An Adaptive User Profile for Filtering News Based on a User Interest Hierarchy

Sarabdeep Singh

Faculty of Computer Science Dalhousie University Halifax, NS Canada B3H 1W5 ssingh@cs.dal.ca

Michael Shepherd (Corresponding Author)

Faculty of Computer Science Dalhousie University Halifax, NS Canada B3H 1W5 shepherd@cs.dal.ca

Jack Duffy

School of Business Administration Dalhousie University Halifax, NS Canada B3H 1W5 jack.duffy@dal.ca

Carolyn Watters

Faculty of Computer Science Dalhousie University Halifax, NS Canada B3H 1W5 watters@cs.dal.ca

A prototype system for the filtering and ranking of news items has been developed and a pilot test has been conducted. The user's interests are modeled by a user interest hierarchy based on explicit user feedback with adaptive learning after each session. The system learned very quickly, reaching normalized recall values of over 0.9 within three sessions. When the user's interests "drifted", the system adapted but the speed with which it adapted seemed dependent on the amount of feedback provided by the user.

Introduction

During the past few years, electronic news has drawn a lot of attention not only in the research world but also in the commercial world. There is a huge amount of news available and it is part of the fabric of most people's lives. Readers are not interested in all news items equally on any given day, i.e., some sort of selection or filtering is done by the reader. In designing news filtering or personalization systems, we must understand what people expect to gain from “getting the news”. Two behavioral theories that have been applied to news reading are *uses and gratification* and *play* or *ludic* (Dozier & Rice, 1984).

The uses and gratification theoretical perspective is based on the assumption that the reader has some underlying goal, outside the reading itself, that reading the news satisfies. Such news reading is an example of extrinsically motivated behaviour in that there is some reward to be gained by engaging in the activity (Deci & Ryan, 1987).

The ludic or play theory of news reading (Stephenson, 1967) is an example of intrinsically motivated behavior (Deci & Ryan, 1987) in that the activity appears to be spontaneously initiated by the person in pursuit of no other goal than the activity itself. This theory asserts that, “... the process of news reading is intrinsically pleasurable, and that intrinsic pleasure is at the root of a mature, orderly, and highly ritualized form of news reading as well as a more casual, spontaneous, and unstructured form of news reading.” (Dozier & Rice, 1984)

The research described in this paper attempts to model the user’s interests for ludic news reading behaviour, i.e., general reading of the news with basic themes of interest that may change slowly over time. The user’s hierarchy of interests is similar to that proposed by Kim & Chan (2003), but is adaptive in the sense that once the initial profile or interest hierarchy is built, the leaf categories of the hierarchy are updated after each session with the explicit feedback of the user. In this update procedure, the weights of existing profile terms are updated, new terms may be added and new leaf categories may be added. This adaptive phase continues the learning and can also model the “drift” in user’s interests over time.

The contribution of this research is the adoption of a learning mechanism for a user interest hierarchy that permits the building of the individual user profile, application of the profile for ranking of news items, and is sensitive to gradual shifts in user interests.

Section 2 of this paper presents background material on news filtering and on knowledge acquisition and modelling. Section 3 describes the hierarchy of the user’s interest and the update procedure. Section 4 presents the evaluation methodology and the results of a pilot user study of this approach. Section 5 summarizes the results and points towards future research.

Related Research

This research really has two components; the filtering of news items based on an adaptive user profile and the profile model itself. The reading of news is a social phenomenon and is both very personal in nature and very community-oriented in nature in that news connects the user to various communities in which the reader participates or has an interest (Asp, 1981). The algorithmic component of this research is to develop a model of the user's interests that can effectively retrieve a broad range of articles of interest and adapt to "drift" in the user's interest over time.

Filtering News Articles

Electronic news delivery systems have been available for several years. Systems may retrieve news items using very coarse-grained profiles from one or more designated sites, like *The Wall Street Journal* or *The New York Times*, and present the items in an integrated format. For example, the reader may choose the business section from *The Wall Street Journal* and the arts section from the *The New York Times*.

The creation of fine-grained personalized editions, where individual stories are chosen from a variety of sites for a particular user, is a very different problem. It is virtually impossible to predict what items a reader will read in today's news based on a history of the items a reader has read over the previous few days (Allen, 1990). There is also a concern that a very narrowly defined user profile will defeat the social and context function of news by filtering out all news except that identified by the profile. The reader will be exposed to no new items of potential interest and the reader may not receive the information necessary to participate fully as a citizen in the local, national, and international community (Asp, 1981). Several research projects have focused on fine-grained filtering of news articles. (Bende, 1994; Bilisus & Pazzani, 1999; Chesnais, Mucklo & Sheena, 1995; Kamba, Bharat, & Albers, 1995; McGillivray, 1995; Ohkubo, Kobayashi & Nakagawa, 1993; Shepherd, et al., 2001). Results from these studies indicate that fine-grained filtering of news items is very difficult and suggest that personal profiles need to be offset by community interests for ludic news reading behavior.

This research does attempt to do fine-grained filtering in the sense that it uses the profile, modeled as a hierarchy of the user's interests, to re-rank a set of news articles. This brings articles "of interest" closer to the front of the queue without necessarily eliminating articles that may be, serendipitously, of interest to the user.

Knowledge acquisition and modeling

The process of acquiring the relevant information from the user in the context of a given task or otherwise is fundamental for any user model to be successful. The knowledge acquisition can be implicit or explicit, long term or short term based, depending on the task domain. The acquired knowledge should be incorporated as needed into the existing user model without causing conflicts and contradictions. In order to achieve this, usually default reasoning and evidential reasoning are used.

The key questions in the current research are whether or not the system learns a profile of the user and whether or not the system can adapt to "drifting" user interests. These questions have been approached from a number of different directions including representing topics as tf.idf vectors (Chen & Sycara, 1998), Bayesian classifiers (Pazzani & Billsus, 1997), neural nets (Tan & Theo, 1998) and combining neural nets with stereotypes (Shepherd, Watters & Marath, 2002).

