Workshop on Artificial Intelligence

A HYBRID RECOMMENDATION SYSTEM WITH VECTOR PROFILE BASED ON FUZZY CLUSTERING

Yi-Cheng Chen and Robert Lai

Department of Computer Engineering & Science
Yuan Ze University, Chung-Li, Taiwan 32026
krlai@cs.yzu.edu.tw

ABSTRCT

To cope with the problems of information overloading and low signal to noise ratio, a number of potential solutions based on information filtering or web mining have been proposed. In this paper, we present a hybrid information filtering mechanism into the development of a recommendation system for distance learning. The web pages and users' logs are represented into vector profiles first. The resulting profiles, then, are classified into peer-group based on fuzzy clustering. The similarity among profiles is characterized as a membership function allowing system to provide recommendation flexibly. Case-based reasoning is also incorporated into the system to support the controlling of browsing order in e-Learning. Besides, in order to catch the rapid changes of users' interests, a corresponding feedback mechanism is also included. Finally, we have also implemented a web site, On-Line Learner (OLLer), and set up several experiments to evaluate the effectiveness of our system.

Keywords: e-Learning, Information Filtering, Fuzzy Clustering, Recommendation System

A HYBRID RECOMMENDATION SYSTEM WITH VECTOR PROFILE BASED ON FUZZY CLUSTERING

Yi-Cheng Chen and Robert Lai

Department of Computer Engineering & Science
Yuan Ze University, Chung-Li, Taiwan 32026
krlai@cs.yzu.edu.tw

ABSTRCT

To cope with the problems of information overloading and low signal to noise ratio, a number of potential solutions based on information filtering or web mining have been proposed. In this paper, we present a hybrid information filtering mechanism into the development of a recommendation system for distance learning. The web pages and users' logs are represented into vector profiles first. The resulting profiles, then, are classified into peer-group based on fuzzy clustering. The similarity among profiles is characterized as a membership function allowing system to provide recommendation flexibly. Case-based reasoning is also incorporated into the system to support the controlling of browsing order in e-Learning. Besides, in order to catch the rapid changes of users' interests, a corresponding feedback mechanism is also included. Finally, we have also implemented a web site, On-Line Learner (OLLer), and set up several experiments to evaluate the effectiveness of our system.

Keywords: e-Learning, Information Filtering, Fuzzy Clustering, Recommendation System

1. INTRODUCTION

Rapid development of Internet makes users differentiating what information is really needed more difficult. Even if users can filter web pages that do not match the query through search engines, the number of resulting web pages is still amazing large. On the other hand, the web where browsing one site from another requires only two or three clicks makes service providers competing with each other fiercely. It is not enough to meet users' diverse tastes with traditional and static contents. In other words, to make information personalized is highly desirable. While facing overloaded information, it is better to allow system with the ability of analyzing users' profiles providing information needed automatically, rather than to make users worried about which query is appropriate. Similarly, for diverse users, the system needs to provide right services to right users as well.

To cope with the problems of information overloading and low signal to noise ratio, a number of potential solutions based on information filtering [1,3,7,10,11,13,15] or web mining [8,9,12] have been proposed. Specifically, a content-based filtering, such as CASPER project [3], provides recommendation by analyzing the content items rated in the past. Then, in [10,15], a collaborative filtering measures the similarity between users' profiles, and gives recommendation with the suggestion of like-minded users. While both of content-based filtering and collaborative filtering are sound approaches, there are shortcomings as well. As a result, various hybrid filtering techniques have been proposed [1,7,11].

In this paper, we present a hybrid information filtering mechanism into the development of a recommendation system for distance learning. The web pages and users' logs are represented into vector profiles first. The resulting profiles, then, are classified into peer-group based on fuzzy clustering. The similarity among profiles is characterized as a membership function allowing system to provide recommendation flexibly. Case-based reasoning is also incorporated into the system to support the controlling of browsing order in e-Learning. Besides, in order to catch the rapid changes of users' interests, a corresponding feedback mechanism is included. Finally, to evaluate the effectiveness of our system, experimental results are also presented.

