Utilizing Hyperlink Transitivity to Improve Web Page Clustering

Jingyu Hou
Department of Mathematics and Computing
University of Southern Queensland
Toowoomba, Qld 4350, Australia
jingyu@usq.edu.au

Yanchun Zhang
Department of mathematics and Computing
University of Southern Queensland
Toowoomba, Qld 4350, Australia
zhang@usq.edu.au

Abstract

The rapid increase of web complexity and size makes web searched results far from satisfaction in many cases due to a huge amount of information returned by search engines. How to find intrinsic relationships among the web pages at a higher level to implement efficient web searched information management and retrieval is becoming a challenge problem. In this paper, we propose an approach to measure web page similarity. This approach takes hyperlink transitivity and page importance into consideration. From this new similarity measurement, an effective hierarchical web page clustering algorithm is proposed. The primary evaluations show the effectiveness of the new similarity measurement and the improvement of web page clustering. The proposed page similarity, as well as the matrix-based hyperlink analysis methods, could be applied to other web-based research areas.

Keywords: World Wide Web, hyperlink analysis, web page similarity, web clustering

1 Introduction

Current web search mainly depends on search engines, such as *Yahoo!*, *AltaVista* and *Google*. In most cases, however, the returned search results are also a large information source from which it is still difficult for users to find required information. Nowadays, with more and more search engine providers claiming that their products are powerful in searching the web (e.g. *Google* covers 2,073,418,204 web pages), how to effectively and efficiently manage and retrieve web-searched information is becoming a challenge problem.

One way to solve this problem is to narrow the concerned information space by finding structures or similarities among the searched pages and constructing a new web information space or community, such as the work in (Kleinberg 1998, Chakrabarti, Dom, Gibson, Kleinberg, Raghavan and Rajagopalan 1998, Bharat and Henzinger 1998, Dean and Henzinger 1999). Another way is to re-organize or cluster the web pages, for instance the work in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996, Wen, Liu, Wen and Zheng 2001, Zamir and Etzioni 1998, Pitkow and Pirolli 1997). Web page clustering makes it possible to use conventional database management techniques to establish index on the web pages, and implement efficient information classification, navigation, storage, retrieval and integration.

The key for implementing effective web page clustering is to find intrinsic relationships, especially similarities, among the pages. For this purpose, web page content, hyperlinks and usage data (server log files) could be utilized. Among them, hyperlink analysis has its own advantages, as hyperlinks convey semantics between web pages in most cases. In fact, with a few exceptions, the authors of web pages create links to other pages usually with an idea in mind that the linked pages are relevant to the linking pages. If a hyperlink is reasonable, it reflects human semantic judgement and this judgement is objective and independent of the synonymy and polysemy of the words in the pages. This latent semantics, once being revealed, could be utilized to find higher-level relationships among the pages. The hyperlink analysis has proven success in many web-related areas, such as page ranking in the search engine *Google* (Brin and Page 1998a, 1998b), web page community construction (Kleinberg 1998, Bharat and Henzinger 1998, Chakrabarti, Dom, Gibson, Kleinberg, Raghavan and Rajagopalan 1998, Hou and Zhang 2002a, Hou, Zhang and Cao 2002) and relevant page finding (Kleinberg 1998, Dean and Henzinger 1999).

One example of directly using hyperlink to cluster web pages can be found in (Pitkow and Pirolli 1997). Its one-level clustering algorithm was based on web page co-citation analysis via hyperlinks. No page similarity was defined for this algorithm. Other examples of using hyperlink analysis, or combining hyperlink and content analyses, to hierarchically cluster web pages can be found in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996, Wang and Kitsuregawa 2001, Pitkow and Pirolli 1997, Marchiori 1997, PirollI, Pitkow and Rao 1996). Most of the work only utilized hyperlinks at the first level, i.e. the hyperlink analysis only focused on direct links between pages.

However, the hyperlinks between web pages usually are transitive. In other words, even though there is no direct link between two pages, they may also have certain indirect semantic relevance via other pages. The page similarity measurement in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996) incorporated hyperlink transitivity, but it defined the similarity directly from hyperlinks with an over-simplified assumption. Marchiori (1997) used hyperlink transitivity to measure page content similarity between a page and the query, in which, however, only out-hyperlinks of pages were considered and no page similarity was directly defined from hyperlinks.

Copyright ©2003, Australian Computer Society, Inc. This paper appeared at the *Fourteenth Australasian Database Conference* (ADC2003), Adelaide, Australia. Conferences in Research and Practice in Information Technology, Vol. 17. Xiaofang Zhou and Klaus-Dieter Schewe, Ed. Reproduction for academic, not-for profit purposes permitted provided this text is included.

On the other hand, the role each page plays in page similarity measurement is different. If a page *within* a certain web page space is dense (i.e. it has higher in-degree or out-degree), its opinion has more impact to other pages and it will play more important role in page similarity measurement within this page space. The authority and hub pages in a web page community (Kleinberg 1998) are the examples. Up to now, however, there is no such web page similarity measurement that incorporates page importance.

In this paper, we propose a hyperlink-based approach to measure web page similarity. It incorporates hyperlink transitivity and page importance. The page similarity is derived from page relevance, rather than direct hyperlinks. This similarity more precisely reflects mutual relationships among the web pages and the nature of the web. With this new similarity measurement, an effective hierarchical web page clustering algorithm is proposed to improve web page clustering.

This paper is organized as follow. The following section 2 gives the new hyperlink-based web page similarity measurement. The hierarchical clustering algorithm based on this new similarity is proposed in section 3. Some primary evaluations of the proposed algorithm are given in section 4. In section 5, some related work and discussions are presented. Finally, section 6 gives the conclusions for this work and further research directions.

2 Web Page Similarity Measurement

A web page similarity usually refers to a certain page space. Since we are concerned about clustering web searched results in this work, we focus on a page space that is related to the user's query topics. In this section, we firstly establish such a page source (space), then, within this page source, we incorporate hyperlink transitivity and page importance to propose a new page similarity measurement.

