# A Survey of Web Matrices for Link Structure Analysis Algorithms

**Dr. Nimisha Modi**
Dept. of Computer Science, VNSGU, Surat, Gujrat, India

**Abstract**
WWW is a huge repository of on-the-click information that is organized as social graph through hyperlinks between them. The spatial locality of a set of web documents reflects their topical locality. Research work in the field of link structure analysis is targeted to improve the efficiency of web information retrieval via exploring the semantic of web document in context of its hyperlinked documents. Basically, link structure analysis algorithms model the web as a graph of social network and represent this graph using various matrices. The measures on the matrices are used to find the importance ranking or to categorize web documents. Each algorithm follows its own perception for a structure of web graph and the matrices that represents the graph. The paper presents the survey on how these algorithms illustrate and derive their web matrices? The approach of algorithms for finding rank vectors for web matrices and practical problems like matrix convergence and uniqueness of rank vector are also reviewed.

**Keywords**
Web Structure Mining, Social Network Graph, Web Matrices, Power Method, Eigenvector

## I. Introduction

The theory of web structure mining is based on the notion that a hyperlink connects related documents i.e. a link from a page p to page q can be viewed as an endorsement of q by p. This indicates a positive judgment by p about the content of q. Thus the quality of a web page can be judged based on the hyperlink structure in which it is embedded. A variety of research works within link-based retrieval strategies are targeted to explore various methods for enriching the judgment about the local document's content using the context of hyperlinked documents. Among that, three significant researches are corresponding to the algorithms - PageRank [1], Hyperlink Induced Topic Search (HITS) [2] and Stochastic Approach for Link Structure Analysis (SALSA) [3]. Major research work in this area is concentrated around these three basic algorithms either to explore these algorithms or to mold these algorithms with various objectives.

Two research students of the Stanford University, Brin and Page, incorporated PageRank into the search engine Google [4] to prioritize the results of keyword based search. Due to good business success of Google, PageRank became very much famous. The algorithm HITS was introduced by Jon Kleinberg [2]. An extension of HITS was used by the search engine Teoma. The link analysis algorithm incorporated by Teoam is now referred to by Ask.com as the Expert Rank [5] algorithm. HITS has been also incorporated into the CLEVER [6] project at IBM Almaden Research Centre. To overcome some of the limitations of HITS and PageRank, Lempel and Moran [3] proposed the variation of link analysis ranking algorithm - Stochastic Approach for Link Structure Analysis (SALSA).

Bibliometrics research uses the citation structure within documents to produce numerical measures that indicates the importance and impact of papers. Link structure analysis is using the concept of co-citation and co-references [7] to produce numerical measures for the importance of web pages. All the three algorithms work to assign a numerical weight to each element of a hyperlinked set of documents on the web graph. This numerical weight is known as rank score and is analogy to notion of prestige within social network analysis. The rank score represents the importance of web pages based on their hyperlinked structure. The vector of rank corresponding to each individual document is known as rank vector.

HITS and SALSA uses the notion of hub and authority. An authority is a web document having many in-links. A hub is a document having many out-links such as portal pages. The algorithms HITS and SALSA assign two separate scores to each document on web – hub score and authority score. PageRank gives the single rank score for each web page that is corresponding to authority score.

The paper presents our analysis of the web graph and matrices that are used by these algorithms for finding rank vectors. We also compare the matrix convergence problem and stability of results for the given algorithms. The paper illustrates the approach of these three algorithms using an example of one small web containing 9 web documents (web pages) that are labeled with number 1 to 9. The linkage structure for the web pages is given in Table 1.

Table 1: Linkage Structure of Small Web

| Hyperlinks From Page | Pointing to Pages |
|---|---|
| 1 | 2,3,7 |
| 3 | 2,7 |
| 5 | 4,6 |
| 6 | 5 |
| 7 | 1,2,9 |
| 8 | 6,5,4 |
| 9 | 4 |

## II. Web Graphs

Link analysis algorithms view the web as social graph where web pages and hyperlinks are represented as nodes and edges respectively. This section discuss the differences between various algorithms with respect to type of graph they select to represents the web documents and their interconnection.

To observe the random walk of web surfer on hyperlinked graph, Page Rank works with the directed neighborhood graph where each web page is a node in graph and hyperlink between two web pages is a directed edge from the referring page to the referred page. The algorithm HITS also represents the web as the neighborhood graph [8]. Fig. 1 shows the neighborhood graph corresponding to web of 9 documents as per linkage structure given in Table 1.

