Mining Web using Hyper Induced Topic Search Algorithm

Manali Gupta[1], **Shweta Rathour**[2]

[1, 2]ITS Engineering College, Greater Noida, India

Abstract: ***There is a lot of data on the Web, some in databases, and some in files or other data sources. The databases may be semi structured or they may be relational, object, or multimedia databases. These databases have to be mined so that useful information is extracted. While we could use many of the data mining techniques to mine the Web databases, the challenge is to locate the databases on the Web. Furthermore, the databases may not be in the format that we need for mining the data. We may need mediators to mediate between the data miners and the databases on the Web. This paper presents the important concepts of the databases on the Web and how these databases have to be mined to extract patterns and trends.***

Keywords: Data Mining, Web Usage Mining, Document Object Model, KDD dataset

1. Introduction

Data mining slowly evolves from simple discovery of frequent patterns and regularities in large data sets toward interactive, user-oriented, on-demand decision supporting. Since data to be mined is usually located in a database, there is a promising idea of integrating data mining methods into Database Management Systems (DBMS) [6]. Data mining is the process of posing queries and extracting patterns, often previously unknown from large quantities of data using pattern matching or other reasoning techniques.

2. Challenges for Knowledge Discovery

Data mining also referred to as database mining or knowledge discovery in databases (KDD) is a research area that aims at the discovery of useful information from large datasets. Data Mining [9] uses statistical analysis and inference to extract interesting trends and events, create useful reports, support decision making etc. It exploits the massive amounts of data to achieve business, operational or scientific goals. However based on the following observations the web also poses great challenges for effective resource and knowledge discovery.

- The web seems to be too huge for effective data warehousing and data mining. The size of the web is in the order of hundreds of terabytes and is still growing rapidly. many organizations and societies place most of their public-accessible information on the web. It is barely possible to set up data warehouse to replicate, store, or integrate of the data on the web.
- The complexity of web pages is greater than that of any traditional text document collection. Web pages lack a unifying structure. they contain far more authoring style and content variations than any set of books or other traditional text based documents. The web is considered a huge digital library; however the tremendous number of documents in this library is not arranged according to any particular sorted order. There is no index by category, nor by title, author, cover page, table of contents and so on.
- The web is a highly dynamic information source. Not only does the web grow rapidly, but its information is also constantly updated. News, stock markets, weather, airports, shopping, company advertisements and numerous other web pages are updated regularly on the web.
- The web serves a broad diversity of user communities. The internet currently connects more than 100 million workstations, and its user community is still rapidly expanding. Most users may not have good knowledge of the structure of the information network and may not be aware of the heavy cost of a particular search.
- Only a small portion of the information on the web is truly relevant or useful. It is said that 99 % of the web information is useless to 99 % of web users. Although this may not seem obvious, it is true that a particular person is generally interested in only a tiny portion of the web, while rest of the web contains information that is uninteresting to the user and may swamp desired such results.

These challenges have promoted search into efficient and effective discovery and use of resources on the internet. There are many index based **Web search engines.** These search the web, index web pages, and build and store huge keyword-based indices that help locate sets of web pages containing certain keywords [1]. However a simple keyword based search engine suffers from several deficiencies. First, a topic of any breadth can easily contain hundreds of thousands of documents. This can lead to a huge number of document entries returned by a search engine, many of which are only marginally relevant to the topic or may contain materials of poor quality. Second, many documents that are highly relevant to a topic may not contain keywords defining them. This is referred to as the **polysemy problem**. For example, the keyword Oracle may refer to the oracle programming language, or an island in Mauritius or brewed coffee. So a search based on the keyword, search engine may not find even the most popular web search engines like Google, Yahoo!, AltaVista if these services do not claim to be search engines on their web pages.

So a keyword-based web search engine is not sufficient for the web discovery, then Web mining should have to be implemented in it. Compared with keyword-based Web search, Web mining is more challenging task that searches for web structures, ranks the importance of web contents, discovers the regularity and dynamics of web contents, and mines Web access patterns. However, Web mining can be used substantially enhances the power of documents, and resolve many ambiguities and subtleties raised in keyword-based web search[2]. Web mining tasks can be classified into three categories:

- Web content mining
- Web structure mining
- Web usage mining.