Although all of these systems perform well on static profiles, the problem with reading news is that people's interests change or drift over time, sometimes very quickly. One promising approach to this problem is the modelling of a user's short-term interests and long-term interests as is done in the News Dude system (Billsus & Pazzani, 1999) and in Alipes system (Widyantoro, Ioerger & Yen, 1999). An interesting approach to this problem is found in the concept of gradual forgetting (Koychev & Schwab, 2000) in which the last observations should be more important for the learning algorithm than the older observations and the importance of the older observations should decrease over time.

There are many different systems and different approaches to these problems and very many open questions around acquisition, modelling, modelling and recognizing drift and how all of these pertain to the filtering of news. The current research is one approach to these questions.

User profile

The user profile in this research is modeled as a hierarchy of the user's interests, closely following the methodology of Kim and Chan (2003). There are three main differences in this research. Kim and Chan (2003) used implicit feedback based on web logs indicating what web pages each user read whereas in our research the user is presented with a set of news articles and is required to indicate if a news article is "of interest" or "not of interest". The second difference is that Kim and Chan (2003) built the hierarchy of interest once and examined the resulting hierarchy in terms of the shape of the tree generated and useful clusters of terms formed, whereas in our research we apply the interest hierarchy to the task of re-ranking of news articles and perform our evaluation on the results of the retrieval experiments. The third difference is that Kim and Chan (2003) built the interest hierarchy once only, whereas in this research an adaptive learning algorithm is applied to update the the leaf categories of the hierarchy after each user session, based on explicit feedback. The results in our research were evaluated by the users, as explained below in section 4.

Structure of profile

The user profile consists of a hierarchy of interests developed from the explicit user feedback as to which news articles were of interest. Thus, each user profile consists of a hierarchical set of categories (Figure 1), where each leaf category consist of a set of pairs, each pair consisting of a keyword and an associated weight. Only the leaf categories of the hierarchy are actually used for learning and for filtering.

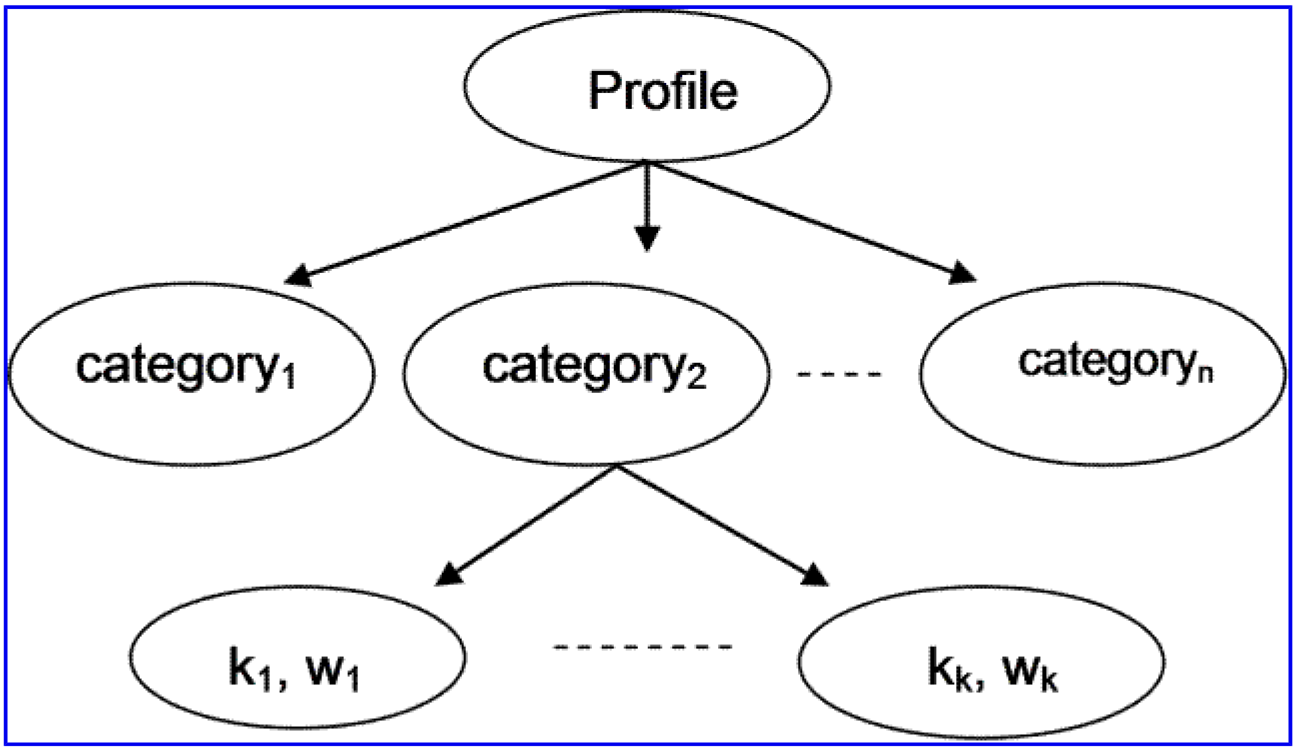

Figure 1. Hierarchy of user's interests

Generating the interest hierarchy

In our research, the user was presented with 100 news articles selected randomly from the Reuters 21578 dataset and was asked to indicate for each article if it was “of interest” or “not of interest” (Figure 4). The users scanned the headlines presented and if an article looked to be of interest a simple click on the headline brought the full text of the article to the screen for reading. By default, all articles were initialized to be not of interest and the user was required to explicitly indicate if the article actually was of interest. This feedback, for all 100 articles, was given to the learning algorithm including, the news article, the feedback, and the learning rate for that article.

