# Building a Recommender Agent for e-Learning Systems

Osmar R. Zaïane
University of Alberta, Edmonton, Alberta, Canada
zaiane@cs.ualberta.ca

## Abstract

*A recommender system in an e-learning context is a software agent that tries to "intelligently" recommend actions to a learner based on the actions of previous learners. This recommendation could be an on-line activity such as doing an exercise, reading posted messages on a conferencing system, or running an on-line simulation, or could be simply a web resource. These recommendation systems have been tried in e-commerce to entice purchasing of goods, but haven't been tried in e-learning. This paper suggests the use of web mining techniques to build such an agent that could recommend on-line learning activities or shortcuts in a course web site based on learners' access history to improve course material navigation as well as assist the on-line learning process. These techniques are considered integrated web mining as opposed to off-line web mining used by expert users to discover on-line access patterns.*

## 1. Introduction

Web-based learning environments are becoming very popular. Typical web-based learning environments, such as Virtual-U [3] and Web-CT [17], include course content delivery tools, synchronous and asynchronous conferencing systems, polling and quiz modules, virtual workspaces for sharing resources, white boards, grade reporting systems, logbooks, assignment submission components, etc. In a virtual classroom, educators provide resources such as text, multimedia and simulations, and moderate and animate discussions. Remote learners are encouraged to peruse the resources and participate in activities. However, it is very difficult and time consuming for educators to thoroughly track and assess all the activities performed by all learners on all these tools. Moreover, it is hard to evaluate the structure of the course content and its effectiveness on the learning process. Resource providers do their best to structure the content assuming its efficacy [18]. When instructors put together an on-line course, they may compile interactive course notes, simulations, demos, exercises, quizzes, asynchronous forums, chat tools, web resources, etc. This amalgam of on-line hyperlinked material could form a complex structure that is difficult to navigate. Designers and instructors, when devising the on-line structure of the course and course material, have a navigation pattern in mind and assume all on-line learners would follow a consistent path; the path put forth in the design and materialized by some hyperlinks. Learners, however, could follow different paths generating a variety of sequences of learning activities. Often this sequence is not the optimum sequence, and probably not the sequence intended by the designer. It is very difficult to assess the on-line learning activities in a web-based system.

What distinguishes the data that results from web-based activities in general and e-learning activities in particular, is the sheer complexity of the information and the vast size of the data collected, as well as the fact that simple information extraction is not possible. Information must be deduced by interactive data mining and associated visualization techniques. Visualization can be used either to visualize the patterns discovered by data mining, and thus is for evaluation and interpretation of the discoveries, or for the visualization of data itself. In this latter case visualization becomes part of data mining as an interactive process [10]. To understand the behaviour of web users accessing on-line resources, visualization is paramount. However, it is not clear how to analyze and visualize web usage data involving long sequences of on-line activities without loosing the big picture [2]. There are specific tools for visualizing web data such as interconnections of web resources [13], tools for visualizing web navigation patterns [4], and tools to visualize the changes in web sites or web usage over time [6]. None of these are good or even adequate for the assessment of e-learning activities or related on-line courses. The problem of efficiently and usefully visualizing e-learning web data remains an open research issue.

Educators, using Web-based learning environments, are in desperate need for non-intrusive and automatic ways to get objective feedback from learners in order to better follow the learning process and appraise the on-line course structure effectiveness. On the learner's side, it would be

Proceedings of the International Conference on Computers in Education (ICCE'02)
0-7695-1509-6/02 $17.00 © 2002 IEEE

very useful if the system could automatically guide the learner's activities and intelligently recommend on-line activities or resources that would favour and improve the learning. The automatic recommendation could be based on the instructor's intended sequence of navigation in the course material, or, more interestingly, based on navigation patterns of other successful learners.

In this paper, we suggest the construction of such automatic recommendation system for Web-based learning environments that takes into account profiles of on-line learners, their access history and the collective navigation patterns, and uses simple data mining techniques, namely association rule mining. The following section presents web mining in the context of e-learning. Section 3 discusses recommendation systems. Association Rule mining is introduced in Section 4 as well as hints on how a recommendation system could be built. Finally, Section 5 presents some concluding remarks.

