

Reorganizing Web Sites Based on User Access Patterns

Yongjian Fu
University of Missouri-Rolla
1870 Miner Circle
Rolla, MO, 65401, USA
yongjian@umr.edu

Mario Creado
University of Missouri-Rolla
1870 Miner Circle
Rolla, MO, 65401, USA
mcreado@umr.edu

Chunhua Ju^{*}
University of Missouri-Rolla
1870 Miner Circle
Rolla, MO, 65401, USA
chunhua@umr.edu

ABSTRACT

In this paper, an approach for reorganizing Web sites based on user access patterns is proposed. The approach consists of three steps: preprocessing, page classification, and site reorganization. In preprocessing, pages on a Web site are processed to create an internal representation of the site, and page access information of its users is extracted from its server log. In page classification, the Web pages on the site are classified into two categories, index pages and content pages, based on the page access information. After the pages are classified, in site reorganization, the Web site is examined to find better ways to organize and arrange the pages on the site. Our experiments on a large real data set show that the approach is efficient and practical for adaptive Web sites.

Keywords

Adaptive Web sites, Web usage mining, page classification, site reorganization, access patterns

1. INTRODUCTION

A Web site is to meet the needs of its users. As the interests of its users change over time, a Web site must change itself accordingly to best serve its users. In other words, Web sites should be adaptive. An adaptive Web site has been defined as a Web site that semi-automatically improves its organization and presentation by learning from visitor access patterns [3].

In this paper, an approach is proposed to build adaptive Web sites which improve their navigation based on access patterns of its users. Our goal is to build adaptive Web sites by evolving site structure to facilitate user access. To be more specific, we aim to build Web sites that provide its users the information they want with less clicks. This minimizes the effort on the user's side. By analyzing the usage

^{*}Visiting from Hangzhou University of Commerce, China.

of a Web site and the structure of the Web site, modifications to the Web site structure are found to accommodate changes in access patterns of its users. These modifications will be suggested to the Webmaster for consideration and implementation.

Our approach consists of three steps: preprocessing, page classification, and site reorganization. In preprocessing, the pages on a Web site are processed to create an internal representation of the site. Page access information of its users is extracted from the Web server log. In page classification, the Web pages on the site are classified into two categories, index pages and content pages, based on the page access information. A page classification algorithm has been developed which uses data about a page's type, structure, and usage to determine its category. After the pages are classified, in site reorganization, the Web site is examined to find better ways to organize and arrange the pages on the site. An algorithm for the reorganization of the site has been developed.

The approach has been implemented and tested on a large real data set. Initial experiments show that the approach is efficient and practical for adaptive Web sites. The reorganized Web site requires fewer clicks for users and is thus easier to navigate.

2. WEB SITE REORGANIZATION BASED ON ACCESS PATTERNS

The process consists of three steps: preprocessing, page classification, and site reorganization, in that order.

2.1 Preprocessing

There are three tasks in preprocessing. The first is Web site preprocessing to obtain the current structure of a Web site, i.e., how the pages are linked. The second is server log preprocessing to organize access records into sessions. The third is to collect access information for the pages from the sessions.

2.1.1 Web Site Preprocessing

The purpose of this phase is to create an internal data structure to represent the Web site. The Web site is represented as a directed graph in which a page is a node and a link is an arc. Each page of the Web site is parsed sequentially and the links in the page (tags beginning with <A HREF>) are extracted. Each page is assigned a unique page identifier

(PID). For each page, PIDs of pages which has a link to it (called its parents) and pages which it links to (called its children) are stored. Currently, the Web pages are assumed to be static. Dynamic pages such as those generated by CGI or other server-side scripts are ignored. All non-HTTP references, e.g., "ftp://", "gopher://", "mailto:", etc, are filtered out because they do not represent site structure. In addition, all references to pages on other sites, e.g., a reference to Adobe site for Acrobat reader, are also removed. Also, multiple links between two pages are treated as one and intra-page links (an intra-page link is a link to the page it is in) are ignored.