After the removal of stop words and stemming, the words were extracted from each news article and a non-directed graph of bigrams was created. A bigram consists of any two words that occur in the same news article. Thus, any word may be part of multiple bigrams representing its co-occurrence with other words in the same articles. Edges between bigrams are weighted with the enhanced Augmented Expected Mutual Information (AEMI-SP) measure (Chan, 1999). The AEMI measure itself is defined as:

$$AEMI(A,B) = P(a,b)\log\frac{P(a,b)}{P(a)P(b)} - \sum_{(A=a,B=\bar{b})(A=\bar{a},B=b)} P(A,B)\log\frac{P(A,B)}{P(A)P(B)} \quad (1)$$

where *A* and *B* are the events for two words and P(A=*a*) is the probability of a document containing a and P(A= $\bar{a}$) is the probability of a document not containing the word *a*. Probabilities for *B* and joint probabilities are defined similarly. The first term computes the supporting evidence that *a* and *b* are related and the second term calculates the counter evidence. This expresses the strength of the relationship between the two words of the bigram.

The enhanced measure, AEMI-SP, is explained fully in (Chan, 1999). It is the AEMI measure enhanced with a specificity function (SP) that is based on the probability of word occurrence in documents. It is similar to the inverse document frequency function in that the more documents in which a term occurs, the lower its value. It is approximated with a sigmoid function where:

$$SP = 1/(1+e^{(0.6\times(m\times10.5-5))}) \quad (2)$$

and m is defined as MAX((P(a), P(b)), and:

$$AEMI - SP = AEMI \times \frac{SP}{2} \quad (3)$$

Once the graph has been generated (Figure 2), edges with lower weights are removed to effectively partition the graph into sub-graphs (Figure 3). This requires determining a threshold AEMI-SP weight below which edges are removed. This research used the maximum child threshold as defined by Kim and Chan (2003). In this method, the highest

AEMI-SP and the AEMI-SP value for the bigram that occurs in only one news article determine the threshold range to be examined. This range is quantized into ten equal regions and the threshold is then determined as the AEMI-SP value which if removed would generate the largest number of children or sub-graphs. This ensures a wider and flatter tree than some other threshold selections might generate. This partitioning is continued in a recursive fashion until either the sub-graph cannot be partitioned further or until the size of the sub-graph is less than some minimum number of elements, set at five in this research.

The result from this exercise is a hierarchy of interests as represented by clusters of terms extracted from the news articles examined by the user. The clusters are based on the AEMI-SP values between words of bigrams.

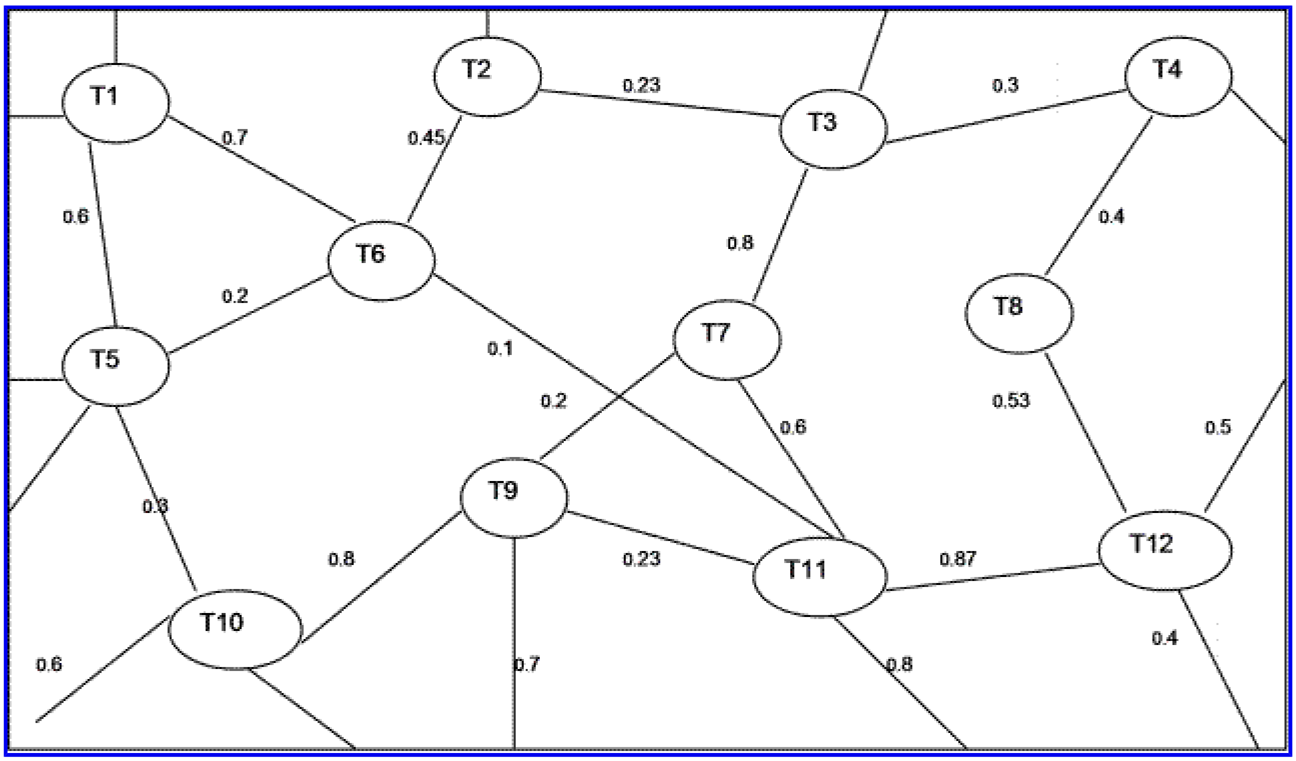

Figure 2. Example bigram graph.

