

Web Document Models for Web Information Retrieval

Michel Beigbeder
G2I department
École Nationale Supérieure des Mines
158, cours Fauriel
F 42023 SAINT ETIENNE CEDEX 2

Abstract

Different Web document models in relation to the hypertext nature of the Web are presented. The Web graph is the most well known and used data extracted from the Web hypertext. The ways it has been used in works in relation with information retrieval are surveyed. Finally, some considerations about the integration of these works in a Web search engine are presented.

1. Web document models

Flat independent pages The immediate reuse of the long lived Information Retrieval (IR) techniques led to the most simple model of Web documents. It was used by the first search engines: Excite (1993), Lycos (1994), AltaVista (1994), etc. In this model, HTML pages are converted to plain text by removing the tags and keeping the text between the tags. Easily, the content of some tags can be ignored. Then pages are indexed as flat plain text. The prevailing IR model used with this document model is the vector model, though AltaVista introduced a combination of a Boolean model to select a bunch of documents which is then ranked with a vector model.

The main advantage of this model is that many of the traditional IR tools and techniques could be straightforwardly used.

Structured independent pages The enhancement from the first model is that some structure about the pages is kept either in the index or considered in the indexing step. For instance, with a Boolean like models, words could be only looked for in the title tag. Such capabilities have been proposed for some time, but, like other Boolean capabilities, did not get much public success. With the vector model the words appearing in the title or sectioning tag (for instance) could receive a greater weight than others. Some

search engines mentioned this peculiarity, but, as far as I know, no details were given and no experiments were conducted to prove the effectiveness of these different weighting schemes.

These uses of the internal structure of the Web documents are very weak compared to the strong internal structure allowed by HTML. But the documents found on the Web are not strongly structured because many structural elements are misused to obtain page layouts. So, the works in IR on structured documents are not useful in the actual Web.

Linked pages In this model the hypertext links represented by the `` tags are used to build a directed graph: the *Web graph*. The nodes of the graph are the pages themselves, and there is one arc from the node P to the node P' iff there is somewhere in the HTML code of P an href link to the page P' . Note that this is a simplification of what is really coded in the HTML, because if there are many href links in P to P' , there is only one arc (otherwise, we would define a *multigraph*).

But the most difficult point here is to define precisely what are the nodes: pages or URL or set of pages. Let us precise the choice.

The pages are identified by their URL, and URL themselves are composed of nine fields:

```
<scheme>://<user>:<passwd>@<host>:<port>/  
<path>;<parameters>?<query>#<fragment>
```

If the `user` and `passwd` fields can be safely ignored, what to do with the `parameters` and `query` ones is not trivial. By ignoring them to define the nodes, a graph with fewer nodes and more connectivity is obtained, but the point is what of the many content is to be associated to the node?

Moreover, using the `fragment` field would lead either to consider the page as composed of smaller units or to consider these smaller units as the documents to be returned by the search engine. Though, due to the poor use of the HTML, many of the opening `` tags are not closed with a ``, so many fragments are not fully

delimited. So I think that this field should be ignored.

Another difficulty is the replication of pages, either actual replication on different servers or replication through different names on a single server. As an example of the second case, both `http://rim.emse.fr/` and `http://www.emse.fr/fr/transfert/g2i/depscientifiques/rim/index.html` point on the same page. When it is possible to recognize this replication, I think it is better to merge the different URL in a single node because a graph with higher density is obtained and as they refer to the same content there is not the problem of choosing or building such a content.

Given some choices regarding the quoted questions, the directed graph can be built. It has been extensively studied [3] [10] and used for information retrieval in particular. We will review some of its usages in section 2.

Anchor linked pages This model takes into account more of the HTML code. Each anchor, delimited with a `` tag and the corresponding `` tag, is used to index the page pointed by the href attribute. This idea is still in use in some search engines. Moreover in the Web context where spidering is an essential part of the information retrieval system (IRS) to keep the index up to date, it allows the association of an index to a document (a page) before it is actually loaded. Variations consist in heuristics that take into account not only the anchor text itself but also its neighboring. Note that this is not very different from the first point exposed in section 2 about relevance propagation.

2. Link usage

The Web graph between pages is used by many works. In relation to IR it has been used for different goals.

Index enhancement and relevance propagation One of the first ideas tested in hypertext environments [8] consists in using the index of neighbors of a node either to index the node in the indexing step, or to use the relevance score values (RSV) of these nodes in the querying step. Both of these methods are based on the idea that the text in a node (a page in the Web context) is not self contained and that the text of the neighbors can give either a context or some precision to the text of the nodes. Savoy conducted many experiments to test this idea. He reports that effectiveness improvements are low with vector and probabilistic models [16] and higher with the Boolean model [17]. Marchiori uses a propagation with some fading for fuzzy metadata [13]. The same scheme could be applied to the term weights in the vector model.

