Mining User Navigation Patterns for Personalizing Topic Directories

Theodore Dalamagas
School of EC Engineering
National Tech. Univ. of Athens
Greece
dalamag@dblab.ntua.gr

Panagiotis Bouros
School of EC Engineering
National Tech. Univ. of Athens
Greece
pbour@dblab.ntua.gr

Theodore Galanis
School of Informatics
University of Edinburgh
Scotland
theodoros.galanis@gmail.com

Magdalini Eirinaki
Dept of Computer Engineering
San Jose State University
USA
eirinaki@faculty.cmpe.sjsu.edu

Timos Sellis
School of EC Engineering
National Tech. Univ. of Athens
Greece
timos@dblab.ntua.gr

ABSTRACT

Topic directories are popular means of organizing information resources in the web. In this work, we introduce a methodology for personalizing topic directories. The key feature of our methodology is that the personalization is based on the mining of navigation patterns extracted from previous user visits. These patterns, expressed in the form of visited categories and retrieved resources, represent the navigation behaviour and interests of different users or user groups. Our work provides a set of mining tasks for user navigation patterns and a set of personalization tasks that customize the organization of the topic directory according to these patterns for certain user groups.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: On-line Information Services - Web-based services; H.2.8 [**Database Management**]: Database applications - Data Mining

General Terms

Algorithms, Experimentation

Keywords

Personalization, Topic Directories, Navigation Patterns, Sequential Patterns

1. INTRODUCTION

Topic directories are popular means of organizing web content. A topic directory is a structure of thematic categories in which web resources are classified. Users can browse a directory to search for resources relevant to certain topics. Compared to traditional keyword search, which is the most popular way of searching information, a topic directory is quite useful for narrowing searching from a broad category to a more specific one. Moreover, a directory can help users understand how topics are related and assist keyword search by narrowing it within the resources of a certain category. Topic directories can be of specialized or of general interest. For example, the Open Directory Project (ODP) Dmoz (http://dmoz.org) is a popular topic directory of general interest used by Google web Directory (http://directory.google.com/).

The growing diversity of web data sources, and the heterogeneity of the user communities accessing them, imposes the need for personalizing the web content and provided services. For instance, content personalization is acknowledged as a key requirement for future digital libraries [6]. To this extend, personalizing topic directories is a challenging issue. Users should be able to access topic directories which are tailored to their specific preferences and needs.

In this work, we introduce a methodology for personalizing topic directories. The personalization is based on mining *navigation patterns* extracted from previous user visits (also referred to as *user sessions*). Since a topic directory covers a broad and heterogeneous set of web resources, the proposed methodology identifies groups of users with similar interests, named *interest groups*, and targets the mining and personalization tasks to each group individually. We should note, however, that a user may belong to more than one interest group.

More specifically, for each interest group, the system discovers navigation patterns such as frequent visits of popular categories of the directory, back and forth visits to the same categories indicating the so-called "indecisive users", and frequent visits of categories that include popular links to external resources. Note that we focus on discovering *sequences* instead of sets of categories to capture the inherent hierarchical organization of the categories in topic directories. The personalization of the topic directory is based on these discovered patterns and results in the generation and insertion of shortcuts to specific categories. Thus, after the personalization process, each group of users is presented with a different "view" of the same topic directory.

WIDM'07, November 9, 2007, Lisboa, Portugal.

Contribution. In a nutshell, the key contributions of this paper are:

- A methodology for discovering interest groups. A user is assigned to an interest group if her navigation patterns during the browsing of topic directories are similar to that of the other users in this group. For this purpose, we propose a metric that estimates the similarity between two navigation patterns and we exploit clustering techniques to generate interest groups.
- A set of mining tasks for interest groups. These tasks are, namely, (a) the detection of indecisive user behaviour, (b) the discovery of sequences of popular categories, and (c) the discovery of sequences of categories with popular resources.
- A set of personalization tasks. These tasks aim at the creation of links, called shortcuts, between categories in the directory. The creation of shortcuts is performed either offline or online. In the offline mode, the system processes the navigation patterns for each interest group and recommends the creation of a number of *static shortcuts*. These shortcuts are presented to all members of an interest group. In the online mode, the system takes into account the categories visited by the user in the current active session and the navigation patterns of her interest groups. Shortcuts are created in real time, and, thus, are called *dynamic shortcuts*.
- An experimental evaluation of the aforementioned methodology. We carry out several experiments to evaluate both mining and personalization tasks. Our results demonstrate the effectiveness of our approach.

We should note that all mining and personalization tasks have been implemented in a fully working prototype maintaining the topic directory of ODP.

Outline. The following section discusses related work. In Section 3, we introduce key concepts regarding the topic directories and the navigation patterns that will be used in the rest of the paper. The proposed mining and personalization tasks for topic directories are presented in Sections 4 and 5, respectively. In Section 6, we present the experimental evaluation of our methods. We conclude in Section 7 with our plans for future work.

2. RELATED WORK

Web usage mining has been extensively used in order to analyze web log data. There exist various methods based on data mining algorithms and probabilistic models. The related literature is very extensive and many of these approaches fall out of the scope of this paper. For more information, the reader may refer to [8, 18, 12].

