Ontology-Based User Profiles for Search and Browsing

Susan Gauch	Jason Chaffee	Alexander Pretschner
Department of EECS	MetaTV, Inc.	Institut fur Informatik
University of Kansas	Mill Valley, CA 94941	Technische Universitat Munchen
Lawrence, KS 66045		Munchen, Germany
sgauch@ku.edu	jasonchaffee@metatv.com	pretschn@in.tum.de

This paper has not been submitted elsewhere in identical or similar form, nor will it be during the first three months after its submission to UMUAI.

ABSTRACT

As the number of Internet users and the number of accessible Web pages grows, it is becoming increasingly difficult for users to find documents that are relevant to their particular needs. Users must either browse through a large hierarchy of con cepts to find the information for which they are looking or submit a query to a publicly available search engine and wade through hundreds of results, most of them irrelevant. The core of the problem is that whether the user is browsing or searching, whet her they are an eighth grade student or a Nobel prize winner, the identical information is selected and it is presented the same way. In this paper, we report on research that adapts information navigation based on a user profile structured as a weighted concept hierarchy. A user may create his or her own concept hierarchy and use them for browsing Web sites. Or, the user profile may be created from a reference ontology by 'watching over the user's shoulder' while they browse. We show that these automati cally created profiles reflect the user's interests quite well and they are able to produce moderate improvements when applied to search results. Current work is investigating the interaction between the user profiles and conceptual search wherein documents are indexed by their concepts in addition to their keywords.

Keywords: ontologies, personalization, browsing, Web navigation, conceptual search

1. INTRODUCTION

The Web has experienced continuous growth since its creation. As of March 2002, the largest search engine contained approximately 968 million indexed pages in its database [SES 02]. As the number of Internet users and the number of accessible Web pages grows, it is becoming increasingly difficult for users to find documents that are relevant to their particular needs. Users of the Internet basically have two ways to find the information for which they are looking: they can browse or they can search with a search engine. Browsing is usually done by clicking through a hierarchy of concepts, or *ontology*, until the area of interest has been reached. The corresponding node then provides the user with links to related Web sites. Search engines allow users to enter keywords to retrieve documents that contain these keywords. The browsing and searching algorithms are essentially the same for all users.

The ontologies that are used for browsing content at a Web site are generally different for each site that a user visits. Even if there are similarly named concepts in the ontology, they may contain different types of pages. Frequently, the same concepts will appear with different names and/or in different areas of the ontology. Not only are there differences between sites, but between users as well. One user may consider a certain topic to be an "Arts" topic, while a different user might consider the same topic to be a "Recreation" topic. Thus, although browsing provides a very simple mechanism for information navigation, it can be time consuming for users when they take the wrong paths through the ontology in search of information.

The alternate navigation strategy, search, has its own problems. Indeed, approximately one half of all retrieved documents have been reported to be irrelevant [Casasola 98]. One of the main reasons for obtaining poor search results is that many words have multiple meanings [Krovetz 92]. For instance, two people searching for "wildcats" may be looking for two completely different things (wild animals and sports teams), yet they will get exactly the same results. It is highly unlikely that the millions of users with access to the Internet are so similar in

their interests that one approach to browsing or searching, respectively, fits all needs. What is needed is a solution that will "personalize" the information selecti on and presentation for each user.

This paper explores the OBIWAN project's [Zhu 99] use of ontologies as the key to providing personalized information access. In Section 3, we describe the automatic creation of user profiles based on a user's browsing be havior. In Section 4, we show how these profiles can be used to improve search results, and in Section 5 we discuss how users can create their own profiles and use them as the basis for personalized browsing. We conclude by summarizing the results of the se investigations and we discuss our current focus on conceptual, personalized search.

2. RELATED WORK

The following section presents related work on ontologies and personalization. Since we create our user profiles automatically using text classification te chniques, we will also review research in this area.

2.1 Classification

Classification is one approach to handling large volumes of data. It attempts to organize information by classifying documents into the best matching concept(s) from a predefined set of concepts. Several methods for text classification have been developed, each with a different approach for comparing the new documents to the reference set. These include: comparison between vector representations of the documents (Support Vector Machines, k-nearest neighbor, linear least-squares fit, tf * idf); use of the joint probabilities of the words being in the same document (Naive Bayesian); decision trees; and neural networks. A thorough survey and comparison of such methods is presented in [Yang 99], [Pazzani 96], and [Ruiz 99].

Classification has been applied to newsgroup articles, Web pages, and other online documents. The system described in [Hsu 99] classifies NETNEWS articles into the best matching news groups. The implementation uses the vector space model to compare new articles to those articles manually associated with each news group. The system presented in [Göver 99] is based on a probabilistic description-oriented representation of Web pages, and a probabilistic interpretation of the *k*-nearest neighbor classifier. It takes into account: 1) features specific to Web pages (e.g., a term appears in a title, a term is highlighted), 2) features standard to text documents, such as the term frequency. The *k*-nearest neighbor approach has also been used by [Larkey 98] in a system that uses classification techniques to automatically grade essays.

2.2 Ontologies

One increasingly popular way to structure information is through the use of ontologies, or graphs of concepts. One such system is *OntoSeek* [Guarino 99], which is designed for content-based information retrieval from online yellow pages and product catalogs. OntoSeek uses simple conceptual graphs to represent queries and resource descriptions. The system uses the *Sensus* ontology [Knight 99], which comprises a simple taxonomic structure of approximately 70,000 nodes. The system presented in [Labrou 99] uses *Yahoo!* [YHO 02] as an ontology. The system semantically annotates Web pages via the use of Yahoo! categories as descriptors of their content. The system uses *Telltale* [Chower 96a, Chower 96b, Pearce 97] as its classifier. Telltale computes the similarity between documents using *n*-grams as index terms.

The ontologies used in the above examples use simple structured links between concepts. A richer and more powerful representation is provided by *SHOE* [Heflin 99, Luke 97]. SHOE is a set of Simple HTML Ontology Extensions that allow WWW authors to annotate their pages with semantic content expressed in terms of an ontology. SHOE provides the ability to define

ontologies, create new ontologies which extend existing ontologies, and classify entities under an "is a" classification scheme.

