From Web Usage Statistics to Web Usage Analysis

Georgios Paliouras,* Christos Papatheodorou, Vangelis Karkaletsis,***
Costantine Spyropoulos,* Panayiotis Tzitziras**
*** Institute of Informatics and Telecommunications,**
**** Division of Applied Technologies,**
National Centre for Scientific Research (NCSR) "Demokritos",
15310, Aghia Paraskevi, Attikis, GREECE
***E-mail:{paliourg, vangelis, costass}@iit.demokritos.gr**
****E-mail:{papatheodor,tzitziras}@lib.demokritos.gr**

ABSTRACT

The World Wide Web has become a major source of information that can be turned into valuable knowledge for individuals and organisations. In the work presented here, we are concerned with the extraction of meta-knowledge from the Web. In particular, knowledge about Web usage which is invaluable to the construction of Web sites that meet their purposes and prevent disorientation. Towards this goal, we propose the organisation of the users of a Web site into groups with common navigational behaviour (user communities). We view the task of building user communities as a data mining task, searching for interesting patterns within a database. The database that we use in our experiments consists of access logs collected from the Web site of the Advanced Course on Artificial Intelligence 1999. The unsupervised machine learning algorithm COBWEB is used to organise the users of the site, who follow similar paths, into a small set of communities. Particular attention is paid to the interpretation of the communities that are generated through this process. For this purpose, we use a simple metric to identify the representative navigational behaviour for each community. This information can then be used by the administrators of the site to re-organise it in a way that is tailored to the needs of each community. The proposed Web usage analysis is much more insightful than the common approach of examining simple usage statistics of the Web site.

1. INTRODUCTION

As the Web is expanding at an increasingly fast rate, embracing a large number of services, the issue of efficient information access is becoming a crucial factor in the design of Web sites. However, the manner in which a user accesses the information available on a Web site is heavily dependent on his/her needs, interests, knowledge and prejudices. As a result, the structure of the site should reflect the requirements of its users.

The first step towards providing efficient information access in a site is to understand its usage. This can be done by monitoring the daily usage of the site and analysing the collected data. Commonly, the data that is collected by the administrators of various sites consists of general-purpose statistical figures, such as the number of users who access a particular page within certain periods of the day. This information can be useful in drawing a few general conclusions on the usage of a site, but does not facilitate the adaptation of the site to needs of the users.

We examine an alternative, more personalised approach to the collection and analysis of usage data. This approach is based on the analysis of access logs, which record the date and time each page is accessed, as well as the IP number of the visitor. We organise the access-log information in sessions, grouping sequentially the pages that were accessed from a particular IP, within a certain period of time. Each session provides a navigational pattern, associating a set of pages in the site.

Access sessions are the basis on which communities of users with common navigational patterns are constructed. The term community was introduced in the system *Doppelgänger* [18]. Here, the construction of communities is done with the use of COBWEB [12], an unsupervised machine learning algorithm. COBWEB belongs in the conceptual clustering family, which is particularly suitable to symbolic training data, as it is the case here, where the training examples are the access sessions.

The resulting communities can be used to improve the services provided in the Web site. However, this can be done effectively only when the generated communities are meaningful, that is if they can be associated with an identifiable navigational behaviour. A sound and objective method for characterising communities is highly desirable, in order for the results to be useful to the site administrator. We use a simple metric to decide on the navigational pattern that is most representative of each community. This metric was introduced in our earlier work, applying user modelling techniques to digital libraries [19]. Here we use this metric to characterise communities of visitors to the Advanced Course on Artificial Intelligence (ACAI '99) Web site.

Section 2 of the paper takes a broader view of user modelling in order to position our problem within this research domain, explains how machine learning techniques can be exploited in user modelling and describes the learning algorithm that is used here. Section 3 discusses the problem of constructing meaningful communities and presents the metric for the characterisation of the community descriptions. Section 4 presents the experimental setting and discusses the experimental results. Section 5 refers briefly to some related work and section 6 summarises the presented work, introducing our plans for the future.

2. RESEARCH BACKGROUND

User Modelling

User Modelling technology aims to make information systems really user-friendly, by adapting the behaviour of the system to the needs of the individual. The importance of adding this capability to information systems is proven by the variety of areas in which user modelling has already been applied: information retrieval [6,7,15], filtering [1,16] and extraction [3] systems, adaptive user interfaces [8,9] and student modelling [26].

A user model consists mainly of the individual preferences of the user. Furthermore, it may contain personal information about the user, such as his/her age, occupation, etc. The latter type of information is not directly necessary for the adaptation of the system to the user, but may be used to categorise the user into a *stereotype*, which in turn allows the system to anticipate some of the user's behaviour. Stereotypes have been introduced in [24], as a means of organising the users of the system into meaningful groups. For instance, a stereotype might state that "graduate students follow a particular path, through the site of ACAI '99, leading to the page on student grants." Thus, a stereotype characterises groups of users, with common behaviour. The characterisation of the group is based on personal information included in the models of the participating users.

Personal information about the users of a system is not always available and therefore the construction of user stereotypes may not be possible. This is especially true of visitors to a Web site. In that case, the organisation of users into groups with common interests can still be useful. Such a group of users is termed a *user community* and corresponds to a stereotype missing the personal information. Clearly, the loss of information in the transition from stereotypes to communities is not without cost. A stereotype can be used to predict the preferences of a user, even when he/she has not explicitly stated any of them. This is not possible with communities, which can only be used to extend/modify an existing user model. Despite that, user communities can be used in several ways to improve the quality of service provided by the information system.

Learning from user models

Machine learning methods have been applied to user modelling problems, mainly for acquiring models of individual users interacting with an information system, e.g. [5,10,22]. In such situations, the use of the system by an individual is monitored and the collected data are used to construct the model of the user, i.e., his/her individual requirements.

We are concerned with a higher level of generalisation of the users' interests: the construction of user communities. One approach to the construction of user communities is by generalising the properties of user models. This approach requires the prior construction of personal user models, which can be done either manually or automatically. However, it is not always possible to identify individual users who visit a Web site, unless the site can only be accessed by registered users. Furthermore it does not seem necessary to construct the intermediate level of models, i.e., the personal user models. One can construct the communities directly from access sessions, by identifying common navigational patterns across different IP numbers. This approach is followed in this paper.

The choice of learning method for the construction of stereotypes and communites depends largely on the type of training data that are available. The main distinction in machine learning research is between *supervised* and *unsupervised* learning methods. Supervised learning requires the training data to be preclassified. This usually means that each training item (*example*) is associated with a unique label, signifying the category in which the item belongs. In our case, this would mean that each user model or access session must be associated with a pre-defined community or stereotype label. Given these data, the learning algorithm builds a characteristic description for each category, covering the examples of this category, i.e., the users belonging in the category, and only them, i.e., none of the users of other categories. The important feature of this approach is that the category descriptions are built conditional to the preclassification of the examples in the training set. In contrast, unsupervised learning methods do not require preclassification of the training examples. These methods form clusters of examples, which share common characteristics. When the cohesion of a cluster is high, i.e., the examples in it are very similar, it is labelled as a category.

Supervised learning seems most appropriate for the construction of user stereotypes, i.e., the characteristic description of user groups based on personal information about the users. In contrast, the construction of user communities is a typical unsupervised learning task. In the work presented here, we have chosen to concentrate on the construction of user communities, since the collection of personal information about the visitors of a Web site is problematic.

Unsupervised learning tasks have been approached by a variety of methods, ranging from statistical clustering techniques to neural networks and symbolic machine learning. In this work, we have opted for a symbolic learning method, because we are interested in the comprehensibility of the results. The branch of symbolic machine learning that deals with unsupervised learning is called *conceptual clustering* and a popular representative of this approach is the algorithm COBWEB.

