
©2005 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
# Matrix Model for Web Page Community

Jingyu Hou  
*School of Information Technology*  
*Deakin University*  
*Burwood, VIC 3125, Australia*  
jingyu@deakin.edu.au

Atul Sajanhar  
*School of Information Technology*  
*Deakin University*  
*Burwood, VIC 3125, Australia*  
atuls@deakin.edu.au

## Abstract

Discovering intrinsic relationships/structures among concerned web information objects such as web pages is important for effectively processing and managing web information. In this work, a set of web pages that has its own intrinsic structure is called a web page community. This paper proposes a matrix model to describe relationships among concerned web pages. Based on this model, intrinsic relationships among pages could be revealed, and in turn a web page community could be constructed. The issues that are related to this model and its applications are investigated and studied. Some applications based on this model are presented, which demonstrate the potential of this matrix model in different kinds of web page community construction and information processing.

## 1. Introduction

The World Wide Web is now a huge information source. The data on the web, however, are neither raw nor very strictly typed as those in conventional database systems. This feature makes it hard to directly apply conventional techniques to process and manage information on the web. For web information processing and management, the main obstacle is the absence of a well-defined underlying data model. One approach to overcome this obstacle is to reveal intrinsic or semantic relationships/structures among concerned web data instead of defining a data model. In this work, we focus on the most commonly used information object (data) on the web - web pages (HTML documents), and define a *web page community* as a set of concerned web pages that has its own *intrinsic* structure.

The key to constructing a web page community is the intrinsic relationships among web pages. In other words, a simple gathering of web pages could not be considered as a community if there is no intrinsic relationship among them. Intrinsic relationship/structure has different meanings for different situations. For example, the pages that are clustered into clusters form a web page community; a set of pages that are relevant to a given page also forms a web page community. In order to uncover intrinsic relationship/structure among web pages, it is necessary to firstly model web pages and their row relationships. The traditional approach is using vector model, i.e. each page is modeled as a keyword vector. The intrinsic relationship of pages such as pages similarity is revealed by performing operations on vectors. Document object model (DOM) [7] is another model for web pages. These models focus on modeling individual web page. Relationships among pages are not directly modeled.

In this paper, we propose a matrix model for web pages and community construction. With this model, web pages, as well as their relationships, are modeled within a matrix framework and intrinsic relationship among web pages could be uncovered via mathematical operations on the matrix rather than on individual vectors, which lays corresponding algorithms on a solid mathematical base. In other words, all concerned pages and their relationships could be considered as a whole in terms of a matrix. Therefore, this model could be used not only for web page community construction, but also for other kinds of information processing. Although a matrix model is widely used, when it is applied within web environment, there are a lot of specific issues to be addressed and resolved because of characteristics of web date.

This paper is organized as follows. In the next section, we propose the matrix model for web pages and community construction. In section 3, 4, 5 and 6, we discuss the issues that are related to this model. The discussion mainly focuses on hyperlink based web page community. The reasons behind it can be found in [12] [9]. In section 7, we present some web applications, as well as some non-web applications, that are based on this model. Finally we conclude this work in section 8.

## 2. Matrix Model

Usually, a community is constructed from a set of concerned information objects, such as web pages and web access logs. For general purposes, we define a *data space* as a set of concerned information objects. Given a data space, how to model it depends on what information is used to express relationships between objects within the
space. For example, given a data space that consists of a set of documents, the relationship between documents could be expressed by keywords. Considering these factors, the matrix model is defined as a framework with the following requirements to be met:

(1) A data (information) space is constructed. For example, in a conventional database system, the data space might be the whole documents within it. But in the context of the web, the situation will be complex. Different web applications require different data spaces.

(2) Two sets of information entities (objects), denoted as \( E_1 \) and \( E_2 \), within the constructed data space are identified. One set should be a reference system to another, i.e. the relationships between entities in \( E_1 \) are determined by those in set \( E_2 \), and vice versa. For example, \( E_1 \) could be a set of documents; \( E_2 \) could be a set of keywords.

The relationship between \( E_1 \) and \( E_2 \) is located in one of three classes:

(i) \( E_1 = E_2 \). Two sets are the same.
(ii) \( E_1 \neq E_2 \). Two sets are different and in different category. For example, \( E_1=\{\text{documents}\} \), \( E_2=\{\text{keywords}\} \).
(iii) \( E_1 \sim E_2 \). Two sets are different but in the same category. For example, \( E_1 \) could be one set of web pages, and \( E_2 \) could be another set of web pages.

