Information Filtering Using the Dynamics of the User Profile

Costin Barbu, Marin Simina

Electrical Engineering and Computer Science Department
Tulane University
New Orleans, LA, 70130
{barbu, simina}@eecs.tulane.edu

Abstract

This paper presents an adaptive algorithm for learning the user profile. The user profile is learned incrementally and continuously based on user's initial profile, his actions and on semantic interpretation of queries using hypernyms extracted by WordNet. A novel model, *time - words vector hyperspace*, is introduced in order to keep track of the user's interests changes. This new model is achieved by adding a temporal dimension to the classical vector hyperspace model. The results of the retrieval experiments using this new algorithm show an improved effectiveness over the current information retrieval techniques.

Keywords.
User profile, information filtering, information retrieval

1. Introduction

In this paper we investigate the role of the user profile in information filtering and we introduce a novel algorithm for learning the user profile.

Information search on the WWW may become a frustrating activity when a search engine returns thousands of documents for a given query. One way to prune irrelevant documents is to take advantage of the user's implicit interests to filter the documents returned by the search engine, or to reformulate the query based on these interests. We can keep track of the user's interests by building an individual user profile and evolving it over time. The issue is to identify what parts (areas of interest) of the user profile are relevant in the current search context.

In this work we propose an adaptive algorithm for learning the changes in user interests. The user profile is learned incrementally and continuously based on his initial profile, his actions and on semantic interpretation of queries using hypernyms extracted by WordNet[1]. In information retrieval, one of the common representations of the documents (and queries) is based on vector hyperspace model (Salton & McGill 1993). We extend the model for the purpose of information filtering by taking into account the user current interests and their decay in time (if interests change). The resulting model, *time - words vector hyperspace*, computes the dynamics of the user profile.

Each dimension of the vector space, but one (the temporal dimension), represents a word and its weight calculated using the classical TF-IDF technique (Salton & McGill 1993). In this space the documents are represented as vectors, having the word-components computed using TF-IDF and the temporal dimension set to zero. Queries are represented as feature vectors but in addition to the TF-IDF weights they have the temporal dimension set to a preset positive initial value that decays in time.

The rest of the paper is organized as follows. Section 2 presents related work and its limitations. Section 3 introduces the user profile learning algorithm. Section 4 discusses experimental results and finally conclusions and future work are approached in the last section.

2. Related Work

Previous work investigated various approaches to learn the user's interests. WebMate is an intelligent agent that keeps track of the user interests while he is surfing the Internet (Chen & Sycara 1998). The user's profile is learned through multiple TF-IDF feature vectors. Categories of interests are learned based on the user's positive feedback. As long as the number of domains is below its upper limit, a new domain category is learned for every feedback. When the maximum limit has been reached, the document to be learned will be employed to change the vector with the greatest similarity. WebMate introduces a "trigger-pair" model to refine the document search.

INFOS is a system that learns automatically by adapting its user model (Mock 1996). The user interests in different domains are represented by feature vectors. Keyword-based and knowledge-based techniques are employed for feature vector manipulation.

The accuracy over keyword approach is improved by the hybrid approach. It also supports domain knowledge and retains the system's scalability.

[1] WordNet is an online lexical reference system available at: http://www.cogsci.princeton.edu/cgi-bin/webwn1.7.1

Balabanovic (1997) proposes an adaptive agent for Web browsing. The user profile is represented by a single feature vector weighted using the TF-IDF technique. The vector weight is increased or decreased based on the explicit positive or negative user's feedback.
Neural network techniques have been used to learn user's profile in papers of many authors such as: Wiener et al (1995), McElligot & Sorensen (1994) or Tan & Teo (1998). Other authors explored genetic algorithms to learn user interests by incremental relevance feedback in NewT (Sheth 1993), and Amalthea (Moukas, A. & Zacharia G. 1997). Widyantoro (1999) developed Alipes, an intelligent agent that learns user's interests and provides personalized news articles retrieved from the Internet.
Although most of the mentioned works deal with learning user's profile, they do not emphasize on the adaptation of their systems to the changing of the user interests, except for the work of Widyantoro. Nevertheless the dynamics and the rate of change of the user interests were not addressed in previous work.
These problems have been addressed by our adaptive algorithm for learning the changes in user interests based on his initial profile, his actions, queries semantic interpretation and on a novel concept for tracking and analyzing dynamics of the user profile: time - words vector hyperspace.