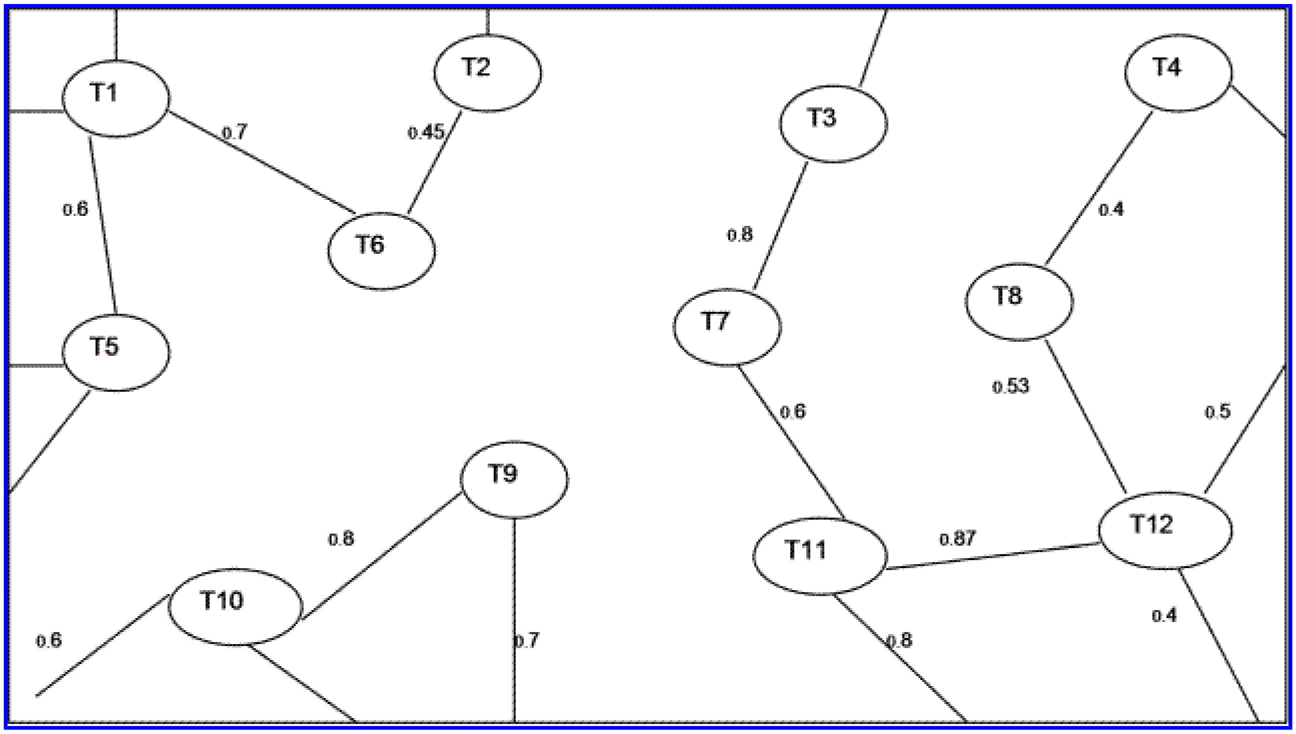

Figure 3. Removal of edges with weight < 0.35

Initial weighting of terms in the hierarchy

Each news article is represented by a vector of terms and associated tf.idf weights and learning rate determined by the user feedback ("of interest" or "not of interest"). An initial set of 100 news articles was reviewed by the user and for each article feedback was given. This information is used by the system to initialize the weights associated with each term in the interest hierarchy.

The initial weight associated with each profile term is 0 and this is adapted iteratively. For each article, *i*, in which term *k* occurs, the weight in the profile associated with *k*, is modified as follows:

$$w_k^p = w_k^p + a_i w_k^i \quad (4)$$

where a_i is the learning rate associated with article *k* and w_k^p is the weight of term *k* in the profile and w_k^i is the weight of term *k* in the term vector representing news article *i*. The learning rate associated with article *k* is determined by the explicit feedback from the user. If the user indicated that the article was "of interest", the a_i associated with that article is +0.9, If the article was "not of interest", the a_i associated with that article is -0.9. The learning rate, a_i, was arbitrarily set to +0.9 and to -0.9 with the intention of fine-tuning the learning rate in further research.

As can be seen from Figure 4, all news articles are presented to the user and initialized as "not of interest". The user scans the list of titles and by clicking on a title that may be of interest, the user can view the entire article. If the article is found to be of interest to the user, the user indicates this by selecting the appropriate radio button beside the title.

The initial profile is generated and all subsequent updates are generated in batch mode, i.e., after the user has finished reading and evaluating a set of news articles, that set is submitted to the profile build/update system.

Filtering news articles using the profile

Once the weights of the terms in the profile have been initialized, the profile can be used for filtering news articles for that user.

Each category in the profile consists of a set of terms with associated weights and thus can be represented as a vector of weighted terms. When new articles are presented to the profile, the cosine similarity is calculated between that article and every category in the profile. The average of these similarity measures is then taken to be the closeness of that document to the user's profile. This averaging is done as profile categories are not developed from individual articles. Rather, they are developed from categories of user interests developed from the bigram graph. As the terms from an article may occur in several different categories, the articles themselves are not associated with a particular category, but are distributed over multiple categories in the profile.

Note that because term weights in the profile may be positive or negative, the similarity values calculated may be positive or negative and the range of the similarity score calculated for each document is [-1, +1].

Updating the profile

The system permits the profile to be adapted based on user feedback. This adaptation or learning is based on user feedback for new sets of news articles.

After the user has evaluated a new set of articles, where each article is either "of interest" or "not of interest", the hierarchy of the user's interests for this new set of articles is generated and leaf categories are merged in with the leaf categories of the existing profile as follows:

