Fuzzy Collaborative Filtering for Web Page Prediction

Chein-Shung Hwang[1] Yu-Pin Chen[2]

Dept. of Information Management, Chinese Culture University,
55 ,Hwa-Kang Rd., Yang-Ming-Shan, Taipei, Taiwan, 111, R.O.C.
[1]E-Mail: cshwang@faculty.pccu.edu.tw [2]E-Mail: speedft@yahoo.com.tw

Abstract

Web page prediction is a popular personalized service on the Web and has attracted much research attention. One of the most successful and widely used approaches is collaborative filtering. Traditional collaborative filtering requires explicit user participation for providing his/her interest to the pages. However, they suffer from some limitations such as additional user effort, user behavior alteration and data sparsity. A possible alternative is to use an implicit indicator which measures the time a user spends on a page. In this paper, we propose a new collaborative filtering algorithm based on the fuzzy set theory, which is designed to better address the sharp boundary problem of discretizing the viewing time. The experimental results show that the proposed methods can achieve a better performance than the Traditional CF without time information.

Keywords: Fuzzy set, Page prediction, Collaborative Filtering

1. Introduction

The explosive growth of the world-wide-web has led to an influx of users and consequently, a huge increase in the volume of available on-line data. The volume of things is considerably more than any person can possibly filter through to find the ones that he/she will like. Many web sites have developed various personalized services to assist users in finding the information they need more quickly and easily. One of them is to predict the next request of a user as he/she visits Web pages. The prediction can be made based on the web contents or the web usage information. Content-based prediction recommends pages based on the page contents that the user has visited previously. On the other hand, usage-based prediction uses the user access behavior to a Web site as base information for its predictive model.

A number of different approaches to web page prediction have been proposed, including collaborative filtering (CF) [4], Markov models [8], association rule mining [6], clustering [3] etc. Among them, the most prevalent technique is collaborative filtering. Collaborative filtering approach finds other users that have shown similar tastes to the current user and recommends what they have liked to that user. Traditional collaborative filtering requires explicit user participation for providing his/her interest to the pages. However, despite their success, the explicit ratings may suffer from some limitations, such as *additional user effort*, *user behavior alteration* and *data sparsity*. To overcome such problems, several researches [1, 7] have investigated the use of implicit interest indicators. An important implicit indicator of the user's navigation path is the time spent on different pages.

Since the viewing time is a continuous-valued data, it must be discretized in advance in order to be integrated with collaborative filtering algorithms. The common used method is to partition the view time into two or several time intervals using single or a set of cut points so that it can be treated as an ordinary user rating. However, the use of the crisp cut points usually causes the sharp boundary problem in which two adjacent numbers are assigned to two different interval classes. Also the suitable position of a cut point is often uncertain and changeable with different application domains.

One possible approach to handle the vagueness of the cut points is the Fuzzy set theory, formulated and proposed by Zadeh in 1965 [9]. Fuzzy set theory can be viewed as an extension of interval arithmetic, which in addition provides possibility theoretic valuation. The method proposed in this paper takes advantage of the concept of the fuzzy logic to solve the above mentioned boundary problem. More precisely, we will propose a new recommender system by integrating fuzzy set and collaborative filtering and apply it for web page prediction.

2. Fuzzy Collaborative Filtering

The CF models can be built based on users or items. User-based CF identifies users whose interests are similar to an active user and recommends items they like. However, they suffer from serious scalability problems which make them unsuitable for on-line processing. On the other hand, item-based CF suggests items that are most similar to the set of items the active user has accessed or rated. Since the relationships between items are relatively static, item-

based approaches can be implemented by first computing the item relationship model and then providing a quick recommendation. In this section, we discuss the way of integrating the fuzzy set theory and the item-based CF algorithm for page prediction.

2.1. Membership Function

The web page prediction problem can be characterized by a set of user sessions $S = \{s_1, s_2, \ldots, s_m\}$ and a set of web pages $P = \{p_1, p_2, \ldots, p_n\}$. Each session $s_i \in S$ can be represented as an n-dimensional vector over the space of page references, i.e., $s_i = \{(p_1, d_1^i), (p_2, d_2^i), \ldots, (p_n, d_n^i)\}$, where d_k^i denotes the duration of page p_k given in session s_i. Suppose that the pages which have been visited by s_i are $A(s_i) \subseteq P$. The aim of the prediction system is to assess the relevance between $A(s_i)$ and every page in $P \setminus A(s_i)$, so as to provide a list of recommendation.

As mentioned beforehand, the duration of a page is a good measure of the user's interest in that page. The higher the time that a user spent on a page, the more likely he/she was interested. We assume that there are two fuzzy membership functions fv^+ and fv^- which map the duration to the fuzzy degree of "like" or "dislike", respectively. These two functions are customized for each page and defined by trapezoidal functions as shown below.

