December 2004

Building an Adaptive Site Map Based on Domain and Usage Information

Jiajia Ye
Fudan University

Weihui Dai
Fudan University

Building an Adaptive Site Map Based on Domain and Usage Information

Jiajia YE	Youwei WANG	Weihui DAI
School of Management Fudan University	School of Management Fudan University	School of Management Fudan University
032025062@fudan.edu.cn	ywwang@fudan.edu.cn	whdai@vip.sina.com

Abstract

As the World Wide Web matures, it makes leaps forward in both size and complexity. Navigating through a large Web site can become a frustrating exercise. Many sites employ Site Maps to help visitors understand the overall structure of the site. However, static Site Maps show all visitors large amounts of irrelevant content. It is very easy for visitors to get lost in hyperspace. Therefore we propose techniques based on domain and usage information that are specialized to guide and recommend visitors, depending on the importance of pages, frequent visiting patterns and so on. Several algorithms and data mining methods have been applied, such as mining server log for visiting patterns and determining the landmarks among context. . A case study of a Web site is carried out to demonstrate the technique of guiding visitors during their navigation by recommending relevant content.

Keywords: Adaptive Site Map, Domain Knowledge, Hyperlink structure, Landmarks

1. Introduction

The World Wide Web has been a very frequently used method when searching for information. But as it matures, it makes leaps forward in both size and complexity. In this expanding environment, the needs and interests of individual users become buried under the explosive viewing choices. So, finding relevant information in this Web site can be tedious and frustrating. Although Web developers commonly apply Site Maps to help visitors understand and navigate complex sites, static Site Maps show all visitors large amounts of irrelevant content, which are not what visitor want, and make visitors facing a lot of shaw but lost in the huge forest.

As mentioned in (Toolan and Kushmetick 2002), there are two challenges to solving this problem. The first is to "determine what content items each visitor is actually interested in." This needs textual analysis to determine similarity between two textual content, and definite standard visitors' interests. The second challenge is to "display these relevant pages in a way that helps visitors understand how the relevant pages are related", which is our focus in this paper. To avoid making visitors aimlessly navigating thought complex Web site, we should emphasize the structure of a Web site, based on the relationship between each page, and view the Web site as a whole.

Many researchers have worked in this field during the last few years. Mining data from log files and analysis the structure of the Web site itself are probably the most familiar, most widely implemented and most mature of the technologies. Mining Web logs can help us find information from past using, i.e. how visitors navigated the Web site, and reveal the inside story of visitors as much as possible. On the contrary, analysis the structure of the existed Web site, i.e. how the pages and relationships between pages compose the whole Web site

together, can reveal latent information of the objective Web site. Toolan and Kushmerick (2002) studied personalized site maps by mining Web logs, aiming to mining the server log for popular path fragments that can be dynamically assembled to reconstruct popular paths. Mukherjea and Hara studied Focus+context views of page nodes (Mukherjea and Hara 1997). It can help to orient visitors when they feel lost in hyperspace, by positioning the current node in overall information space. They emphasized the immediate neighborhood of the current node and its position with respect to the important nodes in the information space, viewing the focus without neglecting context.

The broadest definition of an adaptive site map is a site map that changes based on the way it is used (Kilfoil et al. 2003). Changes can take on many forms, but the original impetus of changes is based on domain and usage information we mined from navigation of visitors. One most popular way of doing this is mining log files of Web sites, to find out how visitors navigating thought Web site before (Chen and Yu 1996; Mobasher et al. 2000). Applying this information to view tracks from user and considering the structure of Web site to search the objective relationship between pages can put forward a solution with a thorough consideration.

The remainder of this paper is organized as follows: section 2 formalizes how to build an adaptive site map based on domain and usage information, which classify all the pages of the Web site into three levels; the detail algorithms and methods are addressed in section 3; section 4 gives out an case of a real Web site to show how to build the adaptive site map; this paper concludes with summary and future directions.