## 2. Use of web mining for developing smart e-learning systems

All accesses to a web site or a web-based application are tracked by the web server in a log containing chronologically ordered transactions indicating that a given URL was requested at a given time from a given machine using a given web client (i.e. browser). Web server log files customarily contain: the domain name (or IP address) of the request; the user name of the user who generated the request (if applicable); the date and time of the request; the method of the request (GET or POST); the name of the file requested; the result of the request (success, failure, error, etc.); the size of the data sent back; the URL of the referring page; the identification of the client agent; and a cookie, a sting of data generated by an application and exchanged between the client and the server. A log entry is automatically added each time a request for a resource reaches the web server. These log entries are not in a format that is usable by mining applications and require to be reformatted and cleansed in order to identify real session information, path completion, etc. [8]. There exist some statistical tools that give rudimentary analysis of the web logs and provide reports on the most popular pages, the most active visitors, etc. in given time periods [9]. However, these tools offer very few insights on what is really happening in the web site, and the analysis provided are shallow [20] since these tools basically count page accesses (called hits) independently. In other words, their ability to help understand the implicit usage information and hidden trends in learners' on-line access behaviour is very limited.

Data mining is a pivotal step in the overall process of knowledge discovery from data. This process, also known as KDD, comprises data collection, data cleaning and preprocessing, data integration, data selection, data mining, and finally the evaluation of discovered patterns, possibly using interactive visualization. Data mining is the step in which, using advanced techniques, interesting and potentially useful patterns are extracted from a set of large, already cleansed data sources [12]. These advanced techniques comprise automatic classification of data, clustering, the discovery of associations and correlation between data, characterization and summarization of data, discovery of discriminant features, identification of outliers, etc.

Applying data mining techniques on web logs to discover useful navigation patterns or deduce hypothesis that can be used to improve web applications is the main idea behind web usage mining. Web usage mining can be used for many different purposes and applications such as user profiling and web page personalization, server performance enhancement, web site structure improvement, pre-fetching, etc. [15]. Some new experimental tools use data mining techniques to extract hidden patterns from the large web logs. Systems such as WebSIFT [7] and WebLogMiner [20] are sets of comprehensive web usage tools that are able to perform many data mining tasks and discover a variety of patterns from web logs. These are versatile systems that discover interesting patterns such as associations between visited web pages, frequent sequential patterns, etc. However these wide-ranging tools are not integrated in e-learning systems and it is cumbersome for an educator who doesn't have extensive knowledge in data mining to use these tools to improve the effectiveness of web-based learning environments [18]. A new web usage mining system dedicated for e-learning is being developed to allow educators to assess on-line learning activities [19]. This system, however, is what we could call Off-line Web Mining. It is an aftermath analysis that could give some hints on how an on-line course is effectively used and how its structure could be improved based on a comparison of the intended usage of the course material vis-à-vis the real usage tracked in the web log. While most data mining algorithms need specific parameters and threshold values to tune the discovery process, the users of web usage mining applications in the context of e-learning, namely educators and e-learning site designers, are not necessarily savvy in the intricate complexities of data mining algorithms [18]. Whereas it is possible, with significant challenge, to design parameter-free algorithms, simplifying the data mining process for the non-experts, it remains a great challenge for database and machine learning researchers [11]. Off-line web usage mining, if enhanced with a good user interfaces, can help educators put in question and validate the learning models they use as well as the structure of the web site as it is perused by the learners. What we advocate in this paper is what we call Integrated Web Mining. With integrated web usage mining, the patterns automatically discovered are fed into an intelligent software system that would assist learners in their on-line learning endeavours. In other words, mined patterns are used on-the-fly by the system to improve the application or its functions. One attractive example is an adaptive web site that makes use of a user's access history to personalize page layouts and web site structure automatically [14]. This could be crucial for customer loyalty for instance. One other example of integrated web mining is a recommendation system that suggests actions or resources to a user, often called recommender agent. This software agent "learns" from past activities of one user or a group of users, and predicts activities or pages that a given user might be interested in before suggesting them to the user. Some have also suggested hyperlink shortcuts by shortening frequent web access sequences discovered in the web log [16].