Page ranking: PageRank [2] and HITS [9] We will not describe once more here these two methods. The first one attribute a (popularity) score to every page, the second one attributes two (*hubbiness* and authority) scores to them. The key point is that these scores are independent of the words used either in the documents or in the query.

Page gathering The page ranking algorithms can be used on any graph, and hence on any subgraph of the Web. The PageRank algorithm has been used to focus gathering on a given topic [5].

Page categorization If some pages are categorized, it can help to categorize their neighbors, this idea has been used in combination with the content analysis of the pages [4].

Page classification Classification is different from categorization in the sense that classes are not predefined. A method based on co-citation, which was first used in library science [18], is presented in the Web context by Prime *et al.* [15], it aims to semi-automatically qualify Web pages with metadata.

Similar page discovery Dean *et al.* [6] proposed two solutions to this problem. The first one is based on the HITS algorithm and the second one is based on co-citation [18].

Replica discovery Bharat *et al.* present a survey of techniques to find replicas on the Web [1]. One of them is based on the link structure of the Web.

Logical Units Discovery The idea here is akin to that of index enhancement: if pages are not self contained, they need to be indexed or searched with other ones. But here, the context is not built with a breadth first search algorithm on the Web graph, but with other algorithms.

Three methods are aimed at augmenting the recall, with the idea that not all the concepts of a conjunctive query are present in a page, but some of them are in neighbor pages [7] [19] [11]. Note that the Dyreson's method [7] does not use the Web graph but a graph derived from it by taking into account the directory hierarchy coded in the URL. These three methods share the drawback that they take place in a boolean framework.

Tajima *et al.* [20] propose to discover the logical units of information by clustering. To take into account the structure, the similarity between two clusters is zero when there are no links between any page of one cluster and any page of the second cluster, otherwise the similarity is computed with Salton's model. So there is not a strong use of the link structure.

Communities discovery Another approach by Li *et al.* [12] attempts to discover *logical domains* — as opposed to the *physical domains* tied to the host addresses. These domains are of a greater granularity than the logical units of the previous paragraph. Their goal is to cluster the results of a search task. In order to build these domains, the first step consists in finding k (an algorithm parameter) *entry points* with criteria that take into account the `title` tag content, the textual content of the URL¹, the in and out degree within the Web graph, etc. In the second step, pages that are not entry points are linked to the first entry point located above considering their URL path (as a result, some pages may stay orphan). Moreover some conditions — minimal size of a domain, radius — influence the constitution of domains.

¹Some words such as *welcome, home, people, user*, etc. are important.

3. Link tools

There are rather few basic methods used in the link usage:

1. graph search (mainly breadth first search);
2. PageRank and HITS algorithms (which are matrix based);
3. co-citation (building the co-citation data is also a matrix; manipulation)
4. clustering (many methods can be used).

4. URL use

We already note that Dyreson [7] does use the URL data to discover logical units. In the study conducted by Mizuuchi *et al.* [14] the URL coded paths are used to discover for every page P one (and only one) *entry page*, *i.e.* a page by which a reader is supposed to get through before arriving at P . A page tree is defined by these entry pages. This tree is used to enhance the index of a page with the content of some tags of the ancestors of P .

5. Conclusion and proposition

IR integration The works quoted above are not all dealing directly with the IR problem. Many of them were not tested with test collections which are standard in the IR community such those of TREC². So some work has to be done on how to integrate and test these methods in a search engine.

Precision enhancement Now, I present some qualitative considerations. Many of these methods are aimed at dealing with the huge size of the Web: everything about some kind of classification or categorization are of this kind. Most often, these methods can be applied either before the query as a preprocessing step or on the results of a query.

While not explicitly in this direction the PageRank algorithm can be considered of this kind. Due to the very huge size of the Web, many queries, especially the very short queries submitted by the Web users, have many, many answers. The polysemy is much higher than in traditional IR collections. So the use of clues external to the vocabulary can be seen as a discrimination factor to select documents when the collection is very large.

Recall enhancement The other usages (Index enhancement and Logical Units Discovery) are aimed on the contrary to enhance *recall*, which is not often required, or not a priority when too many irrelevant answers are given to the queries.

Though, as for me, the Logical Units Discovery methods can be considered in an IR point of view as trying to access to different levels of granularity of documents in the Web space. If we consider that an IR system returns pointers to documents, the notion of document is what is returned by the IR system. So if an IR system returns a Logical Unit which is composed of several pages, this is a higher level of granularity.

²<http://trec.nist.gov/>

Proposition: a hierarchical presentation of the Web

Many of the queries submitted to search engines have many many answers. The IR traditional relevance and the popularity produce lists of answers. But presenting the results as an ordered list increases the likelihood of missing important, and in some sense *rarer*, information. This is true especially if the ranking is only done with popularity as this has the effect that the best ranked documents have the more likelihood to get better ranked.