There exist many approaches for discovering sequences of visits in a web site. Some of them are based on data mining techniques [3, 23, 10], whereas others use probabilistic models, such as Markov models in order to model the users' visits [5, 7, 21, 29, 28]. Such approaches aim at identifying representative trends and browsing patterns describing the activity in a web site and can assist the web site administrators to redesign or customize the web site, or improve the performance of their systems. They do not, however, propose any methods for personalizing the web sites.

There exist some approaches that use the aforementioned techniques in order to personalize a web site [11, 2, 15]. Contrary to our approach, these approaches do not distinguish between different users or user groups in order to perform the personalization. Thus, the methods that seem to be more relevant to ours, in terms of identifying different interest groups and personalize the web site based on these profiles, are those that are based on collaborative filtering.

Collaborative filtering systems [14, 15, 16, 22] are used for generating recommendations and have been broadly used in e-commerce. Such systems are based on the assumption that users with common interests and behaviour present similar searching/browsing behaviour. Thus, the identification of similar *user profiles* enables the filtering of relevant information and the generation of recommendations. Similar to such approaches, we also identify users with common interests and use this information to personalize the topic directory. In our work, however, we do not model the user profiles as vectors in order to find similar users. Instead, we use clustering to group users into interest groups. Moreover, we propose the use of sequential pattern mining in order to generate recommendations. Thus, we also capture the sequential dependencies within users' visits, whereas this is not the case with collaborative filtering systems.

All of the aforementioned approaches aim at personalizing generic web sites. Our approach focuses on the personalization of a specific type of web sites, that of topic directories. Since topic directories organize web content into meaningful categories, we can regard them as a form of digital library or portal. In this context, we also overview here some approaches for personalizing digital libraries and web portals. Some early approaches [9, 24] were based on *explicit user input* and the personalization services provided are limited to simplified search functionalities or alerting services. [25] propose the semi-automatic generation of user recommendations based on *implicit user input*. In those approaches, information is extracted from user accesses in the DL resources, and then is used for further retrieval or filtering. As already mentioned, our approach does not limit its personalization services on identifying the preferences of each individual user alone. Rather, we identify user groups with common interests and behaviour expressed by visits to certain categories and information resources. This is enabled by approaches that are based on collaborative filtering [17, 20]. Those approaches, however, fail to capture the sequential dependencies between the users' visits, as discussed previously.

3. MODELLING TOPIC DIRECTORIES

A topic directory is a hierarchical organization of thematic categories. Each category contains *resources* (i.e., links to web pages). A category may have *subcategories* and/or *related categories*. Subcategories narrow the content of broad categories. Related categories contain similar resources, but they may exist in different places of the directory. Note that the "related" relationship is bidirectional, that is, if category N is related to M, then M is also related to N. A resource cannot belong to more than one category. We consider a graph representation of topic directories.

DEFINITION 3.1. *A* topic directory D *is a labelled graph* $G(V, E)$*, where* V *is the set of nodes and* E *the set of edges, such that: (a) each node in* V *corresponds to a category of* D*, and is labelled by the category name, (b) for each pair of nodes* (n, m) *that corresponds to categories* (N, M)*, where* N *is subcategory of* M *in* D*, there is a directed edge from* m

to n, and (c) for each pair of nodes (n, m) that corresponds to categories (N, M), where N and M are related categories in D, there is a bidirected edge between n and m.

The graph $G(V, E)$ may also have shortcuts*, which are directed edges connecting nodes in V.*

Examples of such graphs are illustrated in Figure 4. The role of shortcuts as a means for personalizing the directory will be further discussed in Section 5.

The case study of Open Directory Project. In our work, we use the Open Directory Project (ODP) as a case study. Figure 1 illustrates a part of the ODP directory. In

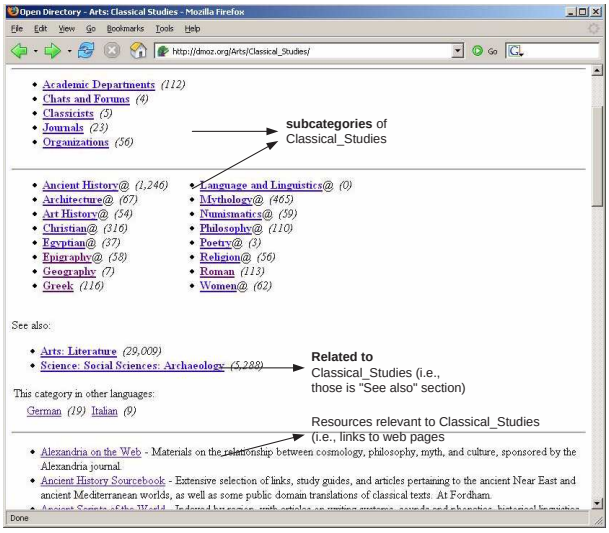

Figure 1: An ODP snapshot and graph representation corresponding categories.

ODP, there are three types of categories: (a) subcategories (to narrow the content of broad categories), (b) relevant categories (i.e., the ones appearing inside the "see also" section, and (c) symbolic categories (i.e., denoted by the @ character after category's name). Symbolic categories are subcategories that are placed in different parts of the directory. We consider relevant categories as related categories, according to the Definition 3.1.

