Easing Operative User Steering through Website Construction Development

R.Konda Reddy#1, Konduru Balakrishna#2
#1 Assoc. Professor, #2 student of M.Tech (C.S) Department of CSE, PBR Visvodaya Institute of Technology & Science, Kavali.

Abstract
Data Mining is a step of Knowledge Discovery in Databases. Clustering can be considered the most important unsupervised learning method so, as every other problem of this kind, it deals with finding a arrangement in a crew of unlabeled data and mining major issue is removing unrelavent data. But here we did not find these type data its fail on removing irrelevant and remove back tracking. For this type of problems Designing well-structured websites to provide operative user steering has long been a challenge. A key reason is that the web speculators understanding of how a website should be structured can be considerably different from that of the users. While different methods have been proposed to relink webpages to increase navigability using user steering data, the completely modernized new structure can be highly impulsive, and the cost of disorienting users after the changes remains unanalyzed. In this we addresses how to improve a website without presenting significant changes. Specifically, we propose a mathematical programming model to improve the user navigation on a website while minimizing alterations to its current structure.

Index Terms—Website design, user Steering, web mining, mathematical programming.

1 Introduction:
The growth of the internet is does not have any limitations. Now there are uncountable pictures, audios, videos, and other data available via internet and still it increasing. Finding required data or any content from websites has become a very difficult task. In order to satisfy the increasing demands from online customers, Firms or organizations are heavily investing in the development and maintenance of their website. Web developer’s does not have any idea about user’s requirements and can only organize pages based on their own judgment. In this paper, we are concerned primarily with transformation approaches. Transformation approach referred as to modify the site structure to ease the navigation for all users. The literature considering transformations approaches mainly focuses on developing methods to completely reorganize the link structure of a website. But this approach has lots of drawbacks. First, the complete reorganization could radically change the location of familiar items, the new website may disorient users [13].Second, the reorganized website structure is not easy to predict, and the cost of unorientation about where you are and how to precede users after the changes endures unanalyzed. This is because a website’s structure is designed by experienced web Developers. Finally, Website reorganization models may A web enhancement model facilitates good user navigation without making more changes to its current structure. Our model is suitable for those websites whose contents are not changing as per user requirement & data within the websites are fixed. For examples companies, hospitals, universities, banks, tourist websites. The presented model may not be appropriate for websites that contents dynamic pages and unpredictable contents. This is because in active web pages approach patterns changes frequently, therefore the weblog data is not used to improve the web site structure [14].In this we perform various tests on a data set collected from a genuine website. The result evidences that our model can improve the site structure with slightest changes. Here we study the problem of improving user steering on a website with minimal changes to the current structure then we calculate the out-degree as an objective function instead of fixed constraints. This tolerates a page to have more links than the out degree threshold if the cost is reasonable and hence offers a good balance between minimizing changes to a website and reducing information overload to users.

2. Related Work and Background
The growth of the Internet has led to numerous studies on improving user navigations with the knowledge mined from web server logs and they can be generally categorized in to web personalization and web transformation approaches [2].

Web personalization is the process of “tailoring” web pages to the needs of specific users using the
information of the users’ navigational behavior and profile data [15]. Perkowitz and Etzioni [2] describe an approach that automatically synthesizes index pages which contain links to pages pertaining to particular topics based on the co-occurrence frequency of pages in user traversals, to ease user navigation. The methods proposed by Mobasher et al. [16], [17], [18] and Yan et al. [19] create clusters of user’s profiles from weblogs and then dynamically generate links for users who are classified into different categories based on their access patterns. Nakagawa and Mobasher [3] develop a hybrid personalization system that can dynamically switch between recommendation models based on degree of connectivity and the user’s position in the site. For reviews on web personalization approaches, see [15] and [20].

Previous studies are divided into web transformation and web personalization. Web personalization means changing the web pages according to particular user or group of users need. Our paper is based on web transformation technique. Wenpu Xing, Ghorbani [1] suggest a weighted Page rank algorithm. This algorithm is the extension of Page rank algorithm. In this paper ranking is given to web pages based on number of time page referred. By using this algorithm, most relevant pages in large number are returned for a search query. Perkowitz and Etzioni [2] present idea of web personalization depending upon occurrence of frequency of pages in user traversal.

