Learning Browsing Patterns for Context-Aware Recommendation

Daniela Godoy and Analía Amandi

ISISTAN Research Institute, UNICEN University
Campus Universitario, CP 7000, Tandil, Argentina
Also at CONICET, Argentina
{dgodoy,amandi}@exa.unicen.edu.ar

Abstract. The success of personal information agents depends on their capacity to both identify relevant information for users and proactively recommend context-relevant information. In this paper, we propose an approach to enable proactive context-aware recommendation based on the knowledge of both user interests and browsing patterns. The proposed approach analyzes the browsing behavior of users to derive a semantically enhanced context that points out the information which is likely to be relevant for a user according to its current activities.

1 Introduction

The main goal of personal information agents is to present relevant information to users based on the knowledge of their interests. In order to enable the adaptation and personalization of information delivered to users, personal agents learn and represent long-term interests into user profiles. Thus, user profiling addresses the issue of modeling interests to determine the relevance of a new, previously unseen piece of information.

In addition to merely choosing the right information, personal agents should also be aware of the user context in order to provide information in the time and in the place it is more relevant to users. User profiles are frequently seen as a way to disambiguate search topics. Even though this use of profiles supports interactive context-aware information retrieval in which relevant documents are gathered upon a direct user request, because of the lack of knowledge about the active goals, it fails at supporting proactive context-aware retrieval in which relevant documents are presented to users according to their activities [3].

In order to enable proactive context-aware information retrieval and recommendation, user behavior patterns as regards interests have to be explicitly modeled into profiles. The extraction of such patterns is fostered by semantically enriched profiles which provide a hierarchical organized view of the concepts a user is interested in. Either ontology-based profiling [8] or conceptual clustering [7] allow agents to obtain these hierarchies starting from examples.

In this paper, we present an approach to augment a hierarchical representation of user interests obtained by conceptual clustering with user behavior patterns extracted from observing the browsing activity. This enables proactive

and adaptable behavior of personal agents which become able to predict and anticipate user information needs. Section 2 describes this approach. Experimental results are summarized in Section 3. Section 4 compares this work with related ones. Finally, concluding remarks are stated in Section 5.

2 Learning and Using Browsing Patterns

The browsing behavior of users is an important resource for inferring contextual information. It can be seen as a sequence of activities that are related to one another not only through evolving information interests that can be described at conceptual level, but also through proximity in time. By activity it is understood a page visit which takes place during the course of browsing, while groups of these activities can be referred to as sessions.

Information agents can take advantage of the knowledge gained from observing user browsing in conjunction with long-term user interests to retrieve context-relevant information. If an agent detects the user is browsing through certain interest categories, it can anticipate the categories in the same session the user is likely to be interested in. The goal of activity-awareness is, therefore, to proactively retrieve Web pages matching the user interests and compute a set of recommendations for the current or active user session.

To accomplish this goal, browsing patterns referring to categories in the user profile that are usually accessed together serve as the basis for recommendation and are mined starting from observation of frequent associations among browsing activities. For extracting navigational patterns, the existence of a conceptual hierarchy constituting the user profile is assumed, so that it can be used to characterize Web pages, i.e. to describe pages in terms of interest categories.

A conceptual clustering algorithm that carries out incremental, unsupervised concept learning over Web documents was used in this work to obtain such hierarchical descriptions of user interests. However, other approaches can be applied within this framework. Hierarchies of concepts produced by this algorithm, named *WebDCC* [7], are classification trees in which internal nodes represent concepts and leaf nodes represent clusters of examples.

In user profiles, browsing habits are represented by association of the form $A \Rightarrow B$, where A and B are groups of categories and the association indicates that, if the user current activities include visiting pages about the categories in A, the next activities are likely to include visiting pages about B.

2.1 Client-Side Sessionization

A browsing session is a set of page references that takes place during one logical period, e.g. the sequence of page accesses that takes place from a log in to a log out of the browser. By identifying the session boundaries, it is ensured that the information collected from one session is within the same context, which provides a good foundation for inferring and applying context in recommendation.