2. Problem Formalization

To formalize an adaptive site map, we must present the visitor where he or she is navigating and the location of whole Web site, and recommend Web pages that the visitor may be interested in (Srikant and Yang 2001). An adaptive site map can be formalized as follows. We take as input a Web site's graph G=(V, E) and its distinguished root (home page) $r \in V$. Each node in V corresponds to a Web page, and directed edge $(u, v) \in E$ represents a hyperlink between the corresponding documents from u to v.

For the directed graph G=(V, E) of a Web site, we should view the overall structure of Web site together. Of course, textual content is also a factor to determine visitors' interest. But we do not study it in this paper; our main attention is paid to the former. The adaptive site maps that integrate the consideration of both site structure and textual content is the direction of our further research. And also, all conclusions are aimed at a certain type of visitors. To simplify the problem here, we postulate that parameters of a certain pattern of visitors are known, that is, how to determine different patterns of visitors is also a known precondition.

For a complex Web Site, usually it contains a large number of nodes and links. Because of this, we must scan nodes and links in another way – an abstractive way to display information at multiple levels (Kim and Yoo 2000). Basing ourselves upon a certain node, we can classify all nodes of a Web site into three levels: local, intermediate and global level, in order to keep balance between local detail and global context. The reason why we classify them separately is that, each level of nodes has special features, which should pay special attention to, and be resolved with special techniques (Nielsen 1999). This will be mentioned in detail in the following parts. The local node is defined as the node that can be reached from and to a

certain node by following at most one link. The intermediate node is defined as the node that must follow two criterions. Firstly, it can be reached from and to a certain node by following at most K links (K>1). Secondly, the nodes are selected by checking session frequency, which is extracted from log files while visitors navigating. Here K is not a fix number, because it depends on the scope of a Web site's structure. If the Web site is very complex, and the number of nodes and links is quite large, apparently a small K is improper, vice versa. The global nodes are defined as the nodes that are frameworks of this Web site. The three levels of nodes are considered from different point of view.

2.1 Local Nodes

The local nodes are the nearest nodes around us, and we can reach or we have just left them immediately, by only one click. We mark the certain node that we are now standing on as *x*. Normally, the node(s), which can be reached from *x*, have already been listed on the page *x*. To avoid putting some nodes aside as isolate ones, we do not take off any link of these nodes. So ours attention is only on the nodes that can reach *x* by one step, and we mark *Y* as the set these nodes. y_i is the i-th node among this set $Y=\{y_i\}$, $i=1,2,\ldots$.

We can reach *x* from y_i directly, but if our statistics from log files of this Web site shows that a lot of visitors navigate from y_i to *x* regularly, it may indicate that both *x* and y_i are very popular among these visitors. So some other visitors visiting *x* may be interested in y_i too. Then a link from *x* to y_i is a must.

2.2 Intermediate Nodes

As we define above, the intermediate node is a node that can be reached from and reach to a certain node by following at most K links (K>1), and can be reflected by session frequency of visitors. This scope restricts the nodes to an area neither too near nor too far (between 1 and K step(s) far away). What is more important is that the criterion of session frequency is from point of view of visitors' usage, not static structure of a Web site.

Both nodes and links compose overall structure of a Web site together. For the intermediate nodes, it is not proper to study features of nodes only, as their relationship between each other is more complex than the local ones. So we should focus our attention to links of these intermediate nodes, besides the features of nodes.

Of course, we can track visitor's whole navigations among the intermediate nodes and analyze them, but there are two problems. Firstly, tracks of navigation may be too long to identify exactly the same tracks, which are popular ones among a certain pattern of visitors. Even we can rank some, whether they are typical enough (this can measured by the percentage of the selected ones) still is doubtable. To resolve these, we can split long tracks of navigation of a certain pattern of visitors into several sessions with proper size (Toolan and Kushmetick 2002). We will discovery that, certain sessions are more frequently visited than others, and it is easy for matching with relative high percentage. By doing this, we can find out popular sessions to identify some characteristic information about a certain pattern of visitors, and use it to recommend others of the same pattern when they navigate. We split long tracks into separate sessions. For example, suppose that A>B>C>D>E and A>B>C>F>G are two paths, where I>J indicates a traversal by a particular visitor from page I to page J. Using the former approach we need to store the two paths completely, which may be too long for a complex and large Web site. However, we could store only A>B>C, D>E and F>G sessions to analyze the popularity and then recreate full paths when we need. This