W. Yan, M. Jacobsen, H. Garcia Molina and U. Dayal [5] suggest an approach in which users are classified based on pages that user visits. Using web pages log, group of users are determined who visits similar web pages. This data is used to improve navigation convenience. According to users group they recommend links dynamically. Fu, M.Y. Shih, M. Creado and C. Ju [8] investigate adaptive web site that automatically improve organizational presentation by learning from visitor’s access pattern. According to access information web pages are. Various attempts were made to evaluate success of a website since the web is invented. The mining of web server log can be used in two ways one is web personalization approach and other is transformation approach. Web personalization is the process of “tailoring” WebPages to the needs of specific users using the information of the users’ navigational behavior and profile data. Perkowitz and Etzioni [6] describe an approach that automatically synthesizes index pages which contain links to pages pertaining to particular topics based on the co-occurrence frequency of pages in user traversals, to ease user navigation.

3. Proposed System

In candidate link set first we find out the existent links. Sometimes it is possible that existent links are found in candidate link set. This is because of location of link or visibility of links is not properly placed. So, first we need to improve those links. Next, we need to find out relevant candidate link set. Relevant candidate links means links having larger path than path threshold. For filtering candidate links first use path threshold. Path threshold means maximum number of path allows reaching target page. Path threshold 1 means we consider only that link by which user reach target page in first path. Then we select those links. We cannot add all these links. We can add only those candidate links that are common for all users. For these we need to find out similarity between links. For finding similarity between links we use Dice’s coefficient index. We apply index on relevant candidate link set. We obtain the dice’s index between 0 and 1. 0 means dissimilar and 1 means exact similar. Then we use KNN classifier. KNN means k nearest neighbor classifier. KNN stores the entire link set and based on similarity measure classify candidate links into two groups. In KNN we use one threshold value to classify candidate links into two groups. One group contains candidate links which are present in more number of users than threshold value. Other group contains candidate links which are present in less number of users than threshold value. We filter second group. And we use the candidate links in first group means the candidate links present in user more than path threshold to re-link or redesign website structure. In other words, we say that if we add that link then problem for more users to locate target page will be solved.

Computational Experiments and Performance Evaluations

Extensive experiments were conducted, both on a data set collected from a real website and on synthetic datasets. We first tested the model with varying parameters values on all data sets. Then, we partitioned the real data into training and testing data. We used the training data to generate Out-Degree Statistics improved site Constructions which were evaluated on the testing data using two metrics that are discussed in detail later. Moreover, we compared the results of our model with that of a heuristic.
Real Data Set
Description of the Real Data Set
The real data set was collected from the Music Machines website (http://machines.hyperreal.org) and contained about four million requests that were recorded in a span of four months. This data set is publicly available and has been widely used in the literature [10]. Table 1 shows the number of pages in the website that had out degrees within a specified range. This website has in total 916 pages, of which 716 have an out-degree of 20 or less, with the majority (83 percent) of the remaining pages having 40 links or less.

Before analysis, we followed the log preprocessing steps described in to filter irrelevant information from raw log files. These steps include: 1) filter out requests to pages generated by Common Gateway Interface (CGI) or other server-side scripts as we only consider static pages that are designed as part of a website Construction, 2) ignore unsuccessful requests (returned HTTP status code not 200), and 3) remove requests to image files (.gif, .jpg, etc.), as images are in general automatically downloaded due to the HTML tags rather than explicitly requested by users [9].

We utilized the page-stay time to identify target pages and to demarcate mini sessions from the processed log files. Three time thresholds (i.e., 1, 2, and 5 minutes) were used in the tests to examine how results changes with respect to different parameter values. Furthermore, we adapted the algorithm proposed in [11] to identify the backtracking pages in mini sessions, which are then used to demarcate the paths traversed to reach the targets. Table 2 lists the number of mini sessions comprising a given number of paths ð> 1Þ for different time thresholds.

Results and Analysis — Real Data Set
We set " ¼ 1:0E 8 and vary the out-degree threshold, the path threshold, and the multiplier for the penalty term to examine how results change with respect to these parameters.

<table>
<thead>
<tr>
<th>Out-degree</th>
<th>Number of pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;100</td>
<td>1</td>
</tr>
<tr>
<td>81–100</td>
<td>2</td>
</tr>
<tr>
<td>61–80</td>
<td>10</td>
</tr>
<tr>
<td>41–60</td>
<td>166</td>
</tr>
<tr>
<td>21–40</td>
<td>538</td>
</tr>
<tr>
<td>11–20</td>
<td>178</td>
</tr>
<tr>
<td>6–10</td>
<td>916</td>
</tr>
</tbody>
</table>

Table 2 reports the experiment results.
5 References

Authors: