

DOMISA: DOM-based Information Space Adsorption for Web Information Hierarchy Mining

Hung-Yu Kao, Jan-Ming Ho*, and Ming-Syan Chen

Electrical Engineering Department
National Taiwan University
Taipei, Taiwan, ROC
E-Mail: {bobby@arbor.ee.ntu.edu.tw,
mschen@cc.ee.ntu.edu.tw}

Institute of Information Science*
Academia Sinica
Taipei, Taiwan, ROC
E-Mail: hoho@iis.sinica.edu.tw

Abstract

Due to the growth of dynamic page generation techniques, the amount and the complexity of Web pages has been increasing explosively, as has the information contained within Web pages. Redundant and irrelevant information is distributed and mixed throughout a page, making it difficult to automatically identify the useful information in that page. Consequently, we propose an **information hierarchy** in this paper, and, from that hierarchy, we can extract the significance and the relationship value of information contained within a Web page. We can then use this hierarchical structure to create a new browsing process. Our **DOM-based Information Space Adsorption (DOMISA)** system applies information theory to map information in a page into an information space, and our **gradient tree adsorption (GTA)** process uses the document object model (DOM) trees of pages to build information hierarchies. Experiments on several commercial news Web sites show high precision and recall rates achieved by DOMISA in determining information clusters of pages which validates its practical applicability to Web sites.

1. Introduction

Due to the growth of dynamic page generation techniques, the amount and the complexity of Web pages has been increasing explosively, as has the information contained within Web pages. Some regions of a Web page contain the main context of the page or provide good links to relevant pages. Some regions, however, contain meaningless information in regard to desired information. These regions are distributed and mixed throughout a page which makes it difficult to identify useful information.

Many Web pages are generated online for the purposes of maintenance, flexibility, and scalability of Web sites. They are usually generated by predefined templates and contents stored in back-end databases. Most commercial Web sites, such as portal sites, search engines, e-commerce stores, news sites, apply a systematic technique to generate Web pages and to adapt various requests from numerous Web users. These Web sites are referred to as *systematic* Web sites in [13]. Information mixing occurs frequently in systematic Web sites, and much of this information is often redundant or irrelevant, such as menu bars or advertisements [1][17]. Redundant information is

information repeated in most pages in a Web site and irrelevant information is irrelevant to the main topic of the page. Navigation panels, advertisements, service catalogs of services, copyright and privacy policy announcements are contained in almost all pages of a systematic Web site, and are redundant and irrelevant to user needs. Information mixing increases the difficulty for search engines, crawlers and information agents to extract the useful information. It is difficult for these applications to easily recognize desired information which results in the need to crawl and index entire Web pages. Moreover, redundant and irrelevant information negatively effects users of small display devices, such as PDAs and mobile phones. According to the general browser design of a PDA, users must scroll continuously when searching because the whole page cannot be displayed in the screen. Users have to scroll past and ignore unnecessary content, while dealing with a low-bandwidth wireless connection. The following examples illustrate these problems.

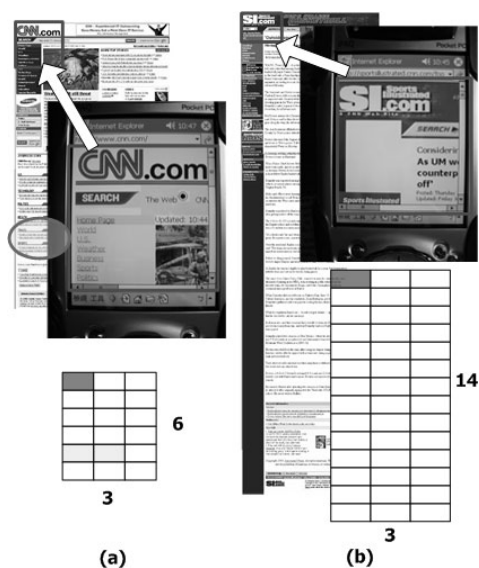


Figure 1: Using a PDA to browse Web pages at a systematic Web site.

Example 1: Figure 1 shows an actual case using a PDA to browse the homepage and one article page of a news Web site, <http://www.cnn.com>. We used a PocketPC with Windows CE 3.0 with a screen resolution of 240x320 to

browse the Web. In case (a), we scrolled about 18 times to view the entire page. The information displayed on the screen was always scattered and non-integrated. We even needed to repeatedly scroll up and down in order to understand the displayed information. The article page, case (b), is always larger than the TOC (Table of Contents) pages. Users dealing with a small screen are likely to get lost on a big page.

Generally, when searching for information in a Web page, blocks likely to contain desired information are searched first, then more fine-grained blocks recursively until the desired information is found. Instead of recursively searching within a page, hierarchical information searching is a more straightforward process for finding and understanding page information. Figure 2 shows an example of searching for and finding information in a systematic Web site. When we want to find today's top sports stories in a news Web site, we first (1) access the block labeled *top stories* (Block A) and search the news items in this block. Unfortunately, in this example, we do not find any information about sports. We therefore (2) find a news catalog block (Block B), and (3) locate the sports catalog. Two top sport news articles are listed in this catalog. We then (4) select one of the articles, or (5) click the SPORTS bar linking to the sports news Web site in order to access more sports news. Note that Blocks D and E, which contain only redundant and irrelevant information, are included in most pages in the CNN news Web site and are ignored in this search flow. If we process the flow on small display devices, we would want to view only the desired information (Blocks A and B) to reduce unnecessary page scrolling.