The major limitation of Kleinberg's HITS is - it follows mutual reinforcement principle and it is susceptible to the Tightly Knit Communities (TKC) [9] effect. SALSA is developed to overcome the TKC effect. So it performs random walks on a bipartite web graph to identify authorities and hubs. For that, Lempel and Moran generates a bipartite undirected graph corresponding to linkage

structures of web documents by constructing two subsets of nodes – $V_a$ and $V_h$ from the directed graph N. Bipartite undirected graph G is defined by three set: $V_a$, $V_h$ and E where –

- $V_a$ is subset of all the nodes with positive in-degree i.e. set of pages {1,2,3,4,5,6,7,9} in given example.
- $V_h$ is subset of all the nodes with positive out-degree i.e. set of pages {1,3,5,6,7,8,9} in given example.
- E is a set of directed edges in N.

The bipartite undirected graph corresponding to given example web is shown in fig. 2.

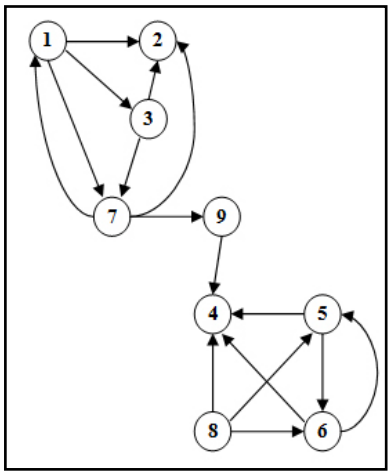


Fig. 1: Directed Neighborhood Graph N

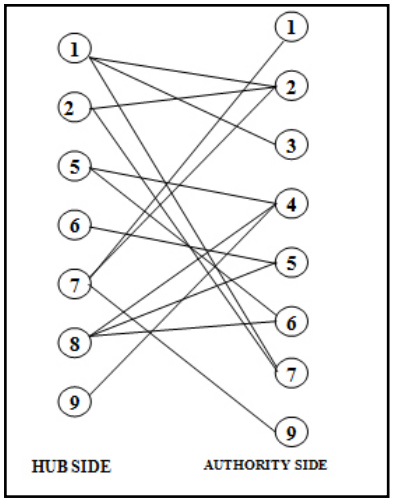


Fig. 2: Undirected Bipartite Graph

## III. Web Matrices

All these three algorithms find the vectors that represents the importance scores (rank scores) for each page in web graph. These vectors are known as rank vectors. The algorithms find rank vectors using the theory of linear algebra [10]. For that, the matrices are derived to express the linkage structure of graph and to find rank vectors analogous to principle eigenvectors of the matrices. The diversity within the approach of various algorithms is reflected in matrices that they derive for computation of rank vectors. This section analyzes - how web matrices are derived for all the three algorithms?

Major link structure analysis algorithms follow either co-citation based approach or random walk based approach.

- The idea behind co-citation analysis is that when two pages A and B both point to some page X, it is reasonable to assume that A and B share a mutual topic of interest. Similarly, when X links to pages A and B both, it is probable that A and B share some mutual topic.
- On the other hand, random-walk based schemes apply a random walk model to the directed web graph. Each Page is then ranked by the probability of visiting that page in the modeled random walk.

The PageRank algorithm is a random walk approach. PageRank makes use of Markov model that assumes web page accesses to be "memoryless" i.e. access statistics are independent of events more than one interval ago. PageRank follows random walk for Markov chain [11] on a corresponding transition probability matrix that is derived from the neighborhood graph. Transition probability matrix P is an n*n matrix, where n is the number of pages in the web.

Each row of the matrix P represents start page, each column represents destination page and each element $P_{ij}$ is the probability that the random surfer navigates from pagei to $page_j$ in one mouse-click.

The value of $P_{ij}$ can be obtained as $P_{ij} = \frac{1}{O_i}$ . Here Oi is the number indicating the total number of out-links from $page_i$.

P is a non-negative matrix with row sums equal to 0 or 1. If $page_i$ contains out-links, the sum of $P_{ij}$ for all outlinks to $page_j$ from $page_i$ is 1. In case, when page does not contain any out-link the sum of $P_{ij}$ for $page_i$ is 0. Pages with no out-links are called the dangling node. PageRank matrix for our example web is given in fig. 3. As documents 2 and 4 are having no out-link, raw 2 and raw 4 having zero for all elements.