2.3 Personalization

Personalization is a broad field of active research. Applications include personalized access to online information such as personalized "portals" to the Web, filtering/rating systems for electronic newspapers [Chesnais 95], Usenet news filtering, recommendation services for browsing, navigation, and search. Usenet news filtering systems include GroupLens [Konstan 97], PSUN [Sorensen 95], NewT [Sheth 94], *Alipes* [Mladenic 98], and SIFT [Yan 95]. SiteIF [Stefani 98] and ifWeb [Asnicar 97] aim to provide personalized search and navigation support. InformationLens [Malone 87] is a tool for filtering and ranking e-mails. Implicit rating and filtering are, among other topics, discussed in [Nichols 97] and [Oard 96]. Finally, [Vivacqua 99] describes a system for expertise location (Java source code). [Pretschner 99a] describes approximately 45 personalization systems and contains a detailed bibliography.

Many personalization projects have focused on navigation. *Syskill & Webert* [Pazzani 96] also recommends interesting Web pages using explicit feedback. If the user rates some links on a page, Syskill & Webert can recommend other links on the page in which they might be interested. In addition, the system can construct a Lycos query and retrieve pages that might match a user's interest. *Wisconsin Adaptive Web Assistant* (WAWA) [Shavlik 98][Shav lik 99] also uses explicit user feedback to train neural networks to assist users during browsing.

Personal WebWatcher [Mladenic 98] is an individual system that is based on *WebWatcher* [Armstrong 95, Joachims 97]. It "watches over the user's shoulder," but it avoids involving the user in its learning process because it does not ask the user for keywords or opinions about pages. *Letizia* [Lieberman 95, Lieberman 97] is a similar individual system that assists a user when browsing by suggesting links that might be of interest and are related to the

page the user is currently visiting. The system relies on implicit feedback including links followed by the user or pages and/or bookmarked pages. *WebMate* [Chen 98] is an individual system based on a stand-alone proxy that can monitor a user's actions to automatically create a user profile. Then the user can enter an URL and WebMate will download the page, check for similarity with the user's profile, and recommend any similar pages. *Amalthaea* [Moukas 96] is a server-based system that employs genetic algorithms to also try to identify Web pages of interest to users.

Most personalization systems are based on some type of user profile, most commonly a set of weighted keywords. Systems that use structured information rather than simple lists of keywords include PEA [Montebello 98] and SiteSeer [Rucker 97], both of which use bookmark information, PSUN [Sorensen 95] which uses K-lines, and SiteIF [Stefani 98] which uses semantic networks. By incorporating temporal information, [Widyantoro 01] uses an extended user profile model that distinguishes between a user's short term and long term interests. Similar to our work, SmartPush [Kurki 99] uses concept hierarchies for user profiles. In contrast, however, these are quite small (40-600 nodes), and weight adjustments are done using data that *explicitly* describes document contents. It is doubtful that hand-made hierarchical content annotation of data will be done on a large scale.

In order to build a user profile, some source of information about the user must be collected. Commercial systems, e.g., MyYahoo, explicitly ask the user to provide information about themself which simply stored to create a profile. Explicit profile creation is not recommended because it places an additional burden on the user, the user may not accurately report their own interests, and the profile remains static whereas the user's interests may change over time. Thus, implicit profile creation based on observations of the user's actions is used in most recent projects. [Chan 00] describes the types of information available. His model considers the frequency of visits to a page, the amount of time spent on the page, how recently a

page was visited and whether or not the page was bookmarked . Similar to our research, the user's surfing behavior is used to create the user profiles in Anatagonomy [Sakagami 97], Letizia [Lieberman 95, Lieberman 97], Krakatoa [Kamba 95], Personal WebWatcher [Mladenic 98], and WBI [Barrett 97].

Our user profiling technique differs from other approaches due to our focus on automatically creating user profiles based on ontologies. In our use of ontologies, we overlap somewhat with initiatives aimed at creating a Semantic Web [Berners-Lee 01]. However, these proposals tend to focus on encoding semantics into the Web pages to describe their content, whereas we use classification techniques to create profiles for users and/or Web sites.

3. AUTOMATIC CREATION OF USER PROFILES

In our system, the user profile is created automatically and implicitly while the users browse. The user profile is essentially a reference ontology in which each concept has a weight indicating the perceived user interest in that concept. Profiles are generated by analyzing the surfing behavior of the user, specifically the content, length, and time spent on each page they visit. No user feedback is necessary.

3.1 Profile Creation

For this study, the reference ontology/user profile consisted of approximately 4,400 concepts (the top level categories from the Magellan Web site). We collected ten documents linked to each concept by the Magellan editors to be used as training data for our vector-space-based classifier. The training documents for each concept were merged to create a collection D containing one super-document per concept. The super-documents were pre-processed to remove high-frequency function words (stopwords) and HTML tags. Finally, the Porter stemmer [Frakes 92] was used to reduce each word to its root and thereby decrease the dimensionality of

the vectors used to represent each concept. The normalized weight of term t_i in super-document d_j, td_{ij}, is given by:

$$td_{ij} = \frac{tf_{ij} * idf_i}{concept\ \ size_j} \quad (1)$$

where

$$tf_{ij} = \text{number of occurences of } t_i \text{ in } d_j \quad (2)$$

$$idf_i = \log\left(\frac{\text{number of documents in } D}{\text{number of documents in } D \text{ that contain } t_i}\right) \quad (3)$$

$$concept\ size_j = \sum_i tf_{ij} * idf_i \quad (4)$$

The files in a Web browser's cache folder were periodically classified into the appropriate concept(s) in the reference ontology. For each of the surfed pages, a term vector was calculated using the same formulae used for the super-document term vectors. The similarity between the page vector, d_j, and the vector associated with concept k, q_k, was calculated using the cosine similarity measure:

$$similarity(d_j, q_k) = \frac{\sum_{i=1}^{n}(td_{ij} * tq_{ik})}{\sqrt{\sum_{i=1}^{n} td_{ij}\text{^}2 * \sum_{i=1}^{n} tq_{ik}\text{^}2}} \quad (5)$$

where

td_{ij} = *the ith term in the vector for document j*

tq_{ik} = *the ith term in the vector for query k*

n = *the number of unique terms in the document collection D*

The concepts with the most similar vectors were assumed to be those most related to the content of the surfed page.

We also investigated the influence of two other factors in concept weight calculation: duration of the visit and page length. Using four different formulae combining theses factors

(see Section 3.2), the strengths of match for the top five concepts reported by the classifier were revised. Intuitively, if a user spends a long time on the page, their interest value in that page should be increased. Also, if the page is long, the time factor should be decreased since the increased time may be due to the amount of information presented, not the level of interest.

The evaluation of our user profile creation algorithm consisted of two parts [Pretschner 99b]. First, we tested our profile creation algorithm to determine whether or not it was able to create a stable user profile. The second experiment validated our automatically generated user profiles against actual user interests.

