May 2001

Structure of the Internet?

Ah Chung Tsoi
University of Wollongong, act@uow.edu.au

Structure of the Internet?

Abstract

We consider a major component in the design of an Internet search engine, viz., how the relevance of a Web page can be determined. A number of methods are described. A number of design issues related to search engines are also discussed.

Keywords

Internet, information analysis, search engines

Disciplines

Physical Sciences and Mathematics

Publication Details

This article was originally published as: Tsoi, AC, Structure of the Internet?, Proceedings of the 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, Hong Kong, 2-4 May 2001, 449-452. Copyright IEEE 2001.

Structure of the Internet ?

Ah Chung Tsoi
University of Wollongong
Northfields Avenue
Wollongong, NSW 2522
Australia

Abstract

In this paper, we will consider a major component of the design of an internet search engine: viz., how the relevance of a web page can be determined. A number of methods are described. A number of design issues related to search engines are also discussed.

Keywords: Search engine, latent semantic analysis, authorities and hubs, probabilistic latent semantic analysis, probabilistic authority and hub analysis.

1 Introduction

The world wide web has been expanding at a tremendous rate. In July 1997, it was estimated that the web contains about 300 million pages, while in late 2000, it is estimated that the web contains over 1 billion pages.

Characteristics of information contained in the web:

1. Heterogenity: the web is heterogenic in its information. It contains information about a large variety of topics, ranging from advertisement of wares from companies to preprints of papers from researchers.
2. Scale. The web contains over 1 billion pages, and is still growing at a tremendous rate. It is estimated that the volume of information is in the region of hundreds of gigabytes.
3. Dynamic nature of the information. The information contained on the web is changing all the time. It is reported that on an average, a web page stays the same on the web for less than 6 months.

Faced with this scale of complexity, information retrieval poses a challenging question. Put simply, the issue of information retrieval is: how to retrieve information from the web "precisely", and "efficiently".

To assist users to "navigate" through the information contained on the internet, a specialized class of software, known commonly as "search engines" has sprung into being. A central question in the design of search engines is how to determine the relevance of a given web page. In this paper, we will indicate the various approaches which have been taken in the consideration of this problem, and will indicate future challenges in the design (Section 2) and refinement in the underlying algorithms (Section 3).

2 Determination of the relevance of a web page

In this section, we will consider one of the main issues in the design of a search engine, viz., how do we determine the relevance of a particular page.

Intuitively, the relevance of a page depends on two factors:

1. Its link structure. This relates to the way in which the page is linked with other pages.
2. Its context. This relates the particular page in the context in which it occurs.

As may be anticipated, it is relatively easier to consider the link structure of a page, as it entails an analysis of the way in which the page is linked to other pages through a graph structure. However, it is relatively more difficult to determine the context in which a page occurs in relation to other pages.

2.1 Feature extraction

There are two feature extraction processes, one corresponding to the link analysis while the other is associated with the analysis of the context.

2.1.1 Feature extraction for link analysis

For link analysis, the web pages are essentially considered as a graph structure.

Consider a page A. If we consider a number of pages relevant to a particular query, e.g., obtained from a commonly used search engine, like AltaVista. This set of web pages is called the "root set". Secondly, the root set of web pages is augmented by pages which link to pages in the root set, and pages which are linked to from pages within the root set. This expanded set of web pages is called the "base set". Assuming that there are a total of N pages in the base set. Construct the $N \times N$ adjacency matrix of the pages in the base set as follows: $A_{ij} = k$ if there are k links from page i to page j. Put it differently, A_{ij} indicates that there are k citations in page i of page j. Otherwise $A_{ij} = 0$. Typically $k = 1$.

2.1.2 Feature extraction for context analysis

This is more difficult as there are many ways in which context can be modelled. A simple method is to construct the term matrix [7]. Assuming that there are N documents in the collection. Assume further

that there are M terms in the dictionary. Typically $N >> M$. Then it is possible to construct a $N \times M$ term matrix B whose elements are given by the occurrence of the term t_i in document d_j. In other words, $B_{ij} = k$, if in document d_j, there are k occurrence of the term t_i.

2.2 PageRank

Brin and Page [2] introduced this method in their design of the "google" search engine.

Consider the following: assuming that there exists a page d_i. This page has d_ℓ, $\ell = 1, 2, \ldots, n_i$ pages pointed to it. If we further assume that each page d_ℓ has n_ℓ, $\ell = 1, 2, \ldots, n_i$ links going out of it. Then the PageRank of page d_i is given by:

$$P_{d_i} = (1 - \alpha) + \alpha \sum_{\ell=1}^{n_i} \frac{P_{d_\ell}}{n_\ell} \quad (1)$$

where α is called a damping factor. P_{d_i} denotes the PageRank of each page d_ℓ, $\ell = 1, 2, \ldots, n_i$.