Learning algorithm

The algorithm that we use in this work performs *conceptual clustering*. Conceptual clustering is a type of learning by observation that is particularly suitable for summarising and explaining data. *Summarisation* is achieved through the discovery of appropriate clusters, which involves determining useful subsets of an object set. In unsupervised learning, the object set is the set of training examples, i.e., each object is an example. *Explanation* involves *concept characterisation*, i.e., determining a useful concept description for each cluster.

COBWEB is an incremental algorithm that uses hill-climbing search to obtain a concept (cluster) hierarchy, partitioning the object space. The term incremental means that objects are incorporated into the concept hierarchy as they are observed. The hill-climbing search makes use of a heuristic called *category utility* [13], which is a probabilistic measure of the cohesion of a cluster of objects.

Each object is a vector of feature-value pairs. In our case, the objects are the access sessions coded appropriately into binary feature vectors. We examine two different ways of encoding the objects: one in which each feature corresponds to a page in the site and one in which each feature is a transition from one page to another. The presence of a page or a transition within a session is signified by the value 1 for the corresponding binary feature. Features in such an encoding can be viewed as components of the path recorded in a session. For convenience, we will henceforth refer to the features in this particular problem as *path components.*

COBWEB is an efficient and flexible algorithm. The complexity of the incorporation of an object in a hierarchy is quadratic to the nodes of the derived hierarchy. Since the size of the hierarchy is log-linearly related to the size of the object set, COBWEB is scalable to large training sets. The algorithm can also deal with missing feature values. However, COBWEB depends on its incremental character, i.e., it is dependent on the order of the observed objects.

3. MEANINGFUL COMMUNITIES

The clusters generated by COBWEB, represent the user communities. The question that arises is whether there is any meaning in the derived communities. Since there is no personal information available about the users, the construction of stereotypical descriptions for the communities is not possible. The only information available is the navigational behaviour of the users in each community. Thus, the natural way to construct meaningful communities is by trying to identify navigational patterns that are representative of the participating users. Ideally, we would like to be able to construct a prototypical model for each community, which is representative of the participating users and significantly different from other communities.

The construction of prototypical models for the communities is a problem in itself. We use a metric to decide on the representative navigational pattern for each community. This metric measures the increase in the frequency of a path component (a page or a transition) within a community, as compared to the default frequency in the whole data set. In [4] the increase in frequency was used as an indication of the increase in the predictability of a feature within the community. Given a component *c*, with default frequency f_c, if the frequency of this component within a community *i* is f_i, the metric is defined as a simple difference of the squares of the two frequencies:

$$FI_c = f_i^2 - f_c^2$$

FI stands here for Frequency Increase. Clearly, when FI_c is negative there is a decrease in frequency and the corresponding path component is not representative of the community. The definition of a representative component for a community, is simply that $FI_c > \alpha$, where α is the required extent of frequency increase. The question that arises is how large the increase should be in order for the path component to be considered as a characteristic one for a community.

In order to see the impact of the parameter on the characterisation of the communities, we vary α and measure two properties of the generated community descriptions:

Coverage: the proportion of components covered by the descriptions. Some of the components will not be covered, because their frequency will not have increased sufficiently.

Overlap: the amount of overlap between the constructed descriptions. This is measured simply as the ratio between the total number of components in the descriptions and the number of distinct components that are covered.

4. CASE STUDY

4.1 Experimental setting

Experimental setting

User communities are constructed by COBWEB as clusters of the sessions extracted from the access logs of the Advanced Course on Artificial Intelligence (ACAI ’99) Web site (http://www.iit.demokritos.gr/skel/eetn/acai99/). The log files consisted of almost 5200 Web-server calls (log file entries) and covered the period between 7 and 26 of May 1999. Each log entry recorded a visitor's access date and time, its computer IP address and domain name, and the target page (URL). In order to construct a training dataset for COBWEB, the data in the log files were pre-processed in two ways:

1. Access sessions were extracted.
2. Two alternative encodings were defined to turn the paths recorded in the access sessions into feature vectors.