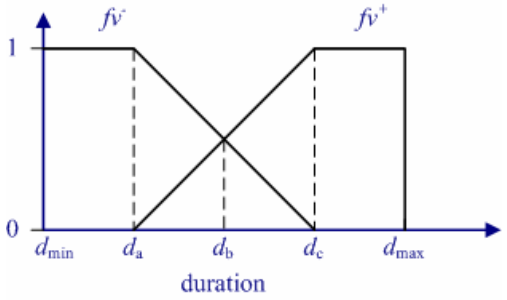

Fig. 1: The membership functions of two fuzzy sets: like and dislike.

In Figure 1, d_a and d_c are two parameters for customizing the membership functions and can be determined subjectively by domain experts or through the use of statistics. The value of d_b can be directly computed from d_a and d_c. The minimum duration $d_{\min}$ is used to remove uninterested page visits. The maximum duration $d_{\max}$ of a web page can be determined based on the page size and reading speed of humans. If the viewing time exceeds the maximum duration, then the time information for that page is invalidated and the maximum value is used.

We can now redefine each session s_i in terms of the fuzzy representation of its duration as follows $s_i = \{(p, fv_i^+(d), fv_i^-(d)) \mid p \in P\}$, where d is the duration of page p and, $fv_i^+(d)$ and $fv_i^-(d)$, the fuzzy values of "like" and "dislike" with respect to duration d, are given by

$$fv_i^+(d) = \begin{cases} 1 & \text{if } d_c \le d \le d_{\max} \\ \dfrac{d - d_a}{d_c - d_a} & \text{if } d_a < d < d_c \\ 0 & \text{if } d_{\min} \le d \le d_a \\ \perp & \text{if } d < d_{\min} \text{ or } d > d_{\max} \end{cases} \quad (1)$$

$$fv_i^-(d) = \begin{cases} 1 & \text{if } d_{\min} \le d \le d_a \\ 1 - \dfrac{d - d_a}{d_c - d_a} & \text{if } d_a < d < d_c \\ 0 & \text{if } d_c \le d \le d_{\max} \\ \perp & \text{if } d < d_{\min} \text{ or } d > d_{\max} \end{cases} \quad (2)$$

where $\perp$ represents an invalid value indicating the current page is not a valid reference.

2.2. Item-Based Fuzzy CF

The main difference between item-based CF and user-based CF is that in the case of item-based CF the similarity is computed based on the page relationships while in the case of user-based CF the similarity is computed based on the user relationships. More precisely, item-based CF first identifies the similarity relationship between different pages and then recommends pages that are most similar to the set of pages the active user has been visited before.

Following this principle, we can define a page profile $p_i \in P$ as an m-dimensional vector over the space of sessions, i.e., $p_i = \{(s_1, fv_1^+(d_1^i), fv_1^-(d_1^i)), (s_2, fv_2^+(d_2^i), fv_2^-(d_2^i)), \ldots, (s_m, fv_m^+(d_m^i), fv_m^-(d_m^i))\}$. Each page-session reference is represented as a triplet describing a session and the fuzzy degree of "like" and "dislike" of that session to a specific page.

Page Inference Calculation

The page inference calculation is an offline process that computes the similarity between each pair of pages based on their page profiles. The similarity is represented by four inference relations over sessions. Each inference relation sums over the fuzzy implication of each session which is defined as the quasi-inverse of *min* function. Thus, for any two page profiles p_i and p_j, the inference relations can be computed as:

$$I^{uv}(i, j) = \sum_{l=1}^{m} FI(fv_l^u(d_l^i), fv_l^v(d_l^j)), \quad \text{for all } u, v \text{ in } \{+, -\} \quad (3)$$

$$FI(fv_l^u(d_l^i), fv_l^v(d_l^j)) = \begin{cases} \min(fv_l^u(d_l^i), fv_l^v(d_l^j)), & \text{if } fv_l^u(d_l^i) \ge fv_l^v(d_l^j) \\ 1, & \text{otherwise} \end{cases} \quad (4)$$

The computation is over all sessions that contain both p_i and p_j. The *min* operator returns zero if any of the fuzzy values is equal to $\perp$. For each page p_i, we only store the relation value of the k most similar pages to page p_i and zero out the rest. The four inference relations are stored as inference matrices and serve as the basis for subsequent page prediction.