The above information searching flow is hard to achieve without human's intervention for information agents or search engines to decide and extract useful and important information in a page. General search engines always crawl and index everything in a page and decide which words or paragraphs are more important by analyzing term frequency and inverse document frequency, a.k.a., *tf-idf* [3]. In the general design of information agents, human's knowledge is first called for decide which blocks are needed for agents to crawl. It is difficult for general search engines and information agents to find important information for users. The reasons are that it is difficult for them to recognize the significance of information in a page and they always consider the contents of Web pages as a linear data stream. In such a data stream, contexts and anchors are treated equally. In Figure 2, words in Block A and Block E are assigned by the same weights if they have the same value of *tf-idf*. Moreover, links in the SPORTS catalog and the EDUCATION catalog are neighboring in the contexts of the page and are usually extracted together.

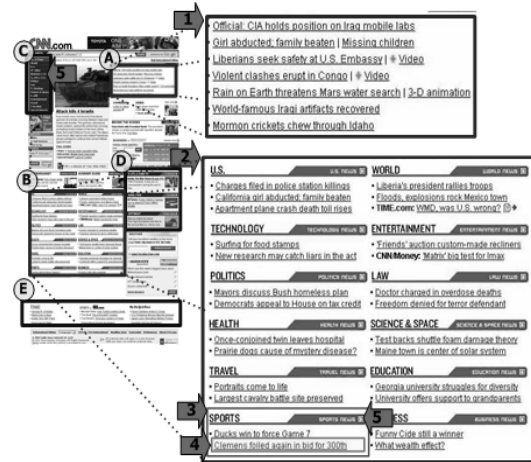


Figure 2: An example of information search flow at www.cnn.com.

As the number and complexity of pages is growing, finding desired information on the Web has become quite important. Extracting and classifying informative regions of Web pages to recognize the significance and relation of different context and links is a crucial issue to increase precision and decrease the cost of search engines and information agents and make the surfing on the small display devices more effectively. The structure of information within a page to represent the significance and relation of information is called the *information hierarchy* of a page. We here focus on the mining of news Web sites to demonstrate the problem and the solution proposed in detail.

Pages in systematic Web sites are similar in structure as they are usually generated from the same template and are assembled according to a set of fundamental information clusters. An *information cluster* is a sub-structure of a page which provides a unique semantic representation to users among pages in a Web site and is composed of information elements or smaller information clusters. An *information element* is one context or an anchor with a non-zero length. Note that the whole content of a page forms the root information cluster of the page. Examples of information clusters include the blocks of searching results in search engines, entries of classified commodities in auction sites, the table of contents (or referred to as TOC) blocks of news categories and news articles, to name a few. In this paper, an information element that provides good information is called an *information authority*. An information element is called an *information hub* if it contains information that links to information authorities. An information cluster contains hybrid characteristics of the information hub and the information authority. The TOC blocks in news Web sites are, for example, information hubs which contain news anchors linking to news articles, which are information authorities, and they also contain the characteristic of information authority due to the inclusion

of abstracts of these news articles. These definitions of the information authority / hub are similar to those in [14], but different from the latter in that here they are not specific to any topic.

Information hierarchy consists of a set of hierarchical information clusters that provide specific configurations for the information of a page. The higher the level of an information cluster, the more generalized information this cluster provides. The information hierarchy of a Web page represents the information in a page by hierarchical configurations and can be considered as the abstraction and summarization of this page. A full-length text document contains a set of subtopics [11]. We believe that a quality summarization should cover as many subtopics as possible. This accounts for the reason that we employ the information hierarchy to represent a Web page in different configurations so as to have a broader coverage of information subtopics than traditional summarization models can. Also, in an HTML document, tags are inserted for purposes of the page layout, content presentation and for providing interactive functions, e.g., form filling and document linking. The tagging structure therefore corresponds to the knowledge of representation and semantics of Web pages. In this paper, we extract and use the knowledge in the tagging tree structure, or referred to the **Document Object Model (DOM)** [21], and apply Information Theory to mine the information hierarchy.

We develop a **DOM-based Information Space Adsorption (DOMISA)** system to automatically build the information hierarchy of each page in a Web site according to both the page information and the structure of a page. The mining flow of the system consists of three main phases: (1) the **Information Space Tree (IST)** building phase, (2) the **Gradient Tree Adsorption (GTA)** phase, and (3) the information hierarchy building phase. The main mining flow in **DOMISA** is to use the information theory to evaluate the information characteristics and scales of content and sub-structures, and then to construct the information hierarchy by applying the specific information adsorption method. The information adsorption method first applies heuristic rules to reduce the original DOM tree to be a concise one and then evaluates the average distance between neighboring clusters and their parent cluster. According to the average distances, the **GTA** process is applied to generate different information configurations. The information hierarchy is constructed by the set of all configurations. Adsorption results show that **DOMISA** effectively extracts the information hierarchy of a page and provides an effective surfing interface for small display devices. Experiments on several real news Web sites show high precision and recall rates achieved by **DOMISA** in determining information clusters of pages which validates its practical applicability on real Web sites.