3. Web Content Mining

The concept of web content mining is far wider than searching for any specific term or only keyword extraction or some simple statics of words and phrases in documents. For example a tool that performs web content mining can summarize a web page so that to avoid the complete reading of a document and save time and energy. Basically there are two models to implement web content mining. The first model is known as local knowledgebase model. According to this model, the abstract characterizations of several web pages are stored locally. Details of these characterizations vary on different systems [3]. For example, there are three categories of web sites: games, educational and others. References to several web sites relating to these categories are stored in a database.

When extracting information, first the category is selected and then a search is performed within the web sites referred in this category. A query language enables you to query the database consisting of information about various categories at several levels of abstraction. As a result of the query, the system using this model for web content mining may have to request web pages from the web that matches the query. The concept of artificial intelligence is highly used to build and manage the knowledgebase consisting of information on various classes of web sites. The second approach is known as agent based model. This approach also applies the artificial intelligence systems, known as web agents that can perform a search on behalf of a particular user for discovering and organizing documents in the web.

4. Web Usage Mining

The concept of web mining that helps automatically discovering user access patterns. For example, there are four products of a company sold through the web site of a company. Web usage mining analyses the behavior of the customers [4]. This means by using a web usage mining tool the nature of the customers that is which product is most popular ,which is less, which city has the maximum number of customers and so on.

5. Web Structure Mining

Denotes analysis of the link structure of the web.web structure mining is used for identifying more preferable documents. For example, the document A in web site X has a link to the document B in the web site Y [5]. According to Web structure mining concept, document B is important to the web site A, and contains valuable information. The hyperlink induced Topic search (HITS) is a common algorithm for knowledge discovery in the web.

6. Mining the Web Page Layout Structure

Compared with traditional plain text, a web page has more structure. Web pages are also regarded as semi-structured data. The basic structure of a web page is its DOM [3] (Document object model) structure. The DOM structure of a web page is a tree structure where every HTML tag in the page corresponds to a node in the DOM tree. The web page can be segmented by some predefined structural tags. Useful tags include<P>(paragraph),<TABLE>(table),<UL>(list),<H1>~< H6>(heading) etc. Thus the DOM structure can be used to facilitate information extraction. Figure 1 illustrates HTML DOM Tree Example [6]:

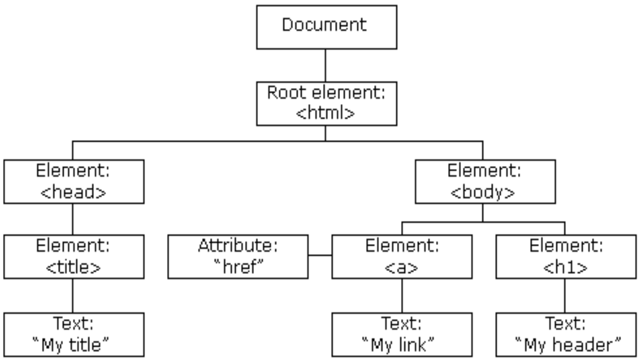

Figure 1: HTML DOM Tree Example

Here's the DOM object tree generated by the code for the TABLE element and its child elements [7]:

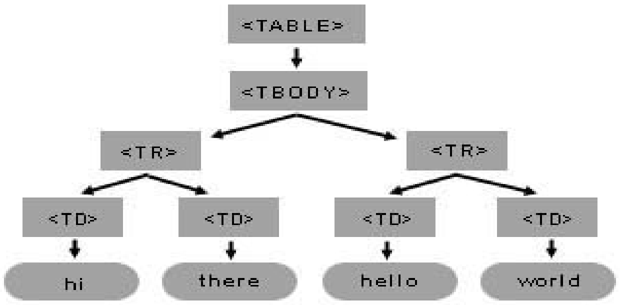

Figure 2: DOM Object Tree Example

Moreover, the DOM tree was initially introduced for presentation in the browser rather than the description of the semantic structure of the web page. For example, even though the two nodes in the Dom tree have the same parent, the two nodes might not be more semantically related to each other than to other nodes.