3.2 Profile Convergence

For the experiments, a group of sixteen users were monitored for 26 days. These sixteen users together surfed 7,664 documents. The mean time spent was 54.6 seconds with a standard deviation of 93.4 seconds. 20% of all pages are visited for less than 5 seconds.

One would assume that each person has a relatively stable collection of interests that may change over time [Lam 96]. We wished to determine how long it takes our system to identify this core set of interests. A user profile is said to be convergent if the number of concepts with non-zero interest values converges over time. Users varied in the number of categories their profiles converged to, most containing between 50 and 200 concepts that account for 95% of the total accumulated personal weight. Low-weighted concepts were ignored to filter "noise" introduced by text classification or user navigation errors.

All profiles showed a tendency to converge after roughly 320 pages, or 17 days, of surfing when the user profile weight was adjusted using either of the following two formulae:

$$\log \frac{time}{\log length} \qquad \text{or} \qquad \log \frac{time}{\log \log length}$$

Two other formulae were evaluated:

$$\log \frac{time}{length} \qquad \text{and} \qquad \frac{time}{length}$$

However, these formulae did not result in profile convergence. This indicates that the length of a surfed page does not much matter when determining the user's interest level. Thus, it seems that users can tell at a glance that a page is irrelevant and, in general, reject it quickly regardless of its length. Time, on the other hand, is important because users do not, in general, spend long on pages in which they have little interest.

3.3 Comparison with Actual User Interests

Although convergence is a desirable property, it does not measure the accuracy of the generated profiles. Thus, the sixteen users were shown the top twenty concepts in their profiles in random order and asked how appropriately these inferred categories reflected their interests. For both the top ten and top twenty concepts, approximately one half of the categories represented actual interests (5.2 and 10.5 respectively), one quarter represented errors and the remaining quarter represented topics of marginal interest. Bearing in mind that the "good" concepts have been chosen out of 4,400 concepts, this result is encouragingly accurate. 75% of the twenty categories chosen reflect actual interests even though these represent only 0.5% of all possible concepts.

4. PERSONALIZED SEARCH RESULTS

Because queries are so short, search engines do not receive enough detail about the user's information need. As a result, many retrieved documents are irrelevant. Although the profiles created as described in Section 4 were not perfect, we evaluated their suitability for improving search results using two different approaches:

1) Re-ranking Re -ranking algorithms apply a function to the match values and/or rank orders returned by the search engine. If that function is well c hosen, it will move relevant documents higher in the list.
2) Filtering Filtering systems determine which documents in the result sets are relevant and which are not. Good filters remove many non -relevant documents and preserve the relevant ones in the results set.

4.1 Evaluation

For a given query, re -ranking was done by modifying the ranking that was returned by the ProFusion meta-search engine [ProFusion 02]. We classified each of the documents in the result set, or rather their title and summary, which, ac cording to [Casasola 98] and [Pazzani 96], is sufficient for classification purposes. The user's average interest in the document's top four concepts was assumed to be an approximation to the actual user interest in the whole document. For each document, *d*, we calculate new match values, new_wt_d, based on the match value returned by the search engine, the strength of association between the document and its non -zero concepts, and the level of user interest in the concept. We evaluated a variety of formul ae combining these factors using multiplication or addition and weighting their contributions differently. The multiplicative formula to is representative of the set of formulae used. For the top four concepts identified by the classifier for document *j*,

$$new\_wt_d = wt_d * (0.5 + \frac{1}{4}\sum_{i=1}^{4} u_{di}) \quad (6)$$

where

wt_d is the weight returned by the search engine

u_{di} is the user's interest in concept d_i (from their profile)

d_i is the *i*th most highly weighted concept for document *d*

To compare the results produced by the different re-ranking formulae, we used the eleven point precision average [Harman 96]. The eleven point precision average evaluates ranking performance in terms of *recall* and *precision*. Recall is a measure of the ability of the system to present all relevant items, and precision is a measure of the ability of a system to present only relevant items. Sixteen users were each asked to submit three queries (48 total). Two queries per users were used for training (32 total) and the third query was reserved for evaluation (sixteen total). The results were presented in random order, and the users were asked to judge each result as being "relevant" or "non-relevant." On average, before re-ranking, only 8.7 of the twenty retrieved pages were considered to be relevant. This is consistent with the findings in [Casasola 98] which reports that roughly 50% of the retrieved documents are irrelevant (with a statistically more significant set of 1,425 queries and 27,598 judged results). According to our results, the multiplicative ranking function (see Formula 6) produced the best performance increase (8%). In particular, the best improvement is occurs within the top-ranked documents. Since the top documents are those most likely to be examined by a user, improvement at the top of the list is encouraging.

We also evaluated the ability of the user profile to filter documents from the result set. After calculating personalized match values (see Formula 6), we excluded documents whose match values fell below a threshold. We evaluated a variety of threshold values and achieved approximately a 2:1 ratio of irrelevant documents removed to relevant documents removed at all values of the threshold. Clearly, as the threshold was raised, more documents of both types were removed.

4.2 Discussion

We were able to create large, structured, user profiles entirely automatically. These profiles were shown to converge and to reflect actual user interests quite well. To evaluate their usability, two applications have been investigated: re-ranking and filtering search results. In

terms of re-ranking, performance increases of up to 8% were achieved. In terms of filtering, roughly a 2:1 ratio of irrelevant documents to relevant documents were removed.

5. PERSONALIZED BROWSING

This system maps between the reference ontology used by the *Ontology Based Informing Web Agent Navigation (OBIWAN)* [Zhu 99] system and the user's personal ontology. OBIWAN spiders and classifies the Web pages of a site using a reference ontology derived from the ontology used by Lycos [Lycos 02]. Just like Yahoo, or any other hierarchically arranged web sites, with OBIWAN, the user can browse the content of a site by clicking up and down a hierarchy of concepts. With OBIWAN, however, all sites appear to be organized conceptually according to the reference ontology, even though each site may be designed around a different hierarchy or arranged non-conceptually (e.g., alphabetically). This work extends OBIWAN so that all sites are browsed using the user's own ontology rather than a system supplied reference ontology.

To create a personal ontology, a user amasses a collection of Web pages that he or she arranges into a hierarchy based on his or her worldview. The system then finds a mapping from the reference ontology concepts to concepts in the personal ontology. Using this mapping, the user can browse any site that has been characterized by OBIWAN with his or her personal ontology without reclassifying the documents. Since OBIWAN will characterize every site in the same manner, and each user's personal ontology reflects their view of the world, they will be able to browse Web pages in a personalized, consistent manner.