path session method by splitting a path into several sessions, can extract more detail common ground among many different sessions, besides compressing the previous sessions much more than storing entire paths. Secondly, as nodes are stringed by links together, visitors are apt to travel back and forth by button click. Generally, a Web site server cannot record the whole track when visitors used “backward” icon and then a forward selection, but split it into several apparently separate paths, which are relative actually. This isolate analysis is not that meaningful, so we must convert the original relative split sessions into a set of traversal ones.

These processed meaningful sessions are composed by a series of sequential nodes, and the sequence and the objective last node are very important for us. To find out the frequently visited objective nodes, we need to determine frequent traversal sessions. The reason why we only care the objective nodes is that, we suppose other nodes on the halfway are only as guider to the objective last one, and what visitors really care is only the objective one. In the intermediate scope, neither can the objective nodes be reached so easily as local ones, nor can they be reached so aimlessly as global ones. The length of session M within this scope can determine how far is the objective node. For example, let M=1 in the extreme, it means the navigation is with great care, and every next node is paid attention to as objective one.

We have several methods to find out frequently visited objective nodes. For example, we can fix the length of each session to L ($1 \leq L \leq K$), select frequently visited sessions by data mining and rank them.

2.3 Global Nodes

With the explosive growth of information that is available on the World Wide Web, it is very easy for the visitor to get lost in hyperspace (Wang and Wang 2001). For an adaptive site map, we should not only recommending pages that the visitor will be interested in or care for, but also helping to orient visitors during their navigation. The disorientation problems arise when you are not familiar with the environment, so you do not know whether the information you need exists and how to find it. While navigating through a Web site, visitors may have the experience, such as not knowing if there are any other relevant pages nearby, or neglecting to return from a digression, or forgetting which pages have been visited or altered. Of course, for a complex Web site, it is not easy and necessary to represent a site map with all pages to a lost visitor. So we must take measures to indicate importance of a certain node, and only show those relatively important nodes, not all of them to the lost visitor. It is just like when we are lost in a big city where we have not been. With a map, we will surely first track various sorts of distinct information that can get easily and help us as a guider, such as fountain squares, tall buildings, central stations and so on, which are all landmarks as symbols. Just the same, we want to find out landmarks among page nodes in a Web site structure, which can guide us in astounding amount of information.

2.4 Nodes with Multi-functions

What should be added here is how to treat nodes with multi-functions. If there is a note that can be treated as both local one and global one, it is prioritized as a local node; a node that can be treated as both local one and intermediate one, it is prioritized as a local node; a node that can be treated as both intermediate one and global one, it is prioritized as an intermediate node.

How to treat nodes with multi-functions is a distinction without a difference essentially. In case some nodes are taken into consideration repeatedly, some rules should be added here.

Take a node that can be treated as both local one and global one for example, as defined above, local nodes are the nodes nearest to the current page for visitors, and there are more chance for them to be visited; the main function of global nodes is help visitor to locate in a relatively macro range. Therefore, if there exist nodes belong to both local and global ones, based on people's conventionality to view objects from near to far, they are considered as local nodes firstly.

All the following parts obey these rules.

3. Algorithms

3.1 Pre-processing

To build an adaptive site map, we must set up a graph G=(V, E) to represent the whole structure of the Web site at first (Wang and Wang 2003). The URL of each page can identify them well. If there is a hyperlink *e* in page *u* that links to page *v*, we note it as a directed edge *e=(u, v)*. We define the length of a path as the number of edges in the path; if *u* and *v* are nodes, the distance from *u* to *v*, written *d (u, v)*, is the minimum length of any path from *u* to *v*. For example, the distance from node *a* to node *e* is 2 (*a*>*b*>*e*, or *a*>*c*>*e*), not 4 (*a*>*c*>*d*>*e*) as in Fig. 1.