## 3. Recommendation Systems for E-Learning

An "e-learning task recommender" is a recommendation system that would recommend a learning task to a learner based on the tasks already done by the learner and their successes, and based on tasks made by other "similar" learners. The similarity of the learners could be established using user profiles, or could be based on common previous access patterns. In principle, there are two major parts in the design of such an agent: a "learning" module that learns from past access patterns and infers an individual or common access model; and an "advising" module that applies the learned model at given times to recommend actions. There are many ways to implement this process, such as data clustering, association rule mining, or collaborative filtering [5], etc.

Software agents that recommend actions, products, or other items have been used in some applications. However, to the best of our knowledge there is no distance leaning system to date that provides such automated facilities to automatically suggest learning activities or resources. In the field of electronic commerce, however, given the lucrative prospects, a significant research effort has been made to devise elaborate methods to take advantage of customers' accesses and purchase behaviours in order to enhance the purchasing experience and customer satisfaction by user profiling and smart recommendations, and thus increase profit. Recommenders are used to boost sales by displaying products or services a consumer is likely to be interested in. For example, systems for recommendation such as Amazon.com that suggests books or other products to purchase related to a current purchase, based on preference information and other users purchases. The techniques used are, however, very simple and not always accurate or even effective. Basically, the program compares the set of items purchased by the current customer with the set of items purchased by other customers, selects the customers with


**Figure 1. Example of e-Learning action recommendation by the Recommender Agent**


**Figure 2. Example of shortcut suggestions by the Recommender to improve navigation**

the bigger item overlap with the current customer's item set, then finally picks some items not yet bought by the customer but present in the baskets of customers with high overlap and presents them as a recommendation list to the current customer. More sophisticated methods take into account ratings of products given by customers and select products for recommendation from customers that rate items the same way or level as the current customer. This technique also used in information retrieval for retrieving text documents that are similar is called collaborative filtering [5]. Recommendation of movies or music compact discs, such as moviefinder.com, uses collaborative filtering by predicting a person's preferences as a linear weighted combination of other people's preferences.

Since we do not have ratings for course material, collaborative filtering is not applicable for devising an accurate recommender agent for e-learning activities. Moreover, we are interested in recommending beneficial learning activities to enhance on-line learning, as well as recommending shortcuts or jumps to some resources to help users better navigate the course materials. Figure 1 and 2 illustrate these two cases from our prototypical e-learning Recommender Agent. These recommendations are triggered after some particular well-defined events. Figure 1 shows an action recommendation that is triggered by the learner attempting to take a test. In other words, the event $start(action, quiz)$ would activate the agent which would compare the learner's profile and previous sequences of actions with the model already learnt at pre-processing time using the web log and other learners' profiles. In this case the learner is advised to consult some unread material. Figure 2 shows a shortcut predictor that suggests a list of direct links the learner may consider following. The prediction is based on the current sequence of activities or pages visited by the learner and the frequent sequences of visited pages or learning activities other users did in the past. This predictor is triggered by an event encompassing a given sequence of actions. In other words, the event $exist(sequence, s)$ where $s$ is the a sequence of activities, would activate the predictor if the current sequence is a prefix of the sequence $s$. Such a software agent can be implemented using association rules.

## 4. Building a Recommender Agent

Web usage mining performs mining on web data, particularly data stored in logs managed by the web servers. The web log provides a raw trace of the learners' navigation and activities on the site. In order to process these log entries and extract valuable patterns that could be used to enhance the learning system or help in the learning evaluation, a significant cleaning and transformation phase needs to take place so as to prepare the information for data mining algorithms [18]. Web server log files of current common web servers contain insufficient data upon which to base thorough analysis. However, they contain useful data from which a well-designed data mining system can discover beneficial information and which can provide a basis for model building. The model we use to construct our recommender system is based on association rules.

