

How Up-to-date should it be? the Value of Instant Profiling and Adaptation in Information Filtering

Daqing He, Peter Brusiloviksy, Jonathan Grady, Qi Li, and Jae-wook Ahn
School of Information Sciences, University of Pittsburgh
{dah44, peterb, jpg14, qil14, jaa38}@pitt.edu

Abstract

In profile-based or content-based adaptive systems, one of the open research questions is how frequently the user's profile and the list of recommended items should be updated. Different systems tend to choose one of the two extremes. Some systems do it once per session (thus called between-session update strategy), whereas some others update whenever there is feedback (called instant update strategy). This paper presents our attempt to assess the value of keeping the list of recommended items up-to-date in the context of task-based information exploration. We conducted controlled studies involving human users performing realistic tasks using two systems that have the same adaptive filtering engine but with the above two different update strategies. Our results show that the between-session strategy helped to find better quality information, and received better subjects' responses about its usefulness and usability. However, it prolonged the selection of useful passages, whereas the instant update strategy helped subjects to obtain almost all of their selected passages (> 98%) within the first 5 minutes. Based on the results, we hypothesize that the best strategy for updating might be a hybrid between the two update strategies, where both adaptability and stability can be achieved.

1. Introduction

With a vast amount of information available electronically on the Internet, people utilize Web search engines for interpretation, synthesis, or evaluation, which are the activities often involved in learning and investigation of certain topics [1]. These searches are called exploratory searches [2], or information exploration. Task-based information exploration is a specific type of exploratory search where the complexity of the information needs and the corresponding search processes are heavily influenced

by the overall task performed by the user [3]. Task-based information exploration is typical for a range of professional users (e.g., intelligence analysts,) and developing systems to support these professional users is an emerging direction of research. We can distinguish two groups of investigations conducted by the research community. The first direction applies Human-Computer Interaction (HCI) techniques to enhancing human abilities with a better interface for information exploration [1]. At the moment, research in this direction focuses mostly on developing visualization systems that help analysts examine the information space [4-7]. Another important direction is to use Artificial Intelligence techniques to build smarter machines that can assist users in collecting useful information. So far, this direction is represented by research work on adaptive systems that utilize user modeling and personalization techniques [8, 9].

We are interested in adaptive information filtering to assist information analysts in their exploration. Our research focuses on profile-based or content-based filtering [10]. The idea of this technology is to observe user actions (such as link following, bookmarking, or ratings) in order to derive a profile of user interests from these actions, and actively recommend relevant information items. Content-based adaptive information filtering is a relatively well-explored area with dozens of research and practical systems reported in the literature (see [11] for a review), yet there are still a number of under-explored research issues.

One of these issues is how frequently the user profile and the list of recommended items should be updated. Different systems tend to choose one of the two extremes. Some systems update the user profile once per session at the end of each session [12, 13] and start the new session with an updated list of recommended items. It is called the between-session update strategy in this paper. Other systems attempt to update the user profile after each newly collected piece of evidence about user interests, keeping the list of

recommended items up-to-date [14, 15]. Here, this is called the instant update strategy. Each extreme has its advantages. User model updates can be relatively slow, especially when advanced machine learning techniques are used [11]. Performing updates once per session allows the system designers to avoid dealing with performance problems. In addition, the system has data about the users' interests and tasks collected from the whole session during its update that, in theory, would lead to more accurate profiling and adaptation. On the other hand, the developers of the systems that adopt the instant update strategy were motivated by the hope that an up-to-date list of recommended items is able to address the user interests better.

But is it really beneficial for the user to obtain constant updates? Are the additional implementation efforts for frequent updates justified by improved system performance? Do users indeed find usefulness in these updates? Do users like this kind of frequent adaptation? These questions are important for our work on adaptive filtering, but we failed to find any answers in the research literature.

This paper presents our attempt to assess the value of the two profile update strategies in the context of task-based information exploration. The research questions that we want to examine are:

- How frequently should an adaptive system update the user profile and its recommended items for better performance?
- Which performance factors can be affected by the frequency of the updates?
- Which measures can be used to examine the interactions and the update frequency?