The remainder of this paper is organized as follows. Section 2 presents the design of a recommendation system for e-Learning, including the system architecture, vector profiles, peer-group generation, inference routine, and feedback mechanism. In Section 3, we, then, discuss the performance evaluation and experimental results followed by some concluding remarks in Section 4.

2. SYSTEM DESIGN

2.1 Architecture

A recommendation system is often meant to provide personalized services based on users' behavior in the past. Conversely, users' active sessions also trigger system to update its knowledge in order to make future services closer to what users really need. Consequently, from a programmer's perspective, a recommendation system can be illustrated as Figure 1.

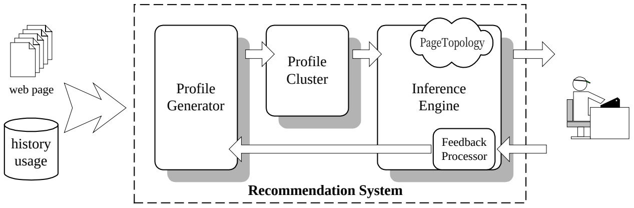


Figure 1: The architecture of a recommendation system for e-Learning

In Figure 1, the task of profile generator is to transform source data into vector profiles; profile clustering, then, analyzes the profiles to uncover relation among users or articles with fuzzy clustering algorithms. Based on the similarity between users' profiles and content profiles, inference engine provides suitable recommendation after some operations. Finally, as users' interests changing, feedback processor is to update the knowledge base to prevent recommendation from being out of date.

In what follows, we give the details of vector profiles, peer-group generation, inference routine, and feedback mechanism in our system.

2.2 Vector Profiles

According to different sources of data, mainly articles, users' registration data, and access records left by users, they can be characterized into proper vector formats as follows.

- Content Feature Vector (CFV)
  Adopted from the function in [13], each article can be translated into a vector $(w_1, w_2,...w_i,..., w_n)$, where w_i denotes the weight of i th keyword in the article.
- User Behavior Vector (UBV)
  While Boolean Model [6] can be utilized to record users' browsing paths generally, it is not scalable. The dimension of model increases rapidly as the web site becomes larger. Moreover, a factor that pages visited are extremely fewer than those of the site makes it storing inefficiently. To address this, besides storing only the articles visited, we also take browsing sequence and time spent into consideration. As a result, the behavior of a particular user can be represented as a tuple:

  $$((a_1,T_{a_1}),...,(a_i,T_{a_i}),...,(a_n,T_{a_n})) \quad (1)$$

  where a_i denotes the visited article; T_{a_i} indicates the time spent in article a_i.
- User Domain Vector (UDV)
  As users enter the system at first time, they will be asked to provide some background (i.e. sex, job, marriage status, age, education degree, and residence, etc). Then, each user's domain information is characterized as $(v_1,v_2,..,v_i,...,v_n)$, where v_i denotes the value of i th attribute.
- User Preference Vector (UPV)
  In order to facilitate clustering and comparison later, the format of UPV is derived from CFV, $(w_1, w_2,...w_i,..., w_n)$. The element of UPV is calculated with following formula

  $$\sum_{a=1}^{n}\frac{T_a}{T}\times CFV_a \quad (2)$$

  where T indicates the total time spent and T_a is time spent in article a . Moreover, CFV_a is the content feature vector of the article a . A UPV represents a user's preference of keywords based on CFVs.

2.3 Peer-Group Generation

Here, we adopt SCM [14] to uncover the relation among objects. Not only does the membership of fuzzy clustering make collection of like-minded users' suggestion more flexible, but also facilitates selection of personalized articles. In distance education, we could have the following peer-groups.

- Article Cluster
  Articles with similar subjects are classified into the same group.
- User Interest Community
  Similarly, users with similar interests belong to the same cluster. The system updates UPV of a particular user in accordance with associated recent usage and re-classified the user into new group, if necessary. In other words, users are not always in fixed interest community.
- User Domain Community
  Without considering browsing behavior, users within the same community have the similar background. Unlike User Interest Community, User Domain Community generated with users' registration data is more stable

2.4 Inference Routine

Based on knowledge base built with resulting peer-groups inference engine provides personalized articles, and its inner structure is shown in Figure 2.