The algorithm HITS works on n*n adjacency matrix L of neighborhood graph where n is the number of pages in the web and each element $L_{ij}$ is set as either 1 or 0. The element $L_{ij}$ is 1 if a there exist a directed edge (i.e. hyperlink) from nodei to $node_j$. The element $L_{ij}$ is 0 if there exists no link from nodei to $node_j$.

To assign dual rank to each web page, HITS finds two separate vectors viz. authority vector and hub vector. The authority vector gives the authority score for individual web pages. Similarly, the hub vector is a vector of the hub score for individual web pages.

HITS follows the mutual reinforcement approach as it derives the hub and authority matrices form adjacency matrix L and transpose of matrix L (i.e. $L^T$). Matrix $L^TL$ is a matrix which is used to find an authority vector, so it is called an authority matrix. Matrix LLT is used to find a hub vector, so it is called a hub matrix. An authority vector and a hub vector are eigenvectors corresponding to highest eigenvalue of the authority matrix and the hub matrix respectively.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0.3333 | 0.3333 | 0 | 0 | 0 | 0.3333 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0.5 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0.5 | 0 | 0.5 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0.3333 | 0.3333 | 0 | 0 | 0 | 0 | 0 | 0 | 0.3333 |
| 0 | 0 | 0 | 0.3333 | 0.3333 | 0.3333 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

Fig. 3: Transition Probability Matrix P for Page Rank

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

Fig. 4: Adjacency Matrix L for Directed Neighborhood Graph N

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 3 | 1 | 0 | 0 | 0 | 2 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 3 | 1 | 2 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 2 | 1 | 2 | 0 | 0 | 0 |
| 0 | 2 | 1 | 0 | 0 | 0 | 2 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Fig. 5: Authority Matrix LTL for HITS

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 3 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 3 | 0 | 0 |
| 0 | 0 | 0 | 0 | 2 | 1 | 0 | 3 | 1 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |

Fig. 6: HUB Matrix LLT for HITS

SALSA takes the idea of random walk from PageRank with the notion of hub/authority from HITS. SALSA calculates the hub and authority scores for web pages using two different Markov chains from bipartite undirected graph G. Conceptually, HITS follows mutual reinforcement approach while SALSA follows stochastic approach.

SALSA differs from HITS in the nature of graph that represents the root set E and the weighting scheme for the authority and the hub matrices. SALSA constructs two transition matrices H and A from adjacency matrix L corresponding to neighborhood graph N.

It obtains Lr from L after dividing each nonzero row by its row sum. Same way obtain Lc from L after dividing each nonzero column by its column sum. Lr may have row containing zero for all columns corresponding to page having no in-links.

Similarly, Lc may have the columns containing zero for all rows corresponding to page having no outlinks.

The hub matrix H is the nonzero rows and columns of $LrLc^T$ and the authority matrix A is the nonzero rows and columns of $Lc^T Lr$. Fig. 7 and fig. 8 shows authority matrix A and hub matrix H for our example web graph respectively. Hub vector is the principle eigenvector of hub matrix H. Similarly, authority vector is the principle eigenvector of authority matrix A.

| | | | | | | |
|---|---|---|---|---|---|---|
| 0.6111 | 0.2778 | 0 | 0 | 0.1111 | 0 | 0 |
| 0.4167 | 0.4167 | 0 | 0 | 0.1667 | 0 | 0 |
| 0 | 0 | 0.4167 | 0 | 0 | 0.4167 | 0.1667 |
| 0 | 0 | 0 | 0.5 | 0 | 0.5 | 0 |
| 0.1111 | 0.1111 | 0 | 0 | 0.7778 | 0 | 0 |
| 0 | 0 | 0.2778 | 0.1667 | 0 | 0.4444 | 0.1111 |
| 0 | 0 | 0.3333 | 0 | 0 | 0.3333 | 0.3333 |

Fig. 7: Hub Matrix H for SALSA

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.3333 | 0.3333 | 0 | 0 | 0 | 0 | 0 | 0.3333 |
| 0.1111 | 0.3889 | 0.1111 | 0 | 0 | 0 | 0.2778 | 0.1111 |
| 0 | 0.3333 | 0.3333 | 0 | 0 | 0 | 0.3333 | 0 |
| 0 | 0 | 0 | 0.6111 | 0.1111 | 0.2778 | 0 | 0 |
| 0 | 0 | 0 | 0.1667 | 0.6667 | 0.1667 | 0 | 0 |
| 0 | 0 | 0 | 0.4167 | 0.1667 | 0.4167 | 0 | 0 |
| 0 | 0.4167 | 0.1667 | 0 | 0 | 0 | 0.4167 | 0 |
| 0.3333 | 0.3333 | 0 | 0 | 0 | 0 | 0 | 0.3333 |

