Web Structure Mining for Usability Analysis

Chun-hung Li and Chui-chun Kit
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong
Kowloon, Hong Kong
chli@comp.hkbu.edu.hk

Abstract

The interaction between usability and how a web site is structured is a complicated issue. In this paper, we discuss a web structure mining algorithm which allows the automatic extraction of navigational structures in a website without performing hypertext analysis. We perform several usability experiments to correlate the usability of web sites and the structural design of the web site. Experimental results show that the structure mining algorithm gives reasonable prediction about several design issues in web structure. The analysis serves as building block in the complex issue of web usability and structure mining..

1. Introduction

The use of hyperlink graph analysis has been very successful in information retrieval. The page rank algorithm by Brin and Page improves significantly on the relevancy of the documents returned by search engines [11]. Similarly the hyperlinks induced topics search by Kleinberg has also been widely used in various information retrieval applications [12]. However, the use of hyperlink graph for navigational and usability analysis has not been very successful in the past. In this paper, we will presents a novel use of the hyperlink graph model and describe our design of a spatial frequent itemset mining algorithm on discovering navigational structure (**NS**) of a website. Navigation structure refers to a common set of hyperlinks which exists across a set of related page to assist user navigate within a set.

Usability test is always regarded as a costly and time exhaustive process but it is becoming more important as many of our business operations and social activities are processed and completed on the Internet directly. We will describe our design of a task-based usability evaluation system in Section 3, followed by a discussion on the experimental results of our web structure and usability analysis, and the correlation of mined website structures with the usability of websites in Section 4.

2. Website Structure Analysis

2.1 Background on Web Structure Analysis

Conventionally, a website structure may be evaluated with the Card Sorting technique; however, this technique is difficult to implement for a large and information-centric website. There are a number of commercial and free evaluation tools available. Most of the tools are based on the user logging results that are stored in the proxy server or through the embedment of client-side programming code to capture the client's events.

A recent publication by Miller and Remington [8] pointed out that the structure of linked pages (the site's information architecture) has a decisive impact on the usability. Previous studies including Shneiderman [10], and Larson and Czerwinski [7] also provided suggestions on how to create the best structure. Larson and Czerwinski [7] found that users took significantly longer time to find items in a structure with depth than breadth. They compared a three-tiered, eight-links-per-page (8 x 8 x 8) structure with two-tiered, 16 and 32 links per page structures (16 x 32 and 32 x 16).

Bernard [1] devised a metric called Hypertext Accessibility Index (HAI) to model the informational accessibility of a particular hypertext structure compared. It explained what Larson and Czerwinski [7] as well as Kiger [6] found about the navigation time of shallow and deep structure in quantitative terms. Kao et al. [5] proposed a LAMIS method with the entropy

Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)
0-7695-2415-X/05 $20.00 © 2005 **IEEE**

analysis to distill the information of a website. Empirically, the probability is directly linked to the transition probability of the web log.

A study on the use of entropy theory to merge the website content was performed by Chen et. al. [2]. Based on the Markov model, Jenamani et al. [4,9] proposed several algorithms to examine (1.) the most accessed pages, (2.) the company's interest, (3.) the visitor's interest pages, (4.) the current visitor's interest pages, (5.) the customized index generation algorithms. However, our system objectives are not to provide a dynamic hyperlink structure, we will concentrate on the study of building a static optimal website structure.

2.2 Hyperlink Graph and Data Mining

A hyperlink model for a website is described by G=(V,E) where V is the set of vertices and E is the set of edges, Eij=1 iff there is a hyperlink from page i to page j, otherwise Eij will be zero. Using this graph model, we can model the hyperlinks embedded in a web page as a list of items. We define Pi to be the sets of hyperlinks of the page i, ie Pi={set of v, s.t. Eiv==1}. We take a novel interpretation of the set Pi and consider the set of hyperlinks as shopping baskets. As navigational structure consists of near-identical hyperlinks that exists in different web pages, these frequent co-occurring items can be extracted via data mining algorithms.