2.1 Page Source Construction

The page source construction is based on the web searched results with respect to the user’s queries. For users, they are usually concerned about a part of searched results, say the first *r* highest-ranked pages returned by the search engine. From the hyperlink analysis point of view, the pages that link to or are linked to these *r* highest-ranked pages are also related to the query topics to some extent (Kleinberg 1998). Therefore, the page source *S* with respect to the user's query topics is constructed as follow:

Step 1: Select *r* highest-ranked pages from the searched results to form a root page set *R*.

Step 2: For each page *p* in *R*, select up to *B* pages, which point to *p* and whose domain names[1] are different from that of *p*, and add them to the back vicinity set *BV* of *R*.

Step 3: For each page *p* in *R*, select up to *F* pages, which are pointed to by *p* and whose domain names are different from that of *p*, and add them to the forward vicinity set *FV* of *R*.

Step 4: Page source *S* is constructed by uniting sets *R*, *BV*, *FV* and adding original links between pages in *S*.

In the above algorithm, parameters *B* and *F* are used to guarantee that the page source *S* is of a reasonable size. For example, we choose value 200 for *B* and *F* from our experimental experience. When constructing sets *BV* and *FV*, it is required that for each page *p* in *R*, the domain names of its parent pages and child pages are different from the domain name of the page *p*. This requirement filters those parent and child pages coming from the same domain where the page *p* is located in, because, as indicated in (Bharat and Henzinger 1998), the links within the same domain are more likely to reveal the inner structure than to imply a certain semantic relationship.

When a page source is constructed, it is possible to bring some mirror pages into the page source. There are several reasons for not being required to remove these mirror pages. Firstly, there is no standard at present to identify whether two pages are mirror pages just by analysing their hyperlinks, and identifying mirror pages will add extra computing cost. Secondly, if two pages are mirror pages, they would have the same hyperlink structure and are most likely to be clustered into one cluster, in which the user or an algorithm can identify them easily. Therefore, keeping a proper mirror page redundancy in the page source *S* is reasonable.

2.2 Page Weight Definition

The role each page plays in similarity measurement is different in a concerned page source *S*. For instance, two kinds of pages need to be noticed. The first one is the page whose *out-link contribution* to *S* (i.e. the number of pages in *S* that are pointed to by this page) is greater than the average out-link contribution of all the pages in the page source *S*. Another kind is the page whose *in-link contribution* to *S* (i.e. the number of pages in *S* that point to this page) is greater than the average in-link contribution of all the pages in the page source *S*. The pages of the first kind are called *index* pages in (Botafogo and Shneiderman 1991) or *hub* pages in (Kleinberg 1998), and those of the second kind are called *reference* pages in (Botafogo and Shneiderman 1991) or *authority* pages in (Kleinberg 1998). These pages are most likely to reflect certain topics related to the query within the concerned page source. If two pages are linked by or link to some pages of these kinds, these two pages are more likely to be located in the same topic group and have higher similarity.

It also needs to be noticed that index web pages in common sense, such as personal bookmark pages and index pages on some special-purpose web sites, might not be the index pages in the concerned page source *S* if their out-link contribution to *S* is below the average out-link contribution in *S*. Similarly, some pages with high in-degrees on the web, such as home pages of commonly used search engines, might not be the reference pages in the concerned page source *S*. For simplicity, we filter the

[1] Page domain name means the first level in the URL string associated with a page.

home pages of commonly used search engines (e.g. *Yahoo!*, *AltaVista*, *Google* and *Excite*) from the concerned page source *S*, since these pages are not related to any specific topics. To measure the importance of each page *within the concerned page source*, we define a weight for each page.

For each page P_i in the page source *S*, we associate a non-negative *in-weight* $P_{i,in}$ and a non-negative *out-weight* $P_{i,out}$ with it. Considering the hyperlink transitivity in the page source, the *in-weight* and *out-weight* for the page P_i in *S* are iteratively calculated as follow (Kleinberg 1998):

$$P_{i,in} = \sum_{P_j \in S, P_j \to P_i} P_{j,out} \ ,$$

$$P_{i,out} = \sum_{P_j \in S, P_i \to P_j} P_{j,in} \ .$$

The in-weight and out-weight vectors are normalized after each iteration.

We denote the average in-weight of *S* as μ, and the average out-degree of *S* as λ. That is

$$\mu = \sum_{P_i \in S} P_{i,in} \ / \ size(S) \ , \ \lambda = \sum_{P_i \in S} P_{i,out} \ / \ size(S) \ ,$$

where *size*(*S*) is the number of pages in *S*. Then the page weight for P_i is defined as

$$w_i = 1 + \max((P_{i,in} - \mu)/(M_{in} - m_{in}), (P_{i,out} - \lambda)/(M_{out} - m_{out})) \qquad (1)$$

where M_{in} , m_{in} , M_{out} and m_{out} are defined as follow:

$$M_{in} = \max_{P_j \in S}(P_{j,in}) \ , \quad m_{in} = \min_{P_j \in S}(P_{j,in}) \ ,$$

$$M_{out} = \max_{P_j \in S}(P_{j,out}) \ , \ m_{out} = \min_{P_j \in S}(P_{j,out}) \ .$$

The page weight definition in (1) indicates that if a page's in-weight and out-weight in *S* are below their corresponding average values μ and λ, its weight will be less than 1, which means its influence to the similarity measurement is relatively less. Similarly, if a page's in-weight or out-weight in *S* is above the average value (e.g. an index page or a reference page), its weight will be greater than 1 and its influence to the similarity measurement is relatively greater. In other words, the page weight defined in (1) reflects the importance of each page's role in the concerned page source.

2.3 Page Correlation Matrix

For each web page, its correlation with other pages, via linkages, is expressed in two ways: one is out-links from it, another is in-links to it. In this work, the similarity between two pages is measured by their own correlations with other pages in the page source *S*, rather than being derived directly from the links between them. For measuring the page correlation, we firstly give the following definitions.