In order to find answers to these questions, we conducted a controlled study involving human users performing realistic tasks. By having subjects interact with two systems that utilize the same adaptive filtering engine but two different update strategies, we attempted to isolate the effect of update frequency. Through the study and the analysis of its results, we hope to find the answers to our research questions.

In the remainder of the paper, we will first talk about the two strategies and the two systems that implement each of them in Section 2. Then we talk about the study design in Section 3. Finally, we present our result analysis and discussion in Sections 4 and 5, and conclude with future work in Section 6.

2. Two Strategies for Profile and Item Update in Adaptive Filtering

Our study focused on examining the effects of two update strategies for adaptive filtering in the context of

task-based information exploration. The study used the same adaptive filtering engine, CAFÉ (see Section 2.1), as the underlying platform, so we can concentrate on the update strategies. The first strategy we studied is a between-session adaptation method, where no feedback is sent back to the filtering engine until a search session has been completed. This gives the adaptive engine all feedback information collected from the session when it tries to update its profiles, but it also means that there is no adaptation within the session. The implementation of this strategy in our study is the SelLite system (see Section 2.2). The second strategy views the instant updating of the profile as important, so the feedback information is sent to the adaptive engine whenever users provide input. The information system using this strategy thus exhibits a very active adaptive behavior. The implementation of this strategy here is the Rosetta system (see Section 2.3).

2.1. CAFÉ Adaptive Filtering Engine

CAFÉ (CMU Adaptive Filtering Engine) was developed at Carnegie Mellon University for utility-based information distillation. It combines a state-of-the-art adaptive filtering system [16] and a top-performing novelty detection system. Its initial profile comes from the rich information in the task description, and the profile is incrementally updated (“adapted”) when user feedback is received. The feedback indicates both relevance and redundancy of the currently processed passages. The user may add new queries or modify the existing queries as a part of the feedback. The adapted profile is used to re-rank passages with the passages already seen removed from the re-ranked list.

CAFÉ uses a regularized regression algorithm for the training of task profiles. It estimates the posterior probability of a task given a passage using a sigmoid function. The selection of a text passage is considered positive feedback by CAFÉ and the removal of an irrelevant passage is negative feedback.

2.2. SelLite: an Implementation of the Between-Session Strategy

The SelLite system is built on top of the CAFÉ engine, employing the between-session strategy for updating CAFÉ profiles. SelLite supports task-based information exploration of intelligence analysts. The exploration procedure is modeled as follows: after given an RFI, an analyst starts to explore large volumes of data from various sources with the help of

CAFÉ, and generates a short summary of information collected, which is typically a two-page point report. Because a task has multiple aspects and is usually related to a seminal event that evolves over time, the exploration is usually complex and multifaceted. In addition, information is collected in the form of text snippets (called passages) rather than whole documents. The final product, the point paper, prompts utility of the information where not only the relevance and novelty are considered, but also the cost of adding certain information into the point paper.

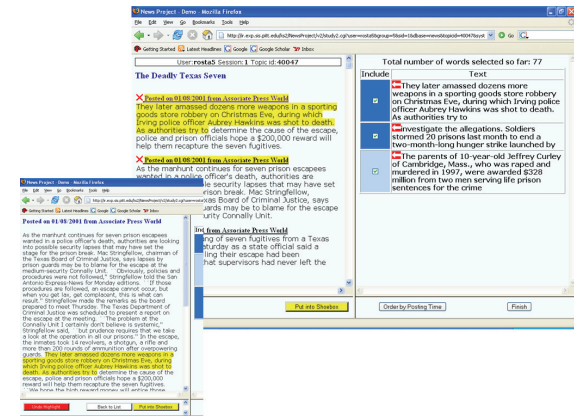


Figure 1: Information foraging in SelLite system: assembling text fragments in the shoebox

The SelLite system excludes an ad-hoc search component and provides the users with a list of passages generated by CAFÉ as the only way to access the information. The system simulates the analyst's activities by providing two separate interfaces: one for collecting or foraging potentially useful passages, and the other for organizing and cleaning selected passages into a final set that resembles the final point paper.