Navigation patterns. To represent the navigation behaviour of users when browsing the directory, we use the notion of *navigation patterns*. A *navigation pattern* is the sequence of categories visited by a user during a session. We note that such patterns may include multiple occurrences of the same categories. This might be the result of users going back and forth within a path in the directory. Finally, we also underline that during a session, a user may pursue more than one topic interests.

4. MINING TASKS

To personalize a topic directory, we propose a set of data mining tasks over users' navigation patterns. For this purpose, the system needs to identify users with similar navigation behaviour and search interests, i.e., users who search for similar information in a similar way. Such users form *interest groups*. All mining tasks are performed for the users of a certain interest group.

4.1 Interest groups

A key issue for the detection of interest groups is the estimation of how similar two navigation patterns are. To estimate this similarity, we calculate the ratio of the number of the common categories (all their occurences) that exist in the navigation patterns to the total number of distinct categories in the patterns.

DEFINITION 4.1. *Let P_1 and P_2 be two navigation patterns. The similarity S between P_1 and P_2 is defined as follows:* $S = \frac{\sum_{forall\, p} occurrence\, of\, p \in P_1 \cap P_2\, in\, P_1 \cup P_2}{|P_1|+|P_2|}$

For example, the similarity between the pattern {`Top,Arts, Classical_studies,Epigraphy,Latin,Epigraphy,Latin`} and pattern {`Top,Arts,Classical_studies,Rome,Latin`} is $S = 9/12 = 0.75$.

We employ the K-means clustering algorithm [19] using this similarity metric. The clustering algorithm detects clusters of navigation patterns. Using these clusters, the system creates the interest groups. An interest group is formed by the users whose navigation patterns are included in the same cluster. Thus, we manage to capture the fact that a user might search for different thematic areas during the same, or different visits to the directory, therefore belonging to more than one interest group.

4.2 Indecisive users, B&F categories and B&F chains

This mining task aims at the identification of the so-called *indecisive users*. We refer to a user as *indecisive* when her navigation patterns includes many "back and forth" visits to the same categories in the directory. This might happen for several reasons. The user might not know in advance the exact information she is searching for, therefore she swings among some categories before refining her search. Another reason might be that the organization of the categories in the directory is different from the user's intuitive categorization of topics. Finally, such behaviour might be an indication of poorly organized topic sub-directories, or of inconsistent category labels. For example, the navigation pattern {`Arts,Music,Pop,Music,Pop,Concerts,Pop,Music,Pop,Concerts`} indicates an indecisive user, since she goes back and forth (*B&F*), visiting categories `Pop` and `Music`. To detect indecisive users, our system quantifies *B&F actions* in the navigation patterns of users.

DEFINITION 4.2. *A set of categories $N_1, N_2, \ldots, N_k$ forms a* B&F action *of length k (denoted as k-*B&F action*) in a navigation pattern P if the pattern $P' = \{N_1, N_2, \ldots, N_{k-1}, N_k, N_{k-1}, \ldots, N_2, N_1, \ldots, N_k\}$ is a subpattern of P. We call P' a* B&F chain *in P.*

For example, categories `Arts` and `Music` are involved in the {`Arts,Music`} 2-*B&F* action in the pattern {`Arts,Music, Music,Pop`}. Similarly, categories `Arts, Music, Pop` are involved in a 3-*B&F* action in the pattern {`Arts,Music,Pop, Music,Arts,Music,Pop,Groups`}. Note that {`Arts,Music, Pop,Music,Arts,Music,Pop`} is the *B&F* chain in the last pattern example.

Figure 2 illustrates how a k-*B&F* action is detected. Assume that we have a navigation pattern P of the form {`A, B,C,D,C,B,A,B,C,D,E,F`}, and we are looking for 4-*B&F* actions, starting from the beginning of P (line 1: function is called with $i = 0$). Condition in line 6 checks whether there

```
1: function CheckBF(navigation pattern P, int k, init position i)
2:     startPosL = i
3:     endPosL = startPosL + (k-1)
4:     startPosR = i+(k-1)+(k-2)+1
5:     endPosR = startPosR + (k-1)
6:     if (subpat(P, startPosL, endPosL) == subpat(P, startPosR,
   endPosR))
7:         and
8:          (subpat(P, startPosL+1, endPosL-1) == subpat(P,
   startPosR-1, endPosL+1)) then
9:         return True
10:    else
11:        return False
12:    end if
13: end function

function subpat(navigation pattern P, startPos, endPos)
    return part of P from startPos to endPos
end function
```

Figure 2: Detection of B&F actions

is a (sub)pattern of 4 categories, appearing twice in P. Variables *startPosL* and *endPosL* (*startPosR* and *endPosR*) refer to the start and the end position of the first (second) appearance of the (sub)pattern in P. In this particular example, the condition involves the (sub)pattern `{A,B,C,D}`, which indeed appears twice (positions 0, 1, 2 and 3 for the first occurrence, and positions 6, 7, 8 and 9 for the second one). This (sub)pattern actually reveals repetition of navigations from A to D (i.e., forward actions). Next, we need to detect backward (i.e., inverse) actions from D to A. Condition in line 8 checks whether there is a (sub)pattern of 2 (i.e., $k-2$) categories appearing in P between the two occurrences of the 4 categories' (sub)patterns mentioned before, indicating a backward action. Indeed, the appearance of categories `C` and `B` in positions 4 and 5, respectively, reveals such an action.