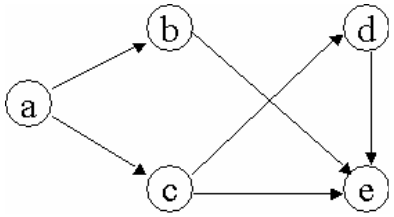

Fig.1 An example for distance between two nodes

Based on the concept of distance, we can list all the paths between every two nodes, if they are connected, or else, we note it as ∞.

It is easy to select from local nodes, as they are the nodes whose distance is only one. But for intermediate nodes, they depend on how we determine *K*. For a large and complex Web site, there are so many nodes and links that, the average distance may be very large. So determining a proper *K* to discriminate intermediate nodes is very important. To do this, we can analyze average distance of sub domain based on a certain topic, or compute average length of visitors' tracks, or something else.

3.2 Local Nodes

As pointed out earlier, for the local node *x*, most of our attention should be paid to those who can reach *x* by only one step, within set *Y*. We suppose that, the most popular nodes are that mostly visited among a certain pattern of visitors. The local nodes are the nearest nodes around us, so we do not need to consider too much criterions to measure them, using *Access frequency* to find out which are the most popularly visited is enough.

- *Access frequency*: Access frequency can indicate how many times the node has been accessed in recent times. The larger access frequency, the more important the node is.

Access frequency can be retrieved from log files of this Web site. We could define: if *access frequency* > *cutoff*, the given node is selected where *cutoff* is a value that can be changed by

the user. Instead of using a cutoff value, there are two other options. Firstly, the *n* most *access frequency* nodes could be chosen. For this option, the user should consider the value of *n* based on how many nodes local nodes are for the current node *x*. Sometimes the value of *n* may be considered again for another current node. Another option is to choose the top *n*% of the local ones (based on access frequency). Note that using just a cutoff value to determine is much simper than these two methods, but the cutoff value may have to be tuned to prevent very few or very large number of nodes becoming selected.

3.3 Intermediate Nodes

Links connect nodes together, but this is a static view of a Web site. For intermediate nodes that are neither too far nor too near, we should also emphasize the dynamic sessions while visitors navigating. We can mine this information from Web log files.

We split each track into several sessions of length *M*, and with overlap nodes value *O*. For example, if M=4, O=2, then the track a>b>c>d>e>f can be split into three sessions as: a>b>c>d, b>c>d>e and c>d>e>f. Because we only care the last node in a session as objective one, a proper M can help us find out it better. We leave to future work a systematic exploration of optimal values for these parameters.

After selecting all sessions as mentioned above, we compute the *importance* of a node, considering the effects from sessions. Popularity of session is determined by *session frequency*. *Session frequency* determines *importance*.

Let:

- S = session frequency

Importance = S.

For intermediate nodes, their *importance* is only determined by *session frequency*. Intermediate nodes are viewed mainly from the usage information.

3.4 Global Nodes

Global nodes spread around. Because of the large or even astounding number of nodes and links, it is not necessary and meaningful to list all the nodes. The conception of landmark in (Mukherjea and Hara 1997) is quite useful here.

A landmark is a kind of symbol of a sub range. Many criterions have to be considered to fairly choose a proper landmark to represent its sub range better and guide visitors who really want it. The criterions are discussed in detail as follows:

- Connectivity: Connectivity is defined as the number of nodes that can be reached from the current node. Since hypertexts are directed edges, the number of nodes that can be reached to the node should also be considered. The former is *outdegree* of a node, and the latter is its *indegree*. Sometimes, outdegree and indegree are different to a great extent, and their meaning is totally different. For example, outdegree of home page is always quite large, in contrast with indegree (we can even consider the indegree is zero, if there is no other pages can reach it and our attention is only devoted within the Web site), as home page is usually the starting point during a navigating. In a hierarchical Web site, the nodes with outdegree zero usually contain detailed information. Compared with home page, whose main function is presenting hyperlinks to visitors, the main function

of the nodes whose outdegree is zero is showing detail content. For some pattern of visitors, the importance of a navigation and content page is different, as for their different functions. But here we only care the total effect of outdegree and indegree, as *connectivity*.