2.3 Navigational Structure Mining

2.3.1 The Adaptive Window Algorithm (AWA). Given a set of *n* web pages of a website, we developed the Adaptive Window Algorithm for discovering the navigational structure(s) in a website. From the hyperlink graph, we can form a *n*-by-*n* adjacency sparse bit matrix *M*, which (*i,j*) element is 1 if page *i* links to page *j*, and 0 otherwise. Since the pages and links are obtained by a web crawler program which parses the web pages in a top-down and breath-first manner, the adjacent links in *M* are assumed to be adjacent links in the actual layout design of the website. AWA attempts to discover navigational structures (e.g. Menu / Templates) in *M*.

The near-identical hyperlinks patterns can be extracted by frequent item-sets data mining algorithms such as Eclat [13] and FP-growth [14]. However, the result of applying these algorithms to *M* is not satisfactory when *M* is dense or large. Menus and web page design templates generally contain 5 to 20 links while the whole website may contain up to ten thousands of links. The support threshold used to generate the frequent link-sets has to be very small and the link-sets generated are usually very bulky. AWA boosts the number of mined NS by working on the adjacent links which locates along the diagonal of *M*. It keeps an adaptive window *W* which slides along the diagonal of *M* from top left to bottom right and computes the frequent link-sets in *W*

We define the width and height of *W* to be twice of the expected size of NS so that any adjacent near-identical link patterns can be covered by *W*. The algorithm first generates a set of page-link baskets $PW(s{:}s{+}w)$, where *s* denotes the first index in *W* and *w* is defined as the width *W*. Furthermore, we defines PW_i to be the set of hyperlinks that is located inside *W*, of the page *i*, ie PW_i={set of *v* in *W*, s.t. E_{iv}==1}.

Frequent itemset mining algorithm, such as Eclat, is then applied to $PW(s{:}s{+}w)$ and the mined frequent link-sets L_k, where $k=1,2,\ldots n$, are used to determine the next starting point and dimensions of *W* in the next round.

2.3.2 NS Selection/ Window Control Rules.

```
Procedure SelectLink (Lk : generated link-sets)
Foreach $L_k$ do
Select $L_k$ if and only if it passes all the NS
    selection rules.
Update $w$ and $s$ for each Window Control Rule.
End
```

Listing 1. The NS selection procedure of AWA

2.3.3 Result of AWA and Eclat in discovering NS. The Eclat algorithm is the first successful algorithm proposed to generate all frequent itemsets in depth-first manna in transaction database [14]. In this problem, each web page is regarded as one transaction in the database, which the Eclat algorithm works on. The AWA is specialized in discovering navigational structure of a web site. It processes a spatial frequent itemsets extraction on the page-link matrix *M*.

Table 1 shows the results of applying Eclat and AWA to two different websites: Dept. of Computer Science of HKBU (M_1) and another CS dept. websites (M_2). M_2 is denser than M_1 as the average number of links of M_1 (4.47 links/page) is smaller than M_2 (6.07 links/page). The performance of Eclat depends on the choice of confidence level σ and Eclat extracts lots of frequent itemsets which are not navigational structures. The result shows that applying Eclat to web link graph is difficult to discover all the underlying NS. The situation goes worse when hyperlink graph is dense. The AWA is able to discover more NS with a higher support threshold in both matrices. The generated linksets of AWA are more likely to be a NS than simply applying Eclat to *M*.