2. Preliminary

2.1 The information space adsorption approach

In Figure 2, there are five information clusters marked in this figure and they are all composed of smaller information clusters or elements. Information Cluster A is the hot news block containing links, each of which is an information element, to the hot news pages on the crawling day. Cluster B contains classified news links and consists of twelve smaller information clusters which have different types of news links. Cluster C is a navigation panel and is composed of several links. The composing models of Clusters D and E are similar to that of Cluster B, but are in essence redundant information and distributed among most pages in the CNN Web site. The page in this figure is a typical TOC page containing links to news articles. All information clusters are information hubs, but differ from one another on their information scales. Information scale of a cluster or element is the amount of relevant information it provides. For example, the TOC blocks (Clusters A and B) in news Web sites have larger information scales than navigation panels and service catalogs (Cluster C). Note that a cluster containing a news article only has a large scale of information authority, but the scale of information hub is zero due to no links contained in it.

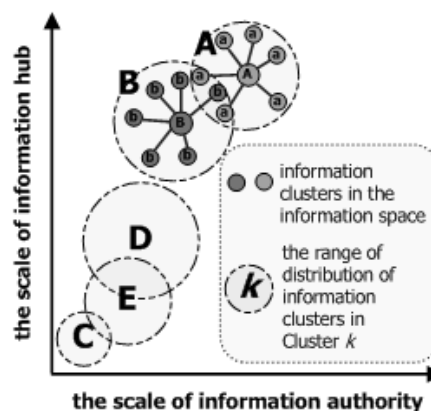


Figure 3: The distribution of information clusters of Figure 2 in the information space.

We use a to denote the scale of the information authority for an information cluster and use h to denote the scale of the information hub. According to this tuple value of the information authority and hub, i.e., (a, h) , an information cluster can be mapped into a two-dimensional space as shown in Figure 3. Nodes in the space represent the scales and characteristics of information clusters. We call this space an **information space** to represent the concept. Clusters A and B contain direct links to news articles and therefore they attain higher h values than those in Clusters C, D and E as shown in Figure 3. The anchor texts in Clusters A and B are often the abstractions of linked news articles, so that they also have higher a values than clusters in C, D and E. When information clusters of a page are all

mapped in the information space, we can discriminate whether any two clusters are similar to each other or not by examining the distance between nodes in the space.

Note that in a Web site, neighboring information clusters usually have similar information scales and characteristics. In view of this, we can merge them into a larger block to represent more generalized information. Neighboring clusters are clusters with the same parent information cluster. This merged structure is more generalized in relation to its merged clusters. This merging process is referred to as *information adsorption* in this paper. Adsorption merges the most similar information clusters first, and the resulting generalized information cluster adsorbs the information from the merged clusters. For example, k information clusters, each of which is one of the top- k searching results generated by search engines, form a generalized cluster of the top- k searching results. Adsorption should not only merge the most similar clusters first, but also meet the hierarchical constraint of a DOM tree. In Figure 3, two smaller clusters in Cluster A, labeled a , are also located into the distribution range of Cluster B. They are more similar to their neighboring b clusters than to other a clusters. However, to meet the hierarchical constraint, these two small clusters will not be merged with their neighboring b clusters, and instead remain in Cluster A. We define a set of information clusters in the information space and the hierarchical constraints as the *information space tree (IST)* of the page. Information adsorption is the node clustering process for the information space tree.

We can apply the information adsorption iteratively on several neighboring clusters that contain the most similar information in order to form a larger cluster in each clustering step. After each step, a page is composed of several disjointed information clusters. We define the configuration formed by the set of information clusters created from the k -th step of adsorption as the k -th level clustering, denoted by L_k . Adsorbing causes the number of clusters to decrease as k increases, and generates more generalized page representation at each step. An information hierarchy is built from Clusterings L_0, L_1, \dots, L_n , where Clustering L_0 is the configuration of all information elements, and Clustering L_n is the configuration of the final converged clustering when $L_n = L_{n+1}$. Consequently, by applying information adsorption to achieve converged clustering, we can build the information hierarchy of a Web page, which contains the different grained information representations. Users can traverse the information hierarchy to obtain the desired information. Moreover, the higher the level of the clustering, the more divergent these clusters are. As such, the redundant and irrelevant information can be more easily discriminated and filtered out to make the surfing more efficient.

2.2 Related work

In a systematic Web site, information clusters are usually generated by an iteration program. Entities in clusters are therefore similar to one another in view of the tag patterns. Frequent substructure mining is a candidate solution to extracting the fundamental information clusters automatically. Recently, frequent substructure mining of the DOM trees of semistructure pages has been studied in [1][8] in which the frequent sub-tree was extracted by using respective pattern mining and noise node concealment methods, such as the wildcard mechanism in [8] and node-skip and edge-skip pruning in [1] respectively. Work has also been done on tree pattern mining in order to extract metadata information in Web pages [9][22]. However, mined blocks that have the *same* tree structure may contain *different* semantic information. We must apply additional information measurement methods to filter out redundant information blocks among useful blocks with the similar tree structure. Moreover, some informative blocks, such as article blocks, are laid out with the unique structures and are indeed difficult to extract by the frequent structure mining.

Research in [16] provides a mechanism to construct the multi-granularity and topic-focused Web site maps. It uses directory paths, page contents, and link structure knowledge to build the semantic site-maps. A constructed site map can be considered a site-level information hierarchy, different from the proposed page-level information hierarchy in the paper and a training dataset needs to be prepared for the classifier generation.