Our system is able to detect k-*B&F* actions, calculate their frequency of occurrence and rank users in descending order of this frequency. The higher the frequency the higher the degree of user indecisiveness.

4.3 Popular categories and sequential navigation subpatterns

The categories of a topic directory are organized hierarchically. This inherent order should be prevalent in the discovered navigation patterns. For this purpose, this task focuses on discovering frequent *sequences* of *popular categories* in user navigation patterns. A category is popular if the number of visits to the category is high (i.e., above a pre-defined threshold). We call these sequences *sequential navigation subpatterns*. Intuitively, frequent sequential navigation subpatterns capture the notion of popular transitions among (not necessary contiguous) categories. The order in a navigation pattern distinguishes the discovery of frequent sequential navigation pattern from the discovery of association rules [1]. For example, consider the navigation pattern p = {`Top,Arts,Classical_Studies,Epigraphy,Latin`} and the popular categories `Arts`, `Epigraphy` and `Latin`. {`Arts,Epigraphy,Latin`} is a sequential navigation subpattern of p, whereas {`Latin,Epigraphy,Arts`} is not, since the three categories do not appear in that order in p.

To identify frequent sequential navigation subpatterns, we adopt the trie-based implementation of Apriori [1] for mining frequent itemsequences [4]. We define the *length* of a sequential navigation subpattern to be the number of categories in the subpattern. A sequential navigation subpattern of length k is called *k-sequential navigation subpattern*. For example, {`Arts,Epigraphy,Arts,Latin`} is a 4-sequential navigation subpattern.

We define the *support* σ of a k-sequential navigation subpattern to be the fraction of navigation patterns that contain this subpattern in an interest group. Intuitively, the support of a sequential navigation subpattern refers to the probability that a user will visit the involved categories in the order specified in the subpattern. A popular category `X` and the 1-sequential navigation subpattern {`X`} have the same support $\sigma(X)$. Thus, the popular categories are 1-sequential navigation subpatterns whose support is at least equal to a given threshold, the so called *minimum support*. Formally:

Definition 4.3. *Let P be the set of navigation patterns of an interest group. Given a set $S = \{S_1, S_2, \ldots, S_n\}$ of k-sequential navigation subpatterns of P, the support of each S_i is defined as follows:*

$$\sigma(S_i) = \frac{|\{nav.\ pattern\ p \in P\text{: } S_i\ is\ subsequence\ of\ p\}|}{|P|}$$

Given a minimum support σ_{min} and a length k, we identify the frequent k-sequential navigation subpatterns as those subpatterns whose support is above σ_{min}.

4.4 L-popular categories and sequential navigation L-subpatterns

This task focuses on discovering frequent *sequences* of *L-popular categories*. A category is L-popular if the number of visits of its resources is high (i.e., above a pre-defined threshold). We call these sequences *sequential navigation L-subpatterns*. Intuitively, frequent sequential navigation L-subpatterns capture the notion of transitions among (not necessary contiguous) categories with popular resources. Note that L-popular categories are not necessarily popular and vice versa. This is due to the fact that a popular category may be exploited only for reaching other categories and not for selecting its resources. Also, a user may visit a category only once but select all its resources, making the category to become L-popular.

Definition 4.4. *Let L be the set of the sequences of resources visited by the users of an interest group. Let C be the set of the sequences of categories that involve those resources, respectively. Given a set $S = \{S_1, S_2, \ldots, S_n\}$ of k-sequential navigation L-subpatterns of P, the support of each S_i is defined as follows:*

$$\sigma(S_i) = \frac{|\{nav.\ pattern\ p \in C\text{: } S_i\ is\ subsequence\ of\ p\}|}{|C|}$$

Intuitively, the support of a sequential navigation L-subpattern refers to the probability that a user of the interest group will visit the involved categories (in the order specified in the subpattern) to select a great number of resources.

To identify sequential navigation L-subpatterns, we exploit the same technique used in the previous section to identify plain sequential navigation subpatterns. Given a minimum support σ_{min} and a length k, we identify the frequent k-sequential navigation L-subpatterns as those subpatterns whose support is above σ_{min}.

5. PERSONALIZATION

In the previous section we presented the mining tasks provided by our system. These tasks assist the personalization

of the topic directory according to the navigation behaviour of the users and their interest groups.

The output of all personalization tasks is a set of *shortcuts* among categories in the directory. A shortcut $A \rightarrow B$ is a direct link from A to B. Using shortcuts, we can provide alternative ways of navigating the directory, depending on the navigation behaviour of different users or interest groups.

We identify two personalization modes, i.e. ways of creating shortcuts: a) the offline mode and b) the online mode.

5.1 Offline mode

In the offline mode, the system processes the navigation patterns for each interest group and recommends the creation of a number of shortcuts. Then, the administrator of the directory decides which of the proposed shortcuts for each group should be finally created. These shortcuts are called *static shortcuts* and are presented to all members of an interest group.

We present three personalization tasks of creating static shortcuts, based on detecting: a) frequent *B&F* chains, b) frequent sequential navigation subpatterns, and c) frequent sequential navigation L-subpatterns.

5.1.1 Personalization based on frequent B&F chains

As mentioned previously, the existence of many *B&F* categories in a user's navigation pattern indicates that the user is indecisive. The more frequent a *B&F* chain is, the more users encounter the same navigation problems in a certain area of the directory.