Extracting access sessions from log files is a complex procedure, in which uncertainty plays significant role. Our methodology is a variant of the log pre-processing phase described in [25]. This process involves the following stages:

1. Group the logs by date and IP address.
2. Define a time-threshold in which the transition from one page to another is "acceptable".
3. Group the pages accessed by the same IP address using the defined time-threshold, to form a session.

In order to define the time-threshold, we created the frequency distribution of the page transitions in minutes. According to this distribution, transitions from one page to another, that are done with a time interval longer than one hour, have almost zero frequency. Thus, a sensible definition for an access session is a sequence of page transitions for the same IP address, where each transition is done at a time interval smaller than one hour. Based on this definition, our log files consisted of 1006 access sessions.

Regarding the encoding of access sessions into feature vectors, we examined two alternative approaches. In the first approach each feature in the feature vector represented the absence or presence of a particular page of the Web site in the session. As there were 41 pages in the site that were visited at least once, the feature vector consisted of 41 binary features. In the alternative encoding, we used transitions between pages, rather than individual pages as the basic path components. Clearly the number of all possible transitions between 41 pages is prohibitively large. Even the number of different transitions that appear in the log files is very large. Thus, we needed a method to reduce the number of features used in this encoding. This reduction was achieved by examining the frequency distribution of the transitions from one page to another (Fig. 1). As can be seen in Fig. 1, the distribution is highly skewed, i.e., there is a small number of very frequent transitions. We decided on a cut-off frequency of 20, which was the point where the distribution was becoming flat. Additionally we removed all transitions from a page to itself. As a result, 27 transitions survived this selection and were used to form the binary feature vector.

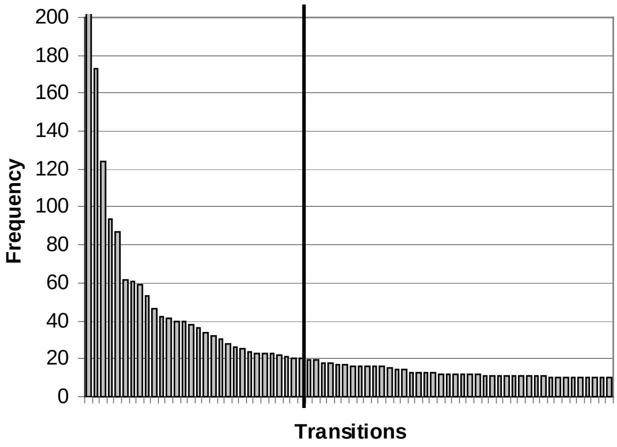

Fig. 1. Distribution of transitions between the pages of the site. Only the leftmost part of the distribution is shown here. The vertical line corresponds to the cut-off point that was chosen. Transitions to the left of this point appear at least 20 times in the dataset.

Results

In the first experiment we adopted a zero-order approach, looking at the Web pages that are included in each session. As explained above, the feature vector in this experiment included 41 binary features, representing the presence or absence of a page in the session. COBWEB was run on the 1006 sessions encoded in this way and generated a hierarchy of 1752 nodes. Starting at the top node and moving down the hierarchy, the set of 1006 sessions is divided into two non-overlapping subsets, each of which is subdivided into three smaller subsets. Thus, at the second level of the hierarchy there are six clusters of sessions. This is a manageable number of clusters to examine the characteristics of the communities that have been formed by COBWEB. For this set of clusters, we used the FI measure to select the most representative web pages for each community and varied the pruning parameter α, measuring the coverage of the communities and the overlap between their descriptions. Figures 2 and 3 present the results of this experiment.