Many works on information extraction have been proposed to extract the informative blocks of a page, which can be considered as the informative clusters in the information hierarchy of a page. Works on wrapper [2][15] provide learning mechanisms to mine the extraction rules of documents. Works in [1][12] also provide auxiliary systems to aid in extracting information boundaries of semistructure documents. However, they need either a pre-marked training set or a considerable amount of human involvement in the process of information extraction.

Gupta et al. [10] uses a rule-based method to deal with the tagging structure and applies a link/text removal ratio to represent the characteristics of blocks. A dictionary of advertisement servers is used to remove the advertisements. Yu et al. [23] uses the vision-based content structure of Web pages to extract relevant blocks. It also uses a hierarchical structure method indicating the different views of a Web page. These two DOM-based extraction methods use tagging structure and tag semantics to recognize block boundaries and block semantics, but they do not consider context semantics, which may be informative or redundant. The work in [13][18] dealt with the mining informative

blocks delimited by <TABLE> tags as opposed to our fine-grained blocks delimited by any kinds of tags.

Buyukkokten et al. [4] use a strategy called *accordion summarization* in which a page can be shrunk or expanded in a tree view and they also discuss a method to transform a Web page into a hierarchy of individual content units called Semantic Textual Units, or STUs, which are built by analyzing syntactic features of an HTML document. These summarization methods only provide an abstraction view for users to surf. Users still need to browse all the abstraction to search for what they want.

Document summarization is a workable approach for fitting Web pages into the small display device screens. Buyukkokten et al. [5] generate an accordion representation for a Web page so that detailed content can be folded or unfolded at the client side. Multiple graphic approaches [19][6] provide a graphic view of the original Web page so users can zoom-in, zoom-out or select the region they want and only the selected region is shown in the device. It is difficult, however, for users to find the location of the desired information from the thumbnail-sized index page. Some commercial approaches have been proposed for formatting Web pages to fit in the small display devices, such as the Opera browser (<http://www.opera.com>). This approach uses the handheld CSS media type. However, basically ends up only reorganizing the content of the Web page to fit in a small display device and still requires scrolling to search for content.

3. DOMISA: DOM-based Information Space Adsorption

Our DOM-based information space adsorption system automatically builds Web page information hierarchy according to page tree structures. The mining flow of the system, shown in Figure 4, consists of three main phases: (1) the *Information Space Tree (IST)* building phase, (2) the *Gradient Tree Adsorption (GTA)* phase, and (3) the information hierarchy building phase. In the first phase, we extract and aggregate useful features from the information of the DOM tree. We use the values of these features to calculate the values of information hubs and information authorities of nodes, as well as to map the DOM tree into two-dimensional information space, i.e., to create an *IST* from the DOM tree. In the second phase, we apply a set of tree reduction rules to remove the dummy nodes from the *IST* and calculate the children diversity for each node. Children diversity is the diversity between the information characteristics and scales of the children nodes. Gradient tree adsorption of phase 2 is then applied to produce information clustering in different levels. *GTA* method converges the reduced *IST* by incrementally merging neighboring information clusters according to Children Diversity (*CDiv*) values. After the clustering process

complete, i.e., Clustering L_n is reached, we then use a hierarchy compression rule to merge non-overlapped clusterings in order to compact the information hierarchy.

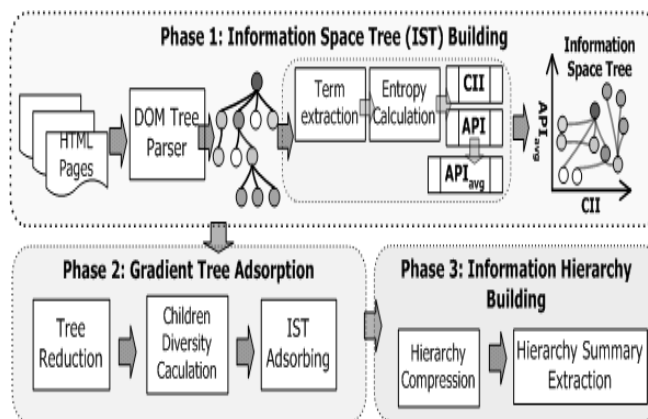


Figure 4: Information adsorption flow.

3.1 Phase 1: Information Space Tree Building

In DOM tree T , each node represents a tag in the Web page and contains tag name information, tag statement attributes and its *innerText*, i.e., the context delimited by the tag. According to the definition of DOM, the *innerText* context of node N includes all node contexts in the sub-tree rooted by node N . We use $T(N)$ to denote the sub-tree rooted by node N . We utilize some node features to indicate the scales of the information authority and the information hub, namely the content information index (*CII*) which is calculated from the entropy values and indicates the amount of information contained in the element or the cluster, and the anchor precision index (*API*) which represents the similarity between the anchor-text and the linked document to indicate that an anchor is informative or redundant.