In contrast to Web usage mining, which focuses on extracting patterns of multiple users within server logs, a more accurate and complete picture of a user Web activity can be obtained from client side data. User actions can be

recorded in an activity log by applications monitoring Web browsers. Thus, the content of Web pages, the access time, the time spent on each page and other information is available for analysis. Furthermore, actions such as opening or closing the browser can be used to start and finish browsing sessions.

From client-side observation, it is possible to reliably recognize sessions in the user activity log to evaluate the user interests as well as to understand user frequent browsing patterns. A session S_j is a list of pages a user accessed to ordered by time-stamp as follows:

$$S_j = \{(p_1, time_1), (p_2, time_2), \ldots, (p_n, time_n)\}$$

where $time_i$ is the time the user accessed the page p_i such that $time_i \leq time_j, \forall i \leq j$. Then, the user browsing activities are partitioned into a set of sessions $S = \{S_1, S_2, \ldots, S_k\}$ containing individual page references.

The process of segmenting the activity of a user into sessions is performed using a time-oriented heuristic in which a time-out establishes a period of inactivity that is interpreted as a signal that the session has ended. If the user did not request any page for a period longer than *max_time* (30 min. is used as default time-out) subsequent requests are considered to be in another session. In addition, the active session is finished when the browser is closed and a new session is started when the browser is re-opened.

2.2 Transaction Identification

The notion of session can be further abstracted by selecting a subset of pages that are significant or relevant for analysis. Each semantically meaningful subset of pages belonging to a user session is referred to as a transaction. Transaction identification assumes that user sessions have already been identified. Hence, the input to this process consists in the page references for a given user session. In Web usage mining there is no convenient method of clustering page references into transactions smaller than an entire user session [6].

To identify semantically meaningful transactions, content pages are considered as those belonging to one or more categories in the profile, unlike content pages in other approaches which are identified simply based on the time spent on a page or on backtracking during the user navigation [5]. Pages not belonging to any category in the profile are considered irrelevant for usage mining since they do no entail information about the user habits regarding interests. Then, a content-only transaction is formed by all the content pages in a session. Figure 1 illustrates the formation of these transactions.

The resulting transactions are further divided using the time window approach, which divides each transaction into time intervals no longer than a specified threshold. This approach assumes that meaningful transactions have an overall average length associated with them. For a large enough specified time window, each transaction will contain an entire user session. If W is the length of the time window, then two pages p_i and p_j are in the same session if:

$$p_i.time - p_j.time \leq W$$

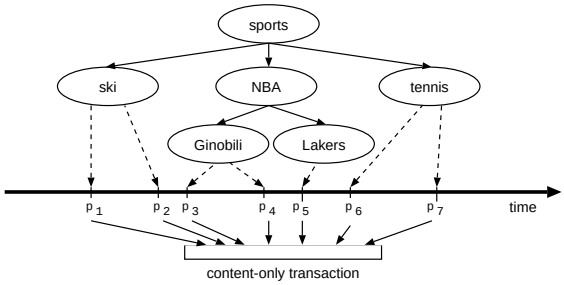

Fig. 1. Example of a content-only transaction

In this way, the set of pages $P = \{p_1, p_2, \ldots, p_n\}$, each with its associated $time_i$, appearing in the set of sessions S are partitioned into a set of m user transactions $T = \{t_1, t_2, \ldots, t_m\}$ where each $t_i \in T$ is a subset of P. The problem of mining association rules is defined over these collection of subsets from the item space where an item refers to an individual page reference.

To incorporate the knowledge of the user interests in pattern extraction, further processing of user activities is needed to map individual Web page references to one or more user interest categories. The enriched version of transactions leads to set of rules that includes categories. Thus, recommendations can be broaden to include any Web page belonging to the involved categories. To integrate content and usage data, each page p_i in a transaction t_j is considered to have an associated set of categories it belongs to, denoted $C_i = \{c_1, c_2, \ldots, c_p\}$, where C_i is extensionally defined by all the categories c_j in the path from the root of the hierarchy to the leaf cluster in which the page p_i was classified into.

