On The Effectiveness of Web Usage Mining for Page Recommendation and Restructuring

Hiroshi Ishikawa, Manabu Ohta, Shohei Yokoyama, Junya Nakayama, and Kaoru Katayama

Tokyo Metropolitan University

Abstract. As an increasing number of Web sites consist of an increasing number of pages, the users more find it more difficult to rapidly reach their own target pages. Ill-structured design of Web sites also prevents the users from rapidly accessing the target pages. In this paper, we describe two complementary approaches to Web usage mining as a key solution to these issues. First, we describe an adaptable recommendation system called the system L-R, which constructs user models by classifying the Web access logs and by extracting access patterns based on the transition probability of page accesses and recommends the relevant pages to the users based both on the user models and the Web contents. We have evaluated the prototype system and have obtained the positive effects. Second, we describe another approach to constructing user models, which clusters Web access logs based on access patterns. The user models also help to discover unexpected access paths corresponding to ill-formed Web site design.

1 Introduction

As an increasing number of Web sites consist of an increasing number of pages, it is more difficult for the users to rapidly reach their own target pages. So the systems supporting the users in navigation of the Web contents such as [9] are in high demand. Furthermore, ill-structured design of Web sites also prevents the users from rapidly accessing the target pages. The support for detecting access paths contrary to the Web site administrators' expectations is also in high demand. The key solution to these issues is Web usage mining. In this paper, we describe two complementary approaches to Web usage mining. First, we describe an adaptable recommendation system called the system L-R (Log-based Recommendation), which constructs user models by mining the user access logs (e.g., IP addresses and access time) and recommends the relevant pages to the users based both on the user models and the Web contents [6]. We construct the user models by classifying Web access logs and by extracting access patterns based on the transition probability of page accesses. Second, we describe another approach to constructing user models, which clusters Web access logs based on access patterns. The user models help to discover unexpected access paths corresponding to ill-formed Web site design as well as play a basic role in recommendation.

In this paper, we explain the functional aspects of the system L-R as a first

approach to Web usage mining in Section 2, describe the experiment and evaluation of the system in Section 3, and describe user model construction based on clustering as a second approach to Web usage mining in Section 4.

2 First Approach to Web Usage Mining: System L-R

Indeed, the recommendation systems [9] are very helpful in directing the users to the target pages. However, applying recommendation systems to existing Web sites requires a large amount of work and knowledge. So we propose a recommendation system that assumes no detailed knowledge about the recommendation system and that can be adapted to any existing Web sites without rewriting the contents.

The proposed system has the following merits:

(1) It is easy for the Web site administrator to incorporate the recommendation facility because the system utilizes only Web usage logs. The system doesn't impose the users to explicitly input their profiles.
(2) It is possible for the Web site administrator to provide multiple recommendation methods by considering the objective of the site and the tendency of the users.
(3) It is possible for the users to interactively choose their favorite recommendation methods.

The system is intelligent in that it can automatically mine the Web log data and construct the user model based on the Web usage and recommend the pages fit for the user.

2.1 Extraction of Web usage logs

Our system recommends the relevant pages to the users based on the probability of the user's transition from one page to another, which is calculated by using the Web access logs. First, we delete the unnecessary log data left by so-called Web robots (i.e., sent by search engines) by heuristically finding them in Web access logs [4]. In general, robots are recommended to have access to a file "robots.txt" created by the Web site administrator as the Web policy about the robot access. Moreover, since robots leave their own names in the Web log as host names or user agents, we prepare the list of robots in advance. Thus, we can find most robots in the Web logs by checking access to the robots text or searching in the robot list. Please note that there is small possibility that non-human data log, such as brand-new robots and page scanning tools, remain in the Web.

Second, we collect a sequence of page accesses by the same user within a predetermined time interval (e.g., 30 minutes) into a user session by using the genuine (i.e., human) Web logs. At the same time, we correct the session by deleting the transition caused by pushing the back button in Web browsers. For example, assume that the user browses the page A and the page B and browses again the page A by pushing the back button and then the page C. We change the transition from B to A to C into the transition from B to C (See Fig.1).