5.1 System Architecture

Each concept has a set of documents that were manually assigned to that concept. For the reference ontology, these documents were collected from the Lycos site. For the personal ontology, the sample documents are provided by the user. The personal browsing system needs

to map from reference ontology concepts to the best matching concept in the personal ontology. To do this, it must calculate the match value between each concept in the reference ontology and the concepts in the personal ontology. Figure 1 shows the system architecture for the personalized browsing system.

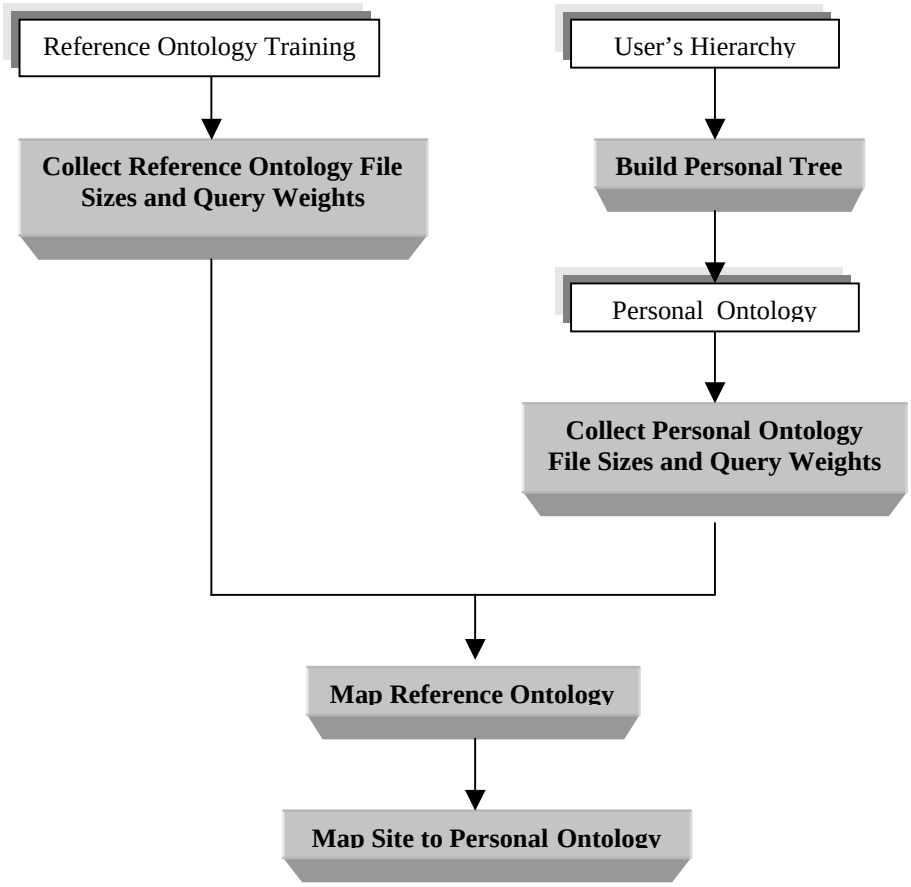

Figure 1. Personalized Browsing Architecture.

5.2 Mapping the Reference Ontology to the Personal Ontology

Each user submits their personal ontology, a hierarchical tree of concepts that represents their view of the world. For our experiments, the tree was required to contain at least ten concepts with at least five sample pages for each concept. The goal of the mapping phase is to map every concept in the reference ontology to a concept in the personal ontology. However, since personal ontologies tend to be much smaller and more narrowly focused than the reference

ontology, many concepts will remain unmapped. Thus, we augment the personal tree with an extra concept called "All-Others" to hold the concepts from the reference ontology that do not map to a corresponding concept in the personal ontology.

We take a multi-phase approach to mapping from each reference ontology concept to the best matching personal ontology concept. While it is possible for a reference ontology concept to map to multiple personal ontology concepts, this would indicate that the personal concepts are more fine-grained than the reference concepts. By choosing an extensive reference ontology, the likelihood of this occurring is decreased. Thus, we simplified our mapping algorithm to focus on mapping each reference concept to the best matching, single personal concept. In practice, our users tended to create concepts that were at least as broad or broader than the reference concept.

The first step maps from the personal ontology concepts to the reference ontology concepts. As described in Section 3.1, vectors are created for each concept in the reference ontology, and the same technique is used to create vectors for each concept in the personal ontology. The similarity between the personal concept vector and reference ontology vector is calculated using the cosine similarity measure (Formula 5) and the top 30 matches are returned. The result of this process is a one-to-many mapping from personal ontology concepts to reference ontology concepts.

After the first step, the same reference ontology concept may appear on multiple lists (i.e., be mapped to more than one personal ontology concept). So, the next step filters the results of step 1 to identify the best matching personal concept for each reference concept. This produces a one-to-one mapping from reference ontology concepts to personal ontology concepts.

The first two steps map individual reference ontology concepts to their best matching personal ontology concept. Since the personal ontology concepts tend to be broader in scope than the reference ontology, we next map any unmapped descendents of mapped nodes to the

same personal ontology concept as their nearest ancestor. Where an unmapped node has multiple mapped ancestors at the same level, the mapping with the highest weight is chosen. This has the effect of mapping entire subtrees rather than just individual concepts. For instance, in Figure 2 it can be seen that the concept "Anime" has ancestors "Animation" and "Arts -&-Entertainment", with "Animation" being the closest ancestor. Therefore, "Anime" has two possible ancestors to which it could be mapped.