Definition 1. If page *A* has a direct link to page *B*, then the *length of path* from page *A* to page *B* is 1, denoted as *l*(*A*,*B*) = 1. If page *A* has a link to page *B* via *n* other pages, then *l*(*A*,*B*) = *n*+1. The *distance* from page *A* to page *B*, denoted as *sl*(*A*,*B*), is the shortest path length from *A* to *B*, i.e. *sl*(*A*,*B*) = *min*(*l*(*A*,*B*)). The length of path from a page to itself is zero, i.e. *l*(*A*,*A*) = 0. If there are no links from page *A* to page *B* (direct or indirect), then *l*(*A*,*B*) = ∞.

It can be inferred from this definition that *l*(*A*,*B*) = ∞ does not imply *l*(*B*,*A*) = ∞, because there might still exist links from page *B* to page *A* in this case.

Definition 2. The *correlation weight* between two pages *i* and *j* ($i \neq j$), denoted as $w_{i,j}$, is the maximal weight of their weights, i.e. $w_{i,j} = max(w_i, w_j)$ where w_i and w_j are the page weights for pages *i* and *j* respectively. If *i* = *j*, $w_{i,j}$ is defined as 1.

The following definition defines how much two pages correlate with each other if there exists a direct link between them.

Definition 3. *Correlation factor*, denoted as *F*, 0<*F*<1, is a constant that measures the correlation rate between two page with direct link, i.e. if page *A* has a direct link to page *B*, then the correlation rate from page *A* to page *B* is *F*.

Similar to the work in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996), the value of *F* in this paper is chosen as 1/2. For general purpose, we still use *F* in the following algorithm to represent this correlation factor.

With the above definitions, a correlation degree between any two pages can be defined. This correlation degree depends on the value of correlation factor *F*, the distance between the two pages (the farther the distance, the less the correlation degree), and the correlation weights of involved pages along the shortest path. The following definition gives this function.

Definition 4. The *correlation degree* from page *i* to page *j*, denoted as c_{ij}, is defined as

$$c_{ij} = w_{i,k1} w_{k1,k2} \cdots w_{kn,j} F^{sl(i,j)} \ , \qquad (2)$$

where *F* is the correlation factor, *sl*(*i*,*j*) is the distance from page *i* to page *j*, and $w_{i,k1}$, $w_{k1,k2}$, …, $w_{kn,j}$ are correlation weights of the pages *i*, *k1*, *k2* , …, *kn*, *j* that form the distance *sl*(*i*,*j*), i.e. $i \to k1 \to k2 \to \ldots \to kn \to j$. If *i* = *j*, then c_{ij} is defined as 1.

For the concerned page source *S*, we suppose the size of the root set *R* is *m*, the size of the vicinity set $V = BV \cup FV$ is *n*. Then the correlation degrees of all the pages in *S* can be expressed in a (*m*+*n*)×(*m*+*n*) matrix $C = (c_{ij})_{(m+n)\times(m+n)}$, called *correlation matrix*. This correlation matrix *C* is a numerical format that converts the hyperlinks (direct or indirect) between pages in *S* into the correlation degrees, incorporating the hyperlink transitivity and page importance.

The distance *sl*(*i*,*j*) in (2) can be computed via some operations on the matrix elements of a special matrix called *primary correlation matrix* $A = (a_{ij})_{(m+n)\times(m+n)}$, which is constructed as follow

$$a_{ij} = \begin{cases} F & \text{if there is a direct link from } i \text{ to } j, i \neq j \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

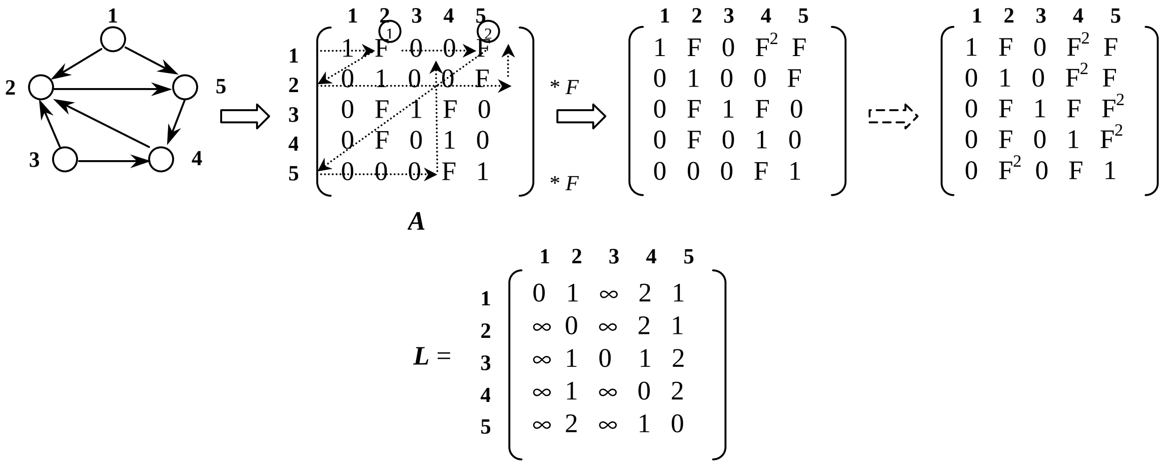

Figure 1. Example of Computing Distance Between Pages

Based on this primary correlation matrix, the algorithm for computing the distance $sl(i,j)$ between any two pages i and j is described as follows:

Step 1: For each page $i \in S$, choose *factor* = F and go to step 2;

Step 2: For each element a_{ij} , if a_{ij} = *factor*, then set k = 1 and go to step 3. If there is no element a_{ij} (j = 1, …, $m+n$) such that a_{ij} = *factor*, then go back to step 1;

Step 3: If $a_{jk} \neq 0$ and $a_{jk} \neq 1$, calculate *factor**a_{jk} ;

Step 4: If *factor**a_{jk} > a_{ik}, then replace a_{ik} with *factor**a_{jk}, change $k = k+1$ and go back to step 3. Otherwise, change $k = k+1$ and go back to step 3;

Step 5: Change *factor* = *factor**F and go to step 2 until there are no changes to all element values a_{ij} ;

Step 6: Go back to step 1 until all the pages in S have been considered.