2.1.2 Server Log Preprocessing

Since a lot of irrelevant information for Web usage mining such as background images is also included in the server log, it has to be processed first. A number of preprocessing algorithms and heuristics exist [1, 2, 4, 5].

The steps involved in preprocessing of the server log are as follows.

1. Records about image files (.gif, .jpg, etc) are filtered as well as unsuccessful requests (return code not 200).
2. Requests from the same IP address are grouped into a session. A timeout of 30 minutes is used to decide the end of a session, i.e., if the same IP address does not occur within a time range of 30 minutes, the current session is closed. Subsequent requests from the same IP address will be treated as a new session.
3. The time spent on a particular page is determined by the time difference between two consecutive requests.

The server log files are transformed into a set of sessions. A session represents a single visit of a user. Each session contains a session ID and a set of (PID, time) pairs, where PID is the page identifier and time is the time the user spent on the page. There are some difficulties in accurately identifying sessions and estimating times spent on pages, due to client or proxy caching of pages, IP sharing, network congestions, and interruptions [2]. Although the server log is not perfect for Web usage mining, it gives us rough idea about page access. Moreover, it is widely available without client-side programming or other intrusive methods. It provides a comprehensive source of access information with reasonable accuracy.

2.1.3 Access Information Collection

In this step, the sessions obtained in server log preprocessing are scanned and the access statistics are computed. The statistics are stored with the graph that represents the site obtained in Web site preprocessing. The following statistics are computed for each page: number of sessions in which the page was accessed, total time spent on the page, and number of times the page is the last requested page of a session. The last item is included because, if a page is the last page of a session, we cannot tell the time spent on the page since there is no page requested after it.

2.2 Page Classification

In this phase, the pages on the Web site are classified into two categories: index pages and content pages. An index page is a page used by the user for navigation of the Web site. A content page is a page containing information the user would be interested in. Its content offers something other than links.

The page classification algorithm uses the following four kinds of heuristics.

1. File type.

An index page must be an HTML file, while a content page may or may not be. If a page is not an HTML file, it must be a content page. Otherwise its category has to be decided by other heuristics.

2. Number of links.

Generally, an index page has more links than a content page. A threshold is set such that the number of links in a page is compared with the threshold. A page with more links than the threshold is probably an index page. Otherwise, it is probably a content page.

3. End-of-session count.

The end-of-session count of a page is the ratio of the number of time it is the last page of a session to the total number of sessions it is in. Most Web users browse a Web site to look for information and leave when they find it. It can be assumed that users are interested in content pages. The last page of a session is usually the content page that the user is interested in. If a page's end-of-session count is larger than a threshold, it is more likely a content page; otherwise, it is more likely an index page.

4. Reference length.

The reference length of a page is the average amount of time the users spent on the page. It is expected that the reference length of an index page is typically small while the reference length of a content page will be large. A cut-off value in reference length is computed based on the average reference length of all pages and an estimation of the overall percentage of pages that are index pages. If a page's reference length is less than the cut-off value, it is more likely an index page, otherwise, it is more likely a content page.

An algorithm for page classification is introduced which combines the heuristics mention above. To determine the category of a page, its file type is first checked. If it is not HTML, the page is certainly a content page and no other testing will be necessary. Otherwise, its end-of-session count, number of links, and reference length, are examined, whose weighted sum decides the page category.

2.3 Site Reorganization

The goal of this phase is to reorganize the Web site so that its users can access the information they desire with fewer clicks. The general idea of reorganization is to cut down the number of intermediate index pages a user has to go through.

To achieve this, we need to place the frequently accessed pages higher up in the Web site structure, i.e., closer to the home page, while pages that are accessed infrequently should be placed lower in the structure. In the meantime, however, we want to preserve the original site structure whenever possible, since it may bear business or organizational logics. Besides, dramatic changes of the site structure may confuse users. As a compromise between these two conflicting requirements, we introduce an evolutionary approach to Web site reorganization.