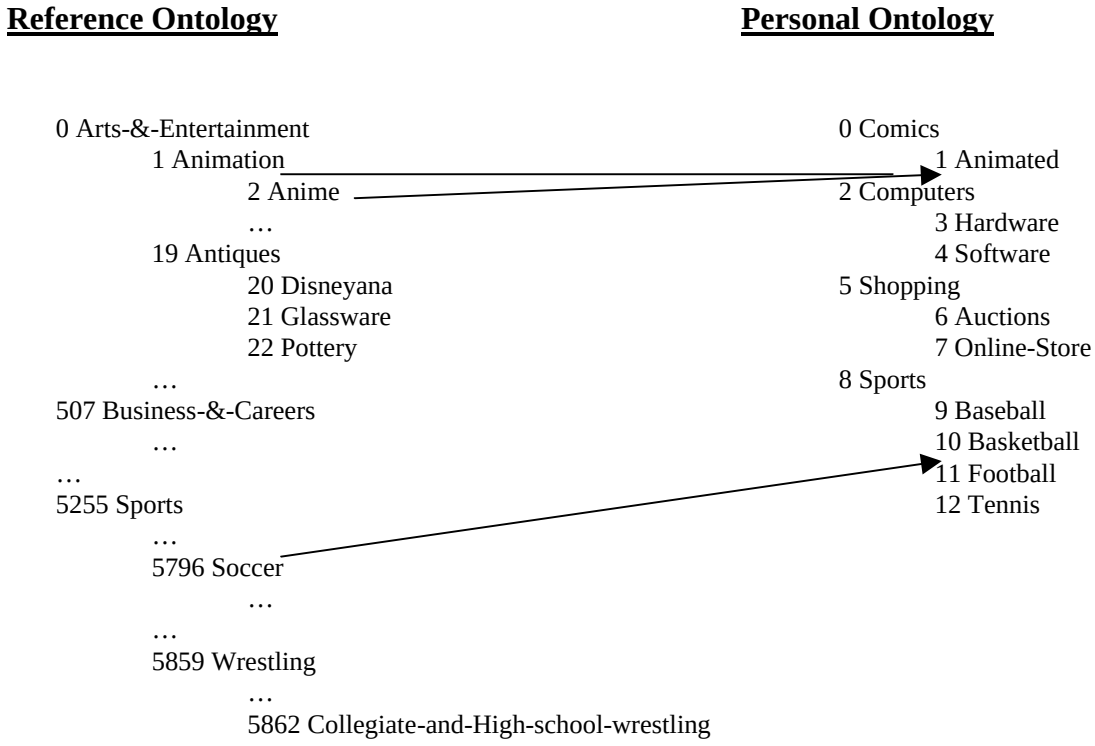

Figure 2. The Reference Ontology Mapped to a Personal Ontology

After the system has mapped a reference ontolog y concept to a personal ontology concept, a mapping factor is calculated which measures the closeness of the match, normalized by the sizes of the mapped concepts and the value of the concept's term vector matched against itself (see Formula 6). For more details, see [Chaffee 00]. The mapping factor can be viewed as a measure of our confidence in the mapping.

$$mapping\ \ factor = \frac{\frac{matching\ \ weight}{file\ size\ of\ \ personalized\ concept}}{\frac{weight\ of\ \ reference\ concept\ \ queried\ \ against\ \ itself}{file\ size\ of\ \ reference\ concept}} \quad (6)$$

5.3 Mapping a Site to the Personal Ontology

Once the mapping file has been created, any site that has had its Web pages spidered and classified into the reference ontology concepts can easily be mapped to the personal ontology. If several concepts in the reference ontology map to one concept in the personal ontology, they are all merged together under the personal concept. If a concept in the reference ontology does not map to any concept in the personal ontology, the pages will remain in the reference ontology concept. Next, the weights must be recalculated for each page that is mapped to the personal ontology. The new weight is calculated by using the matching weight for the page in the reference ontology multiplied by the mapping factor for the reference ontology concept to the personal concept.

$$new\ weight = matching\ \ weight\ \ for\ \ page\ in\ reference\ \ ontology * mapping\ \ \ factor \quad (7)$$

After all pages have been mapped and their weights recalculated, the weights for the concept as a whole are calculated as the sums of the weights of mapped pages plus the weight of any subtrees. Now the site can have its content browsed using the personal ontology rather than OBIWAN's reference ontology.

5.4 Evaluation

The system was evaluated by having five users create personal ontologies. Each user was asked to provide feedback on two different experiments. The first experiment asked each user to compare the reference ontology concept that was mapped to their personal concept and

decide if it was mapped correctly. The second experiment had each user browse a site's Web pages after they had been mapped to their personal ontology (see Figure 3). Each user reported whether or not each page that was mapped to their personal ontology was mapped to the correct concept.

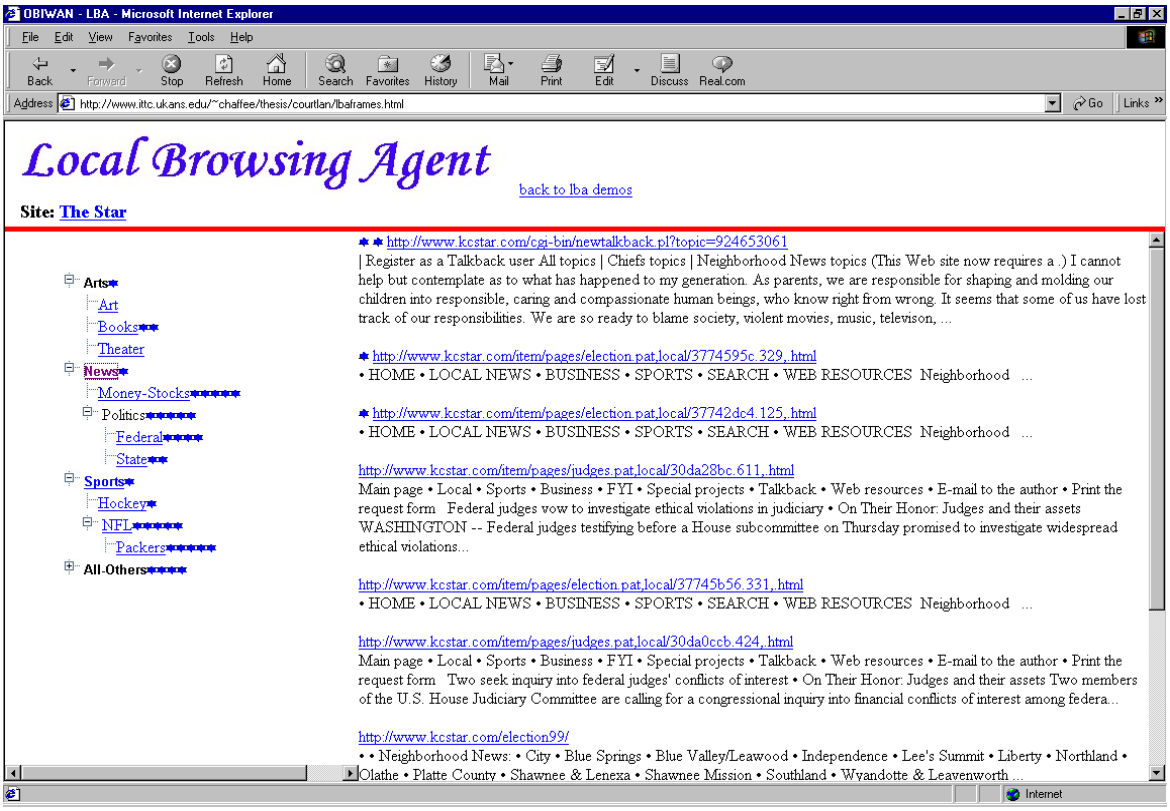

Figure 3. Screen Shot of a Web Site's Content Displayed after being Mapped to the Personal Ontology.

