A Review of Algebraic Link Analysis Algorithms

Mini Singh Ahuja
Assistant Professor
Dept of Computer Science and Engg
GNDU Regional campus, Gurdaspur

Sumit Chhabra
Assistant Professor
Dept of computer science and Applications
Khalsa College for Women, Amritsar

ABSTRACT

The World Wide Web is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them by using hyperlinks. Navigation is the process through which the users can achieve their purposes in using Web site, such as to find the information that they need or to complete the transactions that they want to do. Web mining is the application of data mining techniques to extract knowledge from Web data, where at least one of structure (hyperlink) or usage (Web log) data is used in the mining process (with or without other types of Web data). In this paper we have briefly discussed the web mining technique with major stress to the link analysis algorithms.

Keyword

Data mining, Hyperlink, Information retrieval, Web becons, Web graph, Web log.

1 INTRODUCTION

The concept of WWW was given in 1989 by Tim Berners-Lee while at CERN (the European Laboratory for Particle Physics). Today, www is a popular and interactive medium to interchange information. The Web is an ever growing repository of large amount of information, spread across several servers in a complicated network. To be able to cope with the abundance of available information, users of the Web need assistance of intelligent software agents (often called *softbots*) for finding, sorting, and filtering the available information so it is a good area of research for data mining. Since the web information is different from traditional information containers such as databases, volume topic-coherences web mining is needed. Web mining research has come from many several search communities such as database, information retrieval and artificial intelligence (machine learning and natural language processing).

2 WEB MINIG

Web mining is a technique which uses data mining techniques to discover and extract information from web documents and services. Web mining aims at finding and extracting relevant information that is hidden in Web-related data, in particular in (hyper-) text documents published on the Web. Web mining task is divided into the following sub tasks:

- Resource finding: finding out the required web documents.
- Information selection and preprocessing: automatically selecting and preprocessing specific information from web resources.
- Generalization: discovering general patterns at websites and across multiple sites.
- Analysis: validating and interpreting mind patterns.

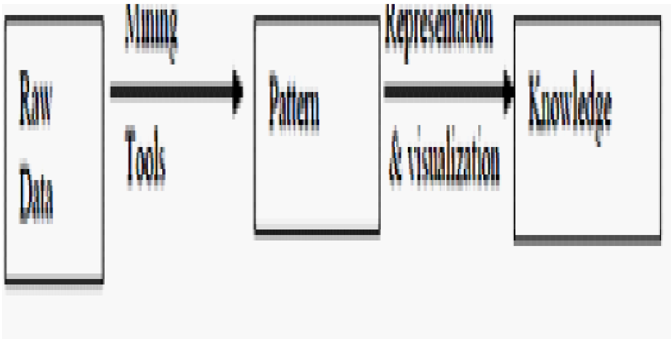


Fig 1: web mining process

Like data mining, web mining is a multi-disciplinary effort that draws techniques from fields like information retrieval, statistics, machine learning, natural language processing, and others.
Web mining is divided into three sub parts:

- Web content mining: application of data mining techniques to unstructured and semi-structured text typically HTML documents.
- Web structure mining: use of hyperlink structure of the web for information.
- Web usage mining: analysis of user behavior while interacting with web server.

A. Web content mining

It is a process in which we extract useful information from the contents of web pages. These contents can be text documents or collection of multiple documents such as images, videos, audio which are included in the web pages. Web content mining can be differentiated in two points of view: the agent based approach and database approach.

In the agent based approach we aim to improve the information finding and filtering. The database approach aims on modeling the data on the web into more structured form.

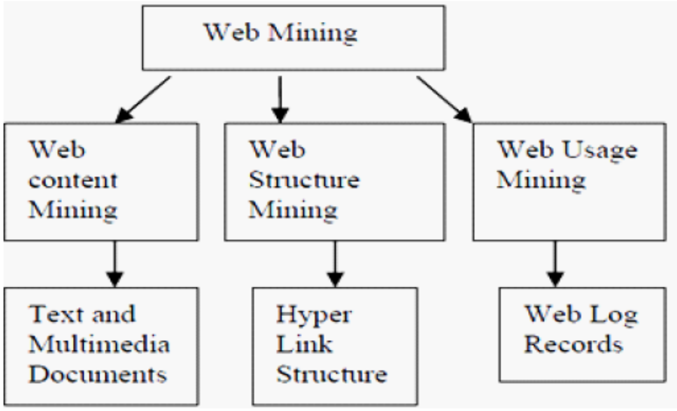


Fig 2: web mining categories

B. Web structure mining

In it we study the link structure of the web pages. The goal is to study the web graph. Web graph is a graphical view of the website. Web graph is a directed graph G= (V, E) where V is the set of nodes representing the web pages and E is the set of edges representing links between web pages. S is the start node of the graph which is the home page of the website. The directed graph should satisfy the condition that all nodes v in V are reachable from the home page.