The basic idea is to locally adjust the site when a frequently accessed page should be promoted. In addition, two thresholds are introduced, that is, maximum number of links in an index page (I) and maximum number of links in a content page (C). An index/content page will not have more than I/C links after site reorganization, unless it has more links before reorganization, in which case its links will be intact. These two thresholds are introduced to achieve two objectives. First is to limit the number of links in a page so its layout will be reasonable. This will prevent extreme cases, for example, a flat site structure where all pages are linked from the home page. Second is to somehow contain the changes in the site structure. The selection of these thresholds can be done by the Webmaster or data analyst.

A page is a frequently accessed page if its frequency is greater than a minimum frequency, F . A page's frequency is defined as the number of sessions it is in divided by the total number of sessions. Frequently accessed pages are promoted to be upper in the site structure. In case such reorganization will cause the number of links in a page to exceed its maximum (I or C), we will try to merge infrequent pages into a larger page. The mergers will reduce the number of clicks by users due to fewer page requests. To prevent spurious results, the merging pages must be HTML files and at most one of them can be a content page.

In the algorithm for site reorganization, the pages are examined sequentially starting from the home page. For each page, we consider its immediate parents and children, where a parent is any page that has a link to it and a child is any page that it has a link to. Depending on the number of children it has, the algorithm will consider three cases: 1 child, 2 children, and 3 or more children. For each case, different actions will be taken according to the frequency and category of the pages involved.

3. EXPERIMENTS

The algorithms have been implemented and tested on the Hyperreal Web site (<http://www.hyperreal.org>). The server log used is available at <http://www.cs.washington.edu/homes/map/adaptive/download.html>.

Some of the findings from our experiments are summarized as follows.

- The algorithms are quite efficient. It takes from a few seconds to a few minutes to process the data set depending on the size of the log we test. The preprocessing seems to be the slowest step.
- The page classification algorithm can correctly classi-

fies the majority (above 80%) of pages. It performs best when the number of links threshold is between 20 and 30, and the estimation of percentage of index pages is between 40% and 60%.

- The reorganization algorithm finds many possible actions to improve the Web site structure. They can reduce the number of pages and links on the Web site by 3% and 2%, respectively.
- If the Web site is reorganized fully according to the recommendations, the number of clicks required for the users to navigate as per log used for evaluation will decrease by 1% to 3%.

4. CONCLUSIONS

An approach to reorganize Web sites based on user access patterns has been proposed. By analyzing the usage of a Web site and the structure of the Web site, modifications to the Web site structure are found to improve the structure of the Web site. Two algorithms, one for page classification and the other for site reorganization, have been developed. The approach has been implemented and tested on a real data set from an actual Web site. Experiments on a large real data set demonstrate that the approach is practical and promising.

We are currently working on more experiments on real and artificial data. They will help us to gain more insight on parameters selection and fine-tuning of the algorithms.

5. ACKNOWLEDGMENTS

The research was partially supported by the Intelligent Systems Center at the University of Missouri-Rolla.

6. ADDITIONAL AUTHORS

Additional author: Ming-Yi Shih (University of Missouri-Rolla, email: mingyi@umr.edu).

7. REFERENCES

- [1] J. Borges and M. Levene. Mining association rules in hypertext databases. In *Proc. 1998 Int'l Conf. on Data Mining and Knowledge Discovery (KDD'98)*, pages 149–153, August 1998.
- [2] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems*, 1, 1999.
- [3] M. Perkowitz and O. Etzioni. Adaptive web sites: An ai challenge. In *Proc. Int. Joint Conf. on AI (IJCAI)*, pages 16–23, 1997.
- [4] C. Shahabi, A. Z. Zarkesh, J. Adibi, and V. Shah. Knowledge discovery from users web-page navigation. In *Proc. of 1997 Int. Workshop on Research Issues on Data Engineering (RIDE'97)*, Birmingham, England, April 1997.
- [5] M. Spiliopoulou, L. Faulstich, and K. Winkler. A data miner analyzing the navigational behaviour of web users. In *ACAI'99 Int. Conf., Workshop on Machine Learning in User Modelling*, Florence, Italy, July 1999.