5.4.1 Evaluating Ontology Mappings

The user was given a Web interface to view each one of their concepts and every concept from the reference ontology that had been mapped to the personal concept. Also, the user was able to view the training data from the reference ontology concepts. The user was asked to give a Yes/No answer to the question of whether or not the reference ontology concept matched the personal ontology concept.

We then used the user responses to determine a threshold. We expected that the percentage of correct mappings would increase if we eliminated mappings below some threshold. Table 1 shows the precision, recall and correctn ess values for each threshold. When the threshold is increased, the number of concepts that are mapped both correctly and incorrectly is reduced. In the extreme, if the threshold is set to 100%, there are no results because there are no mappings. Therefore, another measure was used to measure "correctness" for each threshold. It was found that a threshold of 0.3 produced the highest number of correct mappings.

$$correctnes\ s = \frac{number\ \ kept\ \ that\ \ are\ correct\ + number\ \ dropped\ \ that\ \ are\ incorrect}{total\ \ number\ \ of\ \ concepts\ \ mapped\ \ \ with\ no\ threshold} \qquad (8)$$

Mapping Factor Threshold	**0.0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
Precision	49%	49%	49%	53%	52%	45%	34%	35%	36%	100%
Recall	100%	100%	99%	84%	41%	16%	5%	2%	1%	0%
Correctness	49%	49%	49%	55%	53%	49%	49%	50%	51%	51%
Mapped Correctly (seen)*	585	585	577	491	241	91	29	11	4	1
Mapped Correctly (not seen)**	0	0	8	94	344	494	556	574	581	584
Total Seen***	1192	1192	1179	931	460	202	85	31	11	1

Table 1. Precision, Recall and Correctness values for Various Mapping Thresholds.

* All concepts or pages which were mapped correctly and were not removed due to the threshold.

** All concepts or pages which were mapped correctly and were removed due to the threshold.

*** All concepts or pages that were mapped and were not removed due to the threshold.

5.4.2 Evaluating Site Mappings

The evaluation of the ontology mappings showed that a threshold of 0.3 would produce the most correct mappings from the reference ontology to the personal ontology. Therefore, each user's concept mappings were pruned using a threshold of 0.3 before an individual site's Web pages were mapped to their personal ontologies. Only the top ten mapped pages were kept for any concept in the personal ontology. As with the previous experiment , the user was asked to give a Yes/No answer on whether or not each page that had been mapped to a personal concept belonged there.

We then used the user responses to determine a threshold for the mapping weight of an individual page. We expected that t he percentage of correct mappings would increase if we eliminated mappings below some threshold.

Mapping Weight Threshold	**0**	**100**	**200**	**300**	**400**	**500**	**600**	**700**	**800**	**900**
Precision	50%	50%	50%	46%	37%	25%	19%	20%	21%	15%
Recall	100%	100%	82%	52%	29%	14%	9%	8%	7%	4%
Correctness	50%	50%	50%	45%	41%	36%	36%	38%	41%	42%
Mapped Correctly (seen)*	136	136	111	71	39	19	12	11	9	5
Mapped Correctly (not seen)**	0	0	25	65	97	117	124	125	127	131
Total Seen***	274	273	222	156	105	76	64	56	43	33

Table 2. Precision, Recall and Correctness values for Various Mapping

5.4.3 Discussion

We evaluated the system with two measures, precision and correctness. Precision measures the number of correct pages that were seen vs. the total number of pages that were

seen. Correctness measures the number of correct pages seen plus the number incorrect pages not seen vs. the total number mapped.

It was found that the concepts mapped correctly with a precision of 49% and correctness of 49% with no threshold. The best results were achieved with a mapping threshold of 0.3. This produced a precision of 53% and a corre ctness of 55%. Using a threshold for mapping concepts will reduce the number of reference concepts that actually are mapped, but it will cause the concepts that are mapped to have a higher relevance with the personal concepts. There are several factors that affected the results. First, the concepts that were submitted by the users were not always conceptual in nature, e.g., a user's name. Second, the training data in both the reference ontology and the personal ontologies was not as good as we expected. Although we had what appeared to be an adequate number of pages, many of the pages contained very little content, or the content included a template that added noise to the frequency statistics of words.

We found that individual pages mapped correctly with a precision and correctness of 50% with no threshold. In contrast to the concept mappings, the use of a threshold did not improve precision or correctness. We believe the main source of the low correctness was primarily due to errors introduced when the Web site pages were mapped to the reference ontology concepts rather than when the reference ontology concepts were mapped to the personal ontology concepts.

Currently, the user is asked to provide an ontology for the system. Most users do not want to take the time to create an ontology, especially one that only contains concepts. Therefore, a system that creates the ontology for the user would be beneficial. Finally, the system as described maps from a reference ontology to a personal ontology. It could also be used to map between two commonly found ontologies on the Web. For example, Yahoo!'s ontology could be used as the reference ontology and Lycos' ontology could be the ontology the system will map to. Then, a user could browse Yahoo!'s categories with the Lycos ontology.

6. CONCLUSIONS AND FUTURE WORK

This paper reviews extensions to the OBIWAN project that are working towards the goal of personalized navigation of online information. Our research revolves around using weighted ontologies to represent users and documents conceptually. Ample and accurate training data for each concept, and a reliable and robust classification algorithm is key to the success of this approach. We have used a variety of online subject hierarchies for our reference ontology, but currently we are using approximately 3,000 categories from the Open Directory Project. If ussers create their own ontologies, we can map from their personal ontologies to the reference ontology. Then, users can browse Web sites from their own viewpoint of the world rather than a generic organization of data provided to everyone.

The current focus of our work is on providing personalized search results. Early investigations report on automatically creating user profiles represented as a weighted ontology. These profiles are created implicitly based on the user's surfing behavior. Profiles which emphasize concepts related to documents on which users spend the most time perform the best. It appears that normalization by the length of the document is unimportant. These profiles have been shown to improve search performance by re-ranking the documents returned by the Profusion meta-search engine. Documents that classify into concepts that the user profile indicates the user has an interest in get promoted, moving documents relevant to this user higher up the list.

