Extracting Patterns in Social Network by Means of Link Mining

Hnin Wut Yee Oo,Than Htike Aung
Computer University,(Hinthada)
hninpaper88@gmail.com

Abstract

The World Wide Web serves as a huge, widely distributed, global information service center for news, advertisements, consumer information, financial management, education, government, e-commerce, and many other information services. The Web contains a rich and dynamic collection of hyperlink information which can access usage information and provide rich sources for data mining. It is a highly dynamic information source and web service centers update their web pages regularly. Linkage information and access are also updated frequently. This framework is the distillation of broad search topics, through the discovery of "authoritative" information sources on topics. This using hubs, called HITS (Hyperlink-Induced Topic Search) to find authoritative pages based on the relationship between a set of relevant authoritative pages and the set of "hub pages" that join them together in the link structure. This formulation has connections to the eigenvectors of certain matrices associated with the link graph. This HITS algorithm provides surprisingly good search results for a wide range of query.

Keywords

Link analysis, HITS algorithm, Kleinberg algorithm, hubs, authorities, networks

1. Introduction

Electronic communities evolve around a given area of activity or topic of interest. The World Wide Web serves as a huge, widely distributed, global information service center for variety of information services. The Web also contains rich and dynamic collection of hyperlink information and Web page access and usage information, providing rich sources for data mining. Web mining is an even more challenging task that searches for Web access patterns, Web Structures, and regularity and dynamic of Web contents.

The web structure mining is to discover useful knowledge from the structure of hyperlinks, and web content mining aims to extract useful information or knowledge from web page contents. The complexity of web pages is far greater than that of any traditional text document collection.

The network structure of a hyperlinked environment can be a rich source of information about the content of the environment. In this work, develop a set of algorithmic tools for extracting information from the link structures of such environments, and report on experiments that demonstrate their effectiveness in a variety of contexts on the World Wide Web (www). In particular, focus on the use of links for analyzing the collection of pages relevant to a broad search topic, and for discovering the most "authoritative" pages on such topics.

2. Related Work

Jon Kleinberg, a professor in the Department of Computer Science at Cornell came up with his own solution to the Web Search problem [5]. He developed an algorithm that made use of the link structure of the web in order to discover and rank pages relevant for a particular topic. HITS(hyperlink-induced topic search) is now part of the Ask search engine (www.Ask.com). Link analysis was also used for a search-by-example approach to searching: given one relevant page find pages related to it. Kleinberg proposed using the HITS algorithm for this problem and Dean and Henzinger show that both the HITS algorithm and a simple algorithm on the co-citation perform very well[6]. The idea is that frequent co-citation is a good indication of relatedness and thus the edges with high weight in the co-citation graph tend to connect nodes with are related.

The analysis of link structures with the goal of understanding their social or informational organization has been an issue in a number of overlapping areas.

3. HITS Algorithms

HITS stands for Hyperlink Induced Topic Search. Unlike PageRank which is a static ranking algorithm, HITS is a search query dependent. When a user issues a search query, HITS first expands the list of relevant pages returned by a search engine and then produces two rankings of the expanded set of pages, authority ranking and hub ranking.

HITS computes two different scores for each page of a Web community (or of the whole WWW): a hub score and an authority score. The algorithm can be split into the following steps:

1. Select a starting set of pages. The algorithm is usually focused on a subgraph of the Web, which contains pages on a given topic. This base set can be obtained by sending a specific query to a search engine and selecting the most important results.
2. Extend the starting set of pages, because some of the authoritative sources might not be in the above set.
3. Calculate the scores. Compute an authority measure and a hub measure for each page.

An ***authority*** is a page with many in-links.

- The page may have good or authoritative content on some topic.

A ***hub*** is a page with many out-links.

- The page serves as an organizer of the information on a particular topic and points to many good authority pages on the topic.

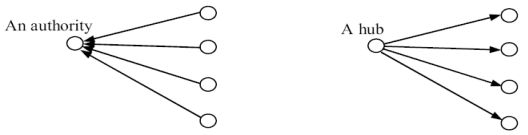


Figure 1: An authority page and hub page

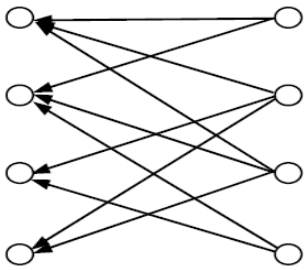


Figure 2: A densely linked set of authorities and hubs

The key idea of HITS is that a good hub points to many good authorities and a good authority is pointed to by many good hubs.

Authorities and hubs have a mutual reinforcement relationship.

