

# Extracting a Website's Content Structure From its Link Structure

Nan Liu

Department of Systems Engineering and  
Engineering Management  
The Chinese University of Hong Kong  
nliu@se.cuhk.edu.hk

Christopher C. Yang

Department of Systems Engineering and  
Engineering Management  
The Chinese University of Hong Kong  
+852 2609 8239  
yang@se.cuhk.edu.hk

## ABSTRACT

Hierarchical models are commonly used to organize a Website's content. A Website's content structure can be represented by a topic hierarchy, a directed tree rooted at a Website's homepage in which the vertices and edges correspond to Web pages and hyperlinks. In this work, we propose an algorithm for extracting a Website's topic hierarchy from its link structure. The proposed algorithm consists of a construction stage and a refining stage, in which we analyze the semantic relationships between web pages based on link structure, web page content and directory structure. We've done extensive experiments using different Websites and obtained very promising results.

## Categories and Subject Descriptors

H.3.3 [Information Systems]: Information Search and Retrieval  
– search process, retrieval models

## General Terms

Algorithms, Experimentation

## Keywords

Content Structure, Website Mining, Topic Hierarchy

## 1. INTRODUCTION

Users looking for information on the web frequently need to explore particular websites carefully to locate individual pages with interesting information. Many websites provide sitemaps to facilitate navigation. Sitemaps usually list the major topics of a website. By taking a look at the site map, one may quickly settle on one or more topics that are of interest. Despite the usefulness of sitemaps, websites with sitemaps only account for a small portion of the entire web. In addition, sitemaps are usually manually constructed and could therefore only cover a limited number of pages.

Individual websites are more organized compared with the entire web. The development of most websites involves content planning and organization. Hierarchical model is a frequent choice for the organization of complex bodies of information on websites because of its simplicity and clarity. Under this model, a large website is first divided into a number of broad topics, which are recursively divided into more subtopics. Since each topic has a corresponding web page, the hierarchical content structure is realized as a directed tree where the web pages are nodes and the hyperlinks point from each topic to its sub topics.

Copyright is held by the author/owner(s).  
CIKM'05, October 31-November 5, 2005, Bremen, Germany.  
ACM 1-59593-140-6/05/0010.

The Stanford Database Group website illustrated in Figure 1 contains several major topics such as *Member* and *Project* which have individual members and research projects as sub topics.

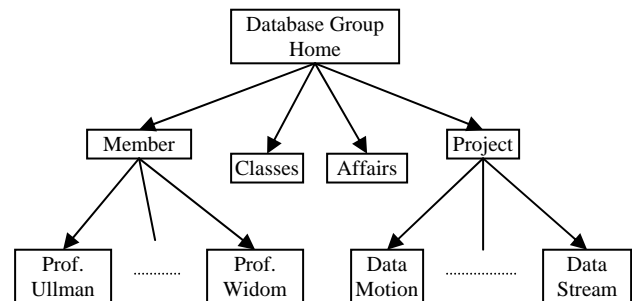


Figure 1: Partial Topic Hierarchy of  
www.db.stanford.edu

We define such hierarchical content structure of a website as its **topic hierarchy**. More formally, a topic hierarchy is a directed tree that is required to be rooted at the homepage of the website and provides a path formed by hyperlinks from the root to every page in the website. In this paper, we study the automatic construction a website's topic hierarchy, in particular, how to extract it from the link structure of the website, which is in the form of a complicated graph.

## 2. TOPIC HIERARCHY MINING

### 2.1 Overview:

The semantic relationships between web pages can generally be classified either as aggregation or association. Aggregation is a kind of directed relationship, in which one page represents a broader concept and subsumes the other. Association is non-hierarchical and symmetric, in which two pages represent parallel concepts. Clearly, the topic/subtopic relationship to be captured by the topic hierarchy corresponds to aggregation. Our algorithm for building the topic hierarchy consists of two stages: a construction stage which produces a primitive tree structure, and a refining stage in which this structure is adjusted for enhancement.

A key feature used in our algorithm is the semantic dissimilarity between two pages, which is inferred based on the contents of the web pages as well as their locations in the website's directory. We model web page's content by the set of distinct terms within its text and measure two page's content dissimilarity is measured based on their vocabulary overlap [3]:

$$d_{cont}(u, v) = 1 - \frac{|S_u \cap S_v|}{|S_u \cup S_v|} \quad (1)$$

where  $S_u$  and  $S_v$  are the sets of terms on  $u$  and  $v$  respectively. Directories are commonly used for organizing large number of documents. Most web designers use folders and subfolders to group related pages and establish boundaries between distinct content units [1, 3]. Given two web pages  $u$  and  $v$  with paths  $u_1/u_2/.../u_n$  and  $v_1/v_2/.../v_m$  respectively, the path dissimilarity between them is computed based on the number of common folders they are both under:

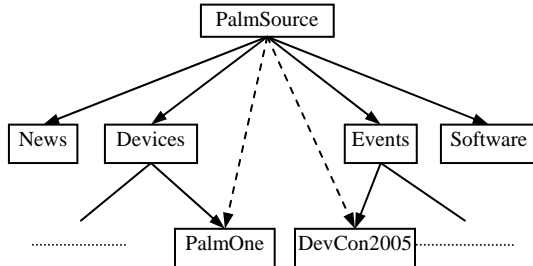
$$d_{dir}(u, v) = 1 - 2 \times \frac{\min\{i \mid 1 \leq i \leq \min(m, n) \wedge u_i \neq v_i\} - 1}{m + n} \quad (2)$$

The overall semantic dissimilarity  $d(u, v)$  is defined as  $\lambda d_{cont}(u, v) + (1 - \lambda) d_{dir}(u, v)$ .