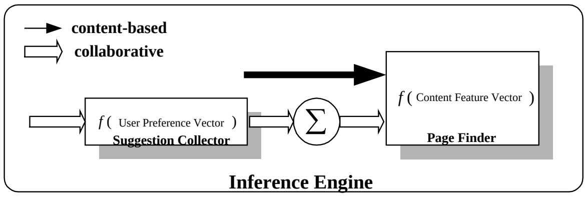


Figure 2: Inner structure of inference engine

In Figure 2, suggestion collector is responsible for gathering the suggestion of like-minded users, and is used only for collaborative filtering. After selecting like-minded users based on memberships of the active user, it sums up their UPVs as a hybrid vector profile. Then, the task of page finder is to choose suitable articles for recommendation. Unlike suggestion collector, it supports content-based and collaborative filtering both. Its input is either a summation of like-minded users' UPVs for collaborative filtering or a CFV for content-based filtering. If its input is a hybrid profile, a peer-group is selected with cosine vector similarity [13] firstly, and then some articles are picked from the peer-group. Nevertheless, for content-based filtering, page finder simply chooses articles which memberships are similar to those of the CFV.

Note that the system does not recommend the resulting articles directly, because it has to be restricted by PageTopology, a component designed specifically for e-Learning. The order of recommended pages in e-Learning is more significant than that of other areas. Clearly, a way from easy articles to difficult ones makes users learning more efficiently. Not only does PageTpology record the most popular learning paths, but also stores some huristic rules. For example, PageTopology contains following page orders, {(wireless technology, GSM, GPRS), (wireless technology, Bluetooth), (wireless technology, wireless LAN)}. Then, given a user read articles about wireless technology and Bluetooth, whereas the articles selected by page finder are about GPRS and wireless LAN. According to PageTopology, the user has better to realize GSM before reading GPRS. As a result, system will provide the articles about GPRS and GSM both, because the user may not involve GSM before. This mechanism ensures that users learn step by step and prevents beginners from starting in difficult articles.

2.5 Feedback Mechanism

With the articles read by users accumulating, the change of UPV will trigger profile cluster to re-classify. On the other hand, a constant between 0 and 1 need to multiply by UPV. This mechanism reduces the effects of users' past preferences because times of multiplication of the order preference is more than those of the newer one. This constant controls the percentage of users' recent preference affecting the recommendation.

Then, before updating PageTopology, we have to clarify the relation of pages in traversal references firstly. MFR algorithm [4] is used to eliminate the case that some pages are revisited because of its location, rather than its content. According the transitivity of paths, we can split a browsing path into several sub-paths in order to facilitate modification. Assume that a page reference {C,B,A} followed by 80% users, i.e. the degree of support is 0.8, and it can be split into two page references {C,B} and {B,A} with same support 0.8. Also, if there is a page reference {A,B,C,D} with support 0.7 in PageTopology, similarly, {A,B},{B,C} ,and {C,D} with support 0.7 are obtained. Then, the paired-pages with higher support replace those with lower support. Consequently, a new page reference {C,B,A} with support 0.8 is generated, and the page reference {C,D} with support 0.7 is reserved. The degree of support is calculated by the formula,

$$S_{o_i} = \frac{count_{o_i}}{N} \qquad (3)$$

where N is the number of users who read the articles of order o_i, and $count_{o_i}$ indicates the number of users who follow the order o_i.