3.1 Information extraction

Given a broad search query q, HITS collects a set of pages as follows:

1. It sends the query q to a search engine system. It then collects t highest ranked pages. This set is called the root set W.
2. It then grows W by including any page pointed to by a page in W and any page that points to a page in W. This gives a larger set called S, called the base set.

*The algorithm restricts its size by allowing each page in W to bring at most k pages.

HITS then works on the pages in S set, assigns every page in S an authority score and a hub score.

Let the number of pages to be studied be n , use

G = (V, E) to denote the (directed) link graph of S.

V is the set of pages (or nodes)

E is the set of directed edges (or links).

L is to denote the adjacency matrix of the graph.

$$L_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Let the authority score of the page i be a(i), and the hub score of page i be h(i). The mutual reinforcing relationship of the two scores is represented as follows:

$$a(i) = \sum_{(j,i)\in E} h(j)$$

$$h(i) = \sum_{(i,j)\in E} a(j)$$

Writing them in the matrix form, using a to denote the column vector with all the authority scores, and use h to denote the column vector with all the hub scores.

$a = (a(1), a(2), \ldots, a(n))^T$,
$h = (h(1), h(2), \ldots, h(n))^T$.

The computation of authority scores and hub scores is basically the same as the computation of the PageRank scores using the power iteration method.

a_k and h_k to denote authority and hub scores at the k^{th} iteration, the iterative processes for generating the final solutions are

$$\boldsymbol{a}_k = \boldsymbol{L}^T \boldsymbol{L} \boldsymbol{a}_{k-1}$$

$$\boldsymbol{h}_k = \boldsymbol{L}\boldsymbol{L}^T \boldsymbol{h}_{k-1}$$

starting with

$$\boldsymbol{a}_0 = \boldsymbol{h}_0 = (1, 1, \ldots, 1).$$

After each iteration, the values are also normalized (to keep them small) so that ..

$$\sum_{i=1}^{n} a(i) = 1$$

$$\sum_{i=1}^{n} h(i) = 1$$

HITS-Iterate(G)
$\boldsymbol{a}_0 \leftarrow \boldsymbol{h}_0 \leftarrow (1, 1, \ldots, 1)$;
$k \leftarrow 1$
Repeat
$\boldsymbol{a}_k \leftarrow \boldsymbol{L}^T \boldsymbol{L} \boldsymbol{a}_{k-1}$;
$\boldsymbol{h}_k \leftarrow \boldsymbol{L}\boldsymbol{L}^T \boldsymbol{h}_{k-1}$;
$\boldsymbol{a}_k \leftarrow \boldsymbol{a}_k / ||\boldsymbol{a}_k||_1$; // normalization
$\boldsymbol{h}_k \leftarrow \boldsymbol{h}_k / ||\boldsymbol{h}_k||_1$; // normalization
$k \leftarrow k + 1$;
until $||\boldsymbol{a}_k - \boldsymbol{a}_{k-1}||_1 < \varepsilon_a$ and $||\boldsymbol{h}_k - \boldsymbol{h}_{k-1}||_1 < \varepsilon_h$;
return $\boldsymbol{a}_k$ and $\boldsymbol{h}_k$

The iteration ends after the 1-norms of the residual vectors are less than some thresholds. *Hence, the*

algorithm finds the principal eigenvectors at "equilibrium" as in PageRank. The pages with large authority and hub scores are better authorities and hubs respectively. HITS will select a few top ranked pages as authorities and hubs, and return them to the user.

The HITS algorithm finds the principal eigenvectors, which in a sense represent the most densely connected authorities and hubs in the graph *G* defined by a query.

However, in some cases, which may also be interested in finding several densely linked collections of hubs and authorities among the same base set of pages.

Each of such collections could potentially be relevant to the query topic, but they could be well-separated from one another in the graph *G*. For example,

The query string may be ambiguous with several very different meanings, e.g., "jaguar", which could be a cat or a car.

The query string may refer to a highly polarized issue, involving groups that are not likely to link to one another, e.g. "abortion".

In each of these examples, the relevant pages can be naturally grouped into several clusters, also called communities.

The smaller clusters (or communities), which are also represented by bipartite subgraphs, can be found by computing non-principal eigenvectors.