Fig. 8: Authority Matrix A for SALSA

## IV. Convergences and Uniqueness with Power Method

Once matrices are derived, the major issue is to calculate the principal eigenvector. As web matrices are large and sparse, power method is more suitable for solving the eigenvector problems. The method is described by the iteration as given in equation 1.

$$b_k+1 = \frac{Ab_k}{\| Ab_k \|} \quad (1)$$

The power iteration algorithm starts with a vector b0, which may be an approximation to the dominant eigenvector or a random vector. At every iteration, the vector bk is multiplied by the matrix A and then normalized. This subsequently converges to an eigenvector associated with the dominant eigenvalue. Even though power iteration can find only the dominant eigenvalue, it is very useful in some specific situations. For matrices that are well-conditioned and sparse as the web matrix, the power iteration method is more efficient than any other methods for finding the dominant eigenvector. This section highlights the issues regarding convergence of matrices and uniqueness of results that are associated with power method and also discusses - how link analysis algorithms deal with them?

For an irreducible stochastic matrix there is only one eigenvalue on the unit circle, all other eigenvalues have modulus strictly less than one [12]. This means that the power method applied to an irreducible stochastic matrix P is guaranteed to converge to the unique dominant eigenvector. It requires that the transition probability matrix must be stochastic. A row (or column) stochastic matrix is a square matrix for which, each of rows (or columns) consists of nonnegative real numbers, and the sum of all elements in the row (or column) is equal to 1.

Due to existence of dangling nodes, transition probability matrix P may have the corresponding rows having 0 for each column and the sum of all elements for that row are also 0. As a consequence, P does not remain a stochastic matrix and can cause a problem for rank sink [12]. Mathematically the problem is fixed by replacing all zero rows with $e^T/n$, where the vector $e^T$ is a raw vector of all ones. Brin and Page obtained the row stochastic matrix $\bar{P}$ using equation where e is vector of all one and x is a dangling node vector (i.e. if the page$_i$ is dangling node $x_i$ =1, otherwise $x_i$ = 0). The stochastic matrix $\bar{P}$ for transition probability matrix can be derived using equation 2.

$$\bar{P} = P + x\, e^T/n \quad (2)$$

Further, irreducibility is a require feature for transition probability matrix to ensure existence of unique stationary distribution vector i.e. PageRank vector. The transition probability matrix is irreducible if there is a path from each nodei to each nodej in graph. As web graph is not strongly connected, some nodei may not have hyperlink to nodej for some $\bar{P}_{ij}$. The elements of $\bar{P}_{ij}$ with value equal to 0 indicate the absence of a path (hyperlink) from pagei to page$_j$. This makes matrix $\bar{P}$ reducible. Brin and Page suggest [1] a way to cheat and alter the matrix, forcing irreducibility and hence guaranteeing existence and uniqueness of the ranking vector. They obtain the irreducible stochastic matrix $\bar{\bar{P}}$ as per equation 3 with $E = ee^T / n$ where e be the vector of all one and d is scalar such as $0 \le d \le 1$

$$\bar{\bar{P}} = d\, \bar{P} + (1 - d)\, E \quad (3)$$

For the above equation, d represents the fraction of the page's rank which is distributed among pages it links to i.e. d $\bar{P}$. Rest of rank (1 - d) is distributed among all pages uniformly such as (1-d) $ee^T$ / n. By adding this damping factor, we can make the

matrix irreducible. In the random surfer model, user gets bored (unhappy) with out-links on a given page and jumps randomly to any other page which is not linked via the current page. Generally the value of d is selected in the range of 0.8 to 0.9.

Langville and Meyer [12] referred this irreducible stochastic matrix $\bar{\bar{P}}$ as Google matrix. The principal eigenvector for Google matrix represent the PageRank vector. The matrix $\bar{\bar{P}}$ is guaranteed to give unique PageRank vector with defined convergence using power method.

Unlike PageRank, HITS and SALSA do not force irreducibility onto the graph. Although the matrices $L^TL$ and $LL^T$ are symmetric and non-negative, proper normalization makes convergence faster. To apply power method for HITS, we argument hub and authority matrices with some seed vector and perform iterative process until convergence. As matrices $L^TL$ and $LL^T$ may be reducible matrices, the principal eigenvectors are not guaranteed to be unique. The results of power method depend on the initial seed vectors given for the calculation of HITS.