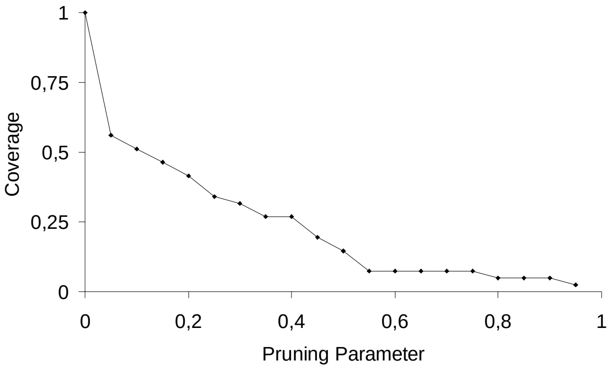

Fig. 2. Coverage of the communities generated at the second level of the hierarchy, using pages as features.

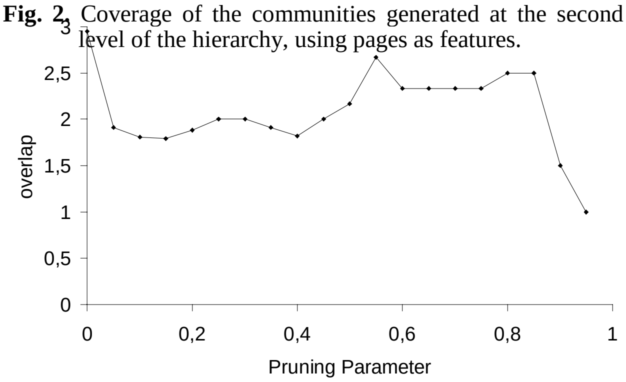

Fig. 3. Overlap between the communities generated at the second level of the hierarchy, using pages as features.

As expected, the coverage decreases, as the criterion for keeping a page in the representative description of a community becomes stricter. At the same time, the overlap between the community descriptions increases, as the average community size decreases, due to a higher level of pruning. In order to examine the quality of the community descriptions that were generated, we chose a small value of α (α=0.05), which provides high coverage and relatively low overlap, and looked at the descriptions of the six communities. Table 1 shows these descriptions.

Table 1. Community descriptions for the second level of the hierarchy, using pages as features.

Community (size)	**Pages**
A (191)	
B (108)	31,1,30
C (492)	1
D (85)	1,22,20,31,27,28,7
E (52)	1,31,22,27,20,2,19,30,9,28,10,23,24,7,15,29,3,14,26,17,11,12,25,
F (78)	1,24,19,23,22,25,31,10,30,14

The first three (A,B,C) and the last three (D,E,F) communities are siblings. In the second column of the table, the pages in each community description are provided in decreasing order of FI, i.e., the most representative pages are near the beginning of the list. Community A serves as a filter for the users that access a small number of pages (typically one). There does not seem to be a particular preference for some pages within this community. Community C groups a large number of users who visited only the first page of the site. In general the three first sibling communities consist of short sessions. The longer sessions are assigned to the last three communities, which are hardly differentiable by the descriptions that are generated. Thus, the main conclusion of this experiment is that looking at the sessions as bags of pages does not help in analysing the navigational behaviour of the visitors of the site.

This observation has led us to consider a different representation of the paths recorded in an access session. Instead of using pages as the path components, we examined the transitions between pages. In section 4 we explained how 27 such transitions were chosen to make up the binary feature vector. COBWEB was run again on the data and generated a hierarchy of 1888 nodes. Focusing again at the top nodes of the hierarchy, the set of 1006 sessions was first split into three subsets, which where further subdivided into eight subsets on the second level. We concentrated our analysis on this set of eight clusters. Figures 4 and 5 show the way in which the coverage and the overlap in the community descriptions varies with the pruning parameter α.

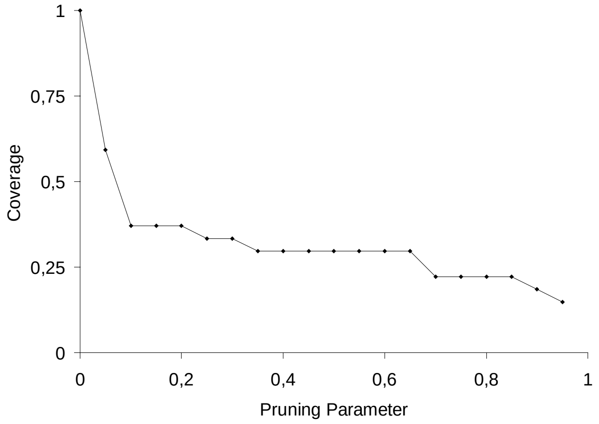

Fig. 4. Coverage of the communities generated at the second level of the hierarchy, using transitions as features.