To calculate these two features, we parse the *innerText* of the root node to extract meaningful terms. A term corresponds to a meaningful keyword or phrase. After extracting terms from all crawled pages, we calculate the entropy value of each term according to its term frequency. From Shannon's information entropy [20], the entropy of term $term_i$ can be formulated as:

$$EN(term_i) = -\sum_{j=1}^n w_{ij} \log_n w_{ij}, \text{ where } n = |D|, D \text{ is the set of pages,}$$

in which w_{ij} is the value of normalized term frequency in the page set. Entropy calculation is applied because terms distributed in more pages in a Web site usually carry less useful information. In contrast, those appearing in fewer pages most likely carry more important information. When entropy values of terms are calculated, we average the entropy values of terms in an *innerText* of node N to get $CII(N)$, the content information index of node N , i.e.,

$$CII(N) = \frac{\sum_{j=1}^k EN(term_j)}{k}, \text{ where } \forall_{j=1-k} term_j \text{ in } innerText \text{ of } N.$$

The CII value of node N represents the amount of information carried in a sub-tree rooted by N . We use the index to represent the information authority scale of a node. In addition, when browsing the Web, people use anchors to get desired information according to the semantics of anchors. Anchor semantics can be comprised of the anchor text, the text surrounding the anchor, an image, and/or other dynamic representations generated by scripts. Anchor semantic is expected to be relevant to the page to which it links. Such relevance is, however, weak in some cases. We therefore define the value of the anchor precision index to indicate the actual correlation of the anchor and its linking page. We use the anchor text to evaluate the value of API . The correlation index API is defined as:

$$API(N) = \sum_{j=1}^m \frac{1}{EN(term_j)},$$

where $term_j$ concurrently appears in both the anchor text of N and the linked page, and m is the number of matched terms. Note that $EN(term_i)$ is always larger than zero because $term_i$ appears in at least two documents. We use the value $API_{avg}(N)$ to indicate the average API value of anchor tags in the sub-tree $T(N)$. The value is normalized by dividing the max $API_{avg}(N)$ value of a page. We use the normalized $API_{avg}(N)$ to specify the scale of information hub of a node. We then map the original DOM tree into the information space after each tuple value ($CII(N)$, $API_{avg}(N)$) for node N is calculated to form the information space tree. The information adsorption process is the clustering of nodes in the space according to the hierarchical constraints of a DOM tree.

3.2 Phase 2: Gradient Tree Adsorption

Before performing tree adsorption, we first apply some heuristics rules to remove dummy nodes in order to reduce the original IST . A node that does not affect the information of a Web page when eliminated from the tree is considered as a dummy node. As shown in Figure 5 (a) and (b), there are two basic types of dummy nodes in our system. Type **D1** is a node which does not contain any meaningful context. Meaningful context is all context excluding symbols, spaces, and stop words. Tags $\langle BR \rangle$ and $\langle HR \rangle$, for example, are type **D1**. Note that when a D1 dummy node is removed, all of its children are also removed because the $innerText$ of a node is a superset of $innerTexts$ of all children by the definition of DOM [21]. A type **D2** node has only one child and contains no more information than its child, i.e., the node and its child have equal $innerText$. We remove this node and connect its child to its parent together. A tag $\langle TR \rangle$ that has only one $\langle TD \rangle$ tag is type **D2**. Some structures combined by **D1** and **D2** also

contain dummy nodes. In Figure 5 (c), when all D1 nodes are removed, some nodes that become to be matched the criterion of type D2 are removed. We apply the heuristic rules on IST recursively until the reduced IST is converged.

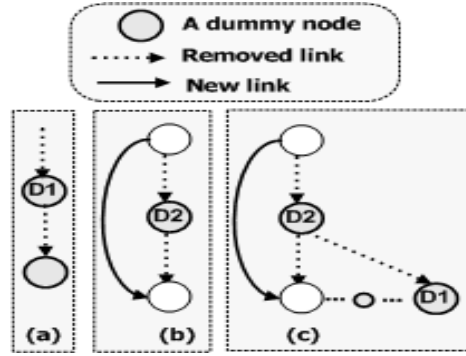


Figure 5: The types of dummy nodes.

After an IST has been reduced, we then calculate the children diversity. Children diversity evaluates the similarity between children. We measure the average distance between children and their parent in information space to represent the evaluation result. Children diversity (referred to as $CDiv$) of Node N is defined as:

$$CDiv(N) = \frac{1}{m} \sum_m \sqrt{(a_m - a_N)^2 + (h_m - h_N)^2},$$

where m is the number of children of N in the reduced IST . Note that $CDiv(N)$ is equal to zero when Node N is a leaf node in the reduced IST .

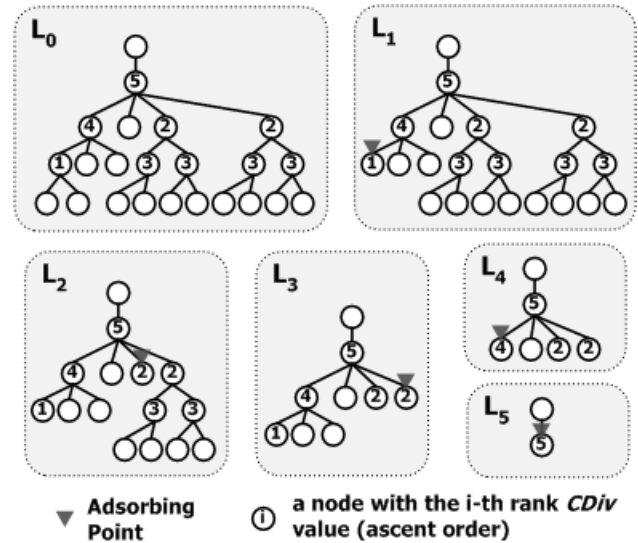


Figure 6: An example of information adsorption.