After element values of matrix A are updated by the above algorithm, the distance from page i to page j is

$$sl(i,\, j) = [\log a_{ij} \,/ \log F]\,.$$

The example in figure 1 gives an intuitive execution demonstration of the above algorithm and the final distance matrix L.

2.4 Page Similarity

In this work, we focus on clustering web-searched pages in the root set R with a new page similarity measurement. For simplicity and better understanding of the new similarity, we divide the correlation matrix C into four blocks (sub-matrices) as follow:

$$C = (c_{ij})_{(m+n)\times(m+n)} = \begin{array}{c|c:c|} & R & V \\ R & \textcircled{1} & \textcircled{2} \\ \hdashline V & \textcircled{3} & \textcircled{4} \end{array}_{(m+n)\times(m+n)}$$

The elements in sub-matrix 1 represent the correlation relationships between the pages in R. Similarly, the elements in sub-matrices 2 and 3 represent the correlation relationships between the pages in R and V, and sub-matrix 4 gives the correlation relationships between the pages in V. It can be seen that the correlation degrees related to the pages in R are located in three sub-matrices 1, 2 and 3. Therefore, the similarity measurement for the pages in R only refers to the elements in these three sub-matrices.

Note: If the similarity between any two pages in the whole source space S is to be measured, the whole correlation matrix C will be used and the similarity definition is the same as follows.

In the correlation matrix C, the row vector that corresponds to each page i in R is in the form of

$$row_i = (c_{i,1}, c_{i,2}, ..., c_{i,m+n}), \quad i = 1,2,..., m\,.$$

From the construction of matrix C, it is known that row_i represents *out-link* relationship of page i in R with all the pages in S, and element values in this row vector indicate the correlation degrees of this page to the linked pages. Similarly, the column vector that is in the form of

$$col_i = (c_{1,i}, c_{2,i}, ..., c_{m+n,i}), \quad i = 1,2,..., m\,,$$

represents *in-link* relationship of page i in R with all the pages in S, and its element values indicate the correlation degrees from the pages in S to page i.

Each page i in R, therefore, is represented as two correlation vectors: row_i and col_i. For any two pages i and j in R, their *out-link similarity* is defined as

$$sim_{i,j}^{out} = \frac{(row_i, row_j)}{\| row_i \| \cdot \| row_j \|}\,,$$

where

$$(row_i, row_j) = \sum_{k=1}^{m+n} c_{i,k} c_{j,k}\,, \qquad \| row_i \| = (\sum_{k=1}^{m+n} c_{i,k}^2)^{1/2}\,.$$

Similarly, their *in-link similarity* is defined as

$$sim_{i,j}^{in} = \frac{(col_i, col_j)}{\| col_i \| \cdot \| col_j \|}\,.$$

Then the similarity between any two pages i and j in R is defined as

$$sim(i,\, j) = \alpha_{ij} \cdot sim_{i,j}^{out} + \beta_{ij} \cdot sim_{i.j}^{in}\,, \tag{3}$$

where α_{ij} and β_{ij} are the weights for out-link and in-link similarities respectively.

The similarity weights α_{ij} and β_{ij} are determined dynamically as:

$$\alpha_{ij} = \frac{\| row_i \| + \| row_j \|}{MOD_{ij}}, \quad \beta_{ij} = \frac{\| col_i \| + \| col_j \|}{MOD_{ij}},$$

where $MOD_{ij} = \| row_i \| + \| row_j \| + \| col_i \| + \| col_j \|$.

3 Hierarchical Web Page Clustering

With the page similarity measurement (3) and the correlation matrix *C*, a hierarchical web page clustering algorithm could be established. This hierarchical clustering algorithm consists of two phases. The first one is single layer clustering, in which the pages in *R* are clustered at the same level without hierarchy. The second phrase is hierarchical clustering, in which the pages in the clusters produced by the first phase are clustered further to form a cluster hierarchical structure. Figure 2 gives this hierarchical clustering diagram.

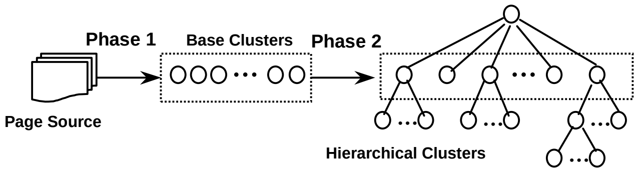

Figure 2. Hierarchical Clustering Diagram

The details of the hierarchical clustering algorithm are described as follows.

Phase 1: Single Layer Clustering

[**Input**]: A set of web pages $R = \{p_1, p_2, \ldots, p_m\}$, clustering threshold *T*.

[**Output**]: A set of clusters $CL = \{CL_i\}$.

[**Algorithm**]: *BaseCluster*(*R*, *T*)

Step 1. Select the first page p_1 as the initial cluster CL_1 and the centroid of this cluster, i.e. $CL_1 = \{p_1\}$ and $CE_1 = p_1$.

Step 2: For each page $p_i \in R$, calculate the similarity between p_i and the centroid of each existing cluster $sim(p_i, CE_j)$.

Step 3: If $sim(p_i, CE_k) = \max_j(sim(p_i, CE_j)) > T$, then add p_i to the cluster CL_k and recalculate the centroid CE_k of this cluster that consists of two vectors

$$CE_k^{row} = \frac{1}{|CL_k|} \sum_{j \in CL_k} row_j,$$

$$CE_k^{col} = \frac{1}{|CL_k|} \sum_{j \in CL_k} col_j,$$

where $|CL_k|$ is the number of pages in CL_k.

Otherwise, p_i itself initiates a new cluster and is the centroid of this new cluster.

Step 4: If there are still pages to be clustered (i.e. pages that have not been clustered or a page that itself is a cluster), go back to step 2 until all cluster centroids no longer change.

Step 5: Return clusters $CL = \{CL_i\}$.