3. Modeling and Learning the User Profile Dynamics

Contextual relevant information, including user profile, has a critical role in information filtering. In this section we shall introduce a new algorithm for dynamic learning of user interests based on his initial profile, his actions and on queries semantic analysis.
Each dimension of the time – words vector hyperspace, but one (the temporal dimension), represents a word. Its weight is calculated using one of the classical techniques: Term Frequency – Inverse Document Frequency (TF-IDF). As mentioned, the documents are represented as vectors with word-components computed using the TF-IDF technique, but the temporal dimension is set to zero.
In this space, queries are represented as TF-IDF feature vectors with an additional temporal dimension (current interest weight) set to a preset positive initial value that decays in time. This fact implies that some specific user interests could decrease as time goes on. However, user interest for a category can be maintained/increased if the user is searching for elements belonging to an already existing category in his profile.

We have modeled the user behavior by developing an adaptive algorithm for dynamic learning of the user profile based on implicit-only user's feedback. This algorithm uses WordNet to enhance the semantic analysis of the queries whenever this is possible.

WordNet is an online lexical reference system developed by the Cognitive Science Laboratory at Princeton University. Its design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives and adverbs are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets. WordNet also provides various senses for a given word and their corresponding hypernyms. A complete sequence of hypernyms starting from one of the senses of a word has been defined in this paper as a "*hypernym chain*".

The scheme proposed in this work keeps track of both the user's Recent and Long-Term Profiles. The input of the algorithm is an explicit or implicit query and the output is one or more triplets (Category C_i, Current Interest Weight W_i, Rate of Interest Change α_i). The user's Long-Term and Recent Profiles are represented by two queues with similar structures, but the Long-Term Profile queue has a larger capacity than the Recent Profile queue. The recent interest categories are added at the rear of the Recent Profile queue (as shown in the example from Figure 1) and stored in the queue as long as the Current Interest Weight W_i is positive. As W_i becomes negative, the corresponding triplet (C_i, W_i, α_i) is moved to the rear of the Long-Term Profile queue. The same action takes place if the Recent Profile queue reaches its capacity. When the Long-Term Profile queue is at its capacity, the triplet (C_i, W_i, α_i) from the front of the queue is deleted. We consider that the Current Interest Weight decays linearly within the Recent Profile period of time and exponentially in the Long-Term Profile time interval. The Rate of Interest Change (α_i) is computed using the cosine similarity between two sequential query feature vectors Q_i and Q_{i-1} as follows

$$\alpha_i = \frac{Q_i \cdot Q_{i-1}}{|Q_i| \times |Q_{i-1}|} \quad (1)$$

Recent Profile

music	show	sport	food	Categories
100	90	85	70	Current Interest Weight
0.75	0.55	0.30	0.10	Rate of Interest Change

Figure 1. User Recent Profile representation

The algorithm introduced in this paper is modeling the user behavior during his information search activity. The user can either input an explicit query (when he types a set of keywords) or he can narrow his search process when he clicks the links on the displayed web page; then he can scroll down during the reading process in case the web page is of interest for him or he clicks on a different link. In the latter situation an implicit query could be inferred based on the user's actions. In case the user does not find what he is looking for, he can type another query and the search process goes on. Therefore two algorithms have been developed, for explicit or implicit queries.

3.1 Learning User Profile from Explicit Queries

We assume that the user has provided a preliminary profile, his Long Term Profile has at most ***L*** domains of interest and his Recent Profile has ***R*** domains of interest. Assume the preset number of elements of a vector is ***M***.

The algorithm (**LearnUserProfileExplicit**) for learning the user profile from explicit queries is defined as follows.