In the sense of human perception, people always view a web page as different semantic objects rather than as a single object. Some research efforts show that users always expect that certain functional parts of a web page appear at certain positions on the page. Actually, when a web page is presented to the user, the spatial and visual cues can help the user unconsciously divide the web page into several semantic parts. Therefore it is possible to automatically segment the web pages by using the spatial and visual cues. Based on this observation there is an algorithm called Vision-based page segmentation (VIPS).VIPS aims to extract the semantic structure of a web page based on its visual presentation. Such semantic structure is a tree structure: each node in the tree corresponds how coherent is the content in the block based on visual perception. The VIPS algorithm makes full use of the page layout feature. It first extracts all of the suitable blocks from the HTML DOM tree, and then it finds the separators between these blocks. Here separators denote the vertical or horizontal lines in a web page that visually cross with no blocks. Based on the separators, the semantic tree of the web page is constructed. A web page can be represented as a set of blocks (leaf nodes of the semantic tree). Compared with the DOM- based methods, the segments obtained by VIPS are more semantically aggregated.

7. Mining the Web's Link Structures to Identify Authoritative Web Pages

Suppose to search for Web pages relating to a given topic, such as financial investing. In addition to retrieving pages that are relevant, the pages retrieved should be of high quality, or authoritative on the topic. the secrecy of authority is hiding in Web page linkages. The Web consists not only of pages, but also of hyperlinks pointing from one page to another. These hyperlinks contain an enormous amount of latent human annotation that can help automatically infer the notion of authority. When an author of a Web page creates a hyperlink pointing to another Web page, this can be considered as the author's endorsement of the other page. The collective endorsement of a given page by different authors on the Web may indicate the importance of the page and may naturally lead to the discovery of authoritative Web pages. Therefore, the tremendous amount of Web linkage information provides rich information about the relevance, the quality, and the structure of the Web's contents, and thus is a rich source for Web mining.

However, the Web linkage structure has some unique features. First, not every hyperlink represents the endorsement we seek. Some links are created for other purposes, such as for navigation or for paid advertisements. Yet overall, if the majority of hyperlinks are for endorsement, then the collective opinion will still dominate. Second, for commercial or competitive interests, one authority will seldom have its Web page point to its rival authorities in the same field. For example, Coca-Cola may prefer not to endorse its competitor Pepsi by not linking to Pepsi's Web pages. Third, authoritative pages are seldom particularly descriptive. For example, the main Web page of Yahoo! may not contain the explicit self-description "Web search engine."

These properties of Web link structures have led researchers to consider another important category of Web pages called a hub. A hub is one or a set of Web pages that provides collections of links to authorities. Hub pages may not be prominent, or there may exist few links pointing to them; however, they provide links to a collection of prominent sites on a common topic. In general, a good hub is a page that points to many good authorities; a good authority is a page pointed to by many good hubs. Such a mutual reinforcement relationship between hubs and authorities helps the mining of authoritative Web pages and automated discovery of high-quality Web structures and resources.

An algorithm using hubs, called HITS (Hyperlink-Induced Topic Search), is a common algorithm for knowledge discovery in the web. HITS is a web searching method where the searching logic partially depends on hyperlinks to identify and locate the documents relating to a topic in the web. The HITS algorithm discovers the hubs and authorities of a community on a specific topic or query. In HITS algorithm the number of links between web sites is measured as weights. For a web site w, the weight of authority denotes the number of web sites containing a hyperlink to the web site w. Similarly the weight of the hub denotes the number of hyperlinks in the web site x pointing to other web sites.