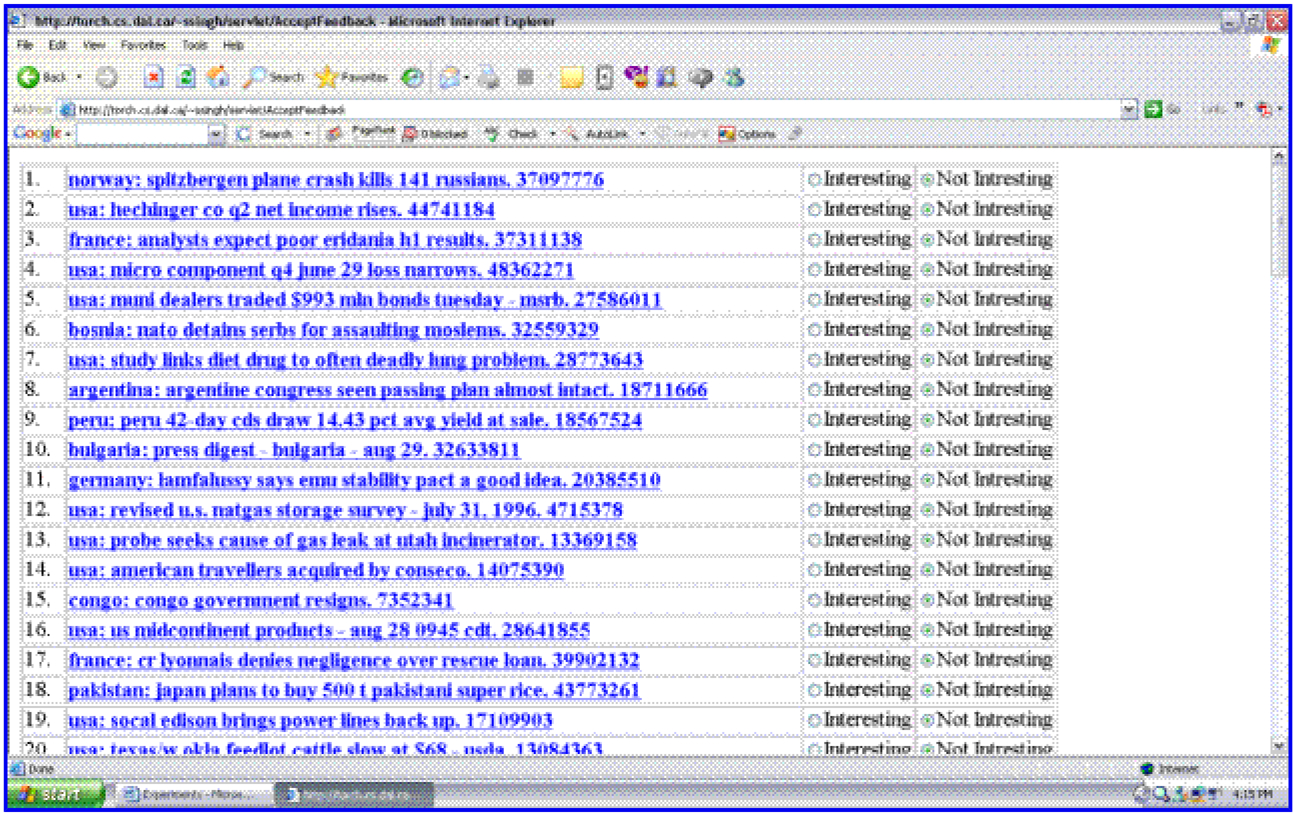

Figure 4. Viewing and evaluating new articles.

For each leaf category in the new hierarchy
Calculate cosine similarity with each leaf category in existing profile
Find profile leaf category with max similarity
If max similarity > Threshold
Merge new leaf category with profile leaf category
Else
Create new leaf category in profile with leaf category from new hierarchy
Endif
Endfor

In this algorithm a new leaf category is merged with an existing profile leaf category. Each term in the new category is associated with one or more news articles, each of which has a learning rate associated with it based on user feedback. The terms from the new category, as represented by their tf.idf weights in the articles in which they appear, and the learning rates associated with those articles are used to update the term weights in the profile category, as per equation 4, above. If a term in the new category does not exist in the profile category, the term is added to the profile category with an initial weight of 0 and then that weight is updated as above.

Evaluation and results

The system was initially trained on profiles that were static, in order to determine that the system did indeed learn and continue to learn with more feedback. The system was then evaluated on profiles that were slightly modified from the original static profiles to evaluate the effect of introducing drift into the user's interests.

Learning static profiles

The data set from which the tests sets were selected was the Reuters 21578 collection of news articles. For each user, five sets of 100 news articles were selected as test sets and one set of 100 articles was selected as a training set. All sets were selected randomly without replacement.

Three researchers associated with the project were asked to develop stable profiles that they might use in "reading the news", i.e., the task is ludic in nature. The researchers were full professors, two in Computer Science and one in the School of Business Administration. One of the computer scientists is female while the other two researchers are male. The profiles were quite different, as shown below:

- User_1:
    - All news items except those dealing with sports and/or finance
- User_2
    - United States congressional politics
    - Tragedies
    - All news about New Zealand, India, Canada and Israel
- User_3
    - Sports, agriculture and all news about Japan and/or Norway

Sets of 100 news articles were randomly selected for each user from the Reuters dataset, without replacement, for the creation of the initial profile and for each subsequent viewing and evaluation.

The three users were asked to evaluate all the news articles in the training set and in each of the five test sets as to whether or not the article was "of interest" or "not of interest", relative to their established profiles. These judgments became the learning rates as described above.

The Normalized Recall measure (R_{Norm}) (Salton, 1968) was calculated for each set of randomly selected articles. This measure assumes that the perfect retrieval system would rank all *n* relevant items in the first *n* positions and the R_{Norm} measure calculates how close the actual system comes to achieving this, with a score between 0 and 1 with a score of 1 being perfect.

Once the “of interest” and “not of interest” news articles were identified for the training set, the initial profile for each user was created. After the initial profiles were created, the following steps were followed for each user for test sets 1 through 5, in order to continue the training of the profile:

For i = 1 to 5
Use profile to re-rank the results of tests sets i through 5
Calculate Normalized Recall for sets i through 5
Update profile using user evaluations of random ordered test set i
Endfor

The results for the three users are listed in Tables 1 through 3. Column 1 of each table shows the number of relevant news articles identified for each test set. The second column indicates the R_{Norm} for each of the test sets before any re-ranking via the profile is done. These sets were selected randomly and, as can be seen from this column, the R_{Norm} is approximately 0.5, as would be expected.

Columns 3 gives the R_{Norm} for all five test sets after the initial profile created from the training set was used to re-rank the five test sets.. Columns 4 through 7 give the R_{Norm} for the test sets as the profile is trained on and applied to the five test sets, incrementally.