The personalized search results reported here are promising, but they exposed two areas of possible improvement. First, the quality of the results is greatly affected by the quality of the classification of documents into concepts that, in turn, is affected by the quality of the training data for each concept. Second, working as a post-process on the search results limits the ability of the system to achieve dramatic gains in search performance. If few of the twenty documents returned by the search engine address the user's information needs, then re-ranking and/or

filtering cannot help. To address the first issue, we have since improved our training phase to collect more trainin g documents and then evaluate them to identify a subset of high -content pages. To address the second issue, we need to integrate the conceptual matching between the user's profile and the document concepts into the retrieval process itself. This is the g oal of our ongoing KeyConcept project .

We have developed the first version of KeyConcept [KeyConcept 02], a conceptual search engine that classifies documents as part of the indexing process. It allows users to specify queries that match their keywords a nd concepts of interest. Currently, the concepts are either explicitly entered by the user or inferred from ancillary text. This system was evaluated on a large collection and a significant increase in top ten precision was found. Our next step is to merge the automatically created user profiles with KeyConcept so that the user profile is submitted along with the query terms. Documents that match the keywords and also the concepts in the user profile will be preferentially retrieved. It is our hope tha t we will thereby make a major step towards truly personalized search.

7. REFERENCES

[Armstrong 95] Robert Armstrong, Dayne Freitag, Thorsten Joachims, Tom Mitchell. WebWatcher: A Learning Apprentice For The World Wide Web. In Proceedings of the AAAI Spring Symposium On Information Gathering, 1995, pp. 6-12.

[Asnicar 97] F. Asnicar and C. Tasso. ifWeb: A Protoype of User Model-Based Intelligent Agent for Documentation Filtering and Navigation in the World Wide Web. In Proceedings of the 6th International Conference on User Modeling, June 1997.

[Barrett 97] R. Barrett, P. Maglio and D. Kellem. How to Personalize the Web. In Proceedings of ACM CHI'97, Atlanta, USA, 1997.

[Berners-Lee 01] Tim Berners-Lee, James Hendler and Ora Lassila. The Semantic Web. In Scientific American, 284 (5), May, 2001, pp. 34 – 43.

[Casasola 98] E. Casasola. ProFusion Personal Assistant: An Agent for Personalized Information Filtering on the WWW. Master's thesis, The University of Kansas, 1998

[Chaffee 00] Jason Chaffee, Susan Gauch. Personal Ontologies For Web Navigation. In Proceedings of the 9th International Conference On Information Knowledge Management (CIKM), 2000, pp. 227-234.

[Chan 00] Philip Chan. Constructing Web User Profiles: A Non-Invasive Learning Approach. In Web Usage Analysis and User Profiling, LNAI 1836, Springer-Verlag, 2000, pp. 39-55.

[Chen 98] L. Chen and K. Sycara. A Personal Agent for Browsing and Searching. In Proceedings of the 2nd International Conference on Autonomous Agents, 1998, pp. 132-139.

[Chesnais 95] P. Chesnais, M. Mucklo and J. Sheena. The Fishwrap Personalized News System. In Proceedings of IEEE 2nd International Workshop on Community Networking: Integrating Multimedia Services to the Home. Princeton, NJ, June 1995.

[Chower 96a] G. Chower and C. Nicholas. Resource Selection in Café: an Architecture for Networked Information Retrieval. In Proceedings of SIGIR'96 Workshop on Networked Information Retrieval, Zurich, 1996.

[Chower 96b] G. Chower and C. Nicholas. Meta-Data for Distributed Text Retrieval. In Proceedings of First IEEE Metadata Conference, 1996.

[Frakes 92] W. B. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, 1992.

[Goecks 99] J. Goecks and J. Shavlik. Automatically Labeling Web Pages Based on Normal User Actions. In Proceedings of the Workshop on Machine Learning for Information Filtering, 1999 International Joint Conference on Artificial Intelligence (IJCAI), Stockholm, Sweden, 1999.

[Goecks 00] J. Goecks and J. Shavlik. Learning Users' Interests by Unobtrusively Observing Their Normal Behavior. In Proceedings of the 2000 International Conference on Intelligent User Interfaces, New Orleans, LA, 2000. pp. 129-132.

[Gover 99] N. Göver, M. Lalmas and N. Fuhr. A Probabilistic Description-Oriented Approach for Categorising Web Documents. In Proceedings of the 8th International Conference on Information and Knowledge Management, 1999, pp. 475-482.

[Guarino 99] N. Guarino, C. Masolo, and G. Vetere, OntoSeek: Content-Based Access to the Web. IEEE Intelligent Systems, 14(3), May 1999, pp. 70-80.

[Harman 96] D. Harman. Evaluation Techniques and Measures. In Proceedings of the 4th Text REtrieval Conference (TREC-4), 1996, pp. A6-A14.

[Heflin 99] Jeff Heflin, James Hendler, and Sean Luke. SHOE: A Knowledge Representation Language for Internet Applications. Technical Report CS-TR-4078 (UMIACS TR-99-71), University of Maryland at College Park, 1999. http://www.cs.umd.edu/projects/plus/SHOE/pubs/techrpt99.pdf

[Hsu 99] Wen-Lin Hsu and Sheau-Dong Lang. Classification Algorithms for NETNEWS Articles. In Proceedings of the 8th International Conference on Information and Knowledge Management, 1999, pp. 114-121.

[Joachims 97] T. Joachims, D. Freitag, T. Mitchell. WebWatcher: A Tour Guide for the World Wide Web. In Proceedings of IJCAI'97, August 1997.

[Kamba 95] T. Kamba, K. Bharat and M. Albers. The Krakatoa Chronicle – An Interactive, Personalized Newspaper on the Web. In Proceedings of the 4th International WWW Conference, 1995, pp. 159-170.

[KeyConcept 02] KeyConcept Project, http://www.ittc.ku.edu/keyconcept

[Knight 99] K. Knight and S. Luk. Building a Large Knowledge Base for Machine Translation. In Proceedings of American Association of Artificial Intelligence Conference (AAAI), 1999, pp. 773-778.

[Konstan 97] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, J. Riedl. GroupLens: Applying Collaborative Filtering To Usenet News. Communications of the ACM, 40(3), March 1997, pp. 77-87.

[Krovetz 92] Robert Krovetz and Bruce W. Croft. Lexical Ambiguity and Information Retrieval. ACM Transactions on Information Systems, 10(2), April 1992, pp. 115-141.

[Kurki 99] T. Kurki, S. Jokela, R. Sulonen and M. Turpeinen. Agents in Delivering Personalized Content Based on Semantic Metadata. In Proceedings of the 1999 AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace, 1999, pp. 84-93.