I suggest that the results should be presented by clusters, with a number of clusters manageable by the user (from ten to one hundred, it could be a user preference). With iterative clustering, any document would be at a $\log(n)$ distance from the root rather than to be at a n distance from the beginning of a sorted list.

To help to do that, many possibilities can be considered:

- some of the clustering techniques could be applied either on the Web, or on the results of a query;
These clustering could be done with similarity based on different clues according to the user information need (text similarity, co-citation similarity, co-occurrence, etc.)
- some categorization could be used (particularly open ones³);
- Entry Points Discovery and Logical Units Discovery could be used to merge several URL in a single node in the graph; Merging several URLs in a single node has two beneficial effects: it both reduces the size of the graph and the resulting graph has a higher density. Reducing the size of the graph has an influence on the run time of the algorithms, which is important due to the size of the Web and the complexity of some algorithms (clustering for example). Increasing the density is important because the Web graph is rather sparse, and a few proportion of pages are cited (and even fewer are co-cited). So the benefit of the algorithms based on the links is not well spread.
- recall enhancement methods could be used when queries give no answers.

References

- [1] K. Bharat, A. Broder, J. Dean, and M. R. Henzinger. A comparison of techniques to find mirrored hosts on the www. *IEEE Data Engineering Bulletin*, 23(4):21–26, 2000.
- [2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.
- [3] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web: experiments and models. In *9th International World Wide Web Conference, The Web: The Next Generation*, 5 2000.
- [4] S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In L. M. Haas and A. Tiwary, editors, *Proceedings ACM SIGMOD International Conference on Management of Data*, pages 307–318. ACM Press, 1998.
- [5] J. Cho, H. Garcia-Molina, and L. Page. The anatomy of crawling through url ordering. *Computer Networks and ISDN Systems*, 30(1–7):161–172, 1998.

³<http://www.dmoz.org/> for instance.

- [6] J. Dean and M. R. Henzinger. Finding related pages in the world wide web. *Computer Networks*, 31(11-16):1467–1479, 1999.
- [7] C. E. Dyreson. A jumping spider: Restructuring the WWW graph to index concepts that span pages. In A.-M. Vercoustre, M. Milosavljevi, and R. Wilkinson, editors, *Proceedings of the Workshop on Reuse of Web Information, held in conjunction with the 7th WWW Conference*, pages 9–20, 1998. CSIRO Report Number CMIS 98-11.
- [8] M. E. Frisse. Searching for information in a hypertext medical handbook. *Communications of the ACM*, 31(7):880–886, 1988.
- [9] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
- [10] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. The web as a graph. In *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 2000.
- [11] W.-S. Li, K. S. Candan, Q. Vu, and D. Agrawal. Retrieving and organizing web pages by "information unit". In M. R. Lyu and M. E. Zurko, editors, *Proceedings of the Tenth International World Wide Web Conference*, 2001.
- [12] W.-S. Li, O. Kolak, Q. Vu, and H. Takano. Defining logical domains in a web site. In *HYPertext '00, Proceedings of the eleventh ACM on Hypertext and hypermedia*, pages 123–132, 2000.
- [13] M. Marchiori. The limits of web metadata and beyond. *Computer Networks and ISDN Systems*, 30(1–7):1–9, 1998.
- [14] Y. Mizuuchi and K. Tajima. Finding context paths for web pages. In *HYPertext '99, Proceedings of the tenth ACM Conference on Hypertext and hypermedia: returning to our diverse roots*, pages 13–22, 1999.
- [15] C. Prime-Claverie, M. Beigbeder, and T. Lafouge. Transposition of the co-citation method with a view to classifying web pages. *Journal of the American Society for Information Science and Technology*, 55(14):1282–1289, 2004.
- [16] J. Savoy. *Citation schemes in Hypertext information retrieval*, pages 99–120. Kluwer Academic Publishers, 1996. in Agosti M. and Smeaton A. editors, *Information Retrieval and Hypertext*.
- [17] J. Savoy. Ranking schemes in hybrid boolean systems: a new approach. *Journal of the American Society for Information Science*, 48(3):235–253, 1997.
- [18] H. Small. Co-citation in the scientific literature: a new measure of the relationship between two documents. *Journal of the American Society for information Science*, 24(4):265–269, 1973.
- [19] K. Tajima, K. Hatano, T. Matsukura, R. Sano, and K. Tanaka. Discovery and retrieval of logical information units in web. In R. Wilensky, K. Tanaka, and Y. Hara, editors, *Proc. of Workshop of Organizing Web Space (in conjunction with ACM Conference on Digital Libraries '99)*, pages 13–23, 1999.
- [20] K. Tajima, Y. Mizuuchi, M. Kitagawa, and K. Tanaka. Cut as a querying unit for WWW, netnews, e-mail. In *HYPertext '98, Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space—structure in hypermedia systems*, pages 235–244, 1998.