Intuitively, PageRank can be considered as a model of user behaviour. Assuming that we have a "random surfer". The random surfer chooses a particular page, and keeps on clicking the links never hitting the back button. Eventually the random surfer gets bored of this process, and decides to choose another page at random and starts the whole process again. The probability of the random surfer in visiting a particular page is its PageRank. The damping factor is the probability that the random surfer will get bored at that particular page, and requests another random page.

2.3 HITS algorithm

The PageRank algorithm uses only the concept of the "authority" of a page, in that there are a number of links pointing to it. In this algorithm [10] the concept of "authority" is extended to include its dual: the "hub". A good "hub" is a page in which it points to "authorative" pages. A page is "authorative" if it is pointed to by a number of good "hubs". On the other hand, a page is a good "hub" if it points to a number of highly "authorative" pages.

If we assume that the authority weights of each page in the base set are collected in a vector $\mathbf{x}^T = [\ x_1 \ \ x_2 \ \ \ldots \ \ x_N\ ]^T$, where T denotes the transpose of a vector or matrix. Similarly the hub weights of each page in the base set is collected in a vector $\mathbf{y}^T = [\ y_1 \ \ y_2 \ \ \ldots \ \ y_N\ ]^T$. Then the authority and the hub weights are computed using the adjacency matrix A in the following iterative manner:

$$\mathbf{x}^{i+1} = A^T \mathbf{y}^i \quad (2)$$

$$\mathbf{y}^{i+1} = A \mathbf{x}^i \quad (3)$$

where $\mathbf{x}^i$ and $\mathbf{y}^i$ denote respectively the i th iterate of the process. In addition, after each step, a normalization step needs to be carried out so that the $\mathbf{x}^{i+1}$ and $\mathbf{y}^{i+1}$ are unit vectors. This process is carried out until it converges.

This method is often referred to as the HITS (Hypertext Induced Topic Selection) algorithm.

The HITS method has been extended in various manner, see e.g., [11].

2.4 PHITS algorithm

This is the formulation of the HITS algorithm in a probabilistic setting [4]. A document d_j, $j = 1, 2, \ldots, N$ in the document collection D is generated with some probability $P(d_j)$. The factor or topic $z_k \in Z$ associated with the document d_j is given by $P(z_k|d_j)$. Given the factor, the citation $c_i \in C$ are generated by the following probabilistic model:

$$P(c_i, d_j) = P(d_j)P(c_i|d_j) \quad (4)$$

$$= P(d_j) \sum_k P(c_i|z_k)P(z_k|d_j) \quad (5)$$

The total log likelihood of the observation is given by

$$L = \sum_i \sum_j \log P(c_i, d_j) \quad (6)$$

The process of finding a model which explains a set of observations reduces to the problem of finding values of $P(d_j)$, $P(z_k|d_j)$, and $P(c_i|z_k)$ such that the log likelihood L is maximized. This can be obtained using an expectation maximization algorithm as follows:

E step : compute the expectation of $P(z_k|d_j, c_i)$:

$$P(z_k|d_j, c_i) = \frac{P(z_k)P(d_j|z_k)P(c_i|z_k)}{\sum_{k'} P(z_{k'})P(d_j|z_{k'})P(c_i|z_{k'})} \quad (7)$$

for each $z_k \in Z$, $d_j \in D$ and $c_i \in C$.

M step : re–estimate $P(z_k)$, $P(c_i|z_k)$, and $P(d_j|z_k)$ to maximize the log likelihood function L:

$$P(z_k) = \frac{\sum_j \sum_i P(z_k|d_j, c_i)}{\sum_{k'} \sum_j \sum_i P(z_{k'}|d_j, c_i)} \quad (8)$$

$$P(d_j|z_k) = \frac{\sum_i P(z_k|d_j, c_i)}{\sum_{j'} \sum_i P(z_k|d_{j'}, c_i)} \quad (9)$$

$$P(c_i|z_k) = \frac{\sum_j P(z_k|d_j, c_i)}{\sum_j \sum_{i'} P(z_k|d_j, c_{i'})} \quad (10)$$

2.5 Latent Semantic Algorithm

This is a very popular method used in bibliometric literature. The latent semantic analysis (LSA) [7] uses a singular value decomposition of the $N \times M$ term matrix B as follows:

$$B = U\Sigma V^T \tag{11}$$

where $U^TU = VV^T = I$, and U, and V are respectively $N \times N$ and $M \times M$ matrices. Typically, $N >> M$. Σ is a $N \times M$ block diagonal matrix with diagonal elements, known commonly as singular values, σ_i. It is customarily to sort the diagonal values of Σ such that $\sigma_i \geq \sigma_{i+1}$.