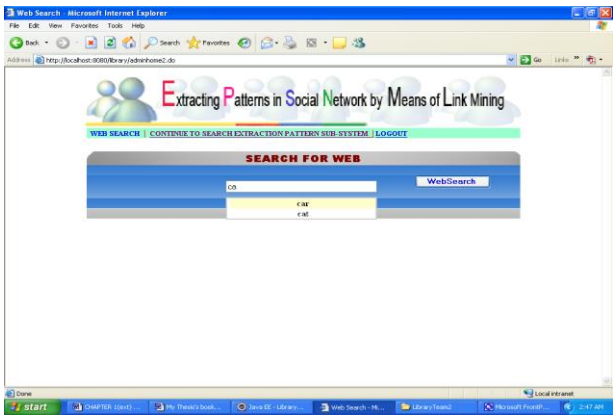
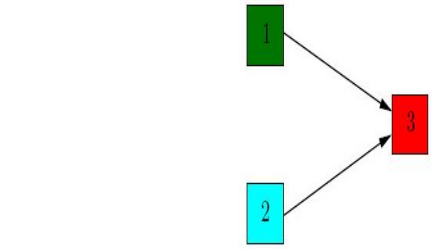


Figure 3: Search for car



The adjacency matrix of the graph is $A = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$, with transpose $A^t = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$. Assume the initial hub weight vector is:

$$u = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

We compute the authority weight vector by:

$$v = A^t \cdot u = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

Then, the updated hub weight is:

$$u = A \cdot v = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix}.$$

Figure 4: Updated authority and hub weight vector

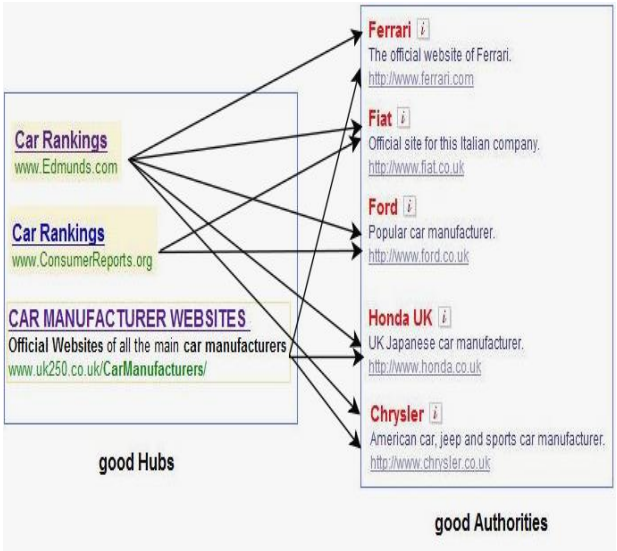


Figure 5: Result of Top Ranked Page Lists

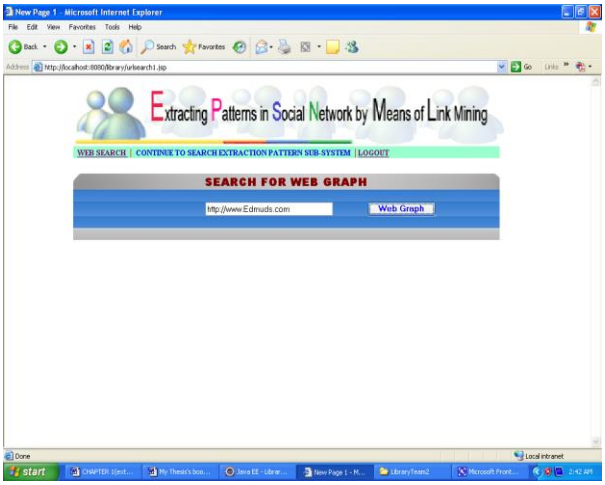


Figure 6: Calculate for Experimental Result

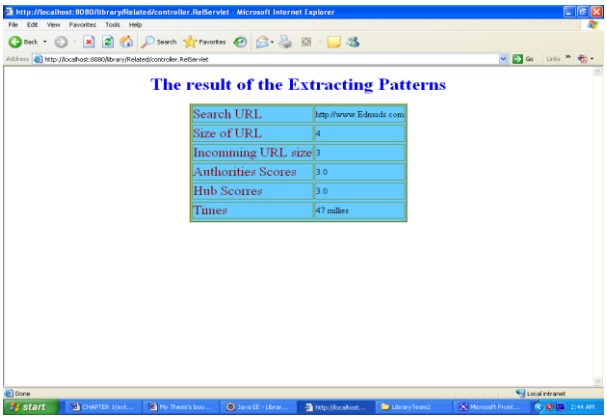


Figure 7: Experimental Resulted of Top Ranked page

4. System Design

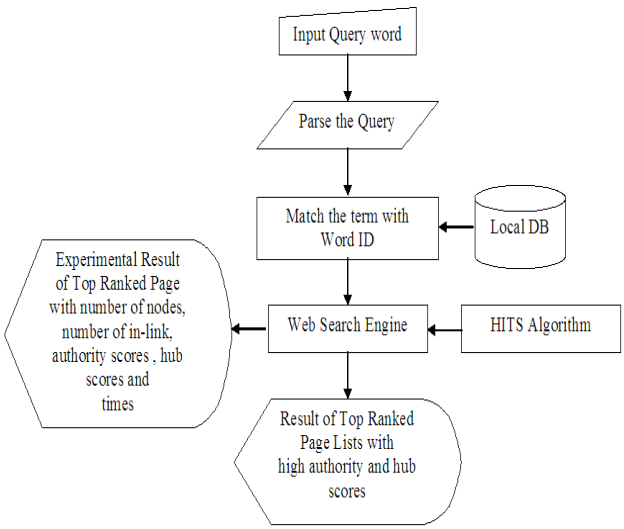


Figure 8: System flow diagram

5. Implementation of the System

Step1: User input with word or URL address to search more relevant pages and to calculate experimental result.

Step2: Parse the query word (read the user input sentence and parse into specific keyword).

Step3: Match the term with word id from Database.

word_id	word
1	artificial intelligence
2	censorship
3	java

document_id	url	page
1	http://wwwaaai.org	American Association for Artificial Intelligence
2	http://www.ai.mit.edu	MIT AI laboratory

document_id	word_id	word
1	1	artificial intelligence
2	1	artificial intelligence

Step4:Web Search engine returned Result of Top Ranked Page and Experimental Result of Top Ranked Page. HITS computes the hub score and an authority score to get the quality and relevant content.

Step5: Displayed the Result of Top Ranked Page Lists with high authority and hub scores provided with the input query term or word. Displayed the Experimental Result of Top Ranked Page with number of nodes, number of in-links authority scores, hub scores and times to find the experimental result by using HITS Algorithm provided with input URL address.

5. Strengths and Weaknesses of HITS

The main strength of HITS is its ability to rank pages according to the query topic, which may be able to provide more relevant authority and hub pages.

The ranking may also be combined with information retrieval based rankings.

HITS has several disadvantages:

1. It does not have the anti-spam capability of PageRank.
2. The query time evaluation is also a major drawback.

6. Conclusion

The nature of this system presents Web structures can be treated as a part of Web contents mining. Finding the Web pages relating to a given topic, the pages retrieved will be high quality, or authoritative on the topic. The Web consists of not only pages but also hyperlinks

pointing from one page to another. These hyperlinks contain an enormous amount of latent human annotation that help to automatically infer the notion authority. Therefore, the tremendous amount of Web linkage information provides rich information about the relevance, quality, and structure of the Web's contents, and thus is a rich source for Web mining. By using this approach, the system can be mined Web's Link Structures to Identify Authoritative Web pages.

References

[1] A. Bonato, A Course on The Web Graph, AMS-AARMS, Graduate Studies in Mathematics v. 89, 2008.http://www.math.ryerson.ca/~abonato/webgraph.html

[2] A. Sime (University of New York College at Brockport, USA), "Web Mining: Applications and Techniques", Idea Group Publishing, Hershey, London, Melbourne, Singapore, ISBN 159140414-2,2004. (WWW1998).http://infolab.stanford.edu/pub/papers/google.pdf

[3] D. Gibson, J. Kleinberg, P. Raghavan, "Inferring Web Communities from Link Topology," Proc. 9th ACM Conference on Hypertext and Hypermedia, 1998

[4] J. Han and M. Kamber (University of Illinois at Urbana-Champaign), "Data Mining: Concepts and Techniques" (2nd Edition), Morgan Kaufmann Publishers, San Francisco, CA, USA, An imprint of Elsevier, ISBN 978-81-312-0535-8, 2006.

[5] J. Kleinberg, Authoritative sources in a hyperlinked environment Discrete Algorithms,1998.http://www.cs.cornell.edu/home/kleinber/auth.pdf.

[6] K. Dean and M. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In Research and Development in Information Retrieval, 1998.

[7] M. Brin, G. Stuck, Introduction to Dynamical Systems. Cambridge University Press, 2002.

[8] S. Brin, L. Page, the Anatomy of a Large-Scale Hypertextual Web Search Engine, Seventh International World-Wide Web Conference (WWW 1998). http://infolab.stanford.edu/pub/papers/google.pdf

[9] S. Chakrabarti, B. Dom, D. Gibson, S.R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins, "Experiments in Topic Distillation," ACM SIGIR Workshop on Hypertext Information Retrieval on the Web, 1998.