The mining task of identifying frequent *B&F* chains helps determining such parts in the directory. The proposed personalization task is performed for each interest group separately. To help users navigate through such problematic areas, our system recommends a shortcut in the directory for each frequent *B&F* chain that exists in a navigation pattern. The start point of the shortcut is the category that appears in the beginning of a frequent *B&F* chain. The end point of the shortcut is the first category c in the navigation pattern such that: (a) c is after the last category involved in the *B&F* chain, and (b) c is different from those involved in the *B&F* chain.

Consider for example the following navigation pattern: {`Music,Easy_Listening,Music,Easy_Listening,Lounge`}. Note that {`Music,Easy_Listening,Music,Easy_Listening`} is a frequent *B&F* chain. Our system will recommend the following shortcut in the directory: `Music`→`Lounge`. Figure 4(b) illustrates the new shortcuts inserted in directory of Figure 4(a), according to the above example. In Figure 3, we formally describe this personalization task.

5.1.2 Personalization based on frequent sequential navigation subpatterns

By identifying the popular categories for an interest group, we have an indication of the topics that are of great interest to the users of the group. The sequential navigation subpatterns, i.e. the sequences of popular categories, indicate popular transitions between these topics. To personalize the topic directory according to this indication, the system recommends a set of static shortcuts. The shortcuts enable users to pursue these transitions by moving directly to the popular categories. Shortcuts are recommended only if there are no edges already connecting the involved categories.

$\mathcal{O}$: the directory
$\mathcal{B}$: the set of k frequent B&F chains
b_1: the category that starts a B&F chain $b \in \mathcal{B}$
b_n: the category that ends a B&F chain $b \in \mathcal{B}$
p_b: the navigation pattern that includes B&F chain b
getNext(b_n): returns the category, other that these in b, that is after b_n in p_b
for each $b \in \mathcal{B}$ **do**
  c =getNext(b_n)
  if there is not an edge from b_1 to c in $\mathcal{O}$ **then**
    create the shortcut $b_1 \rightarrow c$ in $\mathcal{O}$
  end if
end for

Figure 3: Personalizing the directory based on frequent B&F chains.

More specifically, for a given interest group and a given support threshold, the system identifies the frequent 2-sequential navigation subpatterns $\{x, y\}$. Provided that there is not an edge from x to y in the topic directory graph, the system recommends a static shortcut $x \rightarrow y$. We do not need to consider frequent k-sequential navigation subpatterns, with $k \geq 3$, since the Apriori algorithm constructs these subpatterns based on a set of extracted frequent 2-sequential navigation subpatterns.

Consider for example the following frequent sequential navigation subpatterns for a given interest group: {`Arts, Epigraphy`} and {`Epigraphy,Latin`} in the directory of Figure 4(a). The system identifies `Arts`→`Epigraphy` and `Epigraphy`→`Latin` candidate shortcuts. Yet, it recommends only `Arts`→`Epigraphy` shortcut, since there is already an edge from `Epigraphy` to `Latin`. Figure 4(c) illustrates the new shortcut inserted in directory of Figure 4(a), according to the above example.

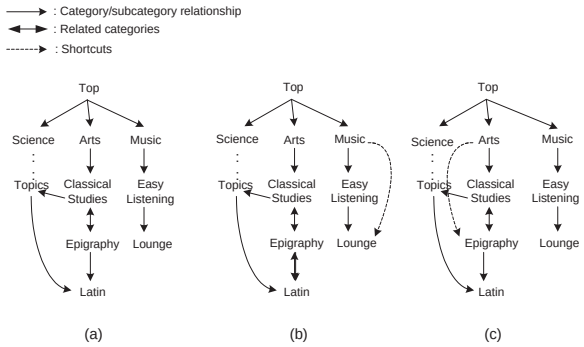

Figure 4: Examples of offline personalization tasks.

5.1.3 Personalization based on frequent sequential navigation L-subpatterns

The frequent sequential navigation L-subpatterns indicate transitions among (not necessary contiguous) categories with popular links. To personalize the topic directory according to this indication, the system recommends a set of static shortcuts to perform these transitions directly (i.e., from one L-popular category to another).

To create the shortcuts we use the same technique used in the previous subsection to personalize the directory based on frequent sequential subpatterns. Note, however, that the shortcuts created to personalize the directory based on frequent sequential L-subpatterns are different that the short-

cuts created to personalize the directory based on plain frequent sequential subpatterns. This is due to the fact that L-popular categories are not necessarily popular (and vice versa).

5.2 Online mode

In the online mode, the system takes into account the categories visited by the user and the navigation patterns of her interest groups. Shortcuts are created in real time, and, thus, are called *dynamic shortcuts*. Note that this mode does not involve any actions from the directory administrator, and that different dynamic shortcuts are presented to each individual user. We present a personalization task for creating dynamic shortcuts, exploiting sequential navigation subpatterns. A similar task has been implemented for creating shortcuts using L-subpatterns.