Lastly, we calculate the probability of the transition from the page A to the page B denoted by $P_{A->B}$ based on the modified Web log as follows:

$P_{A->B}$ = (the total number of transitions from A to B) / (the total number of transitions from A).

Note that the transition probability of the path from A to B to C which we denote by A->B->C (where the length of the path is two) is calculated as follows: $P_{A->B->C} = P_{A->B} * P_{B->C}$.

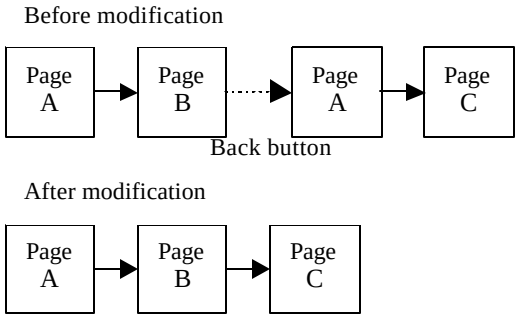


Fig. 1. Log modification w.r.t. back transition

2.2 Classification of users

We classify the Web users by using information extracted from the Web access logs. The information includes IP addresses, domain names, host names, access time, OS, browser names, refers (i.e. the site just previously visited by the user), and keywords for retrieval. We describe the methods of classification based on such kinds of information. We construct user models by applying the methods of classification to the modified Web log data and realize the recommendation fit for the users based on the user models. We show the viewpoints of classification based on the Web log data as follows:

(1) IP address. We count the same IP address to get the number of access times. We recommend pages to the user according to their frequency of access to the Web site. For example, the frequent users are likely to browse the frequently updated pages. New comers are likely to browse "getting started" pages.
(2) Domain and host name. The names of user domains and hosts indicate the place from which the users are accessing the Web site. We classify the users according to the places. For example, the users from companies may browse the company information of EC sites while the general users may browse the product pages
(3) Access time. The users have different tendencies according to the time when the users are accessing the Web site. During the daytime, the housewives and businesspersons often access the site. At midnight, students often access the site.
(4) Day of the week. The weekday and weekend indicate different tendencies. For

example, the pages about the leisure information are more often accessed on weekends than weekdays.

(5) Month and season. Month and season indicate different tendencies. For example, the pages about traveling are more often accessed during holiday seasons.

(6) OS. As a general rule, most users use Windows 98 and 2000. Business users often use Windows NT and Unix. It is possible to recommend different products according to OS.

(7) Browser. From the browser information, we can know the language that the user uses. We can automatically change the display language according to the browser.

(8) Keyword. We can conjecture the pages more correctly that the user intends to access by user-input keywords.

(9) Refer. We can make recommendation more fit to the user by knowing the site that the user visited just before coming to the current site.

Of course, we can know other information such as the types of requests (e.g., get, put) and the size of transferred data from the Web logs, too. However, for the moment, we don't know the effect of the classification based on these kinds of information.

2.3 Recommendation

We provide various methods for recommendation, which can be classified into two groups: pure probability-based methods and weighted probability-based methods.

(1) Pure-probability-based methods. The following recommendation methods are based solely on the transition probabilities calculated from the Web log data.

- **Path recommendation.** The whole path with high transition probability such as pages A, B, and C is recommended to the user (See Fig.2). The recommended path reflects the series of pages, which are usually browsed in sequence such as instructions stretching over manual pages. Of course, the path longer than two can also be recommended.

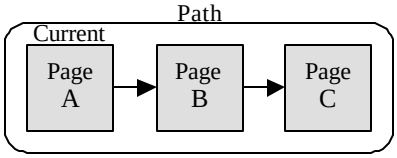


Fig. 2. Path recommendation

- **History-based recommendation.** Based on the pages visited before coming to the

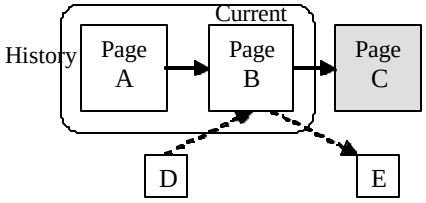


Fig. 3. History-based recommendation

current page, the next page with high probability is recommended. In other words, even if the current page is the same, the next recommended pages differ depending on such histories. For example, either C if A->B or E if D->B is recommended (See Fig.3). Of course, the history path older than two can be utilized too.