	M1	M2

	σ	Itemset extracted	True NS	σ	Itemset extracted	True NS
Eclat	4	0	0	10	0	0
	3	1	1	4	35	0
	1	12	10	3	103	0
	0.5	22	16	1	3260	5
	0.1	402	20	0.5	21771	5
AWA	45	20	20	45	30	30

Table 1. Results of AWA and Eclat in *M*

```
Algorithm AWA
1. Initialization: determine the support threshold $S$ and the window size $w$.
2. Create page-link baskets $PW$ with the links within the area covered by $W$
3. Call frequent itemset mining algorithm Eclat on $PW$.
4. Call SelectLink($L_k$) for each frequent link-set $L_k$, and update $w$ and $s$.
5. Repeat 2-4 until the end of the matrix $M$ is reached.
End
```

Listing 2. The Adaptive Window Algorithm

3. Website Usability Analysis

3.1. Task-Based Usability Evaluation

The task questions and answers are input by the administrator, a set of target web page locations where the answer can be found are also input to the platform. The target locations are needed to determine whether the user has entered the target answer page before answering the question. User activities are logged into the proxy server, followed by filtering and storage in the database.

4. Experimental Results and Discussions

4.1 Results on Task-based usability test

Three commercial web sites of international electronics companies are selected for conducting the usability analysis. Fig. 1 shows the result on number of clicks, average time and failure ratio from 10 tests conducted from 54 users. The number of clicks is highest for W2 which indicate usability problem in W2. W3 has the smallest no. of clicks. The average time spent on a tasks is roughly in the three web sites. In the failure ratio figure, W2 has the highest failure ratio which indicate its usability problem. W1 also has high failure ratio. W3 again has the best performance in failure ratio.

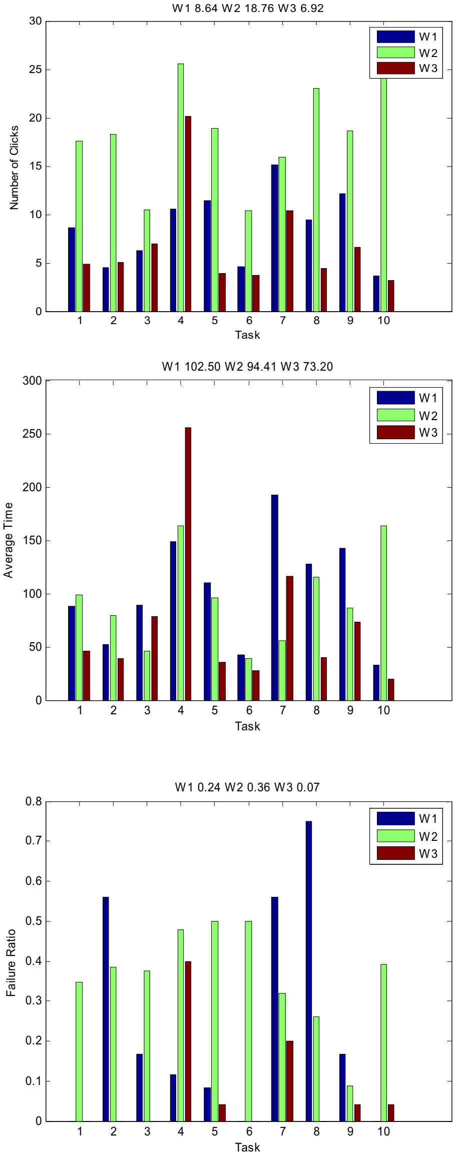

Fig.1 (a) No. of clicks (b) Average time (c) Failure ratio of usability test.

4.2 Results on questionnaire from tester

The testers are asked to give the opinions on the design of the web site in terms of layout design, navigation and contents. W3 has significantly higher score than both W1 and W2, indicating users have much better impression with W3. This result agrees well with the results in the above section on task based usability testing. W1 has slightly lower rating than W2, especially on the general design of web sites.