(3) Original correlation expression between entities that belong to different sets \( E_1 \) and \( E_2 \) is defined and modeled into a matrix. The correlation expression is defined as

\[
(E_1 \triangleright \triangleleft E_2) \leftrightarrow CI,
\]

where \( CI \) stands for correlation information which is the information used to describe the relationship between the entities in \( E_1 \) and \( E_2 \). This expression means the correlations between entities in \( E_1 \) and \( E_2 \) are expressed by defined correlation information \( CI \).

With this expression, each entity of \( E_1 \) is modeled as a row (column) of a matrix, and each entity of \( E_2 \) is modeled as a column (row) of the matrix. The values of matrix elements (intersections of rows and columns) represent the original correlation degrees between entities that belong to \( E_1 \) and \( E_2 \) separately. These original correlations degrees are determined by \( CI \). For example, suppose \( E_1=\{\text{documents}\} \), \( E_2=\{\text{keywords}\} \), we can define \( CI=\{\text{weighted keywords}\} \). Each document in \( E_1 \) is represented as a row of a matrix, and each keyword in \( E_2 \) is represented as a column of the matrix. If one document contains a keyword, the corresponding matrix element value is the weight of the keyword, otherwise it is 0.

The above requirements define the matrix model for information processing and community construction. This model paves the way of revealing intrinsic relationships among information entities through matrix and/or other related mathematic operations. However, when this model is applied to practical situations, especially the web, there are some related issues to be investigated. In this work, the discussion concentrates on web pages and their hyperlinks, i.e. \( E_1 \) and \( E_2 \) are two sets of web pages and \( Cl=\{\text{hyperlinks}\} \) in the above matrix model. The ideas and methods, however, could also be applied to other kinds of correlation expressions. The following sections are dedicated to these issues.

### 3. Data Space Construction

The first requirement of the matrix model is data space construction. In the context of the web, since the web size is very huge and it is impossible to model all pages on the web within a matrix, data space construction is critical to the success of the matrix model. It depends on what the web application requirements are or what kind of web page community to be constructed.

For discussion convenience, we adapt the following concepts: if there is a hyperlink from page \( P \) to page \( Q \), \( P \) is called a parent of \( Q \) and \( Q \) is called a child of \( P \); if two pages have at least one common parent page, these two pages are called siblings. As indicated in [13], in terms of hyperlink, the semantic information about a given page \( u \) is most likely to be given by its parent and child pages. Therefore, the data space construction in terms of hyperlink should focuses on concerned pages and their parent/child pages.

Although there are many ways of constructing hyperlink-based data space [12] [9] [11], they can be classified generally into two categories. The first one is selecting parent/child pages; the second one is selecting parent-child and child-parent pages. The details of these two kinds of data space construction methods are described below.

**Parent/child page selection** This data space construction is usually composed of two steps. The first step is to choose concerned pages to form a root of the data space. Secondly, the parent/child pages of each root page are selected, together with the root pages, to form the data space. This data space also includes hyperlinks between any two pages in the data space, and is considered to be a specific directed graph whose nodes are pages and edges are hyperlinks. The illustration of this method is shown in figure 1. The root of data space is located in the middle of the figure. The solid line arrows represent the hyperlinks that are used to select parent/child pages of the root pages. The dashed line arrows indicate other hyperlinks that exist between pages in the data space. In practical situations, it is necessary to restrict the number of parent/child pages for each root page, such that the size of the data space is reasonable [12] [11].

**Parent-child and child-parent page selection.** This method usually consists of three steps. Firstly, the
concerned pages are selected to form a root of the data space. Secondly, parent and child pages of each root page are selected. Finally, for each selected parent/child page, its child/parent pages are selected. All selected pages together with their corresponding hyperlinks form the data space. The illustration of this method is shown in figure 2. For clearance, this figure only shows one root page. In practical situations, similar to the first method, it is also necessary to restrict the number of each page’s parent/child pages in the data space. Depending on application requirements, this kind of data space could be constructed by only using parent-child or child-parent page selection instead of both at the same time.

4. Noise and Malicious Hyperlink Issue

When constructing a data space for web page communities, it is very likely that some pages that are hyperlinked have no semantic relationship. This kind of hyperlinks/pages is called noise hyperlinks/pages. They should not be included in the data space or their influence on web page community construction should be reduced, otherwise they will distort the nature of communities.

There are two ways of eliminating or reducing the influence of noise hyperlinks/pages in a data space. The first one is to filter noise hyperlinks/pages when constructing a data space. To this end, the hyperlinks in a page are assigned semantics by the keywords around hyperlinks (i.e. anchor text) and page structure information [4]. Then the hyperlink’s semantics are compared with the page semantics. If the similarity is below a certain threshold, then the hyperlink and related page are filtered.