- **Future prediction-based recommendation.** A page ahead by more than one link is recommended based on the path transition probability. This can let the user know what exists ahead in advance. Any link ahead (i.e., any future path farer than two) can be recommended, but it is also possible that the user may miss their true target between the current page and the page far ahead. For example, if the path A->B->C has the highest probability, then the user at A is recommended C (See Fig.4).

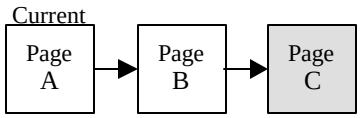


Fig. 4. Future prediction-based recommendation

- **Shortcut recommendation.** The transition by the back button is viewed insignificant and then the path is shortcut. For example, there is a path A->B->A->C where B->A is done by the back button. The path is shortened to A->C.

(2) Weighted probability-based methods. The following recommendation methods are done based on the probability weighted by other aspects such as the time length of stay, the Web policy, and the Web content structures:

- **Weighted by time of stay.** The page where the user stays longest is recommended. It can be known by using the Web log data. For example, if there is a path A->B->C where C is longest referenced, the weight of A and B with respect to C is increased by one. Then C is recommend at A or B.
- **Weighted by most recent access.** The probability is weighted by the importance of the target pages that we judge by most recent access. For example, if there is a path A->B->C where C is most recently referenced, the weight of A and B with respect to C is increased by one. C may be the page with detailed description of products. Then C is recommend at A or B.
- **Weighted by number of references.** If n pages reference a page within the same site, then the weight of the referenced page is increased by n. This method is validated by the observance that many pages link the important pages, which is similar to authorities and hubs in WWW [3].

3 Experiment and Evaluation of System L-R

We have implemented an experimental recommendation system and have evaluated the system by using the logs of the Web server of our department. In this section, we describe the statistical data of the Web logs, the problems and the method for evaluation, the experimental system, and the system evaluation in sequence.

3.1 Statistics

First, we show the statistical information about the Web log data used for the experiments as follows:

Access logs: 384941 records; Web Pages: about 170 HTML files; Log duration: from June to December 2000; Subjects: five In-campus (knowledgeable) students and five out-campus (less knowledgeable) students

Please note that we have excluded non-HTML files such as GIF, JPEG because they need to be handled separately.

3.2 Problems and evaluation method

Before evaluation, we have prepared problems corresponding to the possible objectives of the users visiting the department Web site. They include the following problems:

Problem 1. Find the major field of Prof. Ishikawa.
Problem 2. Find the contents of the course "electric circuit I".
Problem 3. Find the contact information of the member of faculty doing the research on "efficient implementation of pipeline-adaptable filter".
Problem 4. Find the contents of Phd theses for 1995.
Problem 5. Find the contact information of the member of faculty teaching the course on "communication network theory".

Next, we describe the method for evaluating the experimental system. We instruct the subjects to write down the number of transitions (i.e., clicks) and the time to reach the target pages as answers to the above problems. We evaluate the experimental system based on the figures. Each subject solves different problems by different recommendation methods.

3.3 Experimental system

For this time of evaluation, we have implemented the following four recommendation methods effective for our department site:

- Recommendation with back button transitions
- Recommendation without back button transitions
- Future prediction-based recommendation
- Recommendation Weighted by number of references

Please note that the first two methods are simple methods which recommend the next page based on pure transition probability. They are used just for the purpose of comparison with the last two methods as our proposal. We use two user groups based on IP addresses : In-campus students and out-campus students.

We have implemented the recommendation system by using frames (See Fig. 5). The upper frame displays the original Web page and the lower frame displays the recommendation page. Both of the frames change in an interrelated manner. When the user moves from one page to another based on the recommendation, the recommendation page is changed accordingly. The home page allows the user to select favorite recommendation methods. The recommendation page frame displays titles, URLs, and ranks of the recommended pages. The ranks are based on the transition probability calculated from the Web logs. At most five pages are recommended to the user. The user either may or may not accept the recommendation.