In brief, we use a fixed-size sliding window w, called *active navigation window*, for each interest group that the user belongs to. All windows have the same size. Each window slides over the user navigation in the current active session, having the last $|w|$ popular categories visited by the user. To personalize the directory, the system creates shortcuts by matching each w with the sequential navigation subpatterns of each interest group, respectively, of which the user is a member. A similar method has been introduced in [13]. Our approach, though, exploits multiple windows, and matches each of them against the sequential navigation subpatterns of the corresponding interest group, instead of matching a single window to all subpatterns discovered from all users.

We next describe our methodology in detail, considering only one interest group and its corresponding window. We extract and store in advance all sequential navigation subpatterns of at most $|w|+1$ length, given a support threshold and a window size $|w|$. For example, let p_1 = `{Arts,Classical_Studies}`, p_2 = {`Classical_Studies, Latin`} and p_3 = `{Arts,Classical_Studies,Latin}` be the frequent sequential navigation subpatterns for a given interest group with respect to Figure's 4(a) directory.

Given the active navigation window w, we consider the frequent sequential navigation subpatterns of length $|w|+1$ whose prefix contains w. For those sequential navigation subpatterns that match w, the system generates a candidate dynamic shortcut from the last category of w to last category of the subpattern. Assuming for example that w = `{Arts,Classical_Studies}`, the system matches w only with p_3 whose length is equal to $|w|+1$. The candidate dynamic shortcut is `Classical_Studies`→`Latin`.

Finally, the system automatically creates all the candidate shortcuts whose *confidence* value is equal to or greater than a given minimum confidence threshold, provided that there is no edge already connecting them in the topic directory graph. We next define confidence for dynamic shortcuts.

DEFINITION 5.1. *Let $A \rightarrow B$ be a shortcut and w an active navigation window for an interest group such that A is the last category of w. The confidence α of $A \rightarrow B$ is defined as:*

$$\alpha(A \rightarrow B) = \frac{\sigma(w \circ \{B\})}{\sigma(w)}$$

where $\circ$ denotes the concatenation operator.

The confidence of a shortcut $A \rightarrow B$, with A being the last category of the active window w, refers to the conditional probability that the user will visit B given that she has visited all the categories of w.

In the previous example, the support of p_1, p_2 and p_3 is 0.8, 0.7 and 0.6, respectively. The confidence of the candidate dynamic shortcut `Classical_Studies`→`Latin` is: $\alpha(\texttt{Classical\_Studies} \rightarrow \texttt{Latin}) = \frac{\sigma(p_3)}{\sigma(p_1)} = \frac{0.6}{0.8} = 0.75$.

6. EVALUATION

We have implemented a prototype system in order to evaluate the mining and the personalization tasks proposed in this paper. The system includes two basic modules. The first module is actually a web application, where users can register and browse the ODP directory. The ODP categories and the categorized web pages were stored in an RDBMS, after parsing the publicly available RDF dumps of ODP (http://rdf.dmoz.org/) . The second module is a stand-alone application that can be used to execute the mining tasks and, then, perform the personalization tasks (Our system is available at http://casablanca.dblab.ece.ntua.gr/p-miner/) .

Interest groups. We first present the results of the evaluation regarding the quality of the interest groups detected. We asked 12 users (2 groups of 6 users each) to register in our system and browse the directory, searching for web pages relevant to the following topics: (1) video games, (2) William Shakespeare, (3) basketball, and (4) food and cooking. Each topic was assigned to 3 different users. We carefully organized the assignment of these topics, so that users had at least moderate knowledge about them. We repeated the experiment by switching the topics assigned to group 1 with the topics assigned to group 2.

Based on the navigation patterns recorded and the relevance judgment we manually performed, we identified 8 clusters of navigation patterns regarding: (1) video games, (2) food and recreation services, (3) cooking recipies, (4) cooking cheese, (5) healthy food, (6) Shakespeare's theatrical plays, (7) Shakespeare's books, and (8) basketball.

We run the K-means clustering algorithm for several values of K, noticing that meaningful clusters are formed for $K = 10$. The result of the algorithm was the formation of 10 clusters of navigation patterns (i.e., 10 interest groups of users). Taking into consideration the relevance judgment, we calculated *precision* Pr and *recall* R values for the 10 clusters.

In brief, Pr and R are defined as follows (micro-average approach [26, 27]). For an extracted cluster C_i let: (a) a_i be the number of navigation patterns in C_i that were indeed members of that cluster (correctly clustered), (b) b_i be the number of patterns in C_i that were not members of that cluster (misclustered), and (c) c_i be the number of patterns not assigned to C_i, although they should have. Then: $Pr = \sum_i a_i / (\sum_i a_i + \sum_i b_i)$ and $R = \sum_i a_i / (\sum_i a_i + \sum_i c_i)$ High precision implies high accuracy of the clustering task, while low recall means that there are many patterns not assigned to the correct cluster. High precision and high recall indicate excellent clustering quality.

The calculated Pr and R values are shown in Table 1. All navigation patterns wrongly assigned to clusters 3 and 10 were considered as misclustered patterns. The high values for both Pr and R demonstrate the effectiveness of the interest group construction.

Offline personalization mode. To evaluate the personalization tasks in the offline mode, we launched the per-

ClusterNo	macro-avg Pr	macro-avg R
1	0.90	1
2	1.00	0.57
3	0.00	-
4	0.73	0.85
5	1.00	0.86
6	0.78	0.78
7	0.81	0.93
8	1.00	0.92
9	1.00	1
10	0.00	-
—	**micro-avg Pr**	**micro-avg R**
—	0.85	0.89

Table 1: Precision and recall values for clusters (i.e., interest groups) detected.