- Depth: This can indicate at what depth the node resides in the file system hierarchy of the Web locality. The lesser the depth the more important the node is; so the depth of a node can also be used to determine the importance. The depth can be determined from the hierarchy of URL (Mukherjea and Hara 1997). For example, the *depth* of http:\\www.music-machines.org\addressbook\index.html is 2, and the *depth* of http:\\music-machines\bin\swish\src\test.html is 4.

- Access frequency. Just the same as mentioned above.

The procedure for discovering landmarks can be summarized as follows:

1. Let:
   - I = indegree
   - O = outdegree
   - A = access frequency
   - D = depth

2. Calculate:
   For a certain node, its
   importance = $(I + O)/(\text{Max}(I + O))*W_{conn}+A/(\text{Max } A)*W_{access}+W_{depth}/D$
   where $W_{conn} + W_{access} + W_{depth} =1$. So the overall importance of a node is presented by *importance*, which is a number between 0 and 1. The parameters W_{conn}, W_{access}, and W_{depth} determine the importance of connectivity, access frequency and depth respectively, and the user can control them.

3. Choose the top *n* nodes (based on *importance*) as landmarks. *n* is a value that can be changed by the user. However, in this case the value of *n* may have to prevent very few or very large.

Although the criterions of landmarks also include indegree, outdegree and access frequency, it is not a repetition. Because landmarks are relatively static when the structure of the Web site does not change (but they update regularly in accordance with the change of access frequency); in contrast with intermediate nodes that are relatively dynamic, changing at any moment based on the content of log files, that reflects the character of adaptive site map.

By now, we have found out the nodes, which we should present to visitors, based on domains of three different levels, and usage information such as navigating sessions from Web log files. The adaptive site map includes the three parts of hyperlinks decided in the previous sections. The building of an adaptive site map based on domain and usage information is finished.

4. Case Study

In this section, we will give out a real case to illustrate how this method is used. We evaluated our techniques on a Web site: Music Machines (machines.hyperreal.org). It is a real Web site on Internet. Its URL is http://machines.hyperreal.org/. The three years data of its

Web log file (from Feb 1997 to Apr 1999) is available for everyone to download. The number of total requests is 14,722,468 during the three years (Toolan and Kushmetick 2002). It is a Web site to introduce various kinds of music machines to visitors. It is obviously a Web site organized in hierarchy, most pages with shallow depth present hyperlinks of a certain type of music machines, while most pages with deep depth show detail information of a certain product.

We first analyze the structure of this Web site; find out the total 916 pages and 7630 links. Each page can identify by its URL, but to make the following work easy, we map each page to an identity number, with 0 the root of the Web site (home page).

Take the page "\categories\drum-machines\index.html" for example, whose identity number is 100. Its indegree is 2, and outdegree is 9. The detail is shown in Table 1 and Table 2.

Table.1 Indegree of node 100

From	To	From URL
101	100	\categories\drum-machines\info\index.html
527	100	\manufacturers\Moog\Prodigy\mods\index.html

Table.2 Outdegree of node 100

From	To	To URL
100	173	\new\main.html
100	181	\guide\index.html
100	189	\index.html
100	191	\links\index.html
100	391	\manufacturers\index.html
100	881	\MMAgent\browsing.html
100	882	\MMAgent\found.html
100	885	\MMAgent\index.html
100	886	\MMAgent\notfound.html

We compute access frequency, based on data of this Web site from 1998/9/10 to 1998/9/16. We use a value between 0 and 1 to present access frequency. In fact, it is a percentage to measure the access times of certain node to all nodes of the Web site. For example, the access times of node 100 during this week are 434, while the access times of all node of this Web site are 135520. So the access frequency of node 100 is 0.0032 = 434 / 135520.