Fig.5. User interface of recommendation

3.4 Evaluation

We illustrate the graphs indicating the average clicks and time required for solving the five problems (See Fig.6). 1, 2, 3, 4, and 5 indicate no recommendation, recommendation with back button transitions, recommendation without back button transitions, future prediction-based recommendation, and recommendation weighted by number of references, respectively.

First of all, the result shows the positive effect of recommendation because all methods (2-5) decrease the clicks and time in comparison with no recommendation (1). Recommendation without back button transitions (3) is better than recommendation with them (2), so we think that the modified Web logs reflect user access histories more correctly. Future prediction-based recommendation (4) is a little bit worse than recommendation without back button transitions (3). This is probably because the former can skip the user's target mistakenly. Recommendation weighted by number of references (5) is the best of all although recommendation

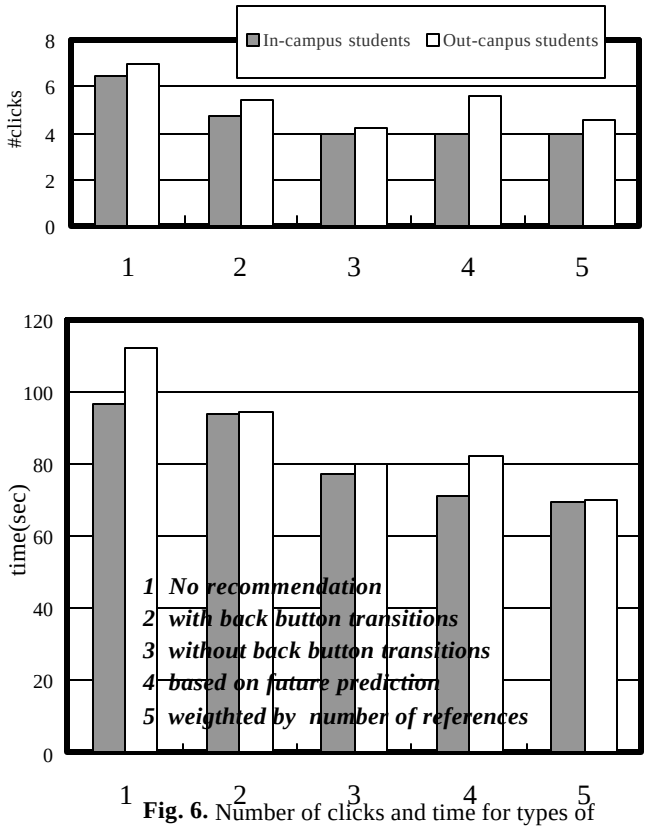


Fig. 6. Number of clicks and time for types of recommendation

without back button transitions (3) is a little bit better in the number of clicks.

The fact that there is difference between in-campus students and out-campus students suggests that we need to classify the user groups. The difference can be shortened by our recommendation system. This indicates that our system is more effective for less-knowledgeable people such as out-campus students.

4 Second Approach to Web Usage Mining: Clustering

4.1 User model based on clustering

We summarize our first approach described earlier. First, we have classified the Web access log data according to the attributes such as in-campus/out-campus students in advance. Then we have extracted access patterns recurring in them and have constructed the user models according to them and have recommended relevant pages

based on the user models.

Here we describe a complementary approach to Web usage mining. First, we cluster Web access log data based on access patterns. Each cluster corresponds to a specific user model with access patterns as its features. Then, we can use the models for recommendation. We can also use the clusters by extracting unexpected access patterns when we redesign Web sites to remedy such inappropriate structures.

We describe how we utilize clusters as the user models for recommendation in more detail. First of all, we match clients with user models (i.e., clusters) and determine the most likely model. Then we recommend the clients relevant pages according to frequent access patterns in the selected cluster. At this time, we can use the various strategies for recommendation described earlier as our first approach. In matching the clients with user models, we can use access patterns (i.e., IP addresses, access date and time, access pages), which we use in clustering user models. In matching, of course, we can also use the other attributes of the Web access log not used for clustering itself. We extract characteristics of such attributes from the constructed clusters by combining them with the original Web access log data.

Next we describe how we use clusters for redesigning Web site structures. We can find access patterns contrary to the expectations of the Web site administrators. It is possible that they indicate ill-formed structures of the Web sites. Thus, we can refine Web sites by resolving such ill-formed structures discovered in cluster analysis.