The above phase 1 of the clustering algorithm produces a set of single layer clusters called *base clusters*. Recursively applying the above algorithm, with increasing clustering threshold *T*, to each base cluster would produce downward hierarchical clusters, as well as the whole hierarchical cluster structure. The procedure is described as the phase 2 of the clustering algorithm.

Phase 2: Hierarchical Clustering

[**Input**]: A set of base clusters $CL = \{CL_i\}$, parameter *NP* and clustering threshold *T* in phase 1.

[**Output**]: Hierarchical clusters $HCL = \{HCL_i\}$.

[**Algorithm**]: *HierarchyCluster*(*CL*, *NP*, *T*)

Step 1: Set $HCL = CL$, and let *CL* to be the set of clusters at layer 1 (base layer), i.e. $CL^1 = \{CL_i^1\} = \{CL_i\}$. Assign $l = 1$ and $T' = T$.

Step 2: Recursively increase T', *l* and call algorithm *BaseCluster*(CL_i^l, T') for those clusters CL_i^l in CL^l that contain more than *NP* pages. Add the clusters at each layer to *HCL*.

Step 3: Return the produced set of hierarchical clusters *HCL*.

The clustering threshold *T* in the algorithm should be chosen such that the pages are clustered into a reasonable number of clusters. For example, *T* could be chosen as the average page similarity of all the pages in *R*. The increase rate for the hierarchical clustering threshold T' could be chosen as a certain percentage of the threshold *T*. The parameter *NP* (e.g. 10) is used to control the number of downward levels of the hierarchical cluster structure. If the number of pages in a cluster ≤ *NP*, this cluster should not be divided into some smaller clusters (at a lower level) any more.

It can be inferred from the phase 1 of the algorithm that a page in *R* only belongs to a cluster. In practice, a page might belong to multiple clusters. This requirement can be easily met by only changing the clustering condition in the step 3 of the phase 1, i.e. changing the condition " If $sim(p_i, CE_k) = \max_j(sim(p_i, CE_j)) > T$ " to " If $sim(p_i, CE_k) > T$ ". For computation simplicity, we still assume that a page only belongs to a cluster.

As stated in (Wen, Liu, Wen and Zheng 2001), for this kind of hierarchical clustering algorithm, it has been proven (Wang 1997) that the algorithm is independent of the order in which the pages are presented to the algorithm if the pages are properly normalized. Since the page normalization is guaranteed in the similarity measurement (3), the above hierarchical clustering algorithm is independent of the page order. It is not difficult to prove that the complexity of this algorithm is *O* (*M*N*logN*), where *M* is the number of generated clusters and *N* is the number of pages to be clustered.

4 Evaluations

Primary clustering experiments were conducted on a real web page source. The page source was for the search topic "*Jaguar*". The search engine we used was *Google*. The number of pages in the root page set was 472, the total number of pages in the page source was 3,540, and the number of hyperlinks in the page source was 17,793.

We named the hierarchical clustering algorithm with static similarity weights, i.e. $(\alpha_{ij}, \beta_{ij}) = (1/2, 1/2)$ in (3), as *HCA*(*S*), and that with dynamic similarity weights as *HCA*(*D*). For comparison, we also implemented the clustering algorithm in (Wang and Kitsuregawa 2001) which was purely based on the hyperlink analysis but did not consider the hyperlink transitivity and page importance. It was declared in (Wang and Kitsuregawa 2001) that their algorithm was better than the Suffix Tree Clustering (STC) algorithm in (Zamir and Etzioni 1998), which was based on the snippets attached with web pages. Since the clustering algorithm in (Wang and Kitsuregawa 2001) was non-hierarchical, we extended this algorithm as a hierarchical algorithm by recursively applying it to each non-hierarchical cluster. Accordingly, we called this extended hierarchical algorithm *WK01A*. All the above algorithms were implemented in Java.

It is a difficult task to measure the effectiveness of a hierarchical clustering algorithm. In this work, we adapt the precision concept in information retrieval (Baeza-Yates and Ribeiro-Neto 1999) and modify it as a notation of clustering accuracy to measure the clustering algorithm effectiveness. Given a page source, we denote its real clusters as the set $\{RC_i\}$ and its experimental clusters as the set $\{EC_j\}$. For an experimental cluster EC_j, its accuracy is defined as

$$Accuracy(EC_j) = \frac{\max_i(|EC_j \cap RC_i|)}{|EC_j|},$$

where $|EC_j|$ is the number of pages in cluster EC_j. For a single-page cluster, its accuracy is defined as 0.

In our primary evaluation, we manually checked each web page to be clustered and gave the (real) clusters according to our judgement. This method might lead to bias in the evaluation though we tried our best to objectively classify the web pages, but it was reasonable to use it as a relative standard for algorithm comparison at this stage. The further user experiment will be conducted in our plan for the future.

In the hierarchical cluster structures produced separately by the above *HCA*(*D*), *HCA*(*S*) and *WK01A* algorithms, three kinds of accuracy comparison were conducted. The first one was average *base* cluster accuracy comparison, the second was average *leaf* cluster accuracy comparison, and the third one was *overall* average cluster accuracy comparison. The results of these three kinds of comparison with different clustering threshold (T) values are shown in figures 3, 4 and 5 separately.

Note: Theoretically, with the increase of clustering similarity threshold, the clustering accuracy should increase accordingly. In this experiment, when the clustering similarity threshold increases, the number of single-page clusters also increases. Since the clustering accuracy definition in this work does not consider single-page clusters, the experimental results here do not follow this accuracy change trend.

It is shown from these results that the algorithm with dynamic similarity weights α_{ij} , β_{ij} , i.e. *HCA*(*D*), usually performs better than that with static similarity weights *HCA*(*S*). In general, the algorithms *HCA*(*D*) and *HCA*(*S*), which adopt the new page similarity, have higher cluster accuracy than the algorithm *WK01A* for all three kinds of comparison. The above evaluation results indicate the effectiveness of the new page similarity and the corresponding hierarchical clustering algorithm in web page clustering improvement.