By the definition, the local nodes of node 100 are {101, 173, 181, 189, 191, 391, 527, 881, 882, 885, 886}. We only care the access of nodes, which can reach node 100 by one step {101, 527}. Their access time and access frequency is list in Table 3. (access frequency = access time / 135520, where 135520 is the times of all nodes within the Web site be accessed) We use a cutoff value to measure access frequency, and let cutoff be 0.0005. So node 527 is selected.

Table.3 Access frequency

Identity Number	Access time	Access frequency
101	65	0.00048
527	91	0.00067

For intermediate nodes of node 100, we determine K = 4, where K is the most steps could follow to reach to and from the current node 100; M = 3, where M is length of each session; O = 2, where O is value of overlap. We select the top three nodes; the result is node 911 (100>189>911), node 858 (100>391>858), and node 758 (100>391>188>758).

When considering global nodes, we determine W_{conn}=0.4, W_{access}=0.5, W_{depth}=0.1, and we can compute from the static structure of the Web site, and the data within the week are: Max (I+O)=913, Max A=0.2519, D=5.

After compute the importance of each node by
importance = $(I + O)/(\text{Max } (I + O))*W_{conn}+A/(\text{Max } A)*W_{access}+W_{depth}/D$
We select the top 10 nodes as landmarks, ranked as follows:

Table.4 Importance

Rank	Identity Number	Importance
1	613	0.38361
2	382	0.38125
3	822	0.38043
4	239	0.37922
5	725	0.37893
6	831	0.37887
7	449	0.37883
8	484	0.36219
9	123	0.35977
10	634	0.12453

Table.5 Mapping between identity number and URL

Identity Number	URL
100	\categories\drum-machines\index.html
101	\categories\drum-machines\info\index.html
123	\manufacturers\Casio\index.html
173	\new\main.html
181	\guide\index.html
189	\index.html
191	\links\index.html
239	\manufacturers\ARP\Odyssey\samples\index.html
382	\guide\index.html
391	\manufacturers\index.html
449	\manufacturers\Korg\MS-synths\schematics\index.html
484	\manufacturers\Latronic\index.html
527	\manufacturers\Moog\Prodigy\mods\index.html

758	\manufacturers\Sequential\Prophet-5\images\index.html
858	\manufacturers\Yamaha\Overview\index.html
613	\links\index.html
634	\manufacturers\Korg\MS-synths\schematics\service-manual\index.html
725	\MMAgent\notfound.html
822	\manufacturers\index.html
813	\MMAgent\found.html
881	\MMAgent\browsing.html
882	\MMAgent\found.html
885	\MMAgent\index.html
886	\MMAgent\notfound.html
911	\samples.html

Table 5 shows the mapping between identity number and URL of nodes that we mentioned here.

So the nodes that should present are all selected, and the links towards these nodes are shown on page node 100. The following two figures, one is an original page of node 100 and the other is a fictitious page with an adaptive site map based on domain and usage information.

Compared with the following two figures, the second one is added with more links, based on this algorithm, analysis the domain and usage information. The links to Web pages that visitors may be interested in are list beside, so visitors can reach pages they may like more easily and directly.

The algorithm mentioned here on building an adaptive site map based on domain and usage information, has considered factors from view of both Web structure and usage information, and divides all the nodes into three categories: local, intermediate and global ones, generally takes the balance between focus and context into account.

Compared with other algorithm without consider intermediate nodes, but only local and global ones, the algorithm introduced here can generally guess more nodes that visitors may be interested in. In the experiment, based on data from log file of one day, top 15 nodes including intermediate and global ones- as group 1, and top 15 global ones- as group 2, are chosen to compare with each other. Refer to the nodes in top 100 most frequently visited sessions, the number of guess nodes within the top 10, 20, 30 nodes in those most frequently visited sessions are shown in the following table. For example, among the top 10 nodes of frequently visited sessions, nodes of group 1 cover 6 nodes; nodes of group 2 cover 5. The group 1 nodes chosen by algorithm mentioned in this paper is better than others without consider intermediate nodes, as the former can guess more than the latter.