If only the cluster a page belongs to is used to describe sessions, the discovered association rules will relate clusters but not categories. Instead, the inclusion of the ancestors in the path from the cluster the page was classified into until the root, makes it possible to find rules at different levels. The result of replacing the elements of the transactions in T by categories in the user profile is a set of transactions $T' = \{t'_1, t'_2, \ldots, t'_m\}$ where each $t'_i \in T'$ is a subset of C. The algorithm for transaction identification can be outlined as follows:

1. For each session $S_i \in S$, create a new transaction t_i in T
2. For each page $p_j \in S_i$, find the set C_j by classifying the page into the current user interest hierarchy
3. If $C_j \neq \emptyset$, add p_j to the transaction t_i since the page is a content page
4. Repeat steps 2 and 3 until all page references have been either added to the transaction or discarded
5. Repeat steps 1 to 4 until all sessions in S have been processed
6. Use the time window approach to partition each $t_i \in T$ into transactions smaller than W
7. For each resulting transaction $t_i \in T$, create the transaction t'_i in T' replacing each page $p_j \in t_i$ by the corresponding C_j

2.3 Mining Association Rules

The association rule mining problem was stated in [1]. Let $\mathcal{I} = \{I_1, I_2, \ldots, I_m\}$ be a set of literals called items, a subset $X \subseteq \mathcal{I}$ is called an itemset and a k-itemset is an itemset that contains k items. Let $\mathcal{D}$ be a database of transactions, where each transaction T is a set of items such that $T \subseteq \mathcal{I}$. Each itemset has a certain statistical significance called support such that an itemset has *support* s in the transaction set $\mathcal{D}$ if $s\%$ of the transactions in $\mathcal{D}$ contain X. An association rule is an implication of the form $X \Rightarrow Y$, where $X \subset \mathcal{I}$, $Y \subset \mathcal{I}$, and $X \cap Y = \emptyset$. The rule $X \Rightarrow Y$ holds in the transaction set $\mathcal{D}$ with *confidence* c if $c\%$ of the transactions in $\mathcal{D}$ that contain X also contain Y.

The problem of mining association rules in $\mathcal{D}$ consists in finding all rules $X \Rightarrow Y$ that have support greater than a user-specified minimum support, called *minsup*, and confidence, called *minconf*. For each rule, the support threshold describes the minimum percentage of transactions containing all items that appear in the rule, whereas the confidence threshold specifies the minimum probability for the consequent to be true if the antecedent is true.

In a hierarchical description of user interests, associations or access patterns may contain interesting regularities at different levels of abstraction including categories or clusters that are related according to the user habits. The problem of mining multiple-level or generalized association rules assumes a hierarchy or taxonomy $\mathcal{T}$ on the items instead of a flat itemset $\mathcal{I}$. A generalized association rule is an implication of the form $X \Rightarrow Y$, where $X \subset \mathcal{I}$, $Y \subset \mathcal{I}$, $X \cap Y = \emptyset$ and no item in Y is an ancestor of any item in X as this would be a trivially valid association. The rule $X \Rightarrow Y$ holds in the transaction set $\mathcal{D}$ with confidence c and support s if $c\%$ of the transactions in $\mathcal{D}$ that support X also support Y and $s\%$ of transactions in $\mathcal{D}$ support $X \cup Y$. These rules are called generalized association rules because both X and Y can contain items from any level of the taxonomy $\mathcal{T}$.

If the problem of determining if a transaction T support an itemset X is considered, for each item $x \in X$ it is necessary to check whether x or some descendant of x is present in the transaction. To simplify this task, all the ancestors of each item in T are added to this transaction to form an extended transaction T'. A straightforward method to find generalized association rules is to run any association rule algorithm on the extended transactions since T supports X if and only if T' is a superset of X. For empirical evaluation of the proposed approach, we used the *Apriori* algorithm over the set of extended transactions obtained as pages are classified in the concept hierarchy.

2.4 Activity-Based Recommendation

From user browsing sessions, patterns representing the user navigational behavior are extracted in the form of association rules, which relate sets of categories or concepts in the user profile. Information agents, therefore, become able to proactively retrieve relevant information to generate recommendations for a user by matching the current user activity against the discovered patterns.

To gather a set of possible recommendations, an agent can perform a Web search to retrieve pages belonging to the concepts the user is interested in. For example, the agent can retrieve pages from some fixed sites (e.g. a newspaper Web site) or find the nearest neighbors of a page in the profile used as query.