4.2 Experiments

(1) Statistics

We have done experiments to validate this complementary approach described above. First, we describe the experiment settings. We use another Web log: The Web site is one for a cinema complex at Tachikawa City. The Web log contains access log records for Sunday, October 7th, 2001 consisting of 6402 records accessing distinct 138 html pages. We extracted 712 user sessions. We determined one session of a consecutive access from the same IP whose intervals are within 40 minutes for this site.

Cluster	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Number of sessions	195	171	144	135	67
Most frequently accessed page/ratio(%)	Road show 17	Road show 26	Road show 18	I-congestion 14	Floor 16
i-mode page ratio (%)	27	20	47	61	33

(a) collection type

Cluster	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Number of sessions	214	161	124	120	93
Most frequently accessed page/ratio(%)	Road show 17	Road show 24	Road show 16	Floor 11	Lineup 16
i-mode page ratio (%)	53	20	36	42	25

(b) set type

Fig. 7. The result of page reference clustering

(2) Session representation

We provide two types of representation to sessions: Page reference and page transition vectors. We represent each page reference vector as a 138-dimensional vector plus IP and last access time, whose each component corresponds to a distinctive referenced page. Further, we provide two sub types of representation to page reference vectors: Set type and collection type. The set type of reference vectors indicate the existence of references to each page (i.e., each component contains either 0 or 1) while the collection type of reference vectors keep records of the times of references to each page (i.e., each component contains 0 or a larger integer). We exclude accesses to only one page as we consider them noisy.

We represent each page transition vector as a 19044 (= 138*138)-dimensional vector plus IP and last access time, whose each component corresponds to a transition from one page to another (i.e., directed one). We also provide two types of representation: Set and collection. Similar to page reference vectors, they indicate the existence and counts of a specific transition, respectively. In other words, we have four different types of clustering, that is, set- and collection-typed page reference clustering and set- and collection-typed page transition clustering. They are selectively used depending on specific cases of clustering. Of course, we exclude accesses to only one page.

(3) Ward clustering

We use Ward's method for clustering sessions: We start with a node in a separate cluster with the node as its center of gravity. In each step, the two clusters that are closest according to a distance measure (e.g., Euclidean distance) are merged and a new center of gravity is computed for this new cluster by using those of the two merged clusters. We repeat this step until we get the target number of clusters.

In Ward's method, we have to determine the number of target clusters. To this end, we provide a visual aid for the user to choose the appropriate number of clusters. The aid graphically plots the minimum values of all cluster-distances vs. the number of clusters. With this help, we have detected a big change of the minimum distance when the number of clusters changes from 5 to 4 at page reference clustering. So we choose 5 as the number of target clusters at page reference clustering. For the same reason, we choose 6 as the target number of clusters at page transition clustering.

(4) Clustering result

First, we illustrate the result of page reference clustering (See Fig. 7). We counted the total number of sessions, the time of the most frequent accesses, the most frequently accessed page and its ratio, and the "i-mode" page (i.e., internet access through mobile phones) ratio for each cluster. We focus on collection-typed clustering (See Fig.7 (a)). The cluster 4 contains the largest number of "i-mode" users with 61%. On the contrary, the cluster 2 contains the largest number of PC users. The users in the cluster 5 most often visit the "floor" page, where the theater facilities are described as expertise. We can use these clusters as the user models on which recommendation is based. For this specific case of clustering, we prefer collection to set because the characteristics extracted from collection-typed clustering are more emphasized. For example, the cluster 2 of set-typed clustering has 161 sessions, "road show" with 24%, and "i-mode" page with 20% (See Fig.7(b)).