C. Web usage mining

It focuses on techniques that predict the behavior of users while they are working on the www. It collects the data from the web logs, web becons, java script tags or packet sniffers. Out of these web logs and Java script tags are most commonly used. Web logs are originally developed to capture the errors generated by web servers but now they are also used to capture more data which web analyst requires. Web usage mining consists of three phases: preprocessing, pattern discovery and pattern analysis.

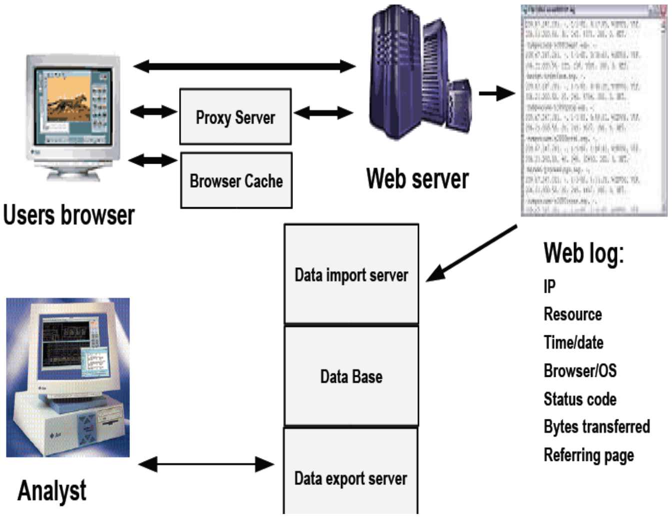


Fig 3: web logs

3 WEB STRUCTURE MINING

Web structure mining deals with the structure of the hyperlinks within the web itself. In the WWW there are web pages and links. Web structure mining has led to the study of social networks and citation analysis. [3] has discussed about Small World Network and Scale Invariance in the structure of the Web graph. Link analysis algorithms are classified into two main categories:

- Algebraic methods: HITS, PageRank, SALSA, hybrid algorithms
- Probabilistic method: CHITS, Bayesian algorithm

Algorithms such as Kleinberg's HITS algorithm, the PageRank algorithm of Brin and Page, and the SALSA algorithm of Lempel and Moran use the link structure of a network to assign weights to each page in the network. These algorithms find a dominant eigenvector of a non-negative matrix that describes the link structure of the given network and use the entries of this eigenvector as the page weights. Probabilistic methods use probabilistic techniques to estimate the rank of pages on a specific topic. In this paper we will study the algebraic methods as they are more popular.

A. HITS

The HITS (Hypertext-Induced Topic Search) algorithm was developed by J. Kleinberg , is now a part of the CLEVER Searching project of the IBM Almaden Research Center . A HIT is an iterative algorithm based on the linkage of the documents on the web. The HITS algorithm takes the result set of a query as input, expands the result set to a base set by adding the immediate neighbors of each result, and constructs a neighborhood graph from the base set by including all edges in the full web graph that connect base set vertices. In HITS Kleinbger has classified two kinds of pages from web hyperlink structure: authorities and hubs. Authorities are the pages that contain a lot of information about a particular topic. A hub is a page that link too many related authorities. A HIT associates a non negative authority weight $X^{<P>}$ and a non negative hub weight $Y^{<P>.}$ The weights are normalized so that their squares sum to 1. According to Kleinbger if p points to many pages with large X values, then it should get large Y values. And if p is pointed to a many pages with large Y values then it should get large X value.

$$X^{<P>} \longleftarrow \sum Y^{<P>}$$

$$Y^{<P>} \longleftarrow \sum X^{<P>}$$

The HITS algorithm first forms N by N adjacent matrix A whose (i,j) element is 1 if page i links to page j and 0 otherwise. Then it iterates the equation 1.

$$a_i^{(t+1)} = \sum_{\{j:j\to i\}} h_j^{(t)};\ h_i^{(t+1)} = \sum_{\{j:i\to j\}} a_j^{(t+1)}$$

…… .(1)

(Where "i --→j" means page i is linked to page j) to obtain a fixed points.