A fixed-size sliding window is used over the active session to capture the current user activity. For a sliding window of size n, the active session ensures that only the last n visited pages influence recommendation. The use of a window is important in discovering context since most users go back and forth while browsing to find the desired information so that earlier portions of the browsing history may refer to no longer valid information needs.

In the recommendation phase, the active session is compared with the discovered rules. If the active session matches the antecedent of an association rule, recommendations are finding by retrieving Web pages belonging to the categories in the rule consequent.

3 Experimental Results

To evaluate the activity-base recommendation approach, a client-side log of visited Web pages in a number of topics a user is interested in and the location of the pages on the user interest hierarchy are needed. Unfortunately, available datasets belong to individual Web sites and record the accesses of several users.

In the absence of client-side data, the content and logs of the *Music Machines*[1] Web site were used for experimentation. In these logs users are anonymized with respect to originating machine, i.e. all hits from one machine on a particular day have the same label. Thus, the browsing behavior of individual users can be interpreted respecting their interest categories within the site. *Music Machines* contains 4582 distinct pages about various kind of electronic musical equipment grouped by manufacturers.

Each access log consists of the user label, request method, accessed URL, data transmission protocol, access time and browser used to access the site. The server logs were filtered to remove those entries that are irrelevant for analysis and those referring to pages that do not exist in the available site copy.

From all users who entered the site after 20/8/98, when the copy of the site was made, the five users having the longest sessions were selected. Then, the experimental procedure simulates users browsing the *Music Machines* site and obtaining recommendations. For each user a profile was built based on both the content of Web pages from the site and the user behavior regarding interest categories. Experiments for each user proceed as follows:

1. Identify the user entries in the log files
2. Extract the URLs of the visited pages and run *WebDCC* algorithm over these pages using the available copy of the *Music Machines* site
3. Identify user sessions in the logs using *max_time*=30 minutes
4. Partition user sessions into transactions and mapping Web page references to categories in the profile

[1] http://www.cs.washington.edu/ai/adaptive-data/

ID	*duration*	*# entries*	*# filtered entries*	*# sessions*	*# pages*	*# clusters*	*# filtered clusters*
1	11:55:29	31404	938	9	1669	115	100
2	23:19:57	3639	2511	23	2427	229	145
3	13:18:47	3511	589	11	1663	129	91
4	21:28:38	3347	1087	10	1782	176	97
5	12:22:52	2862	2114	14	2851	312	176

Table 1. Summary of user data and experimental results

5. Divide the resulting set of transactions into a training (approx. 70%) and a testing set (approx. 30%) for experiments
6. Use the training set to mine association rules regarding categories
7. Use the testing set to simulate active session windows and recommend pages
8. Evaluate the recommendations in terms of precision and coverage

To assess quantitative values of recommendation performance, we used the adaptations of precision and coverage measures proposed by [9]. Given a transaction t and a set of recommendations R produced using a window w such that $w \subseteq t$, the precision and coverage of R with respect to t are defined as:

$$precision(R,t) = \frac{|R \cap (t-w)|}{|R|} \qquad coverage(R,t) = \frac{|R \cap (t-w)|}{|t-w|}$$

Thus, precision measures the degree to which recommendations are accurate for the active session and coverage measures its ability to recommend all the items that are likely to be visited by the user in the active session.

For a given transaction t in the testing set and an active session window of size n, we randomly chose $|t| - n + 1$ groups of items, each having size n, from the transaction as the surrogate active session windows. For each of these active sessions, a set of recommendations are produced based on the extracted rules. The recommendations are compared to the remaining items in the transactions, i.e. $t - w$, to compute performance measures. For each measure, the final score of the transaction t is the average over all of the $|t| - n + 1$ surrogate active sessions associated with this transaction.

The entries remaining after cleaning the logs were used to extract the documents each user accessed in the site. Table 1 summarizes the number of entries, sessions and unique pages accessed in the site. *WebDCC* algorithm was run over the documents each user accessed to identify the interest categories. These documents were partitioned into several clusters although no concepts were extracted by the clustering algorithm. This was mainly due to the site content and structure. It contains few pages referred to many manufacturers, so that different clusters are created for each of them and no generalization is possible.