Next we also focus on the result of collection-typed page transition clustering. We

Cluster	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6
Number of sessions	250	192	83	82	76	29
Most frequently accessed time	10	11	12	18	8	19
Most frequently accessed page/ ratio(%)	Road show-> Lineup 13	Road show-> Next road show 6	I-congestion ->City2 7	Road show-> Lineup 12	Road show-> Lineup 7	Next road show-> Road show 17

(a) collection type

Cluster	Cluster1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6
Number of sessions	279	133	110	71	64	55
Most frequently accessed time	23	12	11	14	13	12
Most frequently accessed page/ ratio(%)	Road show-> Lineup 11	I-congestion ->City1 5	Road show-> Lineup 7	Road show-> Lineup 14	Road show-> Lineup 12	Next road show-> Road show 10

(b) set type

Fig.8. The result of page transition clustering

counted the total number of sessions, the time of the most frequent transitions, the most frequently transitioned page pairs and its ratio for each cluster. At this time, the collection-type also emphasizes the characteristics more clearly than the set-type.

The most frequent transition in the cluster 6 of collection-typed page transition clustering has "next road show->road show" with 17% while the corresponding cluster of set-typed clustering has the same transition with 10% (See Fig. 8). We can interpret this observation as follows: The day of the Web log is late Sunday. Therefore the users have access to the "next road show" page for the next week schedule. However, the page on Sunday is empty and is updated on next Tuesday. Until then, the current schedule page "road show" contains the schedule from Saturday to Friday. Then the users notice the wrong visit and go to the right page, which causes this anomalous transition. Of course, this indicates the inappropriate design of Web pages contrary to the users' expectations. In other words, the observation suggests that the Web site administrator should redesign the Web site structures more understandably.

5 Conclusion

5.1 Summary

Our main contribution is that we have validated the effectiveness of Web usage mining for page recommendation and restructuring through empirical studies. First, we have proposed an experimental recommendation system L-R based on both Web

usage mining (i.e., transition probability calculation) and Web site structures. From the result of the evaluation of the system L-R, we have been able to indicate the following findings:

(1) The recommendation based on Web usage mining is positively effective. All recommendations based on modified Web logs in respect to back button transitions are more effective than that based on pure Web logs.
(2) The recommendation based on user classification is positively effective. We have found that the amount of knowledge about the Web site affects the number of clicks and the time to the destination pages. Recommendation is more helpful for the less-knowledgeable people. So the classification with respect to the point is valid.
(3) The recommendation based on Web page structures is positively effective. The one based on both the modified Web logs and Web page structures (i.e., hybrid approach) is the most effective. This finding is one of our contributions.

Second, we have described user model construction based on automatic clustering. We have been able to indicate the following findings:

(1) Collection-typed page reference clustering extracts features of each cluster more obviously than set-typed one. The result can play a basic role in recommendation.

(2) Collection-typed page transition clustering extracts features more obviously than set-typed one. The result can help to redesign ill-structured Web sites. Our findings about collection-typed clustering techniques are also among our contributions.

However, we also have found remaining issues, in particular, in the implementation of the System L-R as follows.

(1) We have copied the original Web site and Web logs to the virtual site from the experiment and evaluation of the system. So we must rewrite the contents for the virtual site, such as modification of URL. Moreover, the contents in the virtual site are not completely synchronized with those in the original site. However, simple solution such as just incorporating the recommendation system to the original site is also problematic because the user access according to the recommendation modifies the access logs.

(2) The size of the Web logs is no so large that we could not implement the other methods such as path recommendation. This is because the multiplication of the probabilities tends to be quite small in the small-sized Web logs.

(3) The number of problems and subjects in evaluation of the System L-R is not so large enough to give very confident results. In particular, it is difficult to prepare as distinctive types of problems as possible.

5.2 Related work

We will compare our work with relevant works. Cooley et al. [1] have clarified the preprocessing tasks necessary for Web usage mining. Our approach basically follows their steps to prepare Web log data for mining.

Srivastava et al. [10] have surveyed techniques for Web usage mining comprehensively. Among various techniques, we use transition probabilities as a sequential variation of association rules for page recommendation and collection-typed clustering as page restructuring.

Spiliopoulou et al. [11] have proposed Web Utilization Miner (WUM) to find interesting sequential access patterns, which is similar to our approach for page recommendation. Unlike them, however, we construct various types of recommendation from the same set of sequential access patterns.