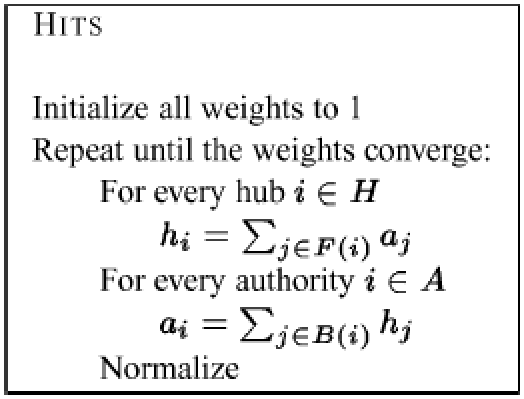
a* = limit t→□□□ a(t) and h *=limit t→□□□ h(t)

The above equation can also be written as:

$a^{(t+1)} = A^T h^{(t)} = (A^T A) a^{(t)}$ ……………(2)

$h^{(t+1)} = A a^{(t+1)} = (A A^T) h^{(t)}$ ……………(3)

When the iterations are initialized with the vector of ones [1,……….1]T, this is the power method of obtaining the principal eigenvector of a matrix.



```
HITS

Initialize all weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \sum_{j \in F(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Fig 4: HITS algorithm

Drawback of HITS:

- *Mutually reinforced relationships between hosts (TKC effect):* we can get even wrong information about good hubs and authorities. e,g sometimes a set of documents on one host points to a single document on a second host or a single document on one host point to set of set of documents on other host.
- *Automatically generated links by tools.*
- *Non relevant nodes:* page may point to non relevant page.

B. PAGE RANK

PageRank Algorithm developed by Brin and Page can provide an overall importance measure of the Web using the structure of the web, so it has a key role in Search Engine. Google uses Page Rank method to measure the importance of a web page. Page Rank algorithm counts in-links and propagates the ranking through links. A page has a high rank, if the sum of the ranks of its in-links is high. Page Rank of a page E is given as :

PR(E) = (1-d)+d(PR(T1)/C(T1)+…PR(Tn)/C(Tn)) …………(4)

where :
d : is the damping factor between 0 and 1 (probability at each page the random user will get bored and request another page)
PR(E) : Page Rank of page E.
PR(Ti) : Page Rank of the page Ti which is linked to page E.
C(Ti) : no of links going out of a page Ti

Page Rank Matrix Mij is

M ij = 1/ C(Ti) if j links to i
0 otherwise

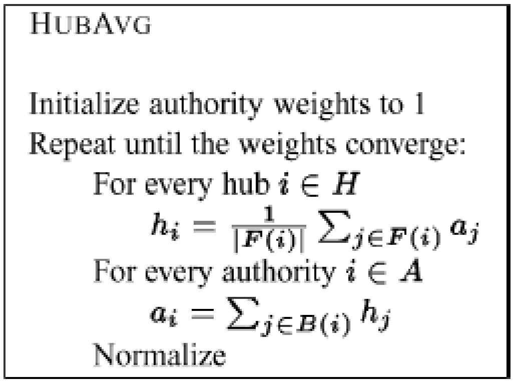
Page rank can be calculated using iterative algorithm and correspond to the principal eigen vector of the normalized link matrix of the web. Kleinbergs algorithm (HITS) is query dependent. Hubs and authorities depend heavily on the subject we are interested in whereas PageRank is query-independent In PageRank all pages on the Web are ranked on their intrinsic value, regardless of topic.

C. SALSA.

The SALSA (Stochastic Approach for Link Structure Analysis) algorithm was developed by Lempel and Moran. It combines the random walk idea of Page Rank with the hub/authority idea of HITS. SALSA is a link-based ranking algorithm that takes the result set of a query as input, extends the set to include additional neighboring documents in the web graph, and performs a random walk on the induced sub graph. Recent large-scale evaluations have shown that some query-dependent link-based ranking algorithms like the SALSA algorithm are more effective than well-known query-independent algorithms such as Page Rank [11, 12]. SALSA performs two random walks on web pages: a random walk by following a backward link and then forward link alternately and another one by following a forward link and then backward link alternately. In SALSA, the TKC effect is overcome through random walks on a bipartite Web graph for identifying authorities and hub. From the web graph G, a bipartite undirected graph T is constructed by building the subset V_a of all the nodes with positive in-degree (authorities) and the subset V_h of all the nodes with positive out-degree (hubs). After selecting V_a and V_h these become the nodes of T. Since some nodes may have both positive in-degree and out-degree the number m of nodes of T satisfies $m \leq 2n$, where n is the number of nodes of G. The (undirected) edges of T are defined from G as follows: if G has a link from i to j, then we put an edge between the nodes corresponding to i in V_h and j in V_a. The algorithm corresponds to a two-step random walk on the graph T. The SALSA authority vector is the stationary probability distribution of the authority walk, and the hub vector is the stationary probability distribution of the hub walk.

D. Hub-Averaging-Kleinberg Algorithm

It is a hybrid of the Kleinberg and SALSA algorithm. It does the authority rating updates just like Kleinberg (giving each authority a rating equal to the sum of the hub ratings of all the pages that link to it). However, it does the hub rating updates by giving each hub a rating equal to the average of the authority ratings of all the pages that it links to. Consequently, a hub is better if it links to only good authorities, rather than linking to both good and bad authorities. The authority weights for the HUBAVG algorithm converge to the principal right eigenvector of the matrix *MHA* = *WTWr* .



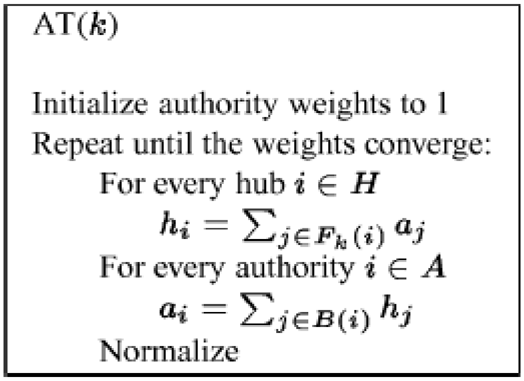
```
HUBAVG

Initialize authority weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \frac{1}{|F(i)|} \sum_{j \in F(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Fig 5: Hub-Averaging-Kleinberg Algorithm

E. The Authority Threshold (AT(k)) Algorithm

There are some situations where even Hub-Averaging-Kleinberg Algorithm can fail .Such situations may arise in practice when a node is simultaneously a strong hub on one topic and a weak hub on another topic. Such hubs are penalized by the HUBAVG algorithm. The Authority-Threshold, AT(*k*), algorithm provides the solution to this situation. It sets the hub weight of node *i* to be the sum of the *k* largest authority weights of the authorities pointed to by node *i*. This is equivalent to saying that a node is a good hub if it points to *at least k* good authorities. The value of *k* is passed as a parameter to the algorithm.



```
AT($k$)

Initialize authority weights to 1
Repeat until the weights converge:
    For every hub $i \in H$
        $h_i = \sum_{j \in F_k(i)} a_j$
    For every authority $i \in A$
        $a_i = \sum_{j \in B(i)} h_j$
    Normalize
```

Fig 6: The *Authority-Threshold*, AT(*k*), algorithm

4 CONCLUSION

In this paper we have reviewed the area of web mining with focus on web structure mining. We have also discussed the link analysis algorithms. Link analysis algorithms are divided into two categories: algebraic and other probabilistic method. Algebraic methods are most popularly used. Since web mining is a vast area, so we hope this paper will provide the starting point for researchers to identify opportunities for further research.

REFERENCES

[1] S. Suman and S. Aggarwal, “WebMine: A tool to uncover the web”, 2005

[2] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal, “Stochastic models for the web graph”, *In Proc. 41st FOCS*, pages 57–65, 2000

[3] S. Brin and L. Page. The anatomy of a large-scale hypertextual (Web) search engine. In *The Seventh International World Wide Web Conference*, 1998. .

[4] Stable algorithms for link analysis, in Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 2001.

[5] M. Henzinger, Link analysis in web information retrieval, IEEE Data Engineering Bulletin, 23 (2000), pp. 3{8

[6] S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg, Mining the link structure of the World Wide Web, IEEE Computer, 32 (1999),

[7] R. Lempel and S. Moran, The stochastic approach for link-structure analysis (SALSA) and the TKC e_ect, in Proceedings of the Ninth International Conference on the World Wide Web, May 2000

[8] Otter, M. and Johnson, H. Lost in hyperspace: metrics and mental models. Interacting with Computers, 13(1), 1-40, 2000

[9] Botafogo, R.A., Rivlin, E. & Shneiderman, B. (1992). Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics. *ACM Trans. on Information Systems 10 (2)*, 142-180.

[10] Srivastava, J., Cooley, R., Deshpande, M. & Tan, P.N. (2000) Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *SIGKDD Explorations 1 (2)*, 1-12

[11] T. Bray, (1996), “ Measuring the web”, In 5th International world Wide Web Conference.Paris,France