Table.6 Number of guess node

	group 1	group 2
Among top 10 nodes of frequently visited sessions	6	5
Among top 20 nodes of frequently visited sessions	9	7
Among top 30 nodes of frequently visited sessions	10	7

Although data of one day is used in the experiment, the result of longer period of time is

similar. The algorithm mentioned here has higher probability of guess visitors' interest than the other algorithm without considering intermediate nodes, but only local and global ones.

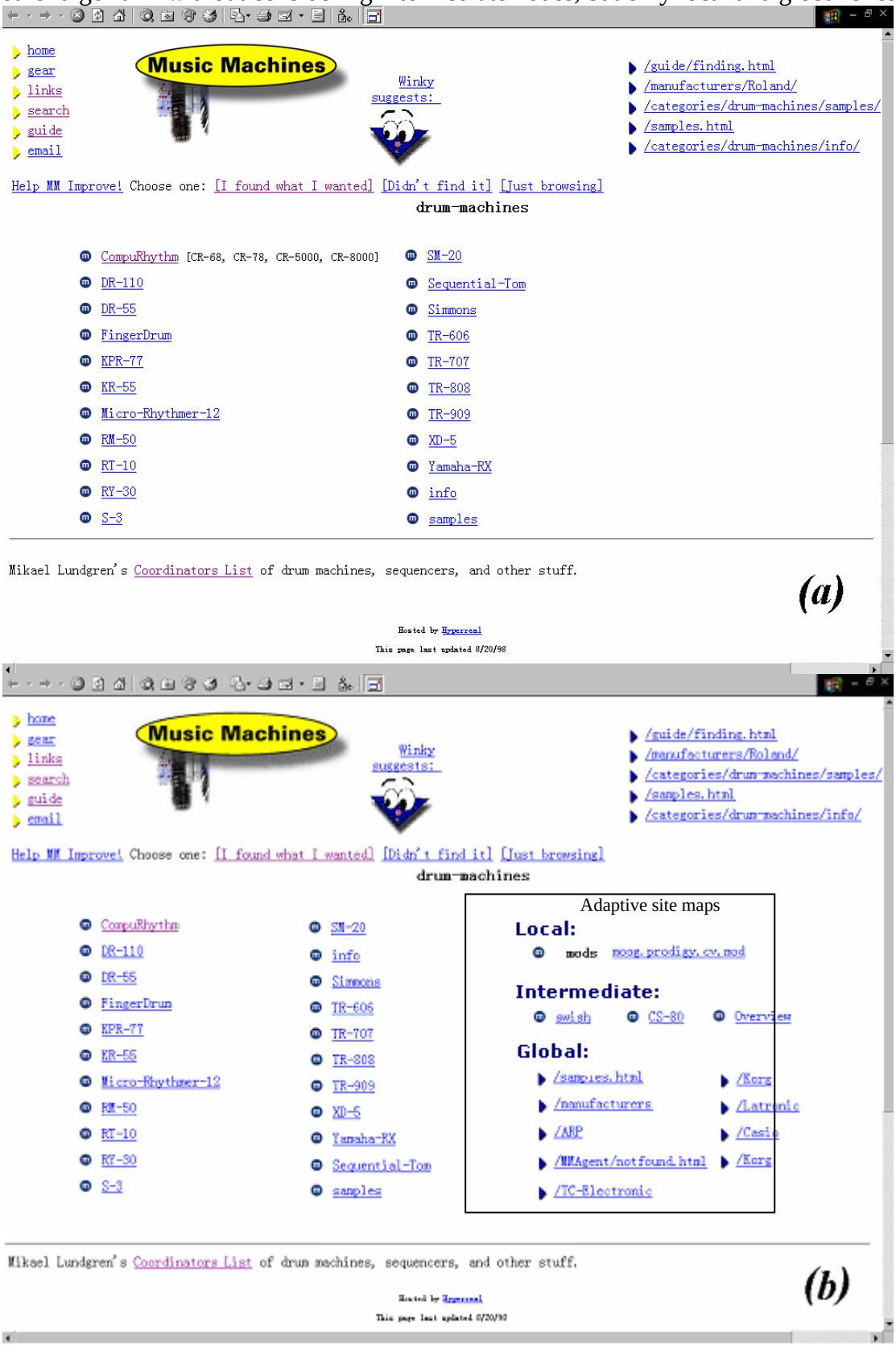

Fig.2 (a) The music-machines.org site map, and (b) a fictitious adaptive version that displays links to the reference pages.