The foraging interface (see Figure 1) consists of two frames. The left frame shows the list of passages ordered by their relevance to the perceived user task. The right frame shows a container called the shoebox, which is a traditional information-processing tool used by analysts to store all useful text selected for their final reports. The users can copy any part of the passages directly to the shoebox, or open a pop-up window to view the complete document and select passages from there. The darker color of the text fragments in the shoebox indicates that they are selected from the passage list directly, and the lighter color represents passages selected from the full text window. Selected passages can be ordered by the documents' posting dates or by the sequence of user's selection. Passages can be removed from the shoebox.

The sense-making interface (see Figure 2) helps the user to organize and clean the selected passages in the shoebox. It provides a word count to help users track the total number of words in the selected passages. There is a 2,000-word limit on the shoebox to resemble the typical word limit of a point paper. Here, to remove the effect of writing reports, the final selected passages serve as the final product of the exploration.

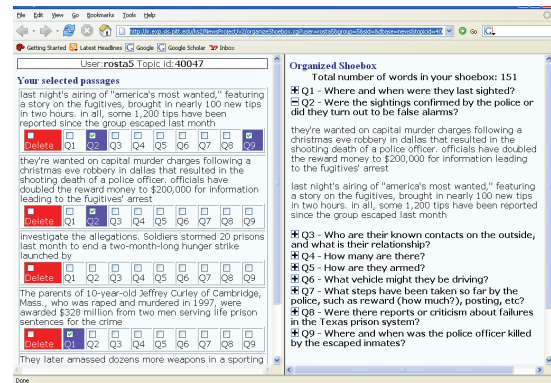


Figure 2: Organizing shoebox in the SelLite system

2.3. Rosetta: an Implementation of the Instant Update Strategy

The Rosetta System has a panel that uses CAFÉ as the underlying adaptive filtering engine. Unlike SelLite, CAFÉ in Rosetta is fully interactive. Users can invoke CAFÉ by issuing a request with important keywords, triggering CAFÉ to return a ranked list of potentially useful passages (see Figure 3).



Figure 3: CAFÉ engine in Rosetta System

Similar to SelLite, Rosetta aims to support intelligence analysts in their task-based information exploration. In Rosetta, users can highlight parts of the passages, select passages, or view the full content of the documents behind passages. However, unlike

Sellite, users' actions are immediately sent to CAFÉ for profile updating. CAFÉ immediately update its profiles of the user, and a new set of results - hopefully better suited to the user's needs - is generated and viewable by clicking on the "Update" button.

Whenever the user selects a passage, the passage is added to a shoebox-like storage device in Rosetta (called "notes.") Users can organize the notes just like the sense-making interface of Sellite. However, the notes device does not have a word count indicator.

3. The Study Design

3.1. TBIE Evaluation Framework

Our study design follows the methodology developed as a part of the TBIE evaluation framework. The framework is constructed for examining systems in task-based information exploration [3]. It shares ideas with human-centered system design in the literature [18, 19], where supporting human users in their tasks is the focus and the criterion for examining the usefulness of the systems. TBIE evaluation framework utilizes task scenarios that simulate the actual tasks of analysts. Under the overall umbrella of task-based information exploration, users' exploration behaviors can be categorized as first information foraging then sense-making for collecting useful information for a complex and evolving task. The framework provides recommendations for evaluation metrics. This includes performance-oriented measures like passage precision of selected passages, and the usability measures about systems' support, especially those examining the interactions between the users and the systems. Examples include the efficiency of selecting useful info, and users' subjective comments.