### 4.1. Association Rule Mining

Association rules are one of the typical rule patterns that data mining tools aim at discovering. They are very useful in many application domains, but are mainly applied in the business world as in market-basket analysis. In a transactional database where each transaction is a set of items bought together, association rules are rules associating items that are frequently bought together. A rule consists of an antecedent (left-hand side) and a consequent (right-hand side). Example: $I_1, I_2, ...I_n \Rightarrow I_\alpha, I_\beta, ...I_\gamma$. The intersection between the antecedent and the consequent is empty. If items in the antecedent are bought then there is a probability that the items in the consequent would be bought as well at the same time. An efficient algorithm to discover these association rules was first introduced in [1]. The algorithm constructs a candidate set of frequent itemsets of length k, counts the number of occurrences, keeps only the frequent ones, then constructs a candidate set of itemsets of length k+1 from the frequent itemsets of smaller length. It continues iteratively until no candidate itemset can be constructed. In other words, every subset of a frequent itemset must also be frequent. The rules are then generated from the frequent itemsets with probabilities attached to them indicating the likelyhood (called support) that the association occurs.

We use this idea of association rules to train our recommender agent to build a model representing the web page access behaviour or associations between on-line learning activities.

### 4.2. E-learning Recommender with Association Rules

A recommender system suggests possible actions or web resources based on its understanding of the user's access. To do so we have to translate the entries in the web log into either known actions (i.e. learning activities such as accessing a course notes module, posting a message on the forum, doing a test, trying a simulation, etc.) or URLs of a web resource. This mapping is a significant processing phase that in itself presents a considerable challenge [20, 19]. Moreover, these identified actions and URLs are grouped into sessions which is yet another difficult and delicate task [18]. These sessions are then modeled into transactions as sets of actions and URLs. The association rule mining technique is applied on such transactions to discover associations between actions, associations between URLs and associations between actions and URLs, as well as associations between sequences of actions and/or URLs. This process usually leads to a very large number of association rules even after filtering out those that do not satisfy the requirement of minimum support [1]. We use other specific filtering approaches to eliminate such discovered rules that associate two URLs that are directly linked from each other. Indeed, it is useless to recommend a page that is directly linked from the current page as a shortcut. Moreover, we give higher weights to rules that have as a consequent a URL or a set of URLs that are frequently towards the end of a session. For the rules that associate actions, we keep only rules that have as a consequent an action that terminated successfully. For instance, if the action is taking an on-line test, it is only useful to recommend that action after a sequence of actions if that test was successful. In other words, actions are labelled whenever possible with "successful" or "unsuccessful" using the users' profiles (i.e. gradebook).

When the recommender agent is activated by a triggering event, the association rules are consulted to check for matches between the triggering event, or sequence of events, with the rule antecedents. When a match is found, the consequent of the rule is suggested. If more matches are found, the suggestions are ranked and only a small set (highest ranked) is displayed.

## 5. Conclusion

A recommender system is a program that sees what a user is doing and tries to recommend courses of action it thinks would be beneficial to the user. This is the idea behind some systems used in electronic commerce sites to recommend products to customers they might wish to purchase based on their previous purchasing history as well as the purchasing history of those who bought similar goods. To date, this hasn't been proposed for on-line learning environments and no known e-learning system uses such a software agent to enhance the on-line learning experience as described in this paper.

We have proposed an approach to build a software agent that uses data mining techniques such as association rules mining in order to build a model that represents on-line user behaviours, and uses this model to suggest activities or shortcuts. These suggestions can help learners better navigate the on-line materials by finding relevant resources faster using the recommended shortcuts and assist the learner choose pertinent learning activities that should improve their performance based on on-line behaviour of successful learners.