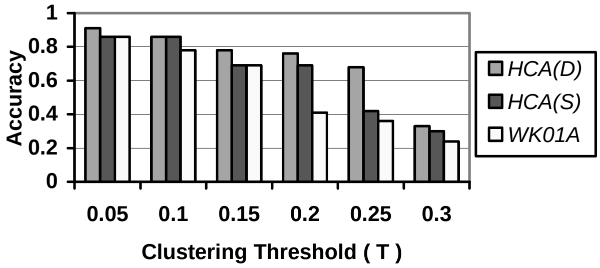

Figure 3. The Average *Base* Cluster Accuracy with Different Clustering Thresholds (T)

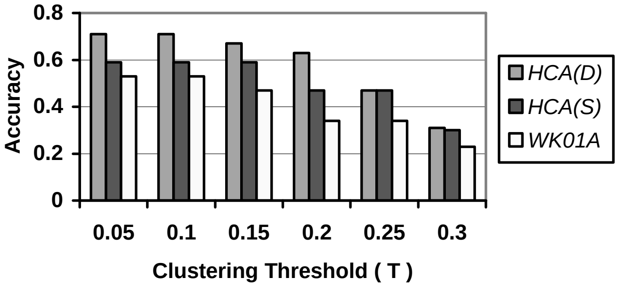

Figure 4. The Average *Leaf* Cluster Accuracy with Different Clustering Thresholds (T)

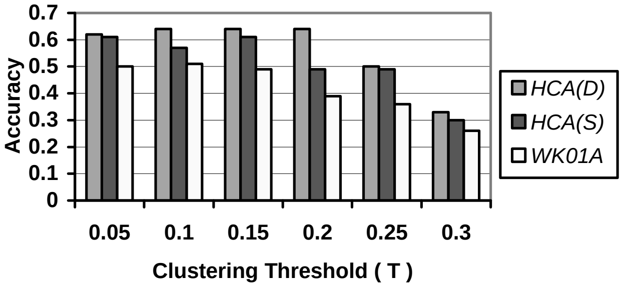

Figure 5. The O*verall* Average Cluster Accuracy with Different Clustering Thresholds (T)

Topic: **Jaguar Game**	
atarijaguardirectory.com	// Atari Jaguar Directory
www.atarihq.com/interactive	// Jaguar Interactive II
www.atari.org	// The Definitive Atari Resource
Topic: **Jaguar Big Cat**	
dspace.dial.pipex.com/agarman/jaguar.htm	//Jaguar
www.animalsoftherainforest.com/jaguar.htm	//Jaguar
www.bluelion.org/jaguar.htm	// Jaguar
Topic: **Jaguar Reef Touring**	
www.jaguarreef.com	// Jaguar Reef Lodge
www.divejaguarreef.com	// Dive Jaguar Reef Lodge
www.belizenet.com/jagreef.html	// Jaguar Reef

Table 1. Examples of Some Major Clusters

Topic: **Jaguar Car and Club**	
www.jaguar.com	// Jaguar Cars Home Page
www.classicjaguar.com	// Classic Jaguar
www.jaguarvehicles.com	// Jaguar Cars Home Page
www.jagweb.com	// A1 JagWeb - Jaguar…
www.jag-lovers.org	// Jag-lovers: …
www.jec.org.uk	// Jaguar Enthusiasts' Club
www.seattlejagclub.org	// Jaguar car club in Seattle
www.jags.org	// Jaguar Associates Group
Topic: **Jaguar Car**	Topic: **Jaguar Car Club**
www.jaguar.com	*www.jec.org.uk*
www.classicjaguar.com	*www.seattlejagclub.org*
www.jaguarvehicles.com	*www.jags.org*
www.jagweb.com	
www.jag-lovers.org	

Table 2. Examples of One Major Cluster with Hierarchical Structure

Finally, we give examples of some major clusters produced by the algorithm *HCA*(*D*) in tables 1 and 2. The table 2 gives examples with a hierarchical structure. The clustering results are satisfactory as the pages in the same cluster share the same topic.

5 Related Work and Discussions

There are many ways to cluster web pages, such as using hyperlink analysis (Chakrabarti, Dom and Indyk 1998, Pitkow and Pirolli 1997, Wang and Kitsuregawa 2001), content analysis (Zamir and Etzioni 1998, Wen, Liu, Wen and Zheng 2001) and link-content analysis (Marchiori 1997, PirollI, Pitkow and Rao 1996, Weiss, Vélez, Sheldon, Nampreмpre, Szilagyi, Duda and Gifford 1996). Here, we present and discuss some representative work that is based on hyperlink analysis.

The early representative work of hyperlink analysis can be found in (Kleinberg 1998) (Bharat and Henzinger 1998) (Chakrabarti, Dom, Gibson, Kleinberg, Raghavan and Rajagopalan 1998) (Dean and Henzinger 1999). In these works, hyperlink analysis was successfully applied to find web page communities, related web pages and more precisely find structures from a set of pages by combining page content analysis. Another successful example of hyperlink analysis application can be seen in the page ranking system of the search engine *Google* (Brin and Page 1998a, 1998b). These works reveal that hyperlinks convey semantics among the web pages and can be used in many areas.

For clustering web pages, Pitkow and Pirolli (1997) proposed two methods that directly used hyperlink analysis. The methods were all based on co-citation (via hyperlink) analysis, which builds upon the notion that when a page *A* contains links to pages *B* and *C*, then *B* and *C* are related in a manner (Figure 6 (a)). Pages *B* and *C* are said to be co-cited. When co-citation analysis was applied to the web page clustering in (Pitkow and Pirolli 1997), firstly, pages whose cited frequencies fell above a specific threshold were selected. Then co-citation pairs of pages with their frequencies of co-occurrence were formed. These co-citation page pairs were considered as the original clusters. One way to further cluster these original clusters was iteratively adding pairs of co-cited pages to the cluster that had at least one page in common with the added pairs. The produced clusters were non-hierarchical. Although this method was simple, the sizes of clusters were large, useful structures could not be revealed and the co-occurrence frequencies of co-cited pairs were not sufficiently exploited.