The second way is to eliminate or reduce the noise hyperlink/page influence in the process of revealing intrinsic relationships. This method usually refers to developing various algorithms, such as SVD based algorithm [11] and co-citation algorithm [8] [5] [9]. Since hyperlinks are dynamic and there is no standard of identifying noise hyperlinks/pages, it can be foreseen that various algorithms will be proposed and research on this issue will still be a challenge.

Malicious hyperlinks are another kind of hyperlinks that need to be addressed when constructing a data space. Malicious hyperlinks are those that are deliberately added in web pages to increase the importance of some web pages on the Web or a web site, even if these added hyperlinks have no semantic relationship with the emphasized pages. This trick will cheat web search engines and unreasonably increase the importance of some pages in the data space.

Before discussing the approaches of reducing influence of malicious hyperlinks, we firstly introduce the following concepts.

Definition 1: Two pages $p_1$ and $p_2$ are back co-cited if they have at least one common parent page. The number of their common parents is their back co-citation degree. Two pages $p_1$ and $p_2$ are forward co-cited if they have at least one common child page. The number of their common children is their forward co-citation degree.

Definition 2: The pages are intrinsic pages if they have same page domain name. Here the domain name is the first level of the URL string associated with a web page.

Definition 3 [5]: Two pages are near-duplicate pages if (a) they each have more than 10 links and (b) they have at least 95% of their links in common.

As indicated in the above section, a data space construction usually begins with selecting a root of the data space, then growing this root to form the data space by adding parent/child pages of each root page. The malicious hyperlinks, therefore, are most likely to be brought into the data space by these parent/child pages. The following is an approach of dealing with malicious hyperlinks by merging intrinsic and near-duplicate parent/child pages.

Suppose we choose a page $u$ in the root of the data space, for pages in a web site (or server) that are hyperlinked deliberately, if some of them are imported into the data space as the parent pages of $u$, their children (the siblings of $u$) most likely come from the same site (or server), and the back co-citation degrees of these children with $u$ would be unreasonably increased. With the merger of intrinsic parent pages, the influence of the pages from the same site (or server) is reduced to a reasonable level (i.e. the back co-citation degree of each child page with $u$ is only 1) and the malicious hyperlinks are shielded off.
For example, in figure 3, suppose the parent pages \( P_1, P_2, P_3 \) and their children \( S_{1,1}, \ldots, S_{1,2} \) be intrinsic pages. In situation (a), the back co-citation degree of page \( S_{2,2} \) with \( u \) is unreasonably increased to 3, which is the ideal situation the malicious hyperlink creators would like. The situation is the same for the pages \( S_{1,2} \) and \( S_{2,1} \). With intrinsic parent page merging, the situation (a) is changed to the situation (b) where \( P \) is a logic page representing the union of parent pages \( P_1, P_2, P_3 \), and the contribution of each child to the back co-citation degree with \( u \) is only 1, no matter how tightly these intrinsic pages are linked together.

\[
\text{Figure 3. Intrinsic parent page merging}
\]

The idea of this approach is the same for merging intrinsic child pages, as well as near-duplicate parent/child pages.

5. Hyperlink Transitivity and Decline Rate

Traditionally [12] in the web context, when mapping the original correlation expression \((E_1 \succ \subset E_2) \leftarrow CI\) into a matrix, where \( E_1 \) and \( E_2 \) are two sets of web pages and \( CI=\{\text{hyperlinks}\} \), each page in \( E_i \) is mapped as a row (column) of the matrix, and each page in \( E_2 \) is mapped as a column (row) of the matrix. The matrix element value is determined as follow: if there is a hyperlink from a page in \( E_1 \) to another page in \( E_2 \), then the corresponding matrix element value is set to 1, otherwise 0. This original correlation matrix is usually called adjacent matrix [12]. However, this adjacent matrix only considers direct (one-step) hyperlinks between any two pages in the data space. In many cases, some pages have no direct hyperlinks between them, but there is still correlation between them through other pages and hyperlinks. This hyperlink transitivity is one of the obvious features of web data, and should be mapped into the matrix model as well.

When considering hyperlink transitivity, it is worth notice that the role each page plays in the data space \( S \) is different. Precisely, two kinds of pages need to be noticed. The first one is a 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 \( S \). Another kind is a 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 \( S \). The pages of the first kind are called index pages in [3] (hub pages in [12]), and those of the second kind are called reference pages in [3] (authority pages in [12]). These pages are most likely to reflect certain topics within the data space \( S \). However, we filter the home pages of commonly used search engines (e.g. Yahoo!, AltaVista, Google and Excite) from \( S \), since these pages are not related to any specific topics. To label the importance of each page within the data space, we define a weight for each page.