From the total number of clusters resulting from running the algorithm, meaningless clusters containing a single instance were filtered out before association rule mining. Then, the pages in each session were partitioned into

W	# transactions	# items	# rules	# recom.	precision	coverage
3	120.20 ± 92.96	13.85 ± 3.62	247.20 ± 199.11	2.34 ± 2.06	75.79 ± 17.31	6.84 ± 6.85
			15423.00 ± 17533.67	3.87 ± 3.61	68.94 ± 20.90	10.13 ± 8.71
5	101.80 ± 98.05	19.73 ± 4.60	486.40 ± 409.05	4.87 ± 5.24	66.15 ± 21.81	8.67 ± 8.20
			16135.80 ± 15442.15	6.10 ± 5.61	59.73 ± 17.71	10.51 ± 8.49
10	45.60 ± 31.19	32.97 ± 6.41	920.00 ± 567.92	5.84 ± 5.48	63.97 ± 18.36	16.02 ± 13.07
			37223.60 ± 39060.43	7.81 ± 5.35	57.83 ± 13.69	19.34 ± 12.42
15	32.00 ± 20.02	46.73 ± 9.38	1255.60 ± 424.38	9.32 ± 4.04	62.32 ± 15.54	17.11 ± 6.90
			47948.40 ± 8319.91	11.44 ± 2.71	55.96 ± 9.32	17.89 ± 4.08

Table 2. Effect of time window in recommendation

transactions and, in turn, the pages in each transaction were mapped into clusters obtaining a set of content-enhance transactions.

There are several parameters influencing the results of recommendation, including the size of the time window W, the size of the sliding window n, the confidence threshold and the size of the itemsets.

The length of the time window W affects primarily the number of transactions obtained from a given session and, consequently, the number of rules and the quality of recommendations. In the first experiment, we investigated the impact of different values of W on the number of rules and the quality of the resulting recommendations for the five users by testing the values 3, 5, 10 and 15 minutes. In association rule mining, all rules having a support greater than 2% were extracted. For recommendation, we used a fixed active window of size 3 and a minimum confidence of 90%. Table 2 summarizes the average and standard deviation of values obtained for the five users in: number of transactions, items per transaction, extracted rules, recommendations, precision and coverage of recommendations. For each value of W in the table, the the results for rules having 1-itemsets and 2-itemsets are shown.

The higher the value W, the lower the number of transactions and the longer its size in terms of the average number of items. Fewer transactions lead to more rules since they are supported by the data, but the quality of these rules is inferior to the quality of rules extracted when more information is available. Indeed, the precision in recommendation decreases from $W = 3$ to $W = 15$. For further experiments we set $W = 3$ min. since the dropping in precision for the immediately next value $W = 5$ is significant (approx. 9%), but the improvement in coverage is rather small (approx. 2%).

The results of 1-itemsets and 2-itemsets, on the other hand, show the same relationship between precision and coverage. The values of precision diminish when 2-itemsets are considered, increasing the coverage of recommendations. In this case, not only the improvement in coverage can be considered small given the loss in precision, but also the number of rules rises drastically. The on-line analysis of such high number of rules becomes too expensive.

To investigate the effect of window size, the portion of the active window session used to produce recommendations, experiments were performed using

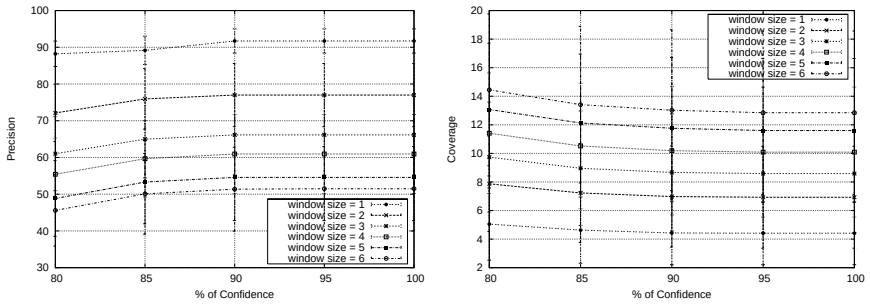

Fig. 2. Impact of window size on precision and coverage of recommendations

window sizes from 1 to 6. Figure 2 shows the impact of window size on precision and coverage of recommendations. In the figures, the results summarize the average and standard deviation of the scores achieved for the five users involved in the experiments varying the confidence threshold.

