not to make changes to them. Our study shows that this approach may be cumbersome as well. The problem is that the subjective acceptance of a profile depends highly on how well the reader recognises the terms and words used in his or her profile. In our experiment, subjects did in general not accept the profiles that were generated from the editor-selected document terms, rather than based on their own expressions of interests. Furthermore, reader ratings did not correlate with filtering quality.

In light of the findings in this study, we would like to advocate two alternative approaches to the design of filtering systems. The first approach is to entirely hide the details of filtering from readers, but instead give them very rich ways to navigate the information space on their own. This kind of design will tend to look a lot like information navigation tools, only that the filter profile affects what is shown when the view is changed. This type of design underlies the approach of adaptive hypermedia, see [9].

When there is need for reader control over the filtering profiles, readers might be helped if the system provides information about filter quality. Readers might be less prone to make changes to profiles, if the system can show them that the profiles indeed perform well. We have taken this approach to interface design in the conference call filter project ConCall briefly mentioned earlier [6]. The domain for this project also concerned calls for paper and participation to conferences and journals. ConCall is a system for researchers, in which they get continuous information about upcoming conferences and journal calls in their own area of research. We believe that as a reader group, researchers are usually reluctant to trust a filter that they do not themselves control. For this reason, we chose a design where readers were entirely in control of their filtering tool, setting it up and changing it manually.

The interface for the ConCall system is shown in figure 7. The figure shows two filters: one is the filter set up by the user and used for filtering. In parallel with this manual filter, the system maintains an automatically generated profile *suggestion*.

To aid the reader in the profile management task, the system continually monitors the performance of the manual filter and compares it to the performance of the automatic filter. Both filters are assessed by their predictive power. The filters are first used to predict

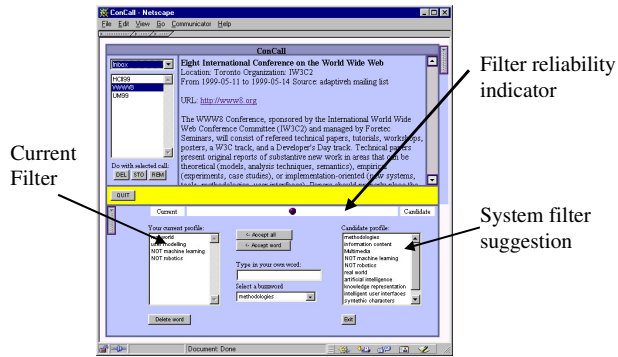


Figure 7. The ConCall user interface.

the reader's interest in a certain document. The filter reliability is then updated depending on whether the prediction was correct or not. If the filter predicts that a document will be read, and the reader does not read it, the reliability of the filter is downgraded. The same thing happens if the filter erroneously predicts that the reader will not read a document. If the prediction was correct, the reliability is upgraded. The ConCall system uses a very simple measurement for reliability: it simply calculates the guess rate for the last fifty documents.

11. Conclusions

We have studied how reader involvement affect filtering performance for a machine-learned filter which is expressed in a human-readable form. The result shows that readers can indeed provide useful information when filtering performance is poor, and that the effects of such reader involvement can persist even after quite extensive training. However, readers were not able to judge filter quality, or make improvements to filters that already had good filtering performance.

In light of these findings, we advocated the use of filter designs where filtering is either made completely invisible, and readers instead can navigate through information, or where readers are made aware both of filtering content and performance. We exemplified the latter approach by the design of the ConCall filtering interface. In this system, readers are entirely in control of their filters, but the filter maintenance is supported by improvement suggestions from the system, as well as by a continuous evaluation of filtering performance.

12. References

- [1] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J..GroupLens: Applying Collaborative Filtering to Usenet News. *Communications of the ACM* 40,3 (1997), 77-87.
- [2] Michael J. Pazzani. Information Filtering, Classification, and Extraction: tutorial notes. Intelligent User Interfaces conference, New Orleans, Louisiana, January 2000.
- [3] Marko Balabanovi'c and Yoav Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM* Vol 40 No 3, 1993, pages 66-72.
- [4] Michael J. Pazzani. Representation of Electronic Mail Filtering Profiles: A User Study. In proceedings of the Intelligent User Interfaces conference, New Orleans, Louisiana, January 2000.
- [5] Sarwar, B., Konstan, J., Borchers, A., Herlocker, J., Miller, B., and Riedl, J..Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. *Proceedings of the 1998 Conference on Computer Supported Cooperative Work*. Nov. 1998.
- [6] Annika Waern, Mark Tierney, Åsa Rudström, Jarmo Laaksoilahti and Torben Mård. "ConCall: Edited and Adaptive Information Filtering". in proceedings of the Intelligent User Interfaces Conference, Los Angeles, Cal., January 1999.
- [7] Munro, A., Höök, K., and Benyon, D. *Social Navigation in Information Space*, Springer Verlag, August 1999.
- [8] Ahmad M. Wasfi. 1999. Collecting User Access Patterns for Building User Profiles and Collaborative filtering. In *Proceedings of the International Conference on Intelligent User Interfaces*, Los Angeles, California, January, ACM.
- [9] Peter Brusilovsky, Elmar Schwarz, & Gerhard Weber. A Tool for Developing Adaptive Electronic Textbooks on WWW. In *Proceedings of WebNet'96 - World Conference of the Web Society*, June 1-22, 1996. Boston, MA, AACE. - pp. 64-69.