According to the values of $CDiv$, the proposed GTA method is applied to generate the set of different configurations of clusterings. In order to construct the hierarchical configurations of information of a Web page, GTA adsorbs

nodes in the sub-tree rooted by Node N with a minimum value of $CDiv$. The term $adsorb$ represents using the information in Node N to substitute for information in the more fine-grained sub-tree $T(N)$. Node N is then deemed an *Adsorbing Point* (AP). The i -th adsorbing point (AP_i) is the selected adsorbing point in Clustering L_i . We illustrate the process in Figure 6. There are six clusterings generated to form the information hierarchy of the sample page, and each of the clusterings represents the original page in a specific configuration.

3.3 Phase 3: Information Hierarchy Building

In this phase, we first compress the information hierarchy generated in Phase 2. We do this by merging non-overlapped successive clusterings into the highest-level clustering. Successive clusterings are non-overlapped when their adsorbing points do not have the same ancestor. Adsorbing points in merged clusterings are moved to the highest-level clustering among these non-overlapped clusterings. When the compression rule is applied on the information hierarchy in Figure 6, i.e., $\{L_0, L_1, L_2, L_3, L_4, L_5\}$, we can attain two compact information hierarchy matching the rule, i.e., $\{L_0, L_3, L_4, L_5\}$ and $\{L_0, L_1, L_4, L_5\}$.

We define the value of the *information gradient* between two clusterings L_i and L_j , i.e., $G_{info}(L_i, L_j)$, as

$$\frac{1}{p} \sum_{k=1-p} CDiv(AP_{jk}) - \frac{1}{q} \sum_{k=1-q} CDiv(AP_{ik}),$$

where AP_{jk} and AP_{ik} are adsorbing points and p and q are the numbers of adsorbing points in L_i and L_j , respectively, and j is larger than i . The function G_{info} is positive and transitive, i.e., $G_{info}(L_i, L_j) = G_{info}(L_i, L_k) + G_{info}(L_k, L_j)$, so that $\sum_{k=0-n-1} G_{info}(L_k, L_{k+1}) = G_{info}(L_n, L_0)$. According to the

characteristic of G_{info} function, we use the entropy value of all successive G_{info} values, i.e., ENG_{info} , to evaluate the goodness of an information hierarchy. Note that the larger its ENG_{info} value is, the better an information hierarchy is. This also means that the information gradient of a Web page is more equally distributed among information clusterings. Consider the two compact information hierarchies $H_1: \{L_0, L_3, L_4, L_5\}$ and $H_2: \{L_0, L_1, L_4, L_5\}$ discussed above, where the five $CDiv$ values are ranked 0.1, 0.2, 0.3, 0.4 and 0.6. Successive G_{info} values of H_1 are

$$\left\{ \frac{0.5}{3}, \frac{0.7}{3}, 0.2 \right\} \text{ and } \left\{ 0.1, \frac{0.5}{3}, \frac{1}{3} \right\} \text{ for } H_2, \text{ so that } ENG_{info}$$

(H_1)=0.992 and ENG_{info} (H_2)=0.893. We thus select H_1 as the compacted information hierarchy.

4. Results and evaluations

News Web sites are typical systematic Web sites. The structures of TOC and article pages are useful in evaluating the proposed method of mining the information hierarchy. We therefore conduct our experiments on Web pages in the

datasets¹ used in [13]. The datasets contain several real news Web sites as described in Table 1 and are composed of HTML documents. We illustrate the results of information space adsorption and compare the characteristics of the resulting hierarchies in Section 4.1. The evaluation of extracting informative information clusters is described in Section 4.2.

Table 1: Datasets for experiments and evaluations of information clusters

| Site Abbr. | URL | Total pages | TOC pages | Answer set [#] (page / block) | |
|------------|---------------------|-------------|-----------|--|---------|
| | | | | TOC | Article |
| CDN | www.cdn.com.tw | 261 | 25 | 22/38 | 60/63 |
| TIMES | news.chinatimes.com | 3747 | 79 | 69/313 | 66/68 |
| CNA | www.cna.com.tw | 1400 | 33 | 29/106 | 50/50 |
| CNET | taiwan.cnet.com | 4331 | 78 | 38/84 | 37/86 |
| CTS | www.cts.com.tw | 1316 | 31 | 19/21 | 53/80 |
| TVBS | www.tvbs.com.tw | 740 | 13 | 12/25 | 50/50 |
| TTV | www.ttv.com.tw | 861 | 22 | 18/20 | 42/75 |
| UDN | udnnews.com | 4676 | 252 | 243/674 | 52/106 |
| TOTAL | | 12035 | 530 | 450/1281 | 411/579 |

#: Domain experts selected the article pages with different and distinctive tagging styles to be the article answer set.

4.1 The results of information space adsorption

We apply *DOMISA* to pages with the marked types to build their information hierarchies. After building *ISTs*, the information space adsorption method is applied to generate different configurations in the information hierarchy of a page. The distribution of the number of clusterings is shown in Figure 7. We find that the numbers of clusterings of TOC pages are larger than those of article pages in six sites and the values of the standard deviation of TOC pages are also larger than those of article pages. This can be explained by the reason that the information gradients in TOC pages are more likely to vary gradually than those of article pages.