The steps in the **HITS algorithm** are:

- Accept the seed set, S, returned by a search engine. The set, S contains n number of web pages, where usually value of n lies between 0 to 200, means, n>0 and n<=200.
- Initialize the weight of the hub to 1 for each web page, p in the set, S. this means, assign hub_weight (p)=1,for each p, where p ε S.
- Initialize the weight of the authority to 1 for each web page, p in the set, S. this means, assign authority_weight(p)=1 for each p, where p ε S.
- Let the expression p→q denote that the web page p has a hyperlink to the web page q.
- Iteratively update weight of the authority, and weight of the hub for each page, p in the set, S. Repeat this step for a predetermined fixed number of times by calculating:

$$\text{authority\_weight}(p)=\sum_{q\to p}\text{hub\_weight}(q) \qquad (1.1)$$

$$\text{hub\_weight}=\sum_{p\to q}\text{authority\_weight}(q) \qquad (1.2)$$

- Stop.

Equation (1.1) implies that if a page is pointed to by many good hubs, its authority weight should increase (i.e., it is the sum of the current hub weights of all of the pages pointing to it). Equation (1.2) implies that if a page is pointing to many good authorities, its hub weight should increase (i.e., it is the sum of the current authority weights of all of the pages it points to).

These equations can be written in matrix form as follows. Let us number the pages {1,2,. . . . ,n} and define their adjacency matrix A to be an n x n matrix where A(i, j) is 1 if page i links to page j, or 0 otherwise. Similarly, we define the authority weight vector a = (a1, a2,. . . ,a_n), and the hub weight vector h = (h1,h2,. . . . ,h_n). Thus, we have

$$\mathbf{h} = A \cdot \mathbf{a} \qquad (1.3)$$

$$\mathbf{a} = A^{T} \cdot \mathbf{h}, \qquad (1.4)$$

where A^T is the transposition of matrix A. Unfolding these two equations k times, we have [3]

$$h = A \cdot a = AA^T h = (AA^T)h = (AA^T)^2 h = \cdot \cdot \cdot = (AA^T)^k h \quad (1.4)$$

$$a = A^T \cdot h = A^T Aa = (A^TA)a = (A^TA)^2 a = \cdot \cdot \cdot = (A^TA)^k a. \quad (1.5)$$

According to linear algebra, these two sequences of iterations, when normalized, converge to the principal eigenvectors of AAT and ATA, respectively. This also proves that the authority and hub weights are intrinsic features of the linked pages collected and are not influenced by the initial weight settings.

Finally, the HITS algorithm outputs a short list of the pages with large hub weights, and the pages with large authority weights for the given search topic. Many experiments have shown that HITS provides surprisingly good search results for a wide range of queries. Although relying extensively on links can lead to encouraging results, the method may encounter some difficulties by ignoring textual contexts. The problems faced in the HITS are:

- This algorithm does not have an effect of automatically generated hyperlinks.
- The hyperlinks pointing to the irrelevant or less relevant documents are not excluded and cause complications for updating hub and authority weights.
- A hub may contain various documents covering multiple topics. The HITS algorithm faces problem to concentrate on the specific topic mentioned by the query. This problem is called drifting.
- Many web pages across various web sites sometimes points to the same document. This problem is referred to as topic hijacking.

Such problems can be overcome by replacing the sums of Equations (1.1) and (1.2) with weighted sums, scaling down the weights of multiple links from within the same site, using anchor text (the text surrounding hyperlink definitions in Web pages) to adjust the weight of the links along which authority is propagated, and breaking large hub pages into smaller units.

By using the VIPS algorithm, we can extract page-to block and block-to-page relationships and then construct a page graph and a block graph. Based on this graph model, the new link analysis algorithms are capable of discovering the intrinsic semantic structure of the Web. Thus, the new algorithms can improve the performance of search in Web context. The graph model in block-level link analysis is induced from two kinds of relationships, that is, block-to-page (link structure) and page-to-block (page layout).

The block-to-page relationship is obtained from link analysis. Because a Web page generally contains several semantic blocks, different blocks are related to different topics.

Therefore, it might be more reasonable to consider the hyperlinks from block to page, rather than from page to page. Let Z denote the block-to-page matrix with dimension n x k. Z can be formally defined as follows:

$$Z_{ij} = \begin{cases} 1/s_i & \text{if there is a link from block } i \text{ to page } j \\ 0, & \text{otherwise,} \end{cases} \quad (1.6)$$

where si is the number of pages to which block i links. $Z_{i\,j}$ can also be viewed as a probability of jumping from block i to page j.
The block-to-page relationship gives a more accurate and robust representation of the link structures of the Web.