Input: explicit query EQ_i
Output: updated user profile P
LearnUserProfileExplicit(EQ_i) => profile P

1. For each query $EQ_i = \{t_{1i}, t_{2i}, t_{3i}, \ldots, t_{ki}\}$, where k = 1…M and t_{ki} are the keywords of query EQ_i
2. Compute the Rate of Interest Change α_i between EQ_i and EQ_{i-1}
3. For each keyword t_{ki}
4. Extract the hypernym chains HC_{ki} for all senses of t_{ki} from WordNet.
5. Do the intersection of the hypernym chains from step 4 with each of the categories' hypernym chains from the Recent Profile. If this intersection is not void then continue with step 6. Else continue with step 7.
6. Select the sense whose hypernym chain HC_{ki} intersected the Recent Profile categories' hypernym chains closest to the keyword's sense, and consider the HC_{ki} corresponding to the selected sense.
7. Do $N_i = \bigcap_k HC_{ki}$ for all keywords t_{ki} of query EQ_i
8. Extract θ from HC_{ki} (where θ is a threshold set of words, i.e. the root and the next level child (hyponym) from the tree lexical structure of WordNet)
9. If $size(N_i) > size(\theta)$ then
   9.1 Extract the closest word from N_i to a keyword t_{ki}
   9.2 Insert it to the Recent Profile as a Category C_i, together with the Current Interest Weight W_i (preset to a positive initial value W) and with the Rate of Interest Change α_i
   9.3 If Rate of Interest Change $\alpha_i > \alpha_{threshold}$ (say $\alpha_{threshold}$ is 0.6) then increase the Current Interest Weight of Category C_i with a positive value ΔW : $W_i = W + \Delta W$
10. Else
   10.1. For all keywords t_{ki} of query EQ_i
   10.2. Do $T_i = HC_{ki} \cap C_j$, where C_j are existing categories from Recent Profile and HC_{ki} have been selected at step 6.
   10.3. If T_i is void then add t_{ki} to the Recent Profile as a new Category C_i, together with the Current Interest Weight W_i (preset to a positive initial value W) and with the Rate of Interest Change α_i
   10.4. Else increase the Current Interest Weight of C_j to the preset positive initial value W.
11. Sort the triplets (C_i, W_i, α_i) from the Recent Profile in ascending order of the Current Interest Weight W_i.
12. Return Updated User Profile.

Note that steps 3 through 6 have been considered in order to better discriminate among possible polysemantic keywords t_{ki} of query EQ_i taking into account the contextual search environment.

3.2 Learning User Profile from Implicit Queries

We shall introduce the algorithm for learning the user profile from implicit queries, as following.

Input: user actions
Output: updated user profile P
LearnUserProfileImplicit(IQ_i) => profile P

1. For each link mouse click do:
2. Preprocess: parse HTML page, deleting the stop words, stemming the plural noun to its single form and inflexed verb to its original form.
3. Extract the words in title as a vector V_{Ti}, and the words in the section titles as a vector V_{STi}
4. Extract the vector V_{Di} for this document using the TF-IDF technique.
5. Compute the implicit query feature vector IQ_i
   $IQ_i = w_1 \cdot V_{Ti} + w_2 \cdot V_{STi} + w_3 \cdot V_{Di}$
   where w_1, w_2, w_3 are weights set to initial values such that $w_1 > w_2 > w_3$
6. Update IQ_i according to user's behavior: if the user scroll down the document for a period of time shorter than an average reading time then w_3 could be increased such that the above inequality holds, since the user has some interest about its content. On the other hand if the user scroll down the document for a period of time longer or equal than an average reading time then he has a real interest on the document and w_3 should be assigned with a greater value than in previous cases; hence $w_3 > w_1 > w_2$
7. Call *LearnUserProfileExplicit(IQ_i)* to learn the profile from implicit query IQ_i.
8. Return Updated User Profile.

---

4. Experimental Results and Discussion

4.1 Experiments for Learning User Profile from Explicit Queries

Example 1

Let's assume the user is typing the Query 1 and his recent profile has the following categories of interests:

Recent Profile = {vehicle, art, sausage, pastry};
Query 1= {Shrimp, Chardonnay, Onion, Dressing};

WordNet provides various senses for each keyword:
3 senses of "Shrimp": - small person
- seafood
- decapod crustacean
2 senses of "Chardonnay": - vinifera grape
- white wine
3 senses of "Onion": - bulb
- alliaceous plant
- vegetable

7 senses of "Dressing": - sauce
- concoction, mixture
- enrichment
- cloth covering
- conversion
- covering
- medical care, medical aid

A hypernym chain example returned by WordNet for sense 3 of keyword "onion" is shown in Figure 2.