It is possible to examine the diagonal values of Σ and decides that for $\sigma_i = 0, i = j+1, j+2, \ldots, N$. In other words we have decided to ignore the contribution of the diagonal values for $\sigma_i, i = j+1, \ldots, N$. The values of σ_i, $i = 1, 2, \ldots, j$ can be considered as the latent dimensions of the sparse dimensional M vector space. It is observed that HITS is related to the LSA approach.

2.6 PLSA algorithm

The Probabilistic LSA algorithm [9] is very similar to the PHITS. In the PLSA, we start with the term document matrix B. The formulation is exactly the same as PHITS except that in Equations (4) and (5), the entity c_i is replaced by t_i and in the log likelihood function we have instead:

$$L = \sum_j \sum_i N_{ij} \log \sum_k P(t_i|z_k)P(z_k|d_j) \tag{12}$$

where N_{ij} denotes the term frequency, i.e., the number of times t_i occurred in document d_j. Again using the EM algorithm it is possible to derive a set of EM algorithms for estimating the values of $P(t_i|z_k)$, $P(d_j|z_k)$ and $P(z_k)$.

2.7 Combined PHITS and PLSA algorithm

Since the PHITS and PLSA operate on different input matrices, it is conceivable that they can be combined [5]. This is because the PHITS works on the citation matrix A, while the PLSA works on the term document matrix B. Both decompose the matrices into factor of mixture models.

The decompositions share the same document specific mixing properties $P(z_k|d_j)$. This connects the probabilities for term and citation: each topic has some probability $P(c_i|z_k)$ of linking to document d_j and some probability $P(t_i|z_k)$ of containing an occurrence of term t_i.

The joint likelihood is given by:

$$L = \sum_j [\alpha \sum_i \frac{N_{ij}}{\sum_{i'} N_{i'j}} \log \sum_k P(t_i|z_k)P(z_k|d_j) + (1-\alpha) \sum_\ell \frac{A_{\ell j}}{\sum_{\ell'} A_{\ell' j}} \log \sum_k P(c_\ell|z_k)P(z_k|d_j)] \tag{13}$$

Using the EM approach, we can derive the updating equations as follows:

$$P(z_k|t_i, d_j) = \frac{P(t_i|z_k)P(z_k|d_j)}{P(t_i|d_j)} \tag{14}$$

$$P(z_k|c_\ell, d_j) = \frac{P(c_\ell|z_k)P(z_k|d_j)}{P(c_\ell|d_j)} \tag{15}$$

$$\begin{aligned} P(t_i|z_k) &= \sum_j \frac{N_{ij}}{\sum_{i'} N_{i'j}} P(z_k|t_i, d_j) \\ &= \sum_j \frac{A_{\ell j}}{\sum_{\ell'} A_{\ell' j}} P(z_k|c_\ell, d_j) \end{aligned} \tag{16}$$

and

$$\begin{aligned} P(z_k|d_j) \propto \alpha \sum_i \frac{N_{ij}}{\sum_{i'} N_{i'j}} P(z_k|t_i, d_j) \\ + (1-\alpha) \sum_\ell \frac{A_{\ell j}}{\sum_{\ell'} A_{\ell' j}} P(z_k|c_\ell, d_j) \end{aligned} \tag{17}$$

3 Issues

There are a number of issues related to the determination of the relevance of a page. These include:

1. Feature extraction to incorporate context. The term matrix is extended to what is known as a term frequency inverse document frequency (TF-IDF) representation as follows:

$$v(t_i) = \frac{N_{ij}}{N_{ij}^{\max}} \log \frac{N}{N_{t_i}} \tag{18}$$

   where $N_{ij}^{\max}$ is the maximum number of occurrences of a term t_i in a document d_j; N_{t_i} is the number of documents in the collection that the term t_i occurs at least once. It is not know if using this measure would make differences to the analysis shown here.

2. Nepotistic links. Content designers may design web pages to take advantages of the way in which web pages are ranked. They intentionally generate many "artificial" links to the page which is to be ranked. This is known as "link–based spam". Some preliminary work in this area is given in [6].