The page-to-block relationships are obtained from page layout analysis. Let X denote the page-to-block matrix with dimension k x n [8]. As we have described, each Web page can be segmented into blocks. Thus, X can be naturally defined as follows:

$$Xij = \begin{cases} f_{pi}(b_j), & \text{if } b_j \, \varepsilon \, p_i \\ 0, & \text{otherwise,} \end{cases} \quad (1.7)$$

where f is a function that assigns to every block b in page p an importance value. Specifically, the bigger fp(b) is, the more important the block b is. Function f is empirically defined below,

$$f_p(b) = \alpha \text{ x } \frac{\text{the size of block } b}{\text{the distance between the center of } b \text{ and the center of the screen}} \quad (1.8)$$

where α is a normalization factor to make the sum of $f_p(b)$ to be 1, that is,

$$\sum_{b \varepsilon p} f_p(b) = 1$$

Note that $f_p(b)$ can also be viewed as a probability that the user is focused on the block b when viewing the page p. Some more sophisticated definitions of f can be formulated by considering the background color, fonts, and so on. Also, f can be learned from some relabeled data (the importance value of the blocks can be defined by people) as a regression problem by using learning algorithms, such as support vector machines and neural networks [9]. Based on the block-to-page and page-to-block relations, a new Web page graph that incorporates the block importance information can be defined as

$$W_P = XZ, \quad (1.9)$$

Where, X is a k x n page-to-block matrix, and Z is a n x k block-to-page matrix. Thus W_P is a k x k page-to-page matrix.

8. Conclusion

This paper has presented the details of tasks that are necessary for performing Web Usage Mining, the application of data mining and knowledge discovery techniques to WWW server access logs [10].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 services. It also contains a rich and dynamic collection of hyperlink information, and access and usage information, providing rich sources for data mining. Web mining includes mining Web linkage structures, Web contents, and Web access patterns. This involves mining the Web page layout structure, mining the Web's link structures to identify authoritative Web pages, mining multimedia data on the Web, automatic classification of Web documents, and Web usage mining [11]. Data mining is an evolving technology going through continuous modifications and enhancements. Mining tasks and techniques use algorithms that are many a times refined versions of tested older algorithms [12]. Though mining technologies are still in their infancies, yet they are increasingly being used in different business organizations to increase business efficiency and efficacy.

References

[1] Definition of Data Mining, http://www.wdvl.com/Authoring/DB/
[2] HTML DOM Tutorial http://www.w3schools.com/htmldom/default.asp
[3] http://www.cs.cornell.edu/home/kleinber/ieee99-web.pdf
[4] Traversing an HTML table with DOM interfaces, https://developer.mozilla.org/en/traversing_an_html_table_with_javascript_and_dom_interfaces
[5] Web Usage Mining http://maya.cs.depaul.edu/~mobasher/papers/webminer-kais.pdf
[6] Data Mining Within DBMS Functionality by Maciej Zakrzewicz, Poznan University.
[7] Data Mining Concepts and Techniques By Jiawei Han and Micheline Kamber, http://www.cs.uiuc.edu/~hanj/bk2/
[8] Data Mining by Yashwant Kanetkar
[9] Databases on web, www.ism-ournal.com/ITToday/Mining_Databases.pdf
[10] "Seamless Integration of DM with DBMS and Applications"by Hongjun Lu
[11] "Mining the World Wide Web - Methods, Applications, and Perspectives".
[12] Wiki links, http://en.wikipedia.org/wiki/Web_mining

Author Profile

Manali Gupta received the B.Tech degree in Information Technology from Uttar Pradesh Technical University and M.Tech degree in Computer Science and Engineering from Amity University, Noida in 2007 and 2013, respectively. Her research interest includes data mining, database management systems and fundamental study of real time operating systems.

Shweta Rathour received the B.Tech degree in Information Technology from H.N.B. Garhwal University and M.Tech degree in Computer Science and Engineering from Uttarakhand Technical University in 2009 and 2011, respectively. Her research interest includes network Security and database management system, web technology.