After inspecting the resultant information hierarchies of the TOC and article pages, we noticed several important characteristics. First, the information hierarchies of pages with the same structure are almost the same in view of the number of clusterings and information clusters in each clustering. This is more prominent in article pages. More than 25% of the article pages in the six Web sites of the dataset attain the same structure of information hierarchy, and so do 87.2% in CNA and 54.9% in CTIMES. Also, the reduction effect of the information hierarchy on the number

¹ Pages of Web sites in datasets are crawled at 2001/12/27 and 2002/4/11. The datasets can be retrieved in our research download site <http://kp06.iis.sinica.edu.tw/isd/index.html>.

of nodes and the tree depth is very noticeable. The average reduction ratios of nodes and the tree depth are 90.7% and 79.8% respectively as shown in Figure 8. This means that users only need to view 10% of the original nodes and select the desired information by 20% of the original depth of traversing in the DOM tree.

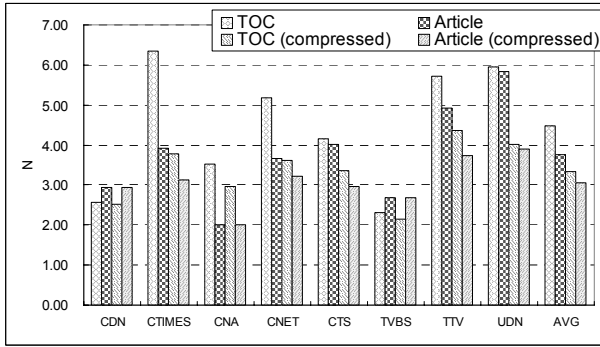


Figure 7: The average number of clusterings.

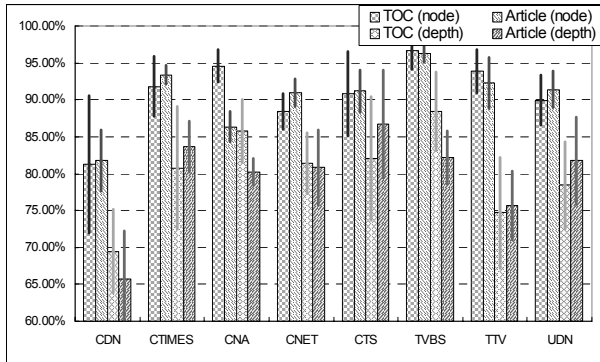


Figure 8: The ratio of the node and depth reduction of Web pages of *DOMISA*.

4.2 Evaluation of informative information clusters

We use TOC blocks and article blocks, to evaluate the precision and recall rates of recognizing informative clusters. These blocks in the answer sets are extracted manually by news domain experts² according to their experience in issuing real-world newspapers. We select most TOC pages and some representative pages among all article pages with different tagging structures to mark. We apply a maximum information searching on the information hierarchy. That is, we traverse the information hierarchy and find a node which meets the pre-assigned threshold of the information scale. We then use two evaluation methods, i.e., significant node coverage (*SNC*) and information coverage (*IC*) to evaluate the precision and recall rates of TOC and article pages, respectively. Explicitly, *SNC*

² News domain experts are researchers at Department of Journalism, National Chengchi University, Taiwan.

evaluates the precision (*P*) and recall (*R*) rates by matching anchor nodes and *IC* matches the string of *innerText*. They are defined as:

$$P_{SNC} = \frac{\text{Number}(\text{Anchor}_{DOMISA} \cap \text{Anchor}_{Answer})}{\text{Number}(\text{Anchor}_{DOMISA})}$$

$$R_{SNC} = \frac{\text{Number}(\text{Anchor}_{DOMISA} \cap \text{Anchor}_{Answer})}{\text{Number}(\text{Anchor}_{Answer})}$$

$$P_{IC} = \frac{\text{Length}(\text{InnerText}_{DOMISA} \cap \text{InnerText}_{Answer})}{\text{Length}(\text{InnerText}_{DOMISA})}$$

$$R_{IC} = \frac{\text{Length}(\text{InnerText}_{DOMISA} \cap \text{InnerText}_{Answer})}{\text{Length}(\text{InnerText}_{Answer})}$$

We also use the F-measure which is the harmonic mean of values of precision and recall and is formulated as $\frac{2 * (R * P)}{R + P}$ to evaluate results in a single efficiency measure.

Figure 9 shows that the information hierarchy built by *DOMISA* is very useful for the article blocks mining of all datasets as well as for TOC blocks of CDN, CTIMES, CNA, CTS, and TVBS. The low values of F-measure on CNET, TTV and UDN are caused by the low accuracy of API values due to the missing of target pages.

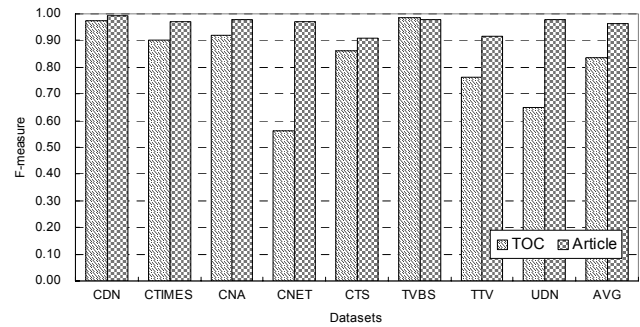


Figure 9: The F-measure of TOC and articles blocks recognition in the information hierarchy.