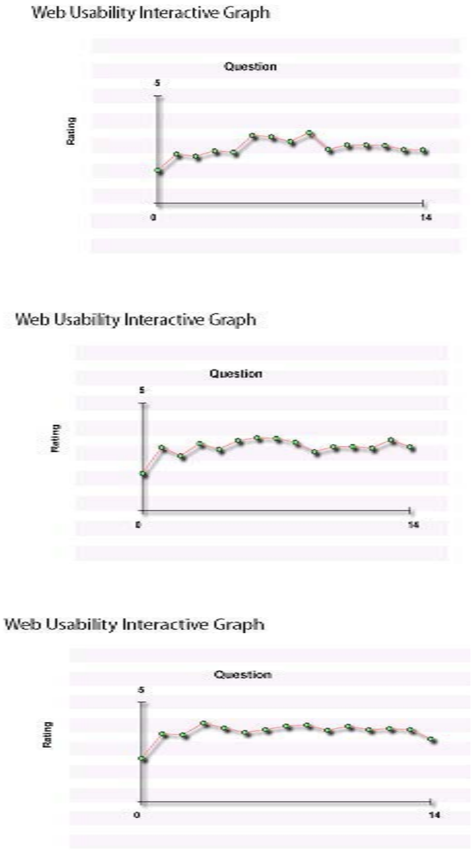

Figure 2. Usability questionnaire results from testers (a) W1 (b) W2 (c) W3.

5. Conclusions

We described the importance of web structure mining and its relationship with web usability. A web structure mining algorithm is described that can effectively extract web structures. Task based usability are conducted on three commercial web sites. Results on task based usability correlates well with the user's questionnaire's results.

6. References

[1] Bernard, M. L., "Examining a Metric for Predicting the Accessibility of Information within Hypertext Structures", *Dissertation Thesis*, Wichita State University, 2002

[2] Chen, Z., Liu, S., Liu, W., Pu, G. and Ma, W., "Building a Web Thesaurus From Web Link Structure", *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, 2003, pp.48-55.

[3] Chi, E. H., Rosien, A., Supattanasiri, G., Williams, A., Royer, C., Chow, C, Robles, E., Dalal, B., Chen, J., Cousins, S., "The Bloodhound Project: Automating Discovery of Web Usability Issues Using the InfoScent™ Simulator", *CHI 2003*, Ft. Lauderdale, Florida, USA, April, 2003.

[4] Jenamani M., Mohapatra P. K. J., and Ghose S., "Online Customized Index Synthesis in Commercial Websites", *IEEE Intelligent Systems*, Vol.17, No.6, 2002, pp.20-26.

[5] Kao, H., Lin, S., Ho, J., Chen, M., "Mining Web Informative Structures and Contents Based on Entropy Analysis", *IEEE Transactions on Knowledge and Data Engineering*, Vol.16, Iss.1, 2004, pp.41 – 55.

[6] Kiger, J. I. , "The Depth / Breadth Tradeoff in the Design of Menu-Driven Interfaces.", *International Journal of Man-Machine Studies*, Vol.20, 1984, pp.201-213.

[7] Larson, K., & Czerwinski, M., "Web page design: Implications of memory, structure and scent for information retrieval", *CHI'98: Human Factors in Computing Systems*, New York: ACM Press, 1998, pp.25-32.

[8] Miller, C. S. and Remington, R. W., "Modeling Information Navigation : Implications for Information Architecture", *Human-Computer Interaction*, Vol.19, No.3, 2004.

[9] Pitkow, J. E. and Pirolli, P. "Mining Longest Repeated Subsequences to Predict World Wide Web Surfing.", *Second USENIX Symposium on Internet Technologies and System, 1999.*

[10] Shneiderman, B., "Designing the User Interface", Strategies for Effective Human-Computer Interaction, 3rd ed, Reading, MA : Addison-Wesley, 1998.

[11] Brin and Page, The autonomy of a Large-Scale Hypertextual Web Search Engine, Computer Networks and ISDN systems, vol. 30, pages 107-117, 1998.

[12] Kleinberg, Authority Sources in a Hyperlinked Environment, Journal of ACM, vol. 46, no. 5, 1999.

[13] Christian Borgelt, "*Efficient Implementations of Apriori and Eclat*" Workshop Information Mining- Navigating Large Heterogeneous Spaces of Multimedia Information German Conference on Artificial Intelligence, Hamburg, Germany, 2003

[14] Bart Goethals, *"Memory issues in frequent itemset mining"* ACM Symposium on Applied Computing, 2004