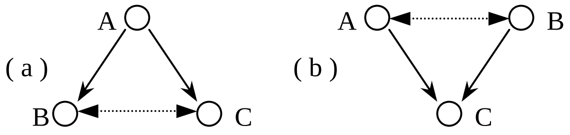

Figure 6. Co-Citation Relationship between Pages

To solve these problems, Pitkow and Pirolli (1997) also proposed another hierarchical clustering method. The co-occurrence frequencies of co-cited pairs were expressed in a co-citation matrix, an *Euclidean distance* matrix was calculated to measure the similarities between pages and then was used to hierarchically cluster the pages. While this work provided two approaches from co-citation analysis to cluster web pages, the co-citation analysis was based on mono-direction linkage. In other words, it only considered the relationship between two pages, e.g. pages *B* and *C* in figure 6(a), that were cited simultaneously by the citing page(s), e.g. page *A* in figure 6(a). From the hyperlink analysis point of view, however, if there exist links between two pages, there would be a certain mutual semantic relationship between these two pages in most cases. Therefore, if pages *A* and *B* have links to some common pages, such as page *C* in figure 6(b), it could also be inferred that *A* and *B* are related to some extent even if there are no direct links between *A* and *B*. Co-citation analysis, as well as the clustering algorithms based on it, should consider the bi-direction linkage relationships between pages, not just mono-direction ones. Meanwhile, the work in (Pitkow and Pirolli 1997) did not take the hyperlink transitivity into consideration.

The work in (Wang and Kitsuregawa 2001) proposed a clustering algorithm for web-searched pages, making use of the bi-direction linkage relationships between pages. Each page to be clustered was expressed as two vectors. One represented out-links of the page to other pages. Another one represented in-links of the page from other pages. The page similarity was measured by the cosine similarity of the vectors, rather than the Euclidean distance measurement. The clusters were also non-hierarchical. However, this algorithm only considered the linkage relationships between the web-searched pages and those pages that have links (linking or being linked) to the searched pages. The linkage relationships among the searched pages were omitted. So, if two searched pages have no common child and parent pages but have links between them, there will be no similarity between them. The reason is that the page linkage relationships were not considered within the whole page space. This work did not consider the hyperlink transitivity either.

Marchiori (1997) was aware of the hyperlink transitivity and made use of this property to improve the content-based web search. In his work, the information a page *A* contained with respect to a query was consisted of two parts: TEXTINFO(*A*) and HYPERINFO(*A*). The

TEXTINFO(*A*) was the textual information measurement of page *A* with respect to a certain query, while HYPERINFO(*A*) was a textual information measurement of other pages that were directly or indirectly pointed to by the page *A*. The HYPERINFO(*A*) is a function of hyperlink distances from *A* to other pages. The hyperlink in this work was actually used to define weights for incorporating other pages' information into the page *A*. The similarity was the content similarity between the page *A* and the query. No page similarity was directly defined from hyperlinks. Although the transitive hyperlink analysis was incorporated in the web page content analysis, the hyperlink analysis was mono-directed (i.e. only hyperlinks from the page *A* to other pages were considered). The work was not for clustering web pages; the page importance was not identified and incorporated in the page content measurement.

The work in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996) proposed a clustering algorithm that combined page content and hyperlink similarities. The hyperlink similarity between two pages was a linear combination of three components. The first component was measured by hyperlinks between the two pages, the second one was measured by common ancestor hyperlinks of the two pages, and the third component was measured by common descendant hyperlinks of the two pages. Precisely, the first hyperlink similarity component of two pages d_i and d_j with the shortest paths between them was defined directly from the hyperlink as

$$S_{ij}^{spl} = \frac{1}{2^{(spl_{ij})}} + \frac{1}{2^{(spl_{ji})}},$$

where spl_{ij} was the shortest path from d_i to d_j , and spl_{ji} was the shortest path from d_j to d_i . From this definition, it can be inferred that if there exits only one directed link from page d_i to d_j, their similarity is 0.5 (50%). Furthermore, for the situation in figure 7, the similarity between pages d_i and d_j is 1 (100%) according to the above similarity definition, which means these two pages can be considered as the same. This similarity measurement between pages is over-simplified.

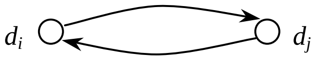

Figure 7. A Special Situation of Similarity Measurement

The algorithm in (Weiss, Vélez, Sheldon, Namprempre, Szilagyi, Duda and Gifford 1996) took the hyperlink transitivity into consideration. However, it regarded the influence of each page to the similarity measurement as the same. The page importance was not considered.

Different from the previous work, the work in this paper effectively incorporates hyperlink transitivity, page importance and bi-direction hyperlink analysis to form a new web page similarity measurement. The effectiveness of the corresponding hierarchical clustering algorithm shows the reasonableness and effectiveness of this new similarity measurement.

6 Conclusions

This work proposes a new web page similarity measurement and a corresponding hierarchical clustering algorithm. This new similarity measurement is purely based on hyperlinks among the pages in the concerned page source, and effectively incorporates hyperlink transitivity, page importance and bi-direction hyperlink analysis. The similarity is measured by the page correlation degrees in the concerned page source. The clustering improvement shown in the primary evaluations demonstrates the effectiveness and reasonableness of this web page similarity, and the effectiveness of the proposed clustering algorithm as well.

The hyperlink-based clustering is intuitive and successful in many cases. However, the hyperlink only partially reveals semantics among the web pages. A proper combination of effective page hyperlink similarities with effective page content similarities might be another approach to greatly increase the effectiveness and efficiency of web page clustering. Meanwhile, some problems in hyperlink analysis still remain to be solved, such as how to more reasonably and precisely determine the page correlation factor *F*. More experiments need to be conducted to further demonstrate the feasibility of the proposed algorithms. The similarity defined in this work could also be applied to other web-related areas, such as web search improvement, related web page finding and XML document clustering. The research in these directions is to be carried out in the near future.