3. Mirror, or duplicate links. Mirror sites are common for duplicating sites locally. Often, the mirroring may not be exact, in that the files in one mirror site may not be exactly the same as those in the original site. The issue is: how do we recognize mirror sites. Some work in this direction is given in [1].

4. Algorithmic issues. Conceptually, it is not too difficult to formulate a Bayesian model for the LSA situation or for the HITS situation. However, Bayesian models are notoriously compute intensive, especially in the evaluation of the posterior probability functions using some kind of sampling techniques, e.g., Markov Chain Monte Carlo technique [13].

5. Computational issues. The deterministic methods, e.g., PageRank, HITS, LSA algorithms are all relatively simple to compute. This is especially true if there is a limit to the total number of documents (or links) considered in the document set. The algorithms will converge relatively rapidly.

   The probabilistic algorithms, e.g., PHITS, PLSA depend on the convergence of the EM algorithm. It is known that the EM algorithm could take time to converge. This is particularly true that if both the number of documents and the number of terms or links are large.

6. Content of the pages. So far we have considered only text versions of the web pages. However, there are many web pages that contain images, video clips, or audio clips. So far, there is relatively fewer work which consider the situation of searching the web which contain multimedia materials.

   It is likely that in considering multimedia materials on the internet, there needs to be additional information incorporated in the pages before effective search is possible. For example, one may consider the possibility that an image is annotated. One may consider that each frame of a video needs to be annotated.

7. Focussed crawling. So far most of the methods discussed are based on the following assumption: it is possible to crawl and categorize the entire internet. With the internet growing at tremendous rate, it is conceivable that the day may come when even with the fastest computing machines available, it is not possible to crawl and categorize the topics contained in the internet. In [8], a focussed crawler is designed by extending the concept of base set to a hierarchy of related document sets.

8. Modification of the current hypertext standards to incorporate feedback. [12] suggests that it might be desirable to modify the current HTML standard to incorporate user feedback in an attempt to have much more refined scores for web pages.

4 Conclusions

In this paper, we have considered one of the central questions in the design of a search engine viz., the determination of the relevance of a web page. We have considered the various ways in which the relevance of a page can be determined. It is shown that there are two approaches, viz., one which is based on link analysis while the other is based on context. It is shown that within each approach, there are a number of almost parallel developments, progressing from a deterministic method based on singular value decomposition, to probabilistic methods. A number of issues are considered which need to be studied before search engines can be made more effective.

References

[1] Bharat, K, Broder, A., Dean, J., Henzinger, M. "A comparison of techniques to find mirrored hosts on the WWW". *Proc of the ACM Digital Library Workshop on Organizing Web Spaces*, 1999.

[2] Brin, S., Page, L. "The anatomy of a large scale hypertextual web search engine". *Seventh International World Wide Web Conference*, Brisbane, 1998.

[3] Borodin, A., Roberts, G., Rosenthal, J., Teaparas, P. "Finding authorities and hubs from link structures on the world wide web". Preprint, 2000.

[4] Cohn, D., Chang, H., "Learning to probabilistically identify authorative documents". *Proc 17th International Conference on Machine Learning*, 2000.

[5] Cohn, D., Hofmann, T. "The missing link – a probabilistic model of document content and hypertext connectivity". *Neural Information Processing Systems*. Nov., 2000.

[6] Davison, B. "Recongizing nepotistic links on the web". *AAAI 2000 Workshop on Artificial Intelligence for Web Search*. 2000.

[7] Deerwester, S., Dumais, S., Harshman, R. "Indexing by Latent Semantic Analysis". *J. of the American Society for Information Science*. Vol. 41, pp 391–407, 1990.

[8] Diligenti, M., Coetzee, F., Lawrence, S., Giles, L., Gori, M. "Focussed crawling using context graphs". *26th International conf on Very Large Databases*, 2000.

[9] Hofmann, T. "Probabilistic latent semantic analysis" *Proc of the 15th Conf. on Uncertainty in AI*. pp 289–296, 1999.

[10] Kleinberg, J. "Authorative sources in a hyperlinked environment". *Proc ACM–SIAM Symposium on Discrete Algorithms*, 1998.

[11] Lempel, R., Moran, S. "The stochastic approach for link structure analysis (SALSA) and the TKC effect". *Proc 9th International World Wide Web Conference*, May, 2000.

[12] Lifantsev, M. "Rank computation methods for Web documents". *Technical Report TR-76*, ECSL, Department of Computer Science, SUNY at Stony Brook, Stony Brook, NY, November 1999.

[13] Robert, C. Casella, G. *Monte Carlo Statistical Methods*. Springer Verlag, New York, 1999.