We are currently testing this recommender system approach on an on-line course and will evaluate the recommendations using questionnaires as well as a log that is keeping track of selected recommendations by the users. The approach is also tested on an on-line system used by novice health care providers at the university hospital at the University of Alberta and will be evaluated based on the time-saving recorded for users who follow the suggested shortcuts in comparison with those that ignore the recommendations.

## References


[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. 1993 ACM-SIGMOD Int. Conf. Management of Data*, pages 207–216, Washington, D.C., May 1993.

[2] B. Berendt. Detail and context in web usage mining: Coarsening and visualizing sequences. In R. Kohavi, B. Masand, M. Spiliopoulou, and J. Srivastava, editors, *WEBKDD 2001-Mining Web Log Data Across All Customer Touch Points*, pages 1–24. Springer Verlag, 2002.

[3] T. C. C. Groeneboer, D. Stockley. Virtual-u: A collaborative model for online learning environments. In *Second International Conference on Computer Support for Collaborative Learning*, Toronto, Canada, December 1997.

[4] I. Cadez, D. Heckerman, and C. Meek. Visualization of navigation patterns on web site using model based clustering. In *ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD'00)*, pages 280–284, Boston, USA, August 2000.

[5] S. Chee, J. Han, and K. Wang. Rectree: An efficient collaborative filtering method. In *3rd Int. Conf. On Data Warehousing and Knowledge Discovery (DAWAK 2001)*, LNCS 2114, pages 141–151, Munich, Germany, September 2001. Springer Verlag.

[6] E. Chi, J. Pitkow, J. Mackinlay, P. Pirolli, and J. Konstan. Visualizing the evolution of web ecology. In *ACM CHI'98 Conference on Human Factors in Computing Systems*, page 644.645, Los Angeles, USA, 1998.

[7] R. Cooley, B. Mobasher, and J. Srivastava. Web mining: information and pattern discovery on the world wide web. In *9th IEEE International Conference on Tools with Artificial Intelligence*, pages 558–567, 1997.

[8] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.

[9] H. A. Edelstein. Pan for gold in the clickstream, March 2001. http://www.informationweek.com/828/mining.htm.

[10] U. Fayyad, G. Grinstein, and A. Wierse. *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann Publisher, 2001.

[11] A. Foss, W. Wang, and O. R. Zaïane. A non-parametric approach to web log analysis. In *Web Mining Workshop in conjunction with the SIAM International Conference on Data Mining*, pages 41–50, Chicago, IL, USA, April 2001.

[12] J. Han and M. Kamber. *Data Mining, Concepts and Techniques*. Morgan Kaufmann, 2001.

[13] T. Munzner and P. Burchard. Visualizing the structure of the world-wide web in 3d hyperbolic space. In *Proceedings of VRML'95*, 1995.

[14] M. Spiliopoulou, L. C. Faulstich, and K. Winkler. A data miner analyzing the navigational behaviour of web users. In *workshop on Machine Learning in User Modeling of the ACAI'99*, Creta, Greece, July 1999.

[15] J. Srivastava, R. Cooley, M. Deshpande, and P. Tan. Web usage mining: Discovery and applications of usage patterns form web data. *SIGKDD Explorations*, 1(2), January 2000.

[16] R. G. T. Zheng, Y. Niu. Webframe: in pursuit of computationally and congnitively efficient web mining. In *6th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 264–275, Taipei, Taiwan, May 2002.

[17] Webct, available on August 2002. http://www.webct.com/.

[18] O. R. Zaïane. Web usage mining for a better web-based learning environment. In *Proc. of Conference on Advanced Technology for Education*, pages 60–64, Banff, AB, June 2001.

[19] O. R. Zaïane and J. Luo. Towards evaluating learners' behaviour in a web-based distance learning environment. In *Proc. of IEEE International Conference on Advanced Learning Technologies (ICALT01)*, pages 357–360, Madison, WI, August 2001.

[20] O. R. Zaïane, M. Xin, and J. Han. Discovering web access patterns and trends by applying OLAP and data mining technology on web logs. In *Proc. Advances in Digital Libraries ADL'98*, pages 19–29, Santa Barbara, CA, USA, April 1998.