Sense 3 of word "Onion"
onion
=> vegetable, veggie
=> produce, green goods, green groceries
=> food
=> solid
=> substance, matter
=> entity, physical thing

Figure 2. Hypernym chain for sense 3 of keyword "onion" as provided by WordNet

The following results are achieved by applying the algorithm for learning the user profile from explicit queries. According to the steps 3, 4, 5, 6 of the algorithm, the hypernym chains for all the senses of keywords from query have been intersected with the hypernym chains of the categories from the Recent Profile and the following keywords hypernym chains have been selected:

Sense 2 of word "Shrimp"
prawn**,shrimp** => seafood => food => solid => substance, matter => entity, physical thing

Sense 2 of word "Chardonnay"
Chardonnay, Pinot Chardonnay => white wine => wine, vino => alcohol, alcoholic beverage, intoxicant, inebriant => beverage, drink, drinkable, potable => food, nutrient => substance, matter => entity, physical thing

Sense 3 of word "Onion"
onion => vegetable, veggie => produce, green goods, garden truck => food => solid => substance, matter => entity, physical thing

Sense 2 of word "Dressing"
stuffing, **dressing** => concoction, mixture, intermixture => foodstuff, food product => food, nutrient => substance, matter => entity, physical thing

The category **Food** is extracted from **Query 1** and added to the Recent Profile according to the steps 7, 8 and 9 of the algorithm.

Example 2

In this example the user has a different profile and he inputs Query 2.
Recent Profile = {vehicle, art, sausage, pastry, food}
Query 2 = {Skating, Mathematics, Anecdote};
WordNet outputs only one sense for each of the keywords of Query 2 and the following hypernym chains:

Sense 1 of Skating
skating => sport, athletics => diversion, recreation => activity => act, human action, human activity

Sense 1 of Mathematics
mathematics, math, maths => science, scientific discipline => discipline, subject, subject area, subject field, field, field of study, study, bailiwick, branch of knowledge => knowledge domain, knowledge base => content, cognitive content, mental object => cognition, knowledge, noesis => psychological feature

Sense 1 of Anecdote
anecdote => report, account => informing, making known => speech act => act, human action, human activity

By applying the algorithm for learning the user profile from explicit queries the hypernym chains for all the senses of keywords from query have been intersected with the hypernym chains of the categories from the Recent Profile. Although no intersection has been found with the existing categories' hypernym chains, the next step we take is doing the intersection of the hypernym chains of the keywords from query, according to the steps 7 of the algorithm. Since $size(\boldsymbol{N_i}) < size(\theta)$ at step 9, we jump to step 10 and compute the set $\boldsymbol{T_i}$. After all these steps, $\boldsymbol{T_i}$ has been found to be void and all the keywords from Query 2 should be added to the Recent Profile as new categories, according to step 10.3 of the algorithm.

Recent Profile = {vehicle, art, sausage, pastry, food, skating, mathematics, anecdote}

4.2. Information Filtering Based on User Profile

Contextual relevant information improves the search performance by filtering the retrieved documents in descending order of the Relevance Score. This Relevance Score can be computed as the cosine similarity between the User Recent Profile feature vector and the feature vectors of the documents retrieved by a classical search engine (i.e. Google). Our preliminary results show that the quality of information filtering based on user recent profile is dramatically improved by taking into account the rate of user's interest change and the polysemantic disambiguation of the query's keywords. More documents relevant to the current interest of the user are retrieved. Sample results are presented in the Table 1.

4.3. Discussion

Another approach of the algorithm for learning the user profile from explicit queries (*LearnUserProfileExplicit*) could be the following.

Assume an explicit query EQ_i has been input by the user, $EQ_i = \{t_{1i}, t_{2i}, t_{3i}, \ldots, t_{ki}\}$, where k = 1…M and t_{ki} are the keywords of the query. Let's consider that steps 1 thru 7 of the algorithm *LearnUserProfileExplicit* have been already performed and a "clustering" threshold has been set at 30%.