Perkowits et al. [8] have proposed Adaptable Web Sites as a method for providing index pages suitable for the users based on Web access logs. They have implemented the algorithm to extract user access patterns from the access logs. They provide virtual sites adapted to individual users by determining the links and their ranks (order) in the index pages based on the algorithm. The Adaptable Web Sites automatically recommend different pages depending individuals while our system recommends pages in several ways chosen by both the Web site administrator and the users.

Mobasher et al. [5] have proposed a method for recommending pages weighted by mining the user access logs. The system recommends pages according to the user access recorded in "cookies" while our system allows the user to choose among several recommendation methods and takes page structures into consideration. The recommended pages change depending on the current page but not on the cookies.

Kiyomitsu et al. [2] [12] have proposed a system for displaying different link structures according to the user access logs and a mechanism for the administrator to easily change the contents. They change pages according to the user access logs while we keep the original contents intact and allow any site to add on recommendation. Their mechanism is solely intended for the Web site administrators. Our recommendation is intended for both the users and the Web site administrators in that several recommendation methods can be chosen by both of them.

Ohura et al. [7] have proposed an approach, which clusters access patterns based only on "set-typed page reference session vectors" and uses them to expand users' queries. Our approach provides four types of clustering to serve different purposes such as page recommendation and restructuring.

5.3 Future work

First, we will improve the evaluation result by the following ways without changing the system:

(1) Increasing the problems and subjects
(2) Increasing the size of the Web logs by using the longer duration of the logs
(3) Classifying users into groups form different viewpoints

Second, we will generalize our recommendation system as follows:
(1) Adapting the system to non-HTML contents such JPEG, MPEG, MP3
(2) Adapting the system to different sites
(3) Using the user models based on automatic clustering of the users

The above three ways to generalization can require the system to change more or less.

Acknowledgements

This work is partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Japan under a grant 14019075. It is also partially supported by the Tokyo Metropolitan University under a president special grant 142.

References

1. R. Cooley, B. Mobasher, and J. Srivastava: Data Preparation for Mining world Wide Web Browsing Patterns, Journal of Knowledge and Information systems, vol.1, no.1, pp.5-32, 1999.
2. H. Kiyomitsu, A. Takeuchi, and K.Tanaka.: Dynamic Web Page Reconfiguration Based on Active Rules, IPSJ Sig Notes, vol.2000, no.69 (2000-DBS-122), pp.383-390, 2000 (in Japanese).
3. J. Kleinberg: Authoritative sources in a hyperlinked environment. Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.
4. T. Mizuhara, T. Nakajima, M. Ohta, and H. Ishikawa: Web Log Data Analysis for Recommendation System, Proc. IPSJ National Convention, 4W-3, 2001 (in Japanese).
5. B. Mobasher, R. Cooley, and J. Srivastava: Automatic Personalization Based on Web Usage Mining, CACM, vol.43, no.8, pp.142-151, 2000.
6. T. Nakajima, T. Mizuhara, M. Ohta, and H. Ishikawa: Recommendation System Using User Models based on Web Logs, Proc. IPSJ National Convention, 4W-4, 2001 (in Japanese).
7. Y. Ohura and M. Kitsuregawa: Examination of the expansion of users' search queries based on clustering Web access logs: Experiments on the i-townpage, IEICE Data Engineering Workshop, A2-1, 2002 (in Japanese).
8. M. Perkowittz and O. Etzioni: Adaptive Web Sites, CACM, vol43, no.8, pp.152-158, 2000.
9. J. Schafer, J. Konstan, and J. Riedl: Recommender Systems in E-Commerce. Proc. ACM Conference on Electronic Commerce (EC-99), pp.158-166, 1999.
10. M. Spiliopoulou, L.C. Faulstich, K.Winkler: A Data Miner Analyzing the Navigational Behavior of Web Users, Proc. Workshop on MachineLearning in User Modeling of ACAI99, 1999.
11. J. Srivastava, R. Cooley, M. Deshpande, P. Tan: Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, SIGKDD Explorations, vol.1, no.2, pp12-23, 2000.
12. A. Takeuchi, H. Kiyomitsu, and K.Tanaka.: Access Control of Web Content Based on Access Histories, Aggregations and Meta-Rules, IPSJ Sig Notes, vol.2000, no.69 (2000-DBS-122), pp.315-322, 2000 (in Japanese).