Prediction Generation

Prediction generation is an on-line process. As a user interacts with the system, his access history is used to find other unvisited pages of similar profiles. To recommend a new page p_j for a user u_a, we look at the pages that have been visited by the user and find out whether the new page is within their k most similar page neighbors. Only those pages that have page p_j in their k most similar page neighbors are concerned. Since a page may receive positive/negative influences from other pages, the prediction score of a page can be computed by integrating its positive and negative contributions as follows.

$$PS(s_a, p_j) = \max(PS^+(s_a, p_j) - \alpha \cdot PS^-(s_a, p_j), 0), \quad (5)$$

where $\alpha \in [0,1]$ is an attenuation coefficient for limiting the importance of negative contributions. The positive and negative contributions to a page are the total extent to which other pages can influence the "like" or "dislike" of the page.

$$PS^+(s_a, p_j) = \sum_i [fv_a^+(d_a^i) \cdot I^{++}(i,j) + fv_a^-(d_a^i) \cdot I^{-+}(i,j)], \quad p_j \in B(p_i) \quad (6)$$

$$PS^-(s_a, p_j) = \sum_i [fv_a^+(d_a^i) \cdot I^{+-}(i,j) + fv_a^-(d_a^i) \cdot I^{--}(i,j)], \quad p_j \in B(p_i) \quad (7)$$

where $B(p_i)$ is the neighborhood of the page p_i in the active session. Note that the summation is over all pages p_i that have page p_j in their neighborhood. These unvisited pages are sorted based on the prediction score and the first N pages are selected.

3. Experimental Evaluation

3.1. Data Preparation

In this research, we use the server log from the web site of department of information management of our institution (www.imd.pccu.edu.tw). The server log was collected over three months from May to July 2005. After data preprocessing, the data set contains 23675 sessions and 1045 pages (URLs). Each session contains at least 5 pages, and each page has been accessed at least 3 times. Since the preprocessing is beyond the scope of this paper, the details of this procedure are not discussed here.

We employ the 5-fold cross-validation approach. For each session in the test data, we move the sliding window over the sequence of page reference in that session at one reference increment. At each window position, our recommender system is evaluated by comparing the Top-N recommendations it makes with the subsequent requests.

In the following experiments, we set d_{min} and d_{max} to 3 and 99 seconds respectively. The parameters d_a and d_c are derived, for each page, by first ordering the durations of all accesses and then finding the ath and the cth percentiles, respectively. Parameter d_b can be computed by averaging d_a and d_c and can be used as the cut point in the crisp-cut approach. A sliding window with size of 2 is used to capture the current session's history depth. There is a trade-off between performance and quality for choosing the size of neighbors k. A small value of k will lead to higher efficiency but lower quality. From the experimental evaluation by [2], a small value between 10 and 30 is suggested. In all of our experiments, k is set to 30. Finally, we let the number of recommended pages N unfixed for comparing the performance of our approach with the traditional click-only CF approach.[1]

3.2. Evaluation Metrics

We use the *hit-ratio* metric and *click-soon-ratio* metric as proposed in [5] to evaluate the quality of a recommendation. Hit-ratio is the fraction of all next requests of the users that are satisfied by the recommendations. Click-soon-ratio is the fraction of recommendations that are requested by the user during the active session.

A hit is declared if any of the recommended pages is the next request of the user and a click-soon is declared if any of the recommended pages matches the subsequent user requests. We compute each metric for each session and the overall average value for the test data is taken as measures of the quality of the recommendation.

3.3. Performance Results

Table 1 shows the results of *hit-ratio* and *click-soon-ratio* for varying values of a and c when the attenuation coefficient α is set to 0.1. In this experiment, we fix the number of recommendation to 3. Both two metrics improve as the range between a and c increases but they reach the maximum performance at a = 10 and c = 90 and any further increment makes no better or even worse results. This is likely due to the outlier effects on the fuzzy membership functions as the value of c closes to 100.

Table 1: Performance comparison for varying values of a and c.

Metric	**Value of *a* and *c***						
	0-100	5-95	10-90	20-80	30-70	40-60	50-50
Hit-Ratio	0.240	0.274	0.284	0.278	0.272	0.268	0.265
Click-Soon-Ratio	0.444	0.482	0.495	0.488	0.479	0.475	0.469

To compare our approach with the click-only CF, we varied the number of recommendation from 1 to 11. We again fix the size of sliding windows to 2 and α to 0.1. Figure 2 shows the performance comparisons between our approach and the click-

[1] Click-only CF refers to the traditional CF without using time information.

only CF. As expected, when the numbers of recommendations increases, both metrics improve gradually. It can also be observed that the fuzzy CF outperforms the click-only CF at all values of *N*. The difference is significant at $N = 3$ to 7, while the significance decreases as *N* becomes smaller or larger.