Passage precision, as the name implies, is calculated at the passage level. It takes advantage of the fact that the list of all useful passages (called ground truth) was manually generated by two human annotators when the TBIE framework was constructed. The formula (1), which is derived from [20], calculates the precision of a passage against the ground truth, where *overlap_length* is the character length of the common text chunk between a user's selection and the ground truth; *weight* is the weight of the ground truth combining the two annotators mark-ups, where the weight can be one of five levels: 0, 0.25, 0.5, 1, 1.25, 2; *miss_length* is the character length of the part of the user's passage that has no overlap with the ground truth. Here 0.5 is the penalty with *miss_length*.

$$\frac{\sum_{i \in \text{passage} \cap \text{groundtruth}} \text{Overlap_length} \times \text{weight}_i}{\sum_{i \in \text{passage} \cap \text{groundtruth}} \text{Overlap_length} \times \text{weight}_i + \sum_{i \in \text{passage} - \text{passage} \cap \text{groundtruth}} \text{miss_length} \times 0.5} \quad (1)$$

The TBIE framework provides a test reference collection that contains 28,390 English documents and 18 task scenarios with two human ground truth annotations on relevance, novelty, and utility respectively. The whole document collection was used in this study, but only two task scenarios were included. Their tasks IDs are 41005, and 41024.

3.2. Subject Profiles

Fifteen subjects participated in the study. Subjects were all native English speakers and have been trained in search (i.e. a master's level course in information retrieval) so that they can best fit the profile of information analysts. All subjects were graduate students at the University of Pittsburgh. 13 subjects were female. On a ten-point scale (10 being the highest), the subjects' mean rating of their search abilities was 8.124 with a mode of 8. All subjects indicated that they used Google as a search engine for finding news, while 9 of the 15 subjects indicated they also used AltaVista, Ask.com, Excite, or Yahoo!.

3.3. Experiment Procedure

This study employed a between-subject design, where two groups of subjects (8 subjects in Sellite group, and 7 subjects in Rosetta group) were recruited to perform the same tasks, but using different interfaces. This design was a natural choice in our formative studies of CAFÉ and Rosetta that consist of a series of experiments. The study on Sellite was conducted in the summer of 2006, where the main purpose was to study the utility of CAFÉ in the context of supporting task-based information exploration. The study of Rosetta was performed in December 2006 as a further experiment on the interaction functionality of Rosetta. This arrangement makes within-subject design difficult to execute. However, because the two groups of subjects both came from the same general population (i.e., graduate students majored in information science), we think that the difference between the groups should not be significantly larger than the difference within the group.

Prior to the experiment, subjects were trained in groups of two or three on using Sellite or Rosetta. Following the training, all subjects performed their corresponding search tasks using the system on which they were just trained. In the Sellite group, subjects

completed their information exploration for each task in 3 sessions with a 2-4 day interval separating each iteration. This interval gave CAFÉ the opportunity to update its models. Each session lasted for 20 minutes. In the case of the Rosetta group, subjects only worked with Rosetta in one long session for each task, which lasted 60 minutes with two short breaks in between. In total, the Rosetta group spent the same time as the SelLite group performing each task. At the end of each task, subjects were asked to complete a questionnaire, followed by a 10-minute exit interview.

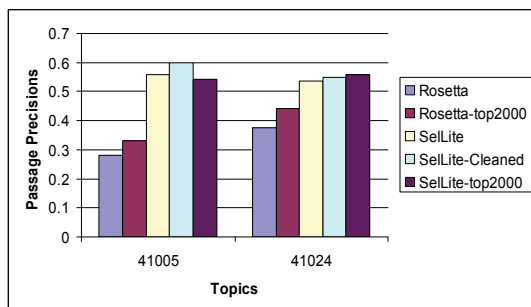


Figure 4: Precision of selected passages

4. Results Analysis

4.1. Passage Precision of Selected Passages

By comparing the precision of the subjects' passage selections using SelLite or Rosetta, we can examine how well subjects performed in their tasks with the help of the systems. Figure 4 shows that the precision of all selected passages from Rosetta (the bar labeled "Rosetta" in Figure 4) is significantly lower (independent samples t-test $p = 0.01$) than those from SelLite (labeled "SelLite").