5. Conclusion

In this paper, we develop a *DOM-based Information Space Adsorption (DOMISA)* system that automatically builds an information hierarchy of Web site pages according to both page information and structure. The mining flow of *DOMISA* consists of three main phases: (1) the *Information Space Tree* building, (2) the *Gradient Tree Adsorption*, and (3) the information hierarchy building. The main mining flow in *DOMISA* uses information theory to evaluate information characteristics and scales of content and sub-structures, and then to construct the information hierarchy by applying our information adsorption and compression method. The attained information hierarchy is useful for search engines, inter-media information agents, and crawlers to index, extract and navigate significant information in a Web site, and for providing the

hierarchical configurations of a page according to the amount of information contained. Results show that *DOMISA* effectively extracts information hierarchies. Experiments on several real news Web sites show the high precision and recall rates of *DOMISA* on finding informative information clusters of pages and also validate its practical applicability to real Web sites. We also develop a simple annotation method in *DOMISA* to utilize the information hierarchy and to provide a novel and effective surfing interface for small display devices.

ACKNOWLEDGEMENT

The authors are supported in part by the Ministry of Education Project No.89-E-FA06-2-4, and the National Science Council Project No. NSC 91-2213-E-002-034 and NSC 91-2213-E-002-045, Taiwan, Republic of China.

References

- [1] B. Adelberg. NoDoSE—a tool for semi-automatically extracting structured and semistructured data from text documents. Proc. of the 1998 ACM SIGMOD international conference on Management of data, 1998.
- [2] T. Asai, K. Abe, S. Kawasoe, H. Arimura, H. Sakamoto and S. Arikawa. Efficient Substructure Discovery from Large Semi-structured Data. SIAM SDM 2002.
- [3] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999.
- [4] O. Buyukkokten, H. Garcia-Molina and A. Paepcke, Accordion Summarization for End-Game Browsing on PDAs and Cellular Phones, In Proc. of the Conf. on Human Factors in Computing Systems, CHI'01, 2001.
- [5] O. Buyukkokten, H. Garcia-Molina and A. Paepcke, Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices, Proc. of 10th World Wide Web Conference, 2001.
- [6] Y. Chen, W.-Y. Ma, H.-J. Zhang, Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices, Proc. of 12th World Wide Web Conf., 2003.
- [7] W. Cohen. Recognizing Structure in Web Pages using Similarity Queries. AAAI 1999.
- [8] G. Cong, L. Yi, B. Liu and K. Wang. Discovering Frequent Substructures from Hierarchical Semi-structured Data. SIAM SDM 2002.
- [9] K. Furukawa, T. Uchida, K. Yamada, T. Miyahara, T. Shoudai and Y. Nakamura. Extracting Characteristic Structures among Words in Semistructured Documents. Proc. of the Sixth PAKDD, 2002.
- [10] S. Gupta, G. Kaiser, D. Neistadt, P. Grimm, DOM-based Content Extraction of HTML Documents, Proc. of 12th World Wide Web Conference, 2003.
- [11] M. A. Hearst, Subtopic Structuring for Full-Length Document Access, In Proc. of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 56-68, 1993.
- [12] C. N. Hsu and M. T. Dung. Generating Finite-state Transducers for Semi-structured Data Extraction from the Web. Information Systems, 23(8):521-538, 1998.
- [13] H.-Y. Kao, S.-H. Lin, J.-M. Ho and M.-S. Chen. Entropy-Based Link Analysis for Mining Web Informative Structures. Proc. of the ACM 11th International Conf. on Information and Knowledge Management (CIKM-02), Nov. 4-9, 2002.
- [14] J. M. Kleinberg, Authoritative sources in a hyperlinked environment. ACM-SIAM Symposium on Discrete Algorithms. 1998.
- [15] N. Kushmerick, D. Weld, and R. Doorenbos, Wrapper Induction for Information Extraction, In Proc. of the 15th Intern'l Joint Conf. on Artificial Intelligence (IJCAI), 1997.
- [16] W. S. Li, N. F. Ayan, O. Kolak and Q. Vu, Constructing Multi-Granular and Topic-Focused Web Site Maps, Proc. of the 10th World Wide Web Conference, 2001.
- [17] X. Li, B. Liu, T. H. Phang and M. Hu, "Using Micro Information Units for Internet Search", Proc. of the ACM 11th International Conf. on Information and Knowledge Management (CIKM-02), 2002.
- [18] S.-H. Lin and J.-M. Ho. Discovering Informative Content Blocks from Web Documents. The 8th ACM SIGKDD, 2002.
- [19] N. Milic-Frayling, and R. Sommerer, SmartView: Flexible Viewing of Web Page Contents, Poster paper at the 11th World Wide Web Conference, 2002.
- [20] C. E. Shannon, A mathematical theory of communication. Bell System Technical Journal, 27:398-403, 1948.
- [21] W3C DOM. Document Object Model (DOM). <http://www.w3.org/DOM/>.
- [22] K. Wang and H. Liu. Discovering Structural Association of Semistructured Data. IEEE Transaction on Knowledge and Engineering, VOL. 12, NO. 3, MAY/JUNE 2000.
- [23] S. Yu, D. Cai, J.-R. Wen, W.-Y. Ma, Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation, Proc. of 12th World Wide Web Conference, 2003.