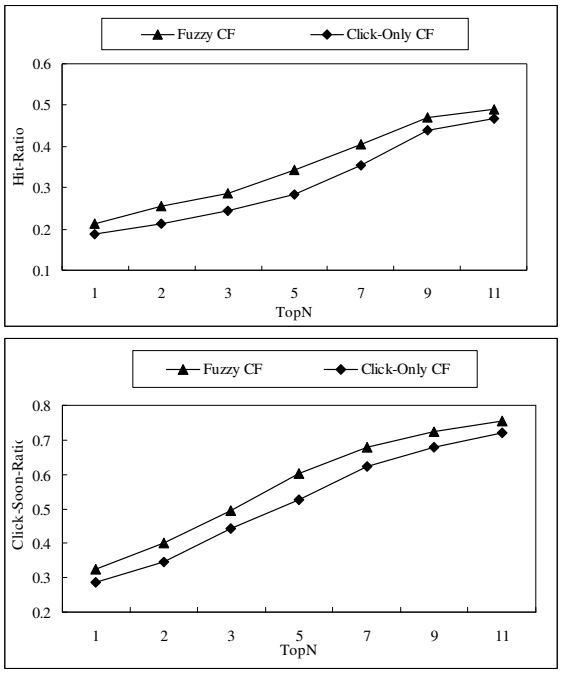

Fig. 2: Hit-ratio and click-soon-ratio comparison between fuzzy CF and click-only CF.

One of the strengths of fuzzy CF is the ability to provide a more personalized recommendation. The fuzzy CF tends to recommend a page that is not only accessed but also liked by a user. To validate this, we add another metric that computes the average viewing time of recommended pages that are also requested by the user. The result is shown in Figure 3. We observe that the time difference between these two approaches decreases as we increase *N*. However, in all circumstances, the fuzzy CF tends to recommend pages with higher viewing time than the click-only CF.

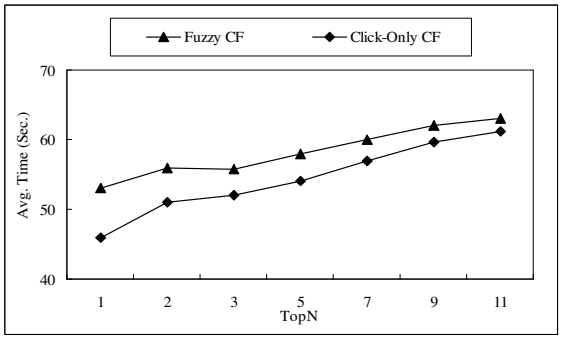

Fig. 3: Average viewing time comparison between fuzzy CF and click-only CF.

4. Conclusion

In this paper we have presented a fuzzy-based approach for the collaborative filtering problem which incorporates the time information into item-based CF algorithms. Each accessed page is characterized by its viewing time in terms of two fuzzy values. We introduce four inference relations over sessions to capture the similarities between pages. These relations compare two page profiles by means of the related sessions and the page fuzzy values. These relations can be built offline and serve as the model of the system for further online page prediction. The experimental results show that our approach not only performs better quality of prediction but also provides a more personalized recommendation than the traditional click-only approach.

5. References

[1] M. Claypool, P. Le, M. Wased and D. Brown, "Implicit Interest Indicators," *Proceedings of 6th international conference on Intelligent User Interfaces*, ACM Press, pp. 33-40, 2001.

[2] M. Deshpande and G. Karypis, "Item-based top-N recommendation algorithms," *ACM Transactions on Information Systems*, vol.22 no.1, ACM Press, pp. 143-177, 2004.

[3] Y. Fu, K. Sandhu, and M. Shih, "Fast Clustering of Web Users Based on Navigation Patterns," *Proceedings of World Multiconference on Systemics*, Cybernetics and Informatics (SCI/ISAS'99), vol. 5, pp. 560-567, 1999.

[4] D. Goldberg, D. Nichols, B. M. Oki and D. Terry, "Using Collaborative Filtering to Weave an Information Tapestry," *Communications of the ACM*, vol. 35, no. 12, ACM Press, pp. 61-70, 1992.

[5] Ş. Gündüz and M. T. Özsu, A Web Page Prediction Model Based On Click-Stream Tree Representation of User Behavior. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM Press, pp. 535-540, 2003.

[6] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, "Effective Personalization Based on Association Rule Discovery from Web Usage Data," *Proceedings of the 3rd ACM Workshop on Web Information and Data Management*, ACM Press, pp. 9-15, 2001.

[7] J. Parsons, P. Ralph, and K. Gallagher, "Using Viewing Time to Infer User Preference in Recommender Systems," *Proceedings of AAAI Workshop on Semantic Web Personalization*, pp. 52-63, 2004.

[8] R. R. Sarukkai, "Link Prediction and Path Analysis Using Markov Chains," *Proceedings of the 9th International World Wide Web Conference on Computer networks*, North-Holland Publishing, pp. 377-386, 2000.

[9] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.