In the experiment of SelLite, subjects were asked explicitly to reduce their total selection of passages for each task to 2000 words. Their manually-cleaned passages are labeled "SelLite-Cleaned" in Figure 4. However, subjects using Rosetta were not asked to perform that same cleaning because the system setting was different. We developed an oracle strategy to clean the passages automatically based on the quality of the passages. Each passage has a precision score by comparing it to the ground truth, allowing us to rank them by precision score in descending order. We then add passages to the cleaned set from the top down until the total size of the cleaned set is above 2000 words. We conducted this automatic selection for both systems, and the results are two sets with label suffix "-top2000". This time, "Rosetta-top2000" improved slightly over the un-cleaned version "Rosetta",

whereas there is basically no change between "SelLite-top2000" and "SelLite". The difference between "Rosetta-top2000" and "SelLite-top2000" is still significant (independent samples t-test $p = 0.01$).

We looked at the distributions of passages according to their passage precision values. Figure 5 shows that the largest group of passages in both "SelLite" and "SelLite-Cleaned" is the group with precisions 0.8 to 1.0, whereas that of "Rosetta" is the group with precisions 0 to 0.2. This clearly shows why both overall passage selections in Rosetta (i.e., "Rosetta") and its oracle selection (i.e., "Rosetta-top2000") are significantly inferior to their SelLite counterparts ("SelLite" and "SelLite-top2000").

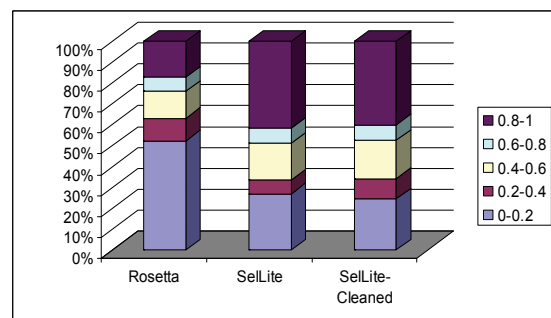


Figure 5: The distribution of passages according to their passage precision values

Interestingly, in all three results presented in Figure 5, the majority of passages are either at very high precision (0.8-1.0) or at very low precision (0-0.2). Few passages are of medium quality. Further study is required to reveal the reason for this.

Table 1: Overlap of Passages between "SelLite-Cleaned" and "SelLite-top2000"

Topics	#passages in SelLite-Cleaned	#passages in SelLite-top2000	#passages in common
41005	44	65	39
41024	56	69	56

Again in Figure 4, we can see that the precision scores between "SelLite-Cleaned" and "SelLite-top2000" are almost identical. By looking into the individual passages in these two results (see Table 1), we can see that most human-selected passages for the final product also appear in our oracle selection of passages. It seems that subjects were very good at identifying high quality passages in their final

selections. This also can be a motivation for studying automatic selection of passages for users' final reports.

4.2. Selection Efficiency

The examination can also be performed on certain aspects of the interaction process - for example, how quickly can subjects obtain useful passages in their exploration, which is called selection efficiency here.

The entire 20-minute SelLite experiment sessions (or sub-session for experiment involving Rosetta) were divided into 4 slots (0 to 5 minutes, 5 to 10 minutes, 10 to 15 minutes, and after 15 minutes,) and the number of passage selections with each time period was counted. Table 2 shows the number of passage selections and their percentages between the two systems in these four time periods. It is evident that subjects who used Rosetta selected more passages within the first 5 minutes, on average, than those using SelLite (38.6 vs. 20.3). In addition, almost all subjects' selections of passages in Rosetta (98%) were done within the first 5 minutes. This is much higher than that in SelLite (only 49%). After 5 minutes, there were almost no passage selections at all in Rosetta, especially after 10 minutes. However, subjects using SelLite still selected passages throughout the 20-minute session, although the total passage selections in Rosetta was only slightly lower, on average, than that in SelLite (39.3 vs. 41.6). The different selection pattern was tested with Pearson's chi-square statistics and the result was significant.