5. Conclusion

In this paper, a method of building an adaptive site map based on domain and usage information is addressed. Two points of view, one is from the structure of a Web site, the other is from the usage of visitors, are all considered.

The following issues still need to be solved for further research:

1) Most of attention of this paper is paid to adding more links to nodes, which visitors should be interested in, and can guild visitors when they feel lost in hyperspace. To avoid lost some nodes as isolate ones, we do not consider deleting some irrelative links or making them invisible. But sometimes reducing links is also a must.

2) If not only the last node of a session is an objective one to visitors, the nodes on the halfway should also be considered. It depended on how to measure the relation between visitors and a node, to what level the visitor is interested in this page, and how to get this information.

3) Parameters K, M, O, w_1, w_2, W_{conn}, W_{access}, W_{depth} and n need to be determined through analysis of historical data.

ACKNOWLEDGEMENTS

This research is partially sponsored by:
1) Libra arts research promotion program, Fudan University: Jinmiao Project(EXH1019325): fuzzy availability-based e-business website structure evaluation and optimization method
2) Youth scientific research funds, School of management, Fudan University(CHH1019070): e-business website updating technology research
The authors also appreciate the assistance of Mike Perkowitz and Oren Etzioni from University of Washington for sharing the Web log data and Music Machine Site files, besides promptly answering questions of us.

REFERENCES

Chen M.S., Park J.S., and Yu. P.S. “Data mining for path traversal patterns in a Web environment,” in: Proc. 16th IEEE International conference on Distributed Computing Systems, 1996, pp. 385-392.

Kilfoil, M., Ghorbani, A., Xing, W., Lei, Z., Lu, J., Zhang, J., and Xu., X. “Toward an adaptive web: the state of the art and science,” In Communication Networks and Services Research (CNSR) 2003 conference, New Brunswick, CA, 2003, pp. 108-119,130.

Kim, J., and Yoo, B. “Toward the optimal link structure of the cyber shopping mall,” Int. J. Human-Computer Studies, 2000.52, pp. 531-551.

Mobasher, B., Dai, H., Luo, T., Sun, Y., and Zhu. “Integrating web usage and content mining for more effective personalization,” In Proceedings of the International Conference on E-Commerce and Web Technologies, Greenwich, UK, J. 2000, pp. 165-176,.

Mukherjea S., and Hara Y. “Focus + Context Views of World – Wide Web Nodes,” In Conference on Hypertext and Hypermedia archive Proceedings of the eighth ACM conference on Hypertext table of contents Southampton, United Kingdom, 1997, pp. 187 – 196.

Nielsen, J. “User interface directions for the web,” Communications of the ACM. 1999.42(1), pp. 65-72.

Srikant, R., and Yang, Y. “Mining web logs to improve website organization,” In the tenth International World Wide Web Conference, Hong Kong. 2001.

Toolan F., and Kushmetick N. “Mining Web Logs for Personalized Site Maps,” In the third International Conference on Web Information Systems Engineering Workshops (WISEw'02) Singapore , 2002, pp. 232-237.

Wang, Y.W., and Wang, D.W. “Design Strategy of Web Page for E-Supermarket,” Jiang Pingyu et. al. editors, International Conference on eCommerce engineering, Xi'an, Sep. 2001.(ISBN:7-900066-54-3, ID:10-01).

Wang Youwei, and Wang Dingwei. “Usability Evaluation and Link Structure Optimization Model of E-Supermarket Website,” Journal of Systems Simulation (In Chinese), 2003.2, Vol.15, No.2, pp. 190-192.