Interest group	Hit ratio
1	0.46
2	0.50
3	—
4	0.77
5	1.00
6	0.75
7	0.53
8	0.88
9	0.46
10	—
Avg for the 8 valid groups	0.67
Number of shortcuts inserted	24

Table 2: Hit ratio of shortcuts per interest group.

sonalization methods of our module to create the directory shortcuts. We then asked the users to browse again the directory, searching for web pages relevant to the same topics used in the previous experiment.

We calculated the *hit ratio* for the created shortcuts per interest group that expresses the utilization of shortcuts. The hit ratio is the number of times that users in the interest group moved from the start to the end category of all shortcuts, using the shortcuts, to the number of times they performed the same movements, regardless using the shortcuts or not. Results are shown in Table 2. The high rates of the hit ratio demonstrate the high utilization of shortcuts, and, thus, the effectiveness of the personalization task.

Online personalization mode. In this experiment, we created a synthetic data set including visits to a part of ODP graph relevant to a certain topic, thus simulating the visits of the users belonging to a specific interest group. For this purpose, we used a crawler for simulating user category visits in a depth-first search way fashion in the ODP hierarchy. The depth of each navigation path is randomly selected. For our experiment, we focus on three subgraphs of ODP, rooted at three subcategories `Poetry`, `World_Literature` and `Drama` of category `Top/Arts/Literature`, respectively. We ran the crawler collecting 60000 navigation patterns starting from category `Poetry` of ODP, 30000 visits starting from category `World_Literature`, and 10000 visits starting from category `Drama`. From the 100000 navigation patterns, we formed a training set of 60000 navigation patterns (36000 for `Poetry`, 18000 for `World_Literature` and 6000 for `Drama`) and a test data set of 40000 navigation patterns (24000 for `Poetry`, 12000 for `World_Literature` and 4000 for `Drama`).

Each navigation pattern in the test data set is divided in two parts. The first part (70% of the pattern) is used for generating the shortcuts using our methodology presented in Section 5.2. Note that the window w includes the last $|w|$ popular categories of this part. The second part (30% of the pattern) is used for the evaluation. In order to evaluate the quality of our personalization method, we compute the ratio of the number of popular categories in the second part, involved in shortcuts as targets, to the number of created shortcuts, for each navigation pattern in the test data set. By averaging the computed values, we calculated the precision of the personalization task.

Figure 5 shows the precision of the personalization task varying the confidence and support threshold, for several values of $|w|$. As expected, the precision goes up as the confidence threshold increases. This is due to the fact that an increased confidence for a shortcut $A \rightarrow B$, with A being the last category w, means that there is high probability that category B exists in the second part of the navigation pattern considered. Similar results are observed for the support values used in the experiment. In both graphs the precision increases as $|w|$ increases. The reason is that, as $|w|$ increases, a more representative part of user navigation behaviour is used to generate the shortcuts.

7. CONCLUSION AND FURTHER WORK

In this paper, we introduce a methodology for personalizing topic directories according to the navigation behaviour of the users. We present a set of mining tasks on user navigation patterns. Navigation patterns actually capture user navigation behaviour during their browsing sessions, representing the users' interests in terms of visited categories and retrieved resources. Moreover, we propose a set of personalization tasks that customize the organization of the topic directory for certain user groups, called interest groups. These tasks aim at the creation of links called shortcuts between categories in the directory. Finally, we run several experiments to evaluate the proposed mining and personalization tasks, showing the effectiveness of our approach.

Our future work will focus on semantically rich topic directories. We plan to investigate how semantic information available in directories (e.g., `is_a` and `partOf` relationships) can assist the mining and the personalization tasks. Furthermore, we will elaborate on the design and development of user-driven profiles to assist the personalization tasks. Finally, we will extend the evaluation method of online personalization by studying real user navigations.

8. REFERENCES

[1] R. Agrawal and R. Srikant, *Fast algorithms for mining association rules*, 20th International Conference on Very Large Data Bases (VLDB) (Santiago, Chile), September 12-15 1994.

[2] C. Anderson, P. Domingos, and D. S. Weld, *Models and their application to adaptive web navigation*, 8th ACM SIGKDD Conference, Canada, 2002.

[3] B. Berendt and M. Spiliopoulou, *Analysing navigation behaviour in web sites integrating multiple information systems*, VLDB Journal **9** (2000), no. 1.

[4] F. Bodon, *Trie-based apriori implementation for mining frequent itemsequences*, ACM SIGKDD Workshop on Open Source Data Mining Workshop (OSDM) (Chicago, IL, USA), 2005.