Table 2: Number of passages in average selected in a specific time period. Percentage numbers are in brackets.

Systems	0-5min	5-10min	10-15min	15-20min
Rosetta	38.6 (98%)	0.7 (2%)	0 (0%)	0 (0%)
SelLite	20.3 (49%)	5.5 (13%)	6.3 (15%)	9.5 (23%)

4.3. Subjects' Feedback Analysis

Our questionnaires obtained subjective views of the two systems from the angles of passage utility, ability to find useful passages, ease of use and overall satisfaction. Chi-square tests were performed on the data to determine significant differences.

For the between systems comparison, our hypothesis was that users would be able to find useful passages faster (and thus be more satisfied) with Rosetta, due to its ability to provide instant model updates to CAFÉ. Although our analyses do show that the subjects were able to find passages much faster, their comments show that this does not mean that they

would be less critical toward Rosetta. In fact, the comments received reveal that subjects gave more positive responses to SelLite across almost all questions. Although there were no significant differences between Rosetta and SelLite in any of the subjective measures, it does raise some interesting questions about possible reasons.

Table 3: Mean post-search questionnaire responses, summarized by session and system. (* p <= 0.05)

	SelLite	Rosetta
Utility of Passages	4.28	3.67
Ability to Find Useful Passages	3.52	3.08
Ease of Use	3.83	4.08
Negative Feedback	3.80	2.79
Overall Satisfaction	3.70	3.25

5. Discussion

In summary, our study uncovers some interesting differences between SelLite, which uses the between-session update strategy, and Rosetta, which uses the instant update strategy. We see that the instant strategy helped subjects to select most of their useful passages within the first 5 minutes of their sessions, significantly faster than those using the between-session strategy. This demonstrates the usefulness of having the instant update of the profiles and recommendations, where subjects could find their information more quickly.

However, we also see that the quality of the selected passages using the instant update strategy is significantly inferior to that of the passages using the between-session update strategy. Subjects' responses to our questionnaires also show more negative feedback toward the instant update strategy. One possible explanation lies in the highly adaptive and less stable behaviors triggered by the instant update strategy. Rosetta's output changes with each selection, and subjects may feel that they needed to exert more cognitive effort in order to manipulate the system, especially if they are disoriented by new passages or passages of changing rank.

It is also true that Rosetta's interface is much more complicated than that of SelLite. In the case of the latter, the subjects only needed to learn how to select passages from a list in order to complete the work. When using Rosetta, subjects had to learn to distinguish CAFÉ in Rosetta from a normal search engine, because CAFÉ includes a text input box very much resembling the search box in a search engine. Some subjects complained that they had difficulties fully understanding the characteristics of CAFÉ in

Rosetta, including its instant adaptation feature of the recommendations. This confusion may have caused subjects to be more critical of Rosetta.

The results seem to indicate that neither of the two update strategies is the best approach. The between-session strategy is slow in response to users' tasks, whereas the instant update strategy generates too many changes for users to handle. Perhaps the best approach lies in between. The profiles and recommendations should be updated along with the users' exploration processes, but less frequently, such as using a small buffer that stores up to a few (2 to 5) items of feedback from the user. Only when the buffer is full will the profile and the recommendations be updated. This strategy gives adaptability and stability to both the system and the user's exploration process.

6. Conclusion and Future Work

In this paper, we presented a study on strategies for updating profiles and recommendations in adaptive systems. Our study concentrated on studying the issues in the context of human subjects working on real task scenarios of high complexity, using information systems with two different update strategies. The underlying adaptive filtering engine is the same for both systems, allowing us to focus on the effects of the update strategies. Our results show that when looking at the quality of selected passages, the between-session strategy helped to find better quality information. The strategy also received better subjects' responses regarding its usefulness and usability, but it did prolong the selection of useful passages such that more than half of the passage selections were done after the first 5 minutes, whereas the instant update strategy helped subjects obtain most of their selected passages (> 98%) within the first 5 minutes of the session. Based on the study results, we hypothesize that the best strategy for updating profiles and recommendations might be a hybrid between the instant update strategy and the between-session strategy, where both adaptability and stability can be achieved at the same time. Our future work includes expanding the study scale to consider more topics and subjects, and testing the hybrid strategy idea.