## 2.2 Topic Hierarchy Construction

A basic model for representing a Website's link structure is the unweighted directed graph model  $G(V, E)$ . Breadth first search (BFS) can efficiently build a directed tree from this graph. However, BFS treats all the links uniformly without regard to the underlying semantics and finds a path to each page by minimizing the number of links. To incorporate more semantics, we adopt the weighted graph model  $G(V, E, \omega)$ , where the cost of a link  $u \rightarrow v$  is equal to the semantic dissimilarity  $d(u, v)$ . We then perform shortest path search (SPS) using Dijkstra's algorithm. The output of SPS is a directed tree rooted at the homepage where the tree paths represent the shortest paths between the homepage and all the other pages.

## 2.3 Topic Hierarchy Refinement



**Figure 2: The correct paths in solid links and the short cuts formed by the dashed arrows.**

In a topic hierarchy, the correct paths to very specific topics usually consist of a number of links. A page could not only be linked from its parent in the hierarchy but also from some of its ancestors. This creates short cuts, which are paths from the homepage to another page that bypass one or more of its ancestors. Short cuts are very common as many web pages contain highlights or promotions that link directly to some of its sub topics. Figure 2 illustrates two particular short cuts of [www.palmsource.com](http://www.palmsource.com). *PalmOne* and *DevCon2005* are sub topics of *Devices* and *Events* respectively. However, the homepage is highlighting these two items and thus creates short cuts. In cases like this, merely measuring path cost may be misleading and even justify the incorrect paths.

In the refining stage, we supplement the path cost measure with an additional feature, which is based on a page's semantic similarities with its siblings in the initial tree. Given an initial topic hierarchy  $TH$ , we adjust the location of a page  $c$  as follows. Let  $subtree(TH, c)$  denote the sub tree of  $TH$  that is rooted at  $c$  and let  $TH/subtree(TH, c)$  be the portion of  $TH$

excluding  $subtree(TH, c)$ . We want to choose a page from  $TH/subtree(TH, c)$  as the best choice for the parent of  $c$ . For a possible parent page  $p$  of  $c$ , we compute the cost of choosing  $p$  as the parent of  $c$ , denoted by  $cost_c(p)$ , which depends on both the **path cost**  $cost_{c,path}(p)$  and the **sibling cost**  $cost_{c,sibling}(p)$ , and choose the parent with minimum  $cost_c(p)$ . The path cost  $cost_{c,path}(p)$  is just the cost of the path from root to  $c$  going through  $p$ , which is measured the same as during construction stage. The newly introduced sibling cost,  $cost_{c,sibling}(p)$  is computed as average semantic dissimilarity between  $c$  and the other offspring of  $p$ . The idea behind this measure is that if correctly located, a page is expected to be grouped with some siblings that are highly semantically similar to it as they all represent sub topics of the same topic. The total cost  $cost_c(p)$  is computed as  $cost_c(p) = \gamma cost'_{c,path}(p) + (1 - \gamma) cost'_{c,sibling}(p)$ .

At refining iteration, pages are examined and adjusted in the breadth-first order. In the experiments, only very few iterations are needed for the tree to stabilize.

## 3. EXPERIMENTS

We tested the algorithm using 5 different web sites. A human judge was asked to manually construct a topic hierarchy for each test website for use as benchmark. Given the benchmark, we evaluate the accuracy of a generated topic hierarchy based on the proportion of web pages which are connected to the same parent as in the benchmark.

The breadth first search (BFS) algorithm is used as the baseline for comparison. We compare BFS with shortest path search (SPS) and shortest path search plus refinement (SPS+REF). For SPS and SPS+REF, we report their performance with the setting  $\lambda=0.5$  and  $\gamma=0.5$ , which was found to be optimal. The accuracies of topic hierarchies generated by these methods are shown in Table 1. Clearly, SPS outperforms BFS significantly, and the proposed refining procedure can make further improvement on SPS as shown by the column of SPS+OPT.

**Table 1: Performance of baseline algorithm**

Web Site	BFS	SPS	SPS+REF
<a href="http://www.db.stanford.edu">www.db.stanford.edu</a>	74.3%	83.1%	89.5%
<a href="http://www.cs.cmu.edu">www.cs.cmu.edu</a>	77.2%	81.4%	86.9%
<a href="http://www.research.ibm.com">www.research.ibm.com</a>	71.9%	78.4%	82.1%
<a href="http://www.whitehouse.gov">www.whitehouse.gov</a>	79.4%	88.5%	96.3%
<a href="http://www.palmsource.com">www.palmsource.com</a>	71.3%	89.7%	94.2%

## 4. CONCLUSION

We have developed a novel technique to build a Website topic hierarchy by analyzing the link structure, directory structure and content similarity. We have compared with previous techniques and showed that our proposed technique outperforms.

## 5. REFERENCES

- [1] W.S. Li, O. Kolak, Q. Vu and H. Takano. Defining Logical Domains in a Website. Proc. of 11<sup>th</sup> ACM Conf. on Hypertext and Hypermedia, San Antonio, 2000
- [2] Z. Chen, S. Liu, W. Liu, G. Pu and W.Y. Ma. Building a Web Thesaurus from Web Link Structure. In Proc. of the 25<sup>th</sup> ACM SIGIR Conference, Finland, 2002
- [3] N. Liu and C. C. Yang. Mining Web Site's Topic Hierarchy. In Proc. of International World Wide Web Conference, Tokyo, Japan, 2005.