The immediate result of the re-ranking is illustrated by Table 4, which contains data for set 5 for User_1. This table shows the R_{Norm} for set 5 before any re-ranking has been done and the rank order of those documents found to be “of interest”. It also presents this information for this set after re-ranking by the profile after initialization of the profile with the training set, and after re-ranking by the profile after the profile has been initialized and continued to be trained with the feedback from test sets 1 through 4.

The results of the evaluation for the static profiles indicate the following:

- Using the gain in normalized recall, there is a significant difference among users ($F = 6.22$, $p = 0.023$, parial $eta^2 = 0.609$)
- Using the gain in normalized recall, there is no significant difference among the test sets ($F = 1.16$, $p = 0.396$, $eta = 0.367$)
- Across all users, the algorithm significantly improves normalized recall ($F = 12.3$, $p < 0.001$, $eta^2 = 0.734$)

- The system stopped learning after the training set and test sets 1 though 3. This implies that the system is stable after this stage, at least for the next two iterations. There was no significant difference:
    - Between this stage and including learning sets 4 and 5 ($t = 0.69$, $df = 43$)
    - Between this stage and including learning test set 4 ($t = 0.5$, $df = 28$)
    - Between this stage and including learning test set 5 ($t = 0.78$, $df = 28$)
    - Between the stage including the training set and test sets 1 through 4 and including learning test set 5 ($t = 0.35$, $df = 28$)
- There is a significant difference between users, indicating that the system learns better for some users ($F = 11.67$, $p < 0.001$, $eta^2 = 0.653$). In this instance, the system learned better for User_1 than for User_3, and better for User_3 than for User_2.

Essentially, the system learned the profiles and improved the results. The learning based on new feedback across the test sets is the single best predictor of the R_{Norm}.

Table 1. Normalized recall for User_1

	Number Relevant	Random R_{Norm}	After training set	Training + set 1	Training + sets 1-2	Training + sets 1-3	Training + sets 1-4
Set 1	21	0.556	0.796				
Set 2	19	0.559	0.799	0.862			
Set 3	22	0.560	0.760	0.788			
Set 4	17	0.500	0.694	0.756	0.795		
Set 5	15	0.438	0.758	0.841	0.864	0.865	0.875

Table 2. Normalized recall for User_2

	Number Relevant	Random R_{Norm}	After training set	Training + set 1	Training + sets 1-2	Training + sets 1-3	Training + sets 1-4
Set 1	13	0.538	0.532				

Set 2	15	0.536	0.470	0.444			
Set 3	12	0.598	0.782	0.724	0.670		
Set 4	16	0.593	0.598	0.522	0.537	0.550	
Set 5	25	0.400	0.698	0.717	0.731	0.734	0.741

Table 3. Normalized recall for User_3

	Number Relevant	Random RNorm	After training set	Training + set 1	Training + sets 1-2	Training + sets 1-3	Training + sets 1-4
Set 1	13	0.490	0.812				
Set 2	13	0.368	0.777	0.813			
Set 3	14	0.703	0.797	0.828	0.826		
Set 4	17	0.446	0.674	0.670	0.685	0.676	
Set 5	13	0.548	0.851	0.829	0.809	0.785	0.794

Table 4. Rank order of articles “of interest”, User_1, test set 5

	RNorm	Ranks of articles “of interest”
Random Selection	0.438	6, 16, 25, 26, 44, 48, 56, 57, 63, 65, 67, 81, 85, 97, 100
After the training set	0.758	6, 7, 11, 12, 15, 16, 17, 18, 25, 27, 28, 43, 52, 75, 77
After the training set + test sets 1-4	0.875	1, 4, 5, 6, 7, 8, 12, 15, 16, 19, 22, 23, 25, 50, 67

Adapting to user interest drift

In order to determine how the system adapted to changing profiles, the three users slightly modified their profiles and evaluated another five sets of 100 randomly selected news articles, without replacement. In this instance, the initial profile was the profile at the end of the training from the first phase, i.e., the profile has been trained on the training set and all five of the original test sets. The modified profiles were as follows:

- User_1
  - All news except those items dealing with the United States
  - No tragedies
  - No "news digest" articles
  - No financial news
  - All sports articles, no matter what country
- User_2
  - No United States articles
  - Tragedies
  - All news about New Zealand, India, Canada, Israel, Australia and Pakistan
- User_3
  - Sports
  - Agriculture
  - Asia
  - Norway
  - tragedies

The results for the three users in this trial are listed in Tables 5, 6 and 7. Column 1 of each table shows the number of relevant news articles identified for each of the five new test sets (sets 6-10). The second column indicates the R_{Norm} for each of these test sets before any re-ranking via the profile is done. Again, these sets were selected randomly and, as can be seen from this column, the R_{Norm} is approximately 0.5, as would be expected.

Columns 3 gives the R_{Norm} for all five test sets after re-ranking based on the profile resulting from the previous training using the training set and the original five test sets.. Columns 4 through 7 give the R_{Norm} for the test sets as the profile is trained on and applied to the five new test sets, incrementally.

Figure 5 is a summary for all three users. On the x-axis, trial 1 represents the R_{Norm} for the random (unranked) order for set 5. Trial 2 shows the R_{Norm} for set 5 after the re-ranking of the news articles as per the profile after the its creation from the training set. Trials 3 through 6 show the R_{Norm} for set 5 after the re-ranking of the news articles after each stage of the updating of the profile using sets 1-4.

Trial 7, the large dip in the graph, shows the R_{Norm} for the random (unranked) order for set 10 (This is a new data set). Trial 8 shows the R_{Norm} for set 10 after the re-ranking of the news articles as per the profile after trial 6. Trials 9 through 12 show the R_{Norm} for set

10 after the re-ranking of the news articles after each stage of the updating of the profile using sets 6-9.

The results of the evaluation of the system when the trained profile is presented with new interests are as follows:

- Using the gain in normalized recall, there is a significant difference among users ($F = 17.03$, $p = 0.003$, partial $eta^2 = 0.85$)
- Using the gain in normalized recall, there is no significant difference among tests sets ($F = 0.746$, $p = 0.563$, $eta = 0.272$)
- The system improves its performance for all users across all trails ($F = 7.51$, $p < 0.013$, $eta^2 = 0.882$)
- The system does significantly better for User_1 than for User_2 over all trials ($F = 39.02$, $p < 0.001$, $eta^2 = 0.819$)
- The system does significantly better for User_3 than for User_2 over all trails and User_1 does better than User_3.

Note that, after an initial drop, User_1 needed only one test set in this second set of experiments to bring the R_{Norm} back up to the level at the end of the first set of experiments. In other words, the system responded very quickly to the slight drift in User_1's interests. This may be due to the fact that the revised profile does not, on the surface, appear to be significantly different from the original profile for User_1 and User_1 identified about the same number of articles to be interest in both stages.

This was not the case for User_2. Although the system continued to learn across all trials for User_2, the level of the R_{Norm} never recovered to the levels at the end of the first set of trials. It should be noted that User_2 indicated far fewer articles to be "of interest" in this second stage than for the first stage. This would mean that the profile was still more heavily weighted towards the first profile, even at the end of the second set of trials.

User_3 on the other hand, had many more articles identified as being "of interest" in the second stage than in the first stage. This may explain why, for User_3, the system performed better than for User_2 but not as good as for User_1.

In this second set of trials, User_1 and User_3 found many more articles to be "of interest" than did User_2, and we suspect that this may have lead to the superior performance of the system for User_1 and User_3 relative to User_2.

Table 5. Normalized recall, User_1, modified profile

	Number	Random	Trained	Trained	Trained	Trained	Trained

	Relevant	R_{Norm}	profile	profile + set 6	profile + sets 6-7	profile + sets 6-8	profile + sets 6-9
Set 6	25	0.549	0.837				
Set 7	27	0.497	0.872	0.880			
Set 8	26	0.486	0.797	0.816	0.83		
Set 9	23	0.560	0.931	0.935	0.933	0.933	
Set 10	24	0.497	0.857	0.863	0.876	0.882	0.881

Table 6. Normalized recall, User_2, modified profile

	Number Relevant	Random R_{Norm}	Trained profile	Trained profile + set 6	Trained profile + sets 6-7	Trained profile + sets 6-8	Trained profile + sets 6-9
Set 6	7	0.533	0.528				
Set 7	12	0.481	0.540	0.541			
Set 8	10	0.400	0.486	0.510	0.510		
Set 9	8	0.530	0.668	0.670	0.672	0.664	
Set 10	15	0.430	0.576	0.589	0.593	0.590	0.593

Table 7. Normalized recall, User_3, modified profile

	Number Relevant	Random R_{Norm}	Trained profile	Trained profile + set 6	Trained profile + sets 6-7	Trained profile + sets 6-8	Trained profile + sets 6-9
Set 6	22	0.521	0.742				
Set 7	27	0.597	0.668	0.671			

Set 8	25	0.459	0.734	0.735	0.757		
Set 9	26	0.573	0.723	0.724	0.739	0.742	
Set 10	23	0.561	0.652	0.656	0.659	0.669	0.674

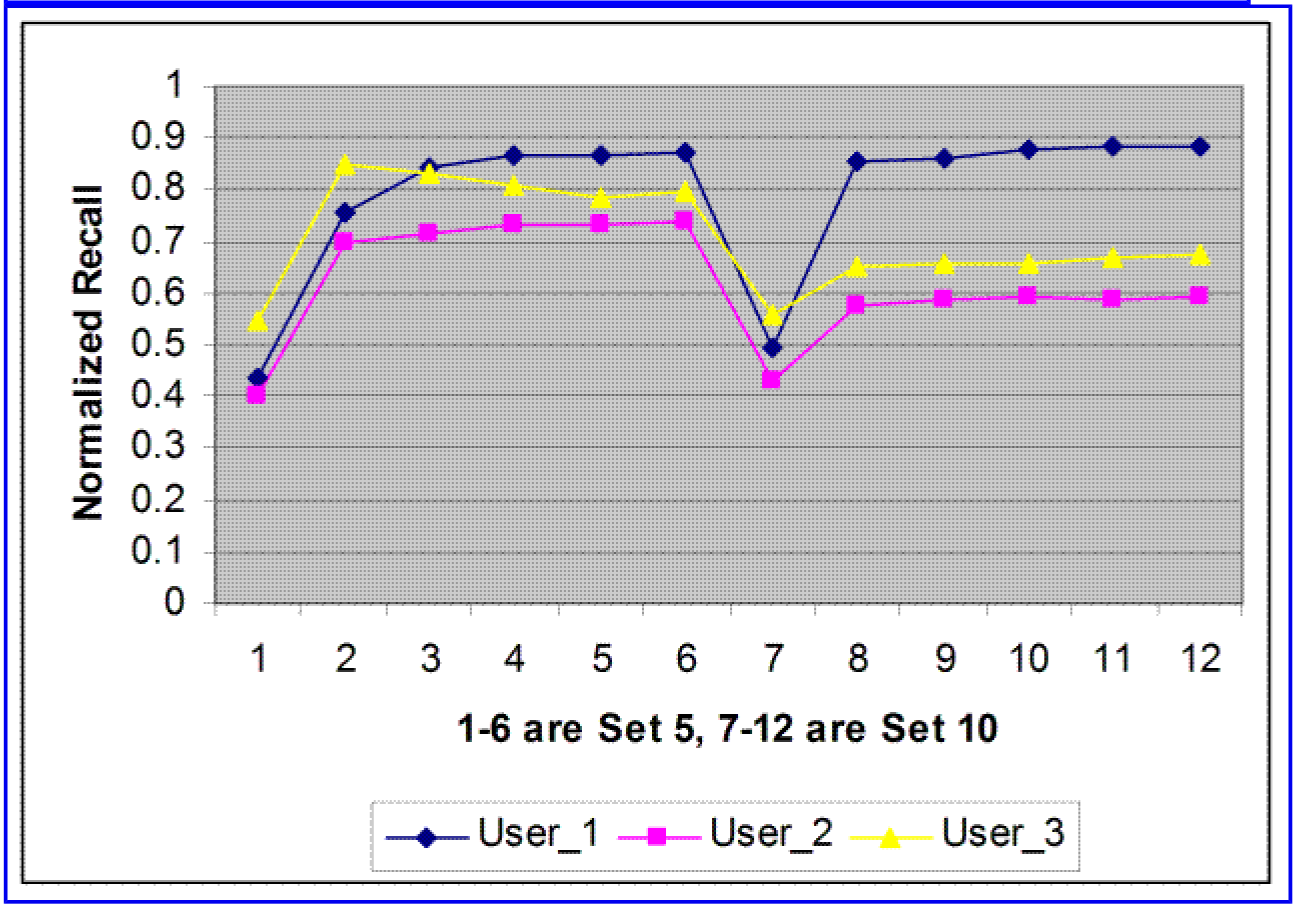

Figure 5. Summary for all users over all trials

Conclusions and future research

Overall, it can be concluded that the user interest hierarchy appears to be a good model on which to base user profiles for re-ranking of news articles. Presumably, this can be broadened out to other types of text documents. The system learned quickly and stabilized after a short number of iterations on static profiles and, given enough feedback (number of articles "of interest"), it did adjust to changes in the user's interests.

There are still a number of open research questions to be investigated. These include developing a finer-grained learning rate. The current learning rate is based on a binary decision, either the article is "of interest" or it is "not of interest" and the learning rate for each term in the article is either +0.9 or -0.9, respectively (Equation 4). The learning rate could be tied to a Likert scale feedback from the user, with Likert-based learning rates in the range of +0.9 to -0.9. A finer-grained learning rate may change the profile more slowly over time.

As can be seen from Figure 5, the results do not recover completely for Users 2 and 3. This may be due to the fact that User_2 gave limited feedback in stage 2, while User_3 gave limited feedback in building the profile in the first stage.

Although the number of users was small, the results are significant based on the number of data points for each user. Three single participant studies were done and the data analyzed for each participant. The system's performance between each user can be compared reliably as there were a thousand data points for each user. While there were only three users, there were 3,000 data points and thus the study has very high power. However, a user study over an extended period of time with a larger and more varied user group may provide better and more interesting statistics, particularly with regards to the effects of the learning rates and the amount of feedback given and how this affects the results.

Of particular interest would be the use of a user hierarchy for collaborative filtering. This may help keep the individual user more connected to other communities, but there may be problems with the hierarchy being skewed to the interests of a user or small group of users who tend to provide more feedback than other users.

References

Allen, R.B. (1990) User models: theory, method, and practice *Int. J. of Man-Machine Studies* 32, 511-543

Asp, K. (1981) Mass media as molders of opinion and suppliers of information In C. Wilhoit & C. Whitney (Eds) *Mass Communciation Review Yearbook* 2, 332-354. Beverly Hills: Sage Publications

Bende, W. (1994) Read all about it in the Daily You *Communicating Business* Winter, 1994/1995. London: Forward Publishing

Billsus, D. & Pazzani,M.J. (1999) A hybrid user model for news classification In Kay J. (Ed) *UM99 User Modeling Proceedings of the Seventh International Conference* 99-108. Wein, New York: Springer-Verlag

Chan, P.K. (1999) A non-invasive learning approach to building web user profiles *Kdd-99 Workshop on Web Usage Analysis and User Profiling* 7-12

Chen, L. & Sycara, K. (1998) Personal agent for browsing and searching *2nd International Conference on Autonomous Agents* 132-139

Chesnais, P.R., Mucklo, M.J. & Sheena, J.A. (1995) The Fishwrap personalized news system. MIT Media Laboratory *Proceedings of the 1995 IEEE Second International Workshop on Community Networking Integrating Multimedia*

Deci, E.L. & Ryan, R.M. (1987) The support of autonomy and the control of behavior *Journal of Personality and Social Psychology* 53, 1024-1037

Dozier D. & Rice R. (1984) Rival theories of electronic newsreading In R. Rice (Ed), *The New Media* 103-128. London: Sage Publications

Kamba, T., Bharat, K. & Albers, M.C. (1995) The Krakatoa Chronicle - An interactive, personalized, newspaper on the Web *Proceedings of the Fourth International World Wide Web Conference* http://www.w3.org.pub/Conferences/WWW4/Papers/93/

Kim, H.R. & Chan, P.K. (2003) Learning Implicit User Interest Hierarchy for Context in Personalization *IUI'03* Miami, Florida, 12-15 January

Koychev, I. & Schwab, I. (2000) Adaptation to drifting user's interests *ECML2000 Workshop: Machine Learning in New Information Age* Barcelona

McGillivray, K.(1995) *Personalizing Electronic News through User Modelling* Honours Thesis. Dept of Mathematics, Statistics, and Computer Science, Dalhousie University, Halifax, Canada

Ohkubo, M., Kobayashi, N. & Nakagawa, T. (1993) Design of an information skimming space *Proceedings of ACM Multimedia'93* 365-371

Pazzani, M. & Billsus, D. (1997) Learning and revising user profiles: The identification of interesting web sites *Machine Learning* 27, 313-331

Salton, G. (1968) *Automatic Information Organization and Retrieval* New York: McGraw-Hill

Shepherd, M., et al. (2001) The role of user profiles in news filtering *Journal of the American Society for Information Science and Technology* 52(2), 149-160

Shepherd, M., Watters, C. & Marath, A. (2002) Adaptive user modeling for filtering electronic news *Proc. of the 35th Hawaii International Conference on System Sciences* Hawaii. 2002

Stephenson, W. (1967) *The Play Theory of Mass Communication* Chicago: The University of Chicago Press

Tan, A. & Theo, C. (1998) learning user profiles for personalized information dissemination *Proc. of the International Joint Conference on Neural Networks* 183-199

Widyantoro, D.H., Ioerger, T.R. & Yen, J. (1999) An adaptive algorithm for learning changes in user interests *CIKM'99* 405-412, Kansas City