References

[1] G. Marchionini, "Exploratory Search: From Finding to Understanding," *Communications of the ACM*, 49:41-6, 2006.
[2] R. W. White, G. Muresan, and G. Marchionini, "Evaluating Exploratory Search Systems," In *Evaluating Exploratory Search Systems*, workshop of SIGIR 2006.

[3] D. He, et. al., "An Evaluation of Adaptive Filtering in the Context of Realistic Task-Based Information Exploration," *Information Processing and Management*, 2007.
[4] S. K. Card, B. Suh, B. A. Pendleton, J. Heer, and J. W. Bodnar, "TimeTree: Exploring Time Changing Hierarchies," presented at IEEE Symposium on Visual Analytics Science and Technology, VAST 2006, Baltimore, MD, 2006.
[5] D. Gotz, et. al. "Interactive Visual Synthesis of Analytic Knowledge," In IEEE Symposium on Visual Analytics Science and Technology, VAST 2006, Baltimore, MD, 2006.
[6] P. C. Wong, G. Chin Jr., H. Foote, P. Mackey, and J. Thomas, "Have Green – A Visual Analytics Framework for Large Semantic Graphs," In IEEE Symposium on Visual Analytics Science and Technology, VAST 2006, 2006.
[7] D. McColgin, et al., "From Question Answering to Visual Exploration" In Workshop on Evaluating Exploratory Search Systems at SIGIR 2006, Seattle, USA, 2006.
[8] E. Santos Jr., et. al. "Impacts of User Modeling on Personalization of Information Retrieval: An Evaluation with Human Intelligence Analysts," In *Evaluation of Adaptive Systems*, workshop of UM2005, 2005.
[9] E. Santos Jr., et. al., "User modeling for intent prediction in information analysis," In 47th Annual Meeting for the Human Factors and Ergonomics Society (HFES-03), 2003.
[10] U. Hanani, B. Shapira, and P. Shoval, "Information filtering: Overview of issues, research and systems.," *User Modeling and User Adapted Interaction*, 11:203-259, 2001.
[11] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*, P. Brusilovsky, K. A., and W. Neidl, Eds.: Berlin Heidelberg New York: Springer-Verlag, 2007, pp. 325-341.
[12] D. Billsus and M. J. Pazzani, "User modeling for adaptive news access," *User Modeling and User Adapted Interaction*, vol. 10, pp. 147-180, 2000.
[13] A. Díaz and P. Gervás, "Personalisation in news delivery systems: Item summarization and multi-tier item selection using relevance feedback," *Web Intelligence and Agent Systems*, vol. 3, pp. 135-154, 2005.
[14] B. Magnini and C. Strapparava, "User modeling for news Web sites with word sense based techniques," *User Modeling and User Adapted Interaction*, 14:239-257, 2004.
[15] J.-W. Ahn, et al., "Open user profiles for adaptive news systems: help or harm?," In the 16th international conference on World Wide Web, WWW '07, 2007.
[16] Y. Yang, S. Yoo, J. Zhang, and B. Kisiel, "Robustness of adaptive filtering methods in a cross-benchmark evaluation.," In 28th ACM SIGIR conference, 2005.
[17] J. Fiscus and B. Wheatley, "Overview of the TDT2004 Evaluation and Results," In *Proceedings of TDT-04*, 2004.
[18] P. Borlund, "The IIR evaluation model: a framework for evaluation of interactive information retrieval systems," *Information Research*, vol. 8, 2003.
[19] R. W. White, et al. "Supporting Exploratory Search," *Communications of the ACM*, vol. 49, pp. 37-39, 2006.
[20] J. Allan, "HARD Track Overview in TREC 2003 High Accuracy Retrieval from Documents," TREC 2003.