For a page \( P_i \) in the data space \( S \), we denote its weight as \( w_i (0 < w_i \leq 1) \). Given weight for each page in \( S \), we are able to define a weight for each hyperlink between any two pages in \( S \). This hyperlink weight is the function of page weights that are linked by this hyperlink. Actually, suppose there are two hyperlinked pages \( P_i \) and \( P_j \) in the data space \( S \) and their page weights are \( w_i \) and \( w_j \) respectively, then their hyperlink weight is defined as \( w_{ij} = f(w_i, w_j) \), where \( f \) is a function and \( 0 < w_{ij} \leq 1 \). How to define web page and hyperlink weight is still a challenge problem. Since we concentrate on hyperlink transitivity here, the discussion of how to define page weight is not covered. The interested reader could refer to [10] for one solution to this problem.

With page and hyperlink weight, we could map transitivity correlations between pages in the data space into a matrix. Before proposing the mapping method, we firstly give the following definitions.

**Definition 4.** 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 (direct or indirect) from page \( A \) to page \( B \), then \( l(A,B) = \infty \).

It can be inferred from this definition that \( l(A,B) = \infty \) does not imply \( l(B,A) = \infty \).

**Definition 5.** Decline rate, denoted as \( F (0 < F < 1) \), is a variable that measures the correlation decline rate between two page with direct link, i.e. if page \( A \) has a direct link to page \( B \) with hyperlink weight \( w_{AB} \), then the correlation degree from page \( A \) to page \( B \) is \( w_{AB} F \).

How to determine the value of decline rate \( F \) to more precisely reflect the correlation relationship between pages is beyond the scope of this work. Further research could be done in this area. For simplicity, we suppose the value of \( F \) is a constant (e.g. \( \frac{1}{2} \) in [15]).

With above definitions, a correlation degree between any two pages can be defined.

**Definition 6.** The correlation degree from page \( i \) to page \( j \), denoted as \( c_{ij} \), is defined as

\[
c_{ij} = w_{i,k_1} w_{k_1,k_2} \cdots w_{k_n,j} F^{l(i,j)}
\]
where \( F \) is the decline rate, \( s(i,j) \) is the distance from page \( i \) to page \( j \), and \( w_{ij1}, w_{ij2}, ..., w_{ijn} \) are hyperlink weights respectively between the adjacent pages \( i, k1, k2, ..., kn, j \) that form the distance \( s(i,j) \), i.e. \( i \rightarrow k1 \rightarrow k2 \rightarrow \ldots \rightarrow kn \rightarrow j \). If \( i = j \), then \( c(i) \) is defined as 1.

For two web page sets \( E_1 \) and \( E_2 \) in a data space \( S \), we suppose the size of \( E_1 \) (i.e. the number of pages in \( E_1 \)) is \( m \), the size of \( E_2 \) is \( n \) and denote \( E = E_1 \cup E_2 \). Then hyperlink-based transitive correlation degrees of all the pages in \( E \) can be mapped into a \((m+n)\times(m+n)\) matrix \( C = (c_{ij})_{(m+n)\times(m+n)} \) called correlation matrix. This mapping incorporates hyperlink transitivity, decline rate and page importance.

The following section proposes an algorithm of computing distance \( s(i,j) \) within a matrix framework.

6. Shortest Path Finding Algorithm

The shortest path (distance) in definition 6 can be found and computed via some operations on elements of a special matrix called primary correlation matrix. The primary correlation matrix \( A = (a_{ij})_{(m+n)\times(m+n)} \) is constructed as follow:

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

Based on this primary correlation matrix, an algorithm of computing distance \( s(i,j) \) between any two pages \( i \) and \( j \) in \( E \) is described as follows:

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

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

**Step 3:** If \( a_{jk} \neq 0 \) and \( a_{jk} \neq 1 \), calculate \( \text{factor} \times a_{jk} \);

**Step 4:** If \( \text{factor} \times a_{jk} > a_{jk} \), then replace \( a_{jk} \) with \( \text{factor} \times a_{jk} \). 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 = \( \text{factor} \times F \) 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 \( E \) have been considered.

**Step 7:** After element values of matrix \( A \) are updated by the above steps, the distance from page \( i \) to page \( j \) is \( s(i,j) = \left[ \log a_{ij} / \log F \right] \).