3. PERFORMANCE EVALUATION

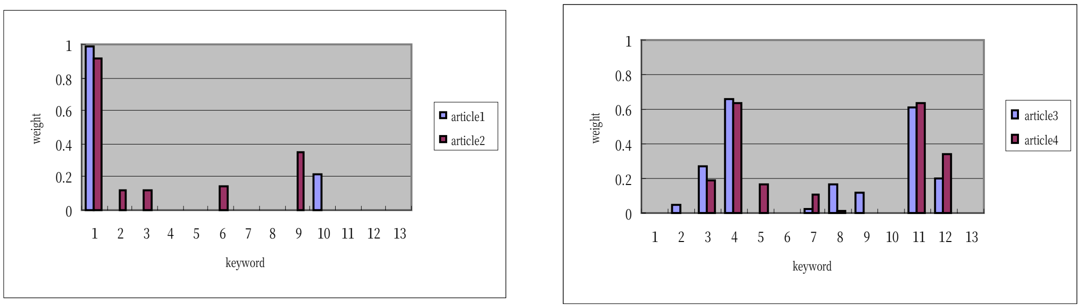
3.1 Experimental Settings

To evaluate the effectiveness of our design, we have implemented a web site, On-Line Learner (OLLer). A dozen of papers about information filtering techniques, and eight articles about data mining [2] are selected for this evaluation. Also, the keyword thesaurus is built based on corresponding bibliographic data. The number of simulated users is almost 10,000, and the majority are college professors and students. Then, we define their behavior as three types: long-term users, short-term users, and naïve users. Besides, the percentage of long-term preference users is 70% among one-month web log files following the format of IIS Server Web Log.

3.2 Experimental Results

We describe several preliminary experiments as follows.

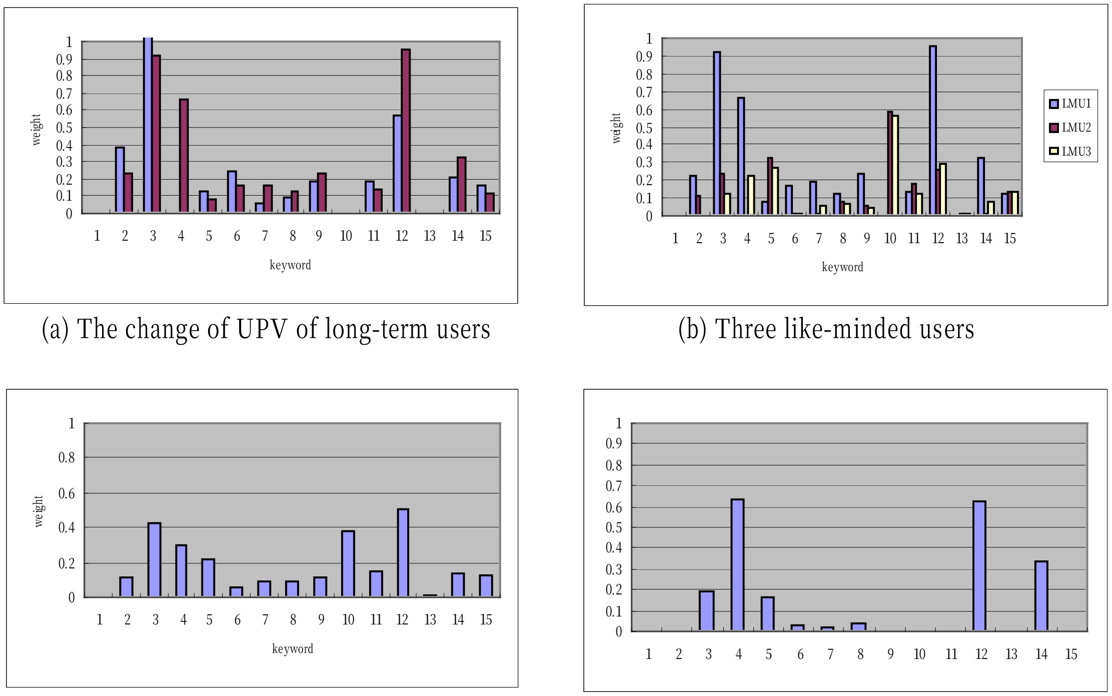
- **Articles Clustering Test**: In this test, OLLer divides articles into three clusters. Then, two articles with high and similar memberships are selected from two clusters respectively, and their CFVs are illustrated with Figure 3. The x-axis of Figure 3 is a set of related keywords; the y-axis indicates the weight of the