7 References

BAEZA-YATES, R. and RIBEIRO-NETO, B. (1999): *Modern Information Retrieval,* Addison Wesley, ACM Press.

BHARAT, K., BRODER, A., HENZINGER, M., KUMAR, P. and VENKATASUBRAMANIAN, S. (1998): The Connectivity Server: Fast Access to Linkage Information on the Web, *Proceedings of the 7th International World Wide Web Conference*, 469-477.

BHARAT, K. and HENZINGER, M. (1998): Improved Algorithms for Topic Distillation in a Hyperlinked Environment, *Proc. the 21st International ACM Conference of Research and Development in Information Retrieval (SIGIR98),* 104-111.

BOTAFOGO, R.A. (1993): Cluster Analysis for Hypertext Systems, *Proceedings of ACM 16th Annual International SIGIR'93,* Pittsburgh, PA.

BOTAFOGO, R.A., RIVLIN, E. and SHNEIDERMAN, B. (1992): Structural Analysis of Hypertexts: Indentifing Hierarchies and Useful Metrics, *ACM Transactions on Information Systems,* Vol 10, No 2, 142-180.

BOTAFOGO, R. A. and SHNEIDERMAN, B. (1991): Identifying Aggregates in Hypertext Structures, *Proceedings of Hypertext'91,* 63-74.

BRIN, S. and PAGE, L. (1998a): The Anatomy of a Large-Scale Hypertextual Web Search Engine, *Proceedings of the 7th International World Wide Web Conference*, Brisbane, Australia.

BRIN, S. and PAGE, L. (1998b): The PageRank Citation Ranking: Bringing Order to the Web, *http://www-db.stanford.edu/~backrub/pageranksub.ps*.

CARRIERE, J. and KAZMAN, R. (1997): WebQuery: Searching and Visualizing the Web through Connectivity, *Proceedings of the 6th International world Wide Web Conference*.

CHAKRABARTI, S., DOM, B., GIBSON, D., KLEINBERG, J., RAGHAVAN, P. and RAJAGOPALAN, S. (1998): Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text, *Proc. the 7th International World Wide Web Conference*, 65-74.

CHAKRABARTI, S., DOM, B. and INDYK, P. (1998): Enhanced Hypertext Categorization Using Hyperlinks, *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'98)*, Seattle, USA, 307-318.

DEAN, J. and HENZINGER, M. (1999): Finding Related Pages in the World Wide Web, *Proc. the 8th International World Wide Web Conference*, 389-401.

DUBES, R. J. and JAIN, A.K. (1988): *Algorithms for Clustering Data*, Prentice Hall.

HOU, J. and ZHANG, Y. (2002a): Constructing Good Quality Web Page Communities, *Proceedings of the 13th Australasian Database Conferences (ADC2002)*, Melbourne, Australia, 65-74.

HOU, J. and ZHANG, Y. (2002b): A Matrix Approach for Hierarchical Web Page Clustering Based on Hyperlinks, *Proceedings of Mining Enhanced Web Search 2002 (MEWS02)*, Singapore.

HOU, J., ZHANG, Y. and CAO, J. (2002): Eliminating Noise Pages for Better Web Page Communities, *Journal of Research and Practice in Information Technology*, (invited submission).

HOU, J., ZHANG, Y., CAO, J., LAI, W. and ROSS, D. (2002): Visual Support for Text Information Retrieval Based on Linear Algebra, *Journal of Applied Systems Studies*, Cambridge International Science Publishing, Vol.3, No.2.

JIANG, H., LOU, W. and WANG, W. (2001): Three-tier Clustering: an Online Citation Clustering System, *Proceedings of the Second international Conference on Web-Age Information Management (WAIM2001)*, Xi'An, China, 237-248.

KLEINBERG, J. (1998): Authoritative Sources in a Hyperlinked Environment, *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms (SODA)*.

LIN, X., LIU, C., ZHANG, Y. and ZHOU, X. (1999): Efficiently Computing Frequent Tree-Like Topology Patterns in a Web Environment, *Proceedings of the 31th international Conference on Technology of Object-Oriented Languages and Systems*, Nanjing, China, IEEE Computer Society Press, 440 – 447.

MARCHIORI, M. (1997): The Quest for Correct Information on the Web: Hyper Search Engines, *Proc. of the 6th International Word Wide Web Conference*.

PIROLLI, P., PITKOW, J. and RAO, R. (1996): Silk from a Sow's Ear: Extracting Usable Structures from the Web, *Proceedings of ACM SIGCHI Conference on Human Factors in Computing*.

PITKOW, J. and PIROLLI, P. (1997): Life, Death, and Lawfulness on the Electronic Frontier, *Proceedings of ACM CHI'97*, Atlanta, USA, 383-390.

TERVEEN, L. and HILL, W. (1998): Finding and Visualizing Inter-site Clan Graphs, *Proceedings of the Conference on Human Factors in Computing Systems (CHI-98): Making the Impossible Possible*, Los Angeles, USA, 448-455.

WANG, L. (1997): On Competitive Learning, *IEEE Transaction on Neural Networks*, Vol.8, No.5, 1214-1217.

WANG, Y. and KITSUREGAWA, M. (2001): Use Link-based Clustering to Improve Web Search Results, *Proceedings of the Second International Conference on Web Information Systems Engineering (WISE2001)*, Kyoto, Japan, 119-128.

WEISS, R., VÉLEZ, B., SHELDON, M.A., NAMPREMPRE, C., SZILAGYI, P., DUDA, A. and GIFFORD, D.K. (1996): HyPursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering, *Proceedings of the Seventh ACM Conference on Hypertext*, 180-193.

WEN, C. W., LIU, H., WEN, W. X. and ZHENG, J. (2001): A Distributed Hierarchical Clustering System for Web Mining, *Proceedings of the Second international Conference on Web-Age Information Management (WAIM2001)*, Xi'An, China, 103-113.

ZAMIR, O. and ETZIONI, O. (1998): Web Document Clustering: A Feasibility Demonstration, *Proceedings of ACM SIGIR'98*, Melbourne, Australia, 46-54.