7. Matrix Model Applications

In this section, we present some applications that are based on the proposed matrix model, with more attentions being paid to the model requirements. For more algorithm details, please refer to the corresponding references.

**Noise Page Elimination** This problem arises from a web application that finds hub and authority pages from a data space [12]. To eliminate noise pages from the data space, Hou et al [11] proposed a noise page elimination algorithm (NPEA) using this matrix model. For NPEA, the data space is the same as that of [12] which is constructed using the parent/child page selection method. Precisely, the root page set is \( R \), and the final data space is \( B \). For eliminating noise pages, two matrices are built to model two correlation expressions: one for \((R \gg \ll R) \leftarrow CI\), another one for \((B-R) \gg \ll B) \leftarrow CI\), where \( CI = \{\text{hyperlinks}\} \). Based on these matrix models, singular value decomposition (SVD) of matrix is incorporated into the algorithm to eliminate noise factors in \( R \) and \( B-R \). This purified \( R \) is then used as a reference system to eliminate noise pages from \( B-R \). The experimental evaluation of this algorithm shows the effectiveness of this algorithm.

**Relevant Page Finding** This problem is described as follow [9] [5]: given a web page \( u \), find a set of pages that are semantically related to it. The critical aspect of this problem is how to construct a data space for this given page such that the data space is rich in semantic related pages and is of a reasonable size. In [9], the data space is constructed from a special root set which only contains this given page \( u \). Then the parent/child and child/parent page selection method is used to construct the required data space. This construction also incorporates techniques of dealing with malicious hyperlinks. Within this data space, \( C = \{\text{child pages of } u\}, P = \{\text{parent pages of } u\}, FS = \{\text{parent pages of } C\} \) and \( BS = \{\text{child pages of } P\} \).

The extended co-citation algorithm in [9] finds relevant pages directly from \( FS \) and \( BS \). Another algorithm, latent linkage information (LLI) algorithm, of [9] is based on the matrix model. Two matrices are built to model two correlation expressions: one for \((FS \gg \ll C) \leftarrow CI\), another is for \((BS \gg \ll B) \leftarrow CI\), where \( CI = \{\text{hyperlinks}\} \). Relevant pages are found by LLI algorithm which takes advantage of SVD of these two matrices. It was found in the experiments that extended co-citation algorithm and LLI algorithm could find more semantic web pages.

**Web Page Clustering** Web page cluster is another important web page community. One of the matrix based clustering algorithms can be found in [10]. For a set of web pages that are to be clustered using their hyperlink information, the data space \( S \) of this algorithm is constructed with the parent/child page selection method. The pages in the data space \( S \) is modeled into a matrix with the correlation expression \((S \gg \ll S) \leftarrow CI\), where \( CI = \{\text{correlation degrees}\} \) which is defined in definition 6.

From this correlation matrix, page similarities are calculated using vector operations on matrix, and a new similarity matrix is constructed from these similarities.
This similarity matrix models another correlation expression $(S) \Rightarrow CI$, where $CI = \{\text{similarities}\}$. Finally, matrix partitioning operations are applied to this similarity matrix iteratively and hierarchical web page clusters are produced. All clustering operations of this algorithm are within the matrix framework and produce satisfactory results in evaluation.

Non-Web Applications One of the representatives of this kind of applications is matrix based textual information retrieval [1] [6], which finds semantic related documents from their keywords even if these documents do not share the same keywords. The corresponding method is called Latent Semantic Indexing (LSI). In LSI, $E_i = \{\text{documents}\}, E_j = \{\text{keywords}\}$ and $CI = \{\text{weighted keywords}\}$. A matrix is constructed to model this correlation expression $(E_i \Rightarrow CI) \Rightarrow CI$. SVD is then applied to this matrix to reveal important associative relationships between keywords and documents that are not evident in individual documents. As a consequence, an intelligent indexing for textual information is implemented. Papadimitriou et al [14] studied the LSI method using probabilistic approaches and indicated that LSI in certain settings is able to uncover semantically "meaningful" associations among documents with similar patterns of keyword usage, even when they do not actually use the same keywords.

8. Conclusions

Matrix model in this work could be widely used in various kinds of information processing, especially in web page community construction. To guarantee the effectiveness and success of this matrix model, the data space should be carefully constructed, and the correlation information for representing the relationship between data items in the data space must be identified. In terms of web page hyperlink analysis, data space construction depends on web application requirements, and correlation information should consider hyperlink transitivity and transitivity decline rate in some cases. Many successful applications demonstrate the effectiveness of this matrix model in web page community construction and other kinds of information processing. The related aspects of this model are also challenge research areas within which many problems need to be solved in the future.

References