From both figures, it can be concluded that the smaller the size of the time window, the higher the precision of recommendation and the lower its coverage. Indeed, the best precision was obtained with a window of size one, but the coverage of recommendations was the poorer too. As the size of the window is enlarged, there are more rules matching the pages inside the window, increasing the number of recommendations and, consequently, their coverage.

There is a trade-off between enlarging enough the window size to recommend most of the pages that are relevant to the current activity context but not enlarge it too much to start recommending pages that were relevant to the previous context of the user in the browsing session. However, precision can be sacrificed in this decision for the sake of an increase in coverage since a lower precision means that the agent is recommending pages which are not contextually relevant but are still content relevant to the user interests.

4 Related Work

Efforts in building user profiles representing user interests have been frequently seen as a method of gathering contextual information since the knowledge contained in a profile persists across retrieval sessions and can be automatically added to queries. However, user profiles by themselves have not means to anticipate information needs given the user current activities and, therefore, do not support proactive context-aware recommendation. In our approach, the current activities act as trigger for retrieval of information matching long-term interests.

WordSieve [2] and *Watson* [4] are systems that use context for information seeking. *WordSieve* is an algorithm to build context profiles which distinguish sets of documents that users tend to access in groups. *Watson* observes the use of standard software tools and generates queries to seek context-relevant information. Instead of retrieving documents based solely on words extracted

from recently consulted documents, our approach extracts information about how users tend to access documents regarding long-term interests to determine what kind of documents are likely to be interesting in a certain context.

The proposed approach differs from Web usage mining techniques in two aspects. First, server-side usage mining provides information about a specific Web site based on correlations among the pages that multiple users have visited. By contrast, our approach extracts rules from the observation of a single user browsing the Web. Second, most Web usage mining approaches obtain association rules that relate single Web pages. By capturing navigational patterns at conceptual level our approach provides more flexibility in recommendation.

5 Conclusions

The user context is an important aspect to take into account in recommendation which, however, has received little attention in personal agents. Most agents are concerned with estimating the interest of new pieces of information, instead of trying to place the relevant information in the right contexts. In this paper, we have described an approach to consider the activity context during profiling to enable context-aware recommendation. Experimental results showed that the extraction of association rules describing browsing patterns at conceptual level helps to predict part of the interests which are relevant to the user in a session.

References

1. R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.
2. T. Bauer and D. Leake. WordSieve: A method for real-time context extraction. In *Proceedings of the Third International and Interdisciplinary Conference on Modeling and Using Context*, pages 30–44, 2001.
3. P. Brown and G. Jones. Context-aware retrieval: Exploring a new environment for information retrieval and information filtering. *Personal and Ubiquitous Computing*, 5(4):253–263, 2001.
4. J. Budzik, K. Hammond, and L. Birnbaum. Information access in context. *Knowledge based systems*, 14(1-2):37–53, 2001.
5. M-S. Chen, J. S. Park, and P. Yu. Efficient data mining for path traversal patterns. *IEEE Transactions on Knowledge and Data Engineering*, 10(2):209–221, 1998.
6. R. Cooley, B. Mobasher, and J. Srivastava. Grouping web page references into transactions for mining world wide web browsing patterns. In *Proceedings of the IEEE Knowledge and Data Engineering Exchange Workshop*, pages 2–9, 1997.
7. D. Godoy and A. Amandi. Modeling user interests by conceptual clustering. *Information Systems*, 31(4-5):247–265, 2006.
8. S. Middleton, N. Shadbolt, and D. Roure. Ontological user profiling in recommender systems. *ACM Transactions on Information Systems*, 22(1):54–88, 2004.
9. B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Discovery and evaluation of aggregate usage profiles for Web personalization. *Data Mining and Knowledge Discovery*, 6(1):61–82, 2002.