(a) Cluster 1

(b) Cluster 2

Figure 3: The features of clusters



(a) The change of UPV of long-term users

(b) Three like-minded users

(c) The hybrid user profile

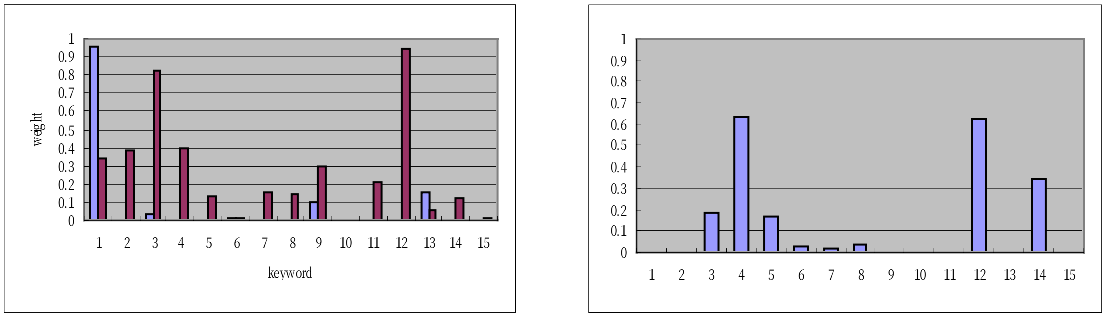
(d) The CFV of collaborative recommendation

Figure 4: The Long-term User Test

keyword. From the value of each keyword, it is obvious that the features of articles within same cluster are similar, and that the features between clusters are different. Based on the articles users read, OLLer will provide related articles by comparing with memberships. Consequently, not only does the results show the clustering validity, but also the quality of content-based recommendation is guaranteed.

- **Long-term Users Test**: During three recent accesses, feedback processor updates the corresponding user profile based on articles the user selected. Figure 4(a) shows the change of the user profile. Since the values of keywords do not change drastically, the user can be considered a long-term user who always read articles with similar topics. The x-axis of Figure 4 is partial keywords and y-axis is the weight of corresponding keyword.

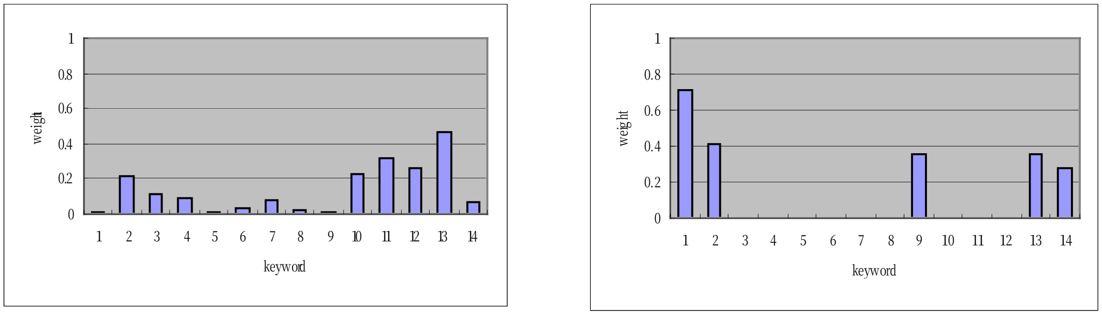
Note that owing to the independence of content-based recommendation and user profiles, this experiment only aims at the collaborative recommendation. Firstly, OLLer will select like-minded users shown in Figure 4(b), and sums up their UPVs into a hybrid profile in Figure 4(c). While the second and the third like-minded users are not similar with the first, all of them have something in common, such as the weights of 11th and 15th keyword. With collaborative filtering, OLLer does not restrict its



(a) The change of UPV of short-term users

(b) The CFV of collaborative recommendation

Figure 5: The Short-term User Test



(a) The UPV of naive users

(b) The CFV of collaborative recommendation

Figure 6: The Naïve Users Test

recommendation only within users' past preference. This way may uncover potential fields users have not involved yet.