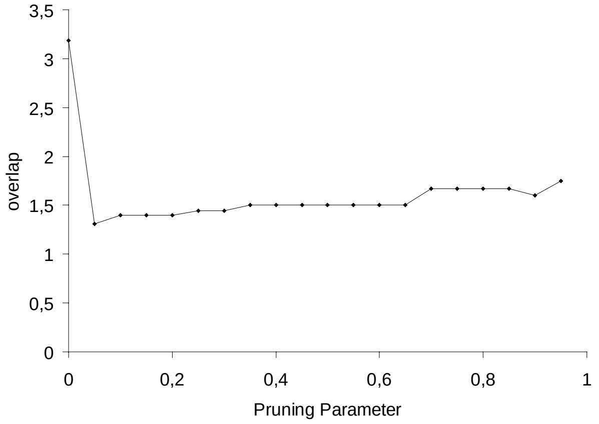

Fig. 5. Overlap between the communities generated at the second level of the hierarchy, using transitions as features.

The results in this second experiment are very different from the first. After a steep fall at the beginning, the coverage of the community descriptions decreases very slowly. Similarly, the overlap between the descriptions remains relatively stable as the pruning parameter increases. Furthermore, the overlap is lower than in the first experiment.

Choosing the same pruning threshold (α=0.05) as in the first experiment, where the overlap is at its minimum and the coverage is still high, we examined the actual community descriptions. Table 2 shows the descriptions of the eight communities.

Table 2. Community descriptions for the second level of the hierarchy, using transitions as features.

Community (size)	Transitions
A (659)	
B (72)	24→25, 23→24, 1→24, 1→19, 19→23
C (41)	1→22, 22→20, 20→31, 31→27, 27→7, 19→23
D (37)	22→31, 1→22
E (36)	22→27, 1→22
F (84)	1→30
G (11)	1→30, 8→1, 1→8
H (66)	30→31, 1→30

The eight communities are separated into siblings as follows: (A,B), (C,D,E) and (F,G,H). The first community is very large and does not have a representative description. It clusters together mainly empty sessions, including the large number of sessions, in which only the first page of the site was visited. The description of the other communities consists of sequences of page transitions, sorted in decreasing order of FI. The first observation is that there are clear differences between the description of different communities. More interestingly, however, the transitions in most of the communities seem to make up paths through the pages. This is not imposed by the representation, which encodes transitions between pairs of pages, rather than complete paths. The representative paths for the communities express very interesting navigational patterns. For instance, the users of the large community B, seem in general to follow the path: (1→19→23→24→25). Page 24 contains registration information for ACAI '99 and page 25 is the registration form. Thus, the community consists of visitors who are interested in registering for ACAI '99. This observation can be utilized in several ways by the administrator of the site, e.g. by adding a direct link from page 1 to page 25, or by suggesting a transfer to page 25, once the transition from page 1 to page 19 is observed.

5. RELATED WORK

This paper proposes the organisation of the users of a Web site into groups, exploiting user modelling and machine learning techniques. Machine learning methods have been applied to user modelling problems, mainly for acquiring models of individual users interacting with an information system. Such techniques are used by systems like FAB [1], IDL [11], Syskill & Webert [21] and Amalthea [17]. The main goal of these systems is to learn and revise user profiles as well as to propose interesting information to a user. WebWatcher [14] is an agent that suggests direct access to specific pages of a Web site that may be of particular interest to the user. LIRA [2] is an agent that autonomously searches the World Wide Web for interesting Web pages.