Similarly for SALSA, the principal eigenvectors may not unique rather they depend on the initial seed vector given for the calculation of rank vectors. The convergence of SALSA is similar to that of HITS. Further, the presence of multiple connected components of bipartite graph G in SALSA make it computationally less complicated than that of HITS. The uniqueness of ranking vector in SALSA depends on the graph structure. If bipartite graph G is not connected, then the authority and the hub vectors are not unique. In such cases G has multiple irreducible components. Each component will coverage on its own and the final result will depended on weight that are assign to each component for calculating global scores.

## V. Conclusion

Google's PageRank algorithm is essentially a Markov chain over the graph of the web. The significance thing is that PageRank considers only forward links while HITS uses both forward link as well as back link. For HITS, the neighborhood graphs that are having reducible $L^TL$ give rise to repeated eigenvalues. HITS calculation on such graphs depends on the initial vector that was chosen to compute the solution. As a consequence, ranking vectors (i.e. results) of this algorithm are not always unique but depend on initial seed vector provided to the algorithm. A modification similar to the Google trick can also be applied to HITS. The damping factor makes the matrix irreducible; the smaller d is the faster the convergence. Although, if d is smaller, the effect of the true hyperlink structure of the web for determining the importance of web pages decreases. Slightly different values for d may produce considerable different PageRank.

PageRank only focuses on authority pages. As some directory like hub pages may have few incoming links, they accumulate low PageRank scores and seldom reported by any search engine which is using PageRank. Such good hubs are essentially important when user is learning a new topic. HITS and SALSA provides hub ranking along with authority ranking. HITS concentrates on TKC, while SALSA tends to mix the authorities of different communities. We proposed [13] the the variation of HITS to find communities on social network of web pages. As a consequence, the SALSA algorithm is better suited for very general search and found relatively useless when searching for a specific detail. SALSA, sometimes, favors small communities that are concerned with the topic being search, while HITS algorithm is better suited for a narrow but deep search.

## References


[1] Brin S., Page L., Motwami R., Terry W.,"The PageRank citation ranking: bringing order to the Web", Technical report, Stanford Digital Library Technologies Project 1998, pp. 1-17.

[2] Kleinberg J.,"Authoritative sources in a hyperlinked environment", Journal of the ACM, 1999, Volume 46, pp. 604-632.

[3] Lempel R., Moran S.,"The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC Effect", at 9th International WWW Conference 2000, [Online] Available: http://www9.org

[4] Brin S., Page L., "The anatomy of a large-scale hyper-textual Web search engine", Proceedings of Seventh International World-Wide Web Conference, 1998.

[5] Loren B.,"Ask.com Spins Teoma into ExpertRank", www.searchenginejournal.com/askcom-spins-teoma-into-expertrank/3001/, 2006.

[6] Chakrabarti S., Dom B., Gibson D., Kleinberg J., Raghavan P., Rajagopalan S.,"Automatic resource list compilation by analyzing hyperlink structure and associated text", Proceedings of 7th International World Wide Web Conference, 1998.

[7] Mandar R., Srinivasa S.,"Co-citations as citation endorsements and co-links as link endorsements", Journal of Information Science, Vol. 36, No.3, pp. 383-400, June 2010

[8] Kleinberg J., Furnkranz,"Web structure mining: Exploiting the Graph Structure of the World-Wide-Web", technical report at Australian Research Institutes for Artificial Intelligence, 2002.

[9] Gareth O. Roberts, Jeffrey S. Rosenthal,"Downweighting Tightly Knit Communities in World Wide Web Rankings", in Advances and Applications in Statistics (ADAS), 2003, Vol. 3, pp. 199-216.

[10] Langville, Meyer,"The Use of Linear Algebra by Web Search Engines", Bulletin of the International Linear Algebra Society, 2005, 33, pp. 2-6.

[11] Wai-Ki Ching, Mechael K. Ng,"Markov Chains Models, Algorithms and Applications", Springer International edition, pp. 49-58

[12] Langville, Meyer,"Deeper inside PageRank", Journal of Internet Mathematics, Vol. 1, pp. 335-380.

[13] Modi N.,"Finding Communities on Social Network of Web Pages Using Eigenvector Method", Journal of Science and Technology, Special Issue on Natural Language Processing and Data Mining 2012, pp. 89-97.