- **Short-term Users Test**: Figure 5(a) depicts the profile of the user whose preference changes in less than three accesses. Similarly, OLLer selects like-minded users, and generate a hybrid profile based on Figure 5 (a). We pick fifteen keywords to illustrate the CFV of collaborative recommendation with Figure 5(b). Even if users' preference varies rapidly, OLLer provides personalized articles by referring the user's new preference.
- **Naïve Users Test**: The targets of this experiment are the users without any particular preference, so called naïve users. The typical characteristics of their user profiles are low value for each keyword shown in Figure 6 (a). Besides, this experiment also demonstrates the effectiveness of PageTopology. According to PageTopology, the original recommended article is difficult for this case, so OLLer prompts to take some requisites before reading the article. Of course, users still can continue learning without reading other articles in advance. The CFV of collaborative recommendation is depicted in Figure 6 (b). Not only does the resulting recommendation cover past preference, but also includes new topics users may be interested in.
- **New Arrival Test**: This experiment demonstrates how to solve the cold-start problem of traditional collaborative systems with User Domain Community and content-based filtering. Collaborative filtering mainly depends on suggestion of like-minded users. Consequently, quality of collaborative filtering is not guaranteed as users' preference is out of ordinary. Especially, this case often occurs when web site is established initially. In this case, OLLer will select the users with similar background rather than like-minded users. Similarly, OLLer obtains a hybrid user profile and pick several articles for recommendation. If it is also difficult to select users with similar background, content-based filtering will provide users related articles in accordance with users' active sessions. In OLLer, both content-based recommendation and User Domain Community eliminate effects from the cold-start problem.

4. CONCLUSION

We have presented a recommendation system for e-Learning based on hybrid filtering and fuzzy clustering. While our approach shows some promising results, many work remain to be done, including extra assessment strategies and multi-media processing.

REFERENCES

[1] Basu, Chumki, Haym Hirsh and William Cohen, 1998, “Recommendation as Classifcation:Using Social and Content-Based Information in Recommendation,” *Proceedings AAAI*, pp. 714-720.

[2] Berry, Michael J.A. and Gordon Linoff, 1997, *Data Ming Techniques For Marketing, Sales, and Customer Support*, *John Wiley & Sons*, NY.

[3] Bradly, Keith, Rachael Rafter and Barry Smyth, 2000, “Case-Based User Profiling for Content Personalization,” *AH*, pp. 62-72.

[4] Chen, Ming-Syan, Jong Soo Park and Yu, P.S., 1998, “Efficient data mining for path traversal patterns,” *in IEEE Trans. on Knowledge and Data Engineering*, vol. 10 Issue 2, pp. 209 –221.

[5] Hoppner, Frank, et al., 1999. *Fuzzy Clustering Analysis*, Wiley, NY

[6] Keith, C.J., 1979, *Information Retrieval, Butterworths*, London.

[7] Kohrs, Arnd and Bernard Merialdo, 2000, “Using Category-Based Collaborative Filtering in the Active WebMuseum,” *Proceedings of IEEE International Conference on Multimedia and Expo*,vol. 1, pp. 351 –354.

[8] Lee, Chung-Hong and Hsin-Chang Yang, 2001, “Developing an Adaptive Search Engine for E-Commerce Using a Web Mining Approach,” *Proceedings of International Conference on Information Technology: Coding and Computing*, pp. 604 –608.

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

[10] Resnick, Paul, et al., 1994, “GroupLens：An Open Architecture for Collaborative Filtering of Netnews,” *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pp. 175-186.

[11] Smyth, Barry and Paul Cotter, 1999, “Surfing the Digital Wave-Generating Personalized TV Listings using Collaborative, Cased-Based Recommendation,” *Lecture Notes in Computer Science*, vol. 1650, pp. 561-572.

[12] Sung, Ho-Ha, Min-Bae Sung and Chan Park Sang, 2000, “Web Mining for Distance Education,” *Proceedings of the IEEE International Conference on Management of Innovation and Technology*, vol.2, pp. 715 –719.

[13] Wu, Kun-Lung, Aggarwal, C.C., Yu, P.S., 2000, “Personalization with Dynamic Profiler,**“** *Third International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems*, pp. 12 –20.

[14] Wu, Kuo Lung, 2000,「Similarity C-Means Clustering Algorithm」,Chung-Yuan Christian University, Master Thesis.

[15] Wu, Yi-Hung, Tong-Chuan Chen and Arbee L. P. Chen, 2001, “Enabling Personalized Recommendation on the Web Based on User Interests and Behaviors,” *Proceedings of Eleventh International Workshop on Research Issues in Data Engineering*, pp.17-24.

[16] Xiao, Jitian and Yanchun Zhang, 2001, “Clustering of web users using session-based similarity measures,” *Proceedings of 2001 International Conference on Computer Networks and Mobile Computing*, pp. 223 –228.