	No Profile Filtering	Total Profile Filtering	Recent Profile Filtering
Total Number Documents Retrieved	73200	51	4436
Relevant Documents in Top 10 Retrieved	6	2	10
Accuracy in Top 10 Documents Retrieved	60 %	20 %	100 %

Table 1. Comparison of information filtering methods

If say at least 70 % of the keywords have hypernym chains that intersect each other and form "clusters" around N different categories, then the rest of the keywords (less than 30 %) from EQi that do not belong to any of the N clusters could be considered "noise" and be ignored. An example of this situation is presented in Figure 3, where the explicit query has 5 keywords. Two categories are extracted from this query (seafood and music) and added to the user's Recent Profile whereas the keyword plane is considered "noise".

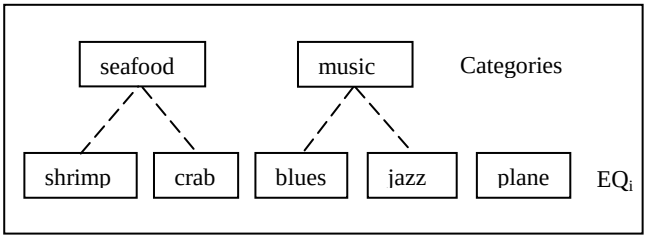

Figure 3. Keywords clustering representation

5. Conclusions

In this paper we presented an adaptive algorithm for learning the changes in user interests based on his initial profile, his actions and on semantic interpretation of queries. We introduced a novel concept, Time - Words Vector Hyperspace to computationally model the rate of interest change and the dynamics of the user profile. We also added adaptive polysemantic disambiguation of the user's query using WordNet. Since our algorithm does not rely on semantic disambiguation for short queries, we avoid the performance degradations mentioned by Voorhees (1993). Our preliminary results show a significant improvement of filtering by employing the user recent profile as opposed to existing approaches (e.g. Chen & Sycara 1998, Balabanovic 1997, Widyantoro et al. 1999) that consider the total profile. Our implementation currently does not handle queries with brand name keywords (i.e.: Sun, computer maker vs. sun, star) since WordNet does not include them. We can overcome this situation in our future work by building and integrating with WordNet a specialized ontology that includes brand names.

References

Balabanovic, M. 1997. An Adaptive Web Page Recommendation Service. In Proceedings of the First International Conference on Autonomous Agents. 378 – 385, New York. N.Y.: ACM

Chen, L., and Sycara, K. 1998. WebMate: Personal Agent for Browsing and Searching. In Proceedings of the Second International Conference on Autonomous Agents, 132-139. New York. N.Y.: ACM

McElligot, M. and Sorensen, H. 1994. An Evolutionary Connectionist Approach to Personal Information Filtering. In Proceedings of the Fourth Irish Neural Network Conference, 141-146, Dublin, Ireland.

Mock, K. J. 1996. Hybrid-Hill-Climbing and Knowledge-based Techniques for Intelligent News Filtering. *In Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*. 48-53. Menlo Park, California: AAAI Press

Moukas, A. and Zacharia G. 1997. Evolving a Multiagent Information Filtering Solution in Amalthea. In Proceedings of the First International Conference on Autonomous Agents, 394-403. New York, N.Y.: ACM

Salton, G., and McGill, M. J. 1993. *Introduction to Modern Information Retrieval*. New York. N. Y.: McGraw-Hill.

Sheth, B. D. 1993. A Learning Approach to Personalized Information Filtering. M.S. diss., Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.

Tan, A. and Teo, C. 1998. Learning User Profile for Personalized Information Dissemination. In Proceedings of 1998 International Joint Conference on Neural Networks, 183-188, Anchorage, AK: IEEE.

Voorhees, E.M. 1993. Using WordNet to Disambiguate Word Senses for Text Retrieval. In Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, 171 -180, Pittsburgh, PA

Widyantoro, D. H., Yin J., El Nasr, M., S., Yang, L., Zacchi, A. and Yen J. 1999. Alipes: A Swift Messenger in Cyberspace. In Proceedings of the Spring Symposium on Intelligent Agents in Cyberspace, 62-67, Palo Alto, CA.

Wiener, E., Pederson, J. and Weigend, A. 1995. A Neural Network Approach to Topic Spotting. In Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval, 317-332, Las Vegas, NV.