There are two main approaches for grouping users exploiting user modelling techniques: community and stereotype modelling [20]. Concerning the construction of user communities there are two main approaches. In the first one the user models themselves are used to reason about the interests of a new user. For instance, we could search for an old user B who shares most of his interests with the interests of the new user A and then use the user model of B to suggest extensions to the model of A. This type of reasoning is called *instance-based* and does not involve any learning, in the sense of drawing general rules from the data. The algorithm that is mostly used for this type of reasoning is called k-nearest-neighbour. The information systems that are based on this approach are called *recommender or collaborative filtering systems* [1,23].

The second approach for constructing user communities is to perform some kind of clustering. Although clustering seems a computationally expensive task compared to the case-based approach, this is not a real problem since communities change far less often than individual user models. On the other hand communities can be very useful since they can be used to re-organise the site in a way that is tailored to the needs of the users. However, this can be done effectively, only when the generated communities are meaningful. That's why we examined the use of the Frequency Indicator metric to characterise communities of visitors to the ACAI '99 Web site.

Although the clustering approach has already been used for the construction of communities [18], this is the first time, as far as we know, that meaningful community descriptions are generated for the visitors of a Web site.

6. CONCLUSIONS

This paper has described an approach to the analysis of Web usage data that goes beyond the widely-used Web usage statistics. The work presented here belongs in the research area of data mining as applied to data on the Web (also referred to as Web mining). The aim of the work is to identify communities of Web users that exhibit similar navigational behaviour with respect to a particular Web site. Towards this goal we have employed techniques from the areas of machine learning and user modelling. We have evaluated our method on the Web site of the Advanced Course on Artificial Intelligence 1999 and have shown how interesting navigational patterns can be discovered and utilised, in order to improve the services provided in the site.

The work presented here has raised numerous issues for further work. One such issue is the representation of training cases for the learning algorithm. We have shown the advantages of using page transitions, rather than treating access sessions as bags of pages. However, we have not considered higher-order representations that may give even more interesting results. A thorough study of the appropriate representation is necessary.

However, the main source of open questions is the utilisation of the results generated by an approach such as the one presented here. Concrete suggestions on the modification of the site is one option. An interesting alternative, is the construction of adaptive Web sites, which modify their behaviour according to the community in which they classify their visitors.

In conclusion, we believe that the employment of machine learning and user modelling techniques for web usage analysis is a very promising solution to the problem of making access to on-line information more efficient. This issue is becoming crucial as the size of the Web increases at breathtaking rates.

References

[1] M. Balabanovic and Y. Shoham, "Content-Based, Collaborative Recommendation," *Communications of the ACM,* Vol. 4, No. 3, 1997, pp. 66-72.

[2] M. Balabanovic and Y. Shoham, "Learning Information Retrieval Agents: Experiments with Automated Web Browsing," *Proceedings of AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Resources*, Stanford, CA 1995.

[3] E. Benaki, V. Karkaletsis, and C.D. Spyropoulos, "Integrating User Modelling Into Information Extraction: The UMIE Prototype" *Proceedings of the User Modelling conference UM'97*, 1997, pp.55-57.

[4] G. Biswas, J.B. Weinberg, and D. Fisher, "ITERATE: A Conceptual Clustering Algorithm for Data Mining," *IEEE Transactions on Systems, Man and Cybernetics,* Vol. 28, 1998, pp. 100-111.

[5] E. Bloedorn, I. Mani, and T.R. MacMillan, "Machine Learning of User Profiles: Representational Issues," *Proceedings of the National Conference on Artificial Intelligence (AAAI),* 1996, pp. 433-438.

[6] G. Brajnik, and C. Tasso, "A Shell for Developing Non-monotonic User Modelling Systems," *International Journal of Human-Computer Studies,* Vol. 40, 1994, pp. 31-62.

[7] G. Brajnik, G. Guida and C. Tasso, “User Modelling in Intelligent Information Retrieval,” *Information Processing and Management,* Vol. 23, 1987, pp. 305-320.