[Labrou 99] Yannis Labrou, Tim Finin. Yahoo! As An Ontology – Using Yahoo! Categories To Describe Documents. In Proceedings of the 8th International Conference On Information Knowledge Management (CIKM), 1999, pp. 180-187.

[Lam 96] W. Lam, S. Mukhopadhyay, J. Mostafa and M. Palakal. Detection of Shifts in User Interests for Personalized Information Filtering. In Proceedings of ACM SIGIR'96, Zurich, Switzerland, 1996.

[Larkey 98] Leah S. Larkey. Automatic Essay Grading Using Text Categorization Techniques. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1998, pp. 90-95.

[Lieberman 95] Henry Lieberman. Letizia: An Agent That Assists Web Browsing. In Proceedings of the 14th International Joint Conference On Artificial Intelligence, 1995, pp. 924-929.

[Lieberman 97] Henry Lieberman. Autonomous Interface Agents. In Proceedings of the ACM Conference on Computers and Human Interaction (CHI'97), May 1997.

[Luke 97] S. Luke, L. Spector, D. Rager and J. Hendler. Ontology-Based Web Agents. In Proceedings of the First International Conference on Autonomous Agents (AA'97), 1997.

[Lycos 02] Lycos. http://www.lycos.com

[Malone 87] T. Malone, K. Grant, F. Turbak, S. Brobst and M. Cohen. Intelligent Information Sharing Systems. Communications of the ACM, (30), May 1987, pp. 390-402.

[Mladenic98] Dunja Mladeni. Personal Web Watcher: Design and Implementation. Technical Report IJS-DP-7472, J. Stefan Institute, Department for Intelligent Systems, Ljubljana, Slovenia, 1998.

[Montebello 98] M. Montebello, W. Gray and S. Hurley. A Personable Evolvable Advisor for WWW Knowledge-Based Systems. In Proceedings of the 1998 International Database Engineering and Application Symposium (IDEAS'98), July 1998, pp. 224-233.

[Moukas 96] Alexandros Moukas. Amalthaea: Information Discovery And Filtering Using A Multiagent Evolving Ecosystem. In Proceedings of the Conference on the Practical Application of Intelligent Agents and MultiAgent Technology, 1996. http://moux.www.media.mit.edu/people/moux/papers/PAAM96

[Nichols 97] D. Nichols. Implicit Rating and Filtering. In Proceedings of the 5th DELOS Workshop on Filtering and Collaborative Filtering, November 1997.

[Oard 96] D. Oard and G. Marchionini. A Conceptual Framework for Text Filtering. Technical Report EE-TR-96-25 CAR-TR-830 CLIS-TR-9602 CS-TR-3643, University of Maryland, May 1996.

[Pazzani 96] Michael Pazzani, Jack Muramatsu, Daniel Billsus. Syskill & Webert: Identifying Interesting Web Sites. In Proceedings of the 13th National Conference On Artificial Intelligence, 1996, pp. 54-61.

[Pearce 97] Claudia Pearce, Ethan Miller. The TellTale dynamic hypertext environment: Approaches to scalability. In Advances in Intelligent Hypertext, Lecture Notes in Computer Science. Springer-Verlag, 1997.

[ProFusion 02] ProFusion. http://www.profusion.com

[Pretschner 99a] Alexander Pretschner. Ontology Based Personalized Search. Master's thesis, University of Kansas, June 1999.

[Pretschner 99b] Alexander Pretschner, Susan Gauch. Ontology Based Personalized Search. In Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), November 1999, pp. 391-398.

[Rucker 97]. James Rucker, Marcos J. Polanco. Siteseer: Personalized Navigation For The Web. Communications of the ACM, 40(3), March 1997, pp. 73-75.

[Ruiz 99] Miguel Ruiz, Padmini Srinivasan. Hierarchical Neural Networks For Text Categorization. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 1999, pp. 281-282.

[Sakagami 97] H. Sakagami and T. Kamba. Learning Personal Preferences on Online Newspaper Articles From User Behaviors. In Proceedings of the 6th International WWW Conference, 1997, pp. 291-300.

[Salton 89] G. Salton. Automatic Text Processing. Addison-Wesley, 1989. ISBN 0-201-12227-8.

[Shavlik 98] J. Shavlik and T. Eliassi-Rad. Intelligent Agents for Web-Based Tasks: An Advice-Taking Approach. In Working Notes of the AAAI/ICML-98 Workshop on Learning for text categorization, Madison, WI, 1998.

[Shavlik 99] J. Shavlik, S. Calcari, T. Eliassi-Rad, and J. Solock. An Instructable, Adaptive Interface for Discovering and Monitoring Information on the World Wide Web. In Proceedings of the 1999 International Conference on Intelligent User Interfaces, Redondo Beach, CA, 1999.

[SES 02] Search Engine Showdown: Size statistics. http://www.searchengineshowdown.com/stats/sizeest.shtml

[Sheth 94] B. Sheth. A Learning Approach to Personalized Information Filtering. Master's thesis, Massachusetts Institute of Technology, February 1994.

[Sorensen 95] H. Sorensen and M. McElligott. PSUN: A Profiling System for Usenet News. In Proceedings of CIKM'95 Workshop on Intelligent Information Agents, December 1995.

[Stefani 98] A. Stefani and C. Strappavara. Personalizing Access to Web Sites: The SiteIF Project. In Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia HYPERTEXT'98, June 1998.

[Vivacqua 99] A. Vivacqua. Agents for Expertise Location. In Proceedings of the 1999 AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace, 1999, pp. 9-13.

[Widyantoro 01] D. H. Widyantoro, T. R. Ioerger, and J. Yen. Learning User Interest Dynamics with a Three-Descriptor Representation. Journal of the American Society of Information Science and Technology (JASIST) , Vol 52, No. 3, 2001, pp. 212-225,.

[Yan 95] T. Yan and H. García-Molina. SIFT – A Tool for Wide-Area Information Dissemination. In Proceedings of USENIX Technical Conference, 1995, pp. 177-186.

[Yang 99] Yiming Yang, Xin Liu. A Re-Examination Of Text Categorization Methods. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 1999, pp. 42-49.

[YHO 02] Yahoo! http://www.yahoo.com

[Zhu 99] Xiaolan Zhu, Susan Gauch, Lutz Gerhard, Nicholas Kral, Alexander Pretschner. Ontology-Based Web Site Mapping For Information Exploration. In Proceedings of the 8th International Conference On Information Knowledge Management (CIKM), 1999, pp. 188-194.