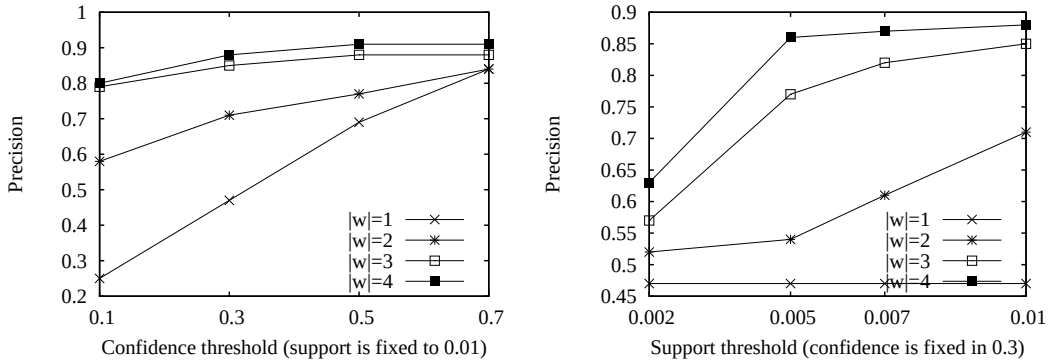

Figure 5: Precision of the personalization task varying the confidence/support threshold for several values of $|w|$.

[5] J. Borges and M. Levene, *Data mining of user navigation patterns*, Lecture Notes in Computer Science, 1836 (1999).

[6] J. Callan, A. Smeaton, M. Beaulieu, P. Borlund, P. Brusilovsky, M. Chalmers, C. Lynch, J. Ried, B. Smyth, U. Straccia, and E. Toms, *Personalisation and recommender systems in digital libraries*, May 2003, Joint NSF-EU DELOS Working Group Report.

[7] M. Deshpande and G. Karypis, *Selective markov models for predicting web-page accesses*, ACM Transactions on Internet Technology **4** (2004), no. 2.

[8] M. Eirinaki and M. Vazirgiannis, *Web mining for web personalization.*, ACM Trans. Internet Techn. **3** (2003), no. 1.

[9] L. Fernández, J. Alfredo Sánchez, and A. García, *Mibiblio: personal spaces in a digital library universe.*, In The 5th ACM Conference on Digital Libraries (ACM DL), 2000, pp. 232–233.

[10] G. Hooker and M. Finkelman, *Sequential analysis for learning modes of browsing*, 6th WEBKDD Workshop, Seattle, 2004.

[11] E. Manavoglu, D. Pavlov, and C.L. Giles, *Probabilistic user behaviour models*, 3rd Intl. Conference on Data Mining (ICDM 2003), 2003.

[12] B. Mobasher, *Data mining for personalization. in the adaptive web: Methods and strategies of web personalization.*, (2006).

[13] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, *Using sequential and non-sequential patterns in predictive web usage mining tasks.*, International Conference on Data Mining (ICDM), 2002.

[14] B. Mobasher, H. Dai, T. Luo, Y. Sung, and J. Zhu, *Discovery of aggregate usage profiles for web personalization*, WEBKDD Workshop, Boston, 2000.

[15] M. Nakagawa and B. Mobasher, *Model based on site connectivity*, WEBKDD Workshop, Washington DC, 2003.

[16] M. Papagelis, I. Rousidis, D. Plexousakis, and E. Theoharopoulos, *Incremental collaborative filtering for highly-scalable recommendation algorithms*, Foundations of Intelligent Systems, 15th International Symposium (ISMIS), May 25-28 2005.

[17] W. Park, W. Kim, S. Kang, H. Lee, and Y.-K. Kim, *Personalized digital e-library service using users' profile information.*, 10th European Conference on Digital Libraries (ECDL), 2006, pp. 528–531.

[18] D. Pierrakos, G. Paliouras, C. Papatheodorou, and C. D. Spyropoulos, *Web usage mining as a tool for personalization: A survey.*, User Model. User-Adapt. Interact. **13** (2003), no. 4, 311–372.

[19] E. Rasmussen, *Clustering algorithms*, 1992, in: W. Frakes, R. Baeza-Yates (Eds.), Information Retrieval: Data Structures and Algorithms, Prentice Hall.

[20] M. E. Renda and U. Straccia, *A personalized collaborative digital library environment: a model and an application.*, Inf. Process. Manage. **41** (2005), no. 1, 5–21.

[21] R. R. Sarukkai, *Link prediction and path analysis using markov chains*, Computer Networks **33** (2000), no. 1-6.

[22] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, *Item-based collaborative filtering recommendation algorithms.*, In the 10th International World Wide Web Conference (WWW), 2001.

[23] M. Spiliopoulou, L. C. Faulstich, and K. Wilkler, *A data miner analyzing the navigational behaviour of web users.*, Workshop on Machine Learning in User Modelling, 1999.

[24] N. Spyratos, Y. Tzitzikas, and V. Christophides, *On personalizing the catalogs of web portals*, Fifteenth International Florida Artificial Intelligence Research Society Conference (FLAIRS), May 14-17 2002.

[25] M. Theobald and C.-P. Klas, *BINGO! and DAFFODIL: Personalized exploration of digital libraries and web sources*, 7th International Conference on Computer-Assisted Information Retrieval (RIAO 2004), 2004.

[26] C. J. van Rijsbergen, *Information retrieval*, Butterworths, London, 1979.

[27] Y. Yang, *An evaluation of statistical approaches to text categorization*, Information Retrieval **1** (1999), no. 1.

[28] A. Ypma and T. Heskes, *Categorization of web pages and user clustering with mixtures of hidden markov models*, WEBKDD Workshop, Canada, 2002.

[29] J. Zhu, J. Hong, and J. G. Hughes, *Using markov models for web site link prediction*, ACM HYPERTEXT, 2002.