[8] P. Brusilovsky, E. Schwarz, “User as Student: Towards an Adaptive Interface for Advanced Web Applications,” *Proceedings of the User Modelling conference UM’97*, 1997, pp. 177-188.

[9] D.N. Chin, “KNOME: modelling what the user knows,” *User models in dialog systems*, Kobsa and Wahlster (eds): Springer-Verlag, Berlin, 1989, pp. 74-107.

[10] P. Chiu, “Using C4.5 as an Induction Engine for Agent Modelling: An experiment of Optimisation,” *Proceedings of the User Modelling conference UM’97,* Workshop on Machine Learning for User Modelling, 1997.

[11] F. Esposito, D. Malerba, G. Semeraro, N. Fanizzi, S. Ferilli, “Adding Machine Learning and Knowledge Intensive Techniques to a Digital Library Service,” *Int. Journal on Digital Libraries*, Vol. 2, 1998, pp. 3-19.

[12] D.H. Fisher, “Knowledge Acquisition via Incremental Conceptual Clustering,” *Machine Learning,* Vol. 2, 1987, pp. 139-172

[13] M.A. Gluck, and J.E. Corter, “Information, Uncertainty and the Utility of Categories,” *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*: Lawrence Erlbaum Associates, 1985, pp. 283-287.

[14] T. Joachims, D. Freitag, and T. Mitchell, “WebWatcher: A tour guide for the World Wide Web,” *Proceedings of the International Joint Conference in Artificial Intelligence*, 1997.

[15] J. Kay, “The um Toolkit for Cooperative User Modelling,” *User Modelling and User Adapted Interaction*, Vol. 4, 1995, pp. 149-196.

[16] P. Maes, “Agents that Reduce Work and Information Overload,” *Communications of the ACM,* Vol. 37, No. 7, 1994, pp. 31-40.

[17] A. Moukas, “Amalthaea: Information Discovery and Filtering using a Multiagent Evolving Ecosystem,” *Applied Artificial Intelligence: An International Journal*, Vol. 11, No. 5, 1997, pp. 437-457.

[18] J. Orwant, “Heterogeneous Learning in the Doppelgänger User Modelling System,” *User Modelling and User-Adapted Interaction,* Vol. 4, 1995, pp. 107-130.

[19] G. Paliouras, C. Papatheodorou, V. Karkaletsis, C.D. Spyropoulos and V. Malaveta, “Learning User Communities for Improving the Services of Information Providers,” *Lecture Notes in Computer Science*, No. 1513: Springer-Verlag, 1998, pp. 367-384.

[20] G. Paliouras, V. Karkaletsis, C. Papatheodorou, C.D. Spyropoulos, “Exploiting Learning Techniques for the Acquisition of User Stereotypes and Communities,” *Proceedings of the International Conference on User Modelling (UM ’99).*

[21] M. Pazzani and D. Billsus, “Learning and Revising User Profiles: The Identification of Interesting Web Sites,” *Machine Learning,* Vol. 27, 1997, pp. 313-331.

[22] B. Raskutti and A. Beitz, “Acquiring User Preferences for Information Filtering in Interactive Multi-Media Services,” *Proceedings of the Pacific Rim International Conference on Artificial Intelligence,* 1996, pp. 47-58.

[23] P. Resnick and H.R. Varian, “Recommender Systems,” *Communications of the ACM,* Vol. 4, No. 3, 1997, pp. 56-58.

[24] E. Rich, “Users are Individuals: Individualizing User Models,” *International Journal of Man-Machine Studies*, Vol. 18, 1983, pp. 199-214.

[25] M. Spiliopoulou, “The laborious way from data mining to web mining,” *Int. Journal of Comp. Sys., Sci. & Eng*., Special Issue on “Semantics of the Web,” March 1999.

[26] M. Virvou, J. Jones and M. Millington, “Virtues and Problems of an Active Help System for Unix,” *Journal of Artificial Intelligence Review*, 1999.