Web Content Extraction to Facilitate Web Mining

Anjali Singh
Department of Information Technology
CDAC Noida, GGSIP UNIVERSITY
angelikasingh.18@gmail.com

Abstract- Internet continuously strives to become the prime source of knowledge and Information, used in almost every sphere of life. As the volume and complexity of the Information shared on WEB is increasing, various forms of representation of this data has been emerged. In order to deal with different forms of data, different technologies have been discovered to efficiently provide the Information to the end users. With advent of such technologies the web content is reforming from simple HTML pages to highly complex, sophisticated bunch of data representation. A web page typically contains a mixture of many kind of information e.g. main contains, advertisements, navigational panels, copyright blocks etc. For a particular End User only part of information is useful and the rest could be regarded as noise. These all results into web applications which contain irrelevant and redundant Information, This can seriously harm web mining. The goal of this paper is to explore the use of formal methods for filtration of noise from web pages. Filtration of noise from web pages is a difficult task which in turn leads to difficulty in segmentation. Various automatic techniques use various algorithms of segmentation, which are mainly based on web source code (HTML) including template based analysis. Our insight is to use the DOM structures of web documents to efficiently implement a technique to remove irrelevant data to optimize the WEB mining process. In this approach, we firstly build the Semantic Tree to partition the web page into the content parts/elements based on the web page tags. The main focus is a need to develop a technique that keep common navigation structure as it is, but removes images, advertisement and improve surfing efficiency

Keywords – Web Content Extractor, DOM tree, InnerHTML, outerHTML

I. INTRODUCTION

The proliferation, ubiquity and increasing power of computer technology has increased data collection, storage and manipulations. Data mining has become increasingly common in both the public and private sectors. Organizations use data mining as a tool to survey customer information, reduce fraud and waste, and assist in medical research. Generally; data mining is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Noise removal technique is concerned with content published on internet, usually as Html, Plain text, Xml documents. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases formation. Data mining is seen as an increasingly important tool by modern business to transform data into business intelligence giving an informational advantage.

This technology is popular with many businesses because it allows them to learn more about two important and active areas of current research are data mining and the World Wide Web. Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years. However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost. A natural combination of the two areas sometimes referred to as Web mining. The term Web mining has been used in three distinct ways:

The first, which is referred to as Web content mining, describes the process of information or resource discovery from millions of sources across the World Wide Web.

The second, which we call Web usage mining, is the process of mining Web access logs or other user information user browsing and access patterns on one or more Web localities.

The third, which is referred to as web structure mining to generate structural summary about the Web site and Web page. Structure mining can be used to reveal the structure of Web pages; this would be good for navigation purpose and make it possible to compare/integrate Web.
Web Content Extraction to Facilitate Web Mining

Data mining commonly involves many techniques:

- **WEB MINING**
  - Web usage mining
  - Web content mining
  - Web structure mining

- **SPATIAL MINING**

- **TEMPORAL MINING**

Web mining is the integration of information gathered by traditional data mining methodologies and techniques with information gathered over the World Wide Web. Web mining can be used in social networking, web advertisement, search engine, E-banking and online auction. Web data mining has grown out of the large volumes of data freely available on the web. Prior to data mining becoming a stand-alone task, business analysts and statisticians extracted and analyzed datasets. However, the large volume and technical nature of data necessitated the creation of data mining tools designed specifically for web data mining. A commercial Web page typically contains many information blocks. Apart from the main content blocks, it usually has such blocks as navigation panels, copyright and privacy notices, and advertisements (for business purposes and for easy user access). We call these blocks that are not the main content blocks of the page the noisy blocks. We show that the information contained in these noisy blocks can seriously harm Web data mining. Eliminating these noises is thus of great importance.

In the same way, we can extract some meaningful and specific data from the large data of the Webpage’s. The existing noise removal techniques are mainly used to extract knowledge from Web data, in which at least one of structure or usage data is used in the mining process. These are some of the things we can mine the specific content from the webpage’s. Web data mining can also be used to extract data from elsewhere in your industry, such as price lists and user data. The techniques used till now does not have guarantee of 100% efficiency and the specific content of web pages does not comes at a first search. Once web information is collected it can be used to improve the results of our website. There is a need of content specific and efficient technique which is better than others and gives us perfect search results. In this approach, we use the DOM tree representation as the base since we can obtain many kinds of features by accessing the DOM nodes. Also, by using DOM nodes, we still have the information of the document structure which can be helpful for some reasons e.g. to detect certain pattern structure in the document, to traverse to the other part of the document etc.

II. RELATED WORK

Various methods and Approaches have been devised by researchers which contrasts’ with the approach that is followed here. However there are several restrictions and dissimilarities in the previous approaches used. Various scholars and researches have proposed related work in content identification and information retrieval that attempts to solve similar problems using various other techniques.

Aidan Finn [12] discusses in his research work “Fact or fiction: Content classification for digital libraries”, methods for content extraction from “single-article” sources, where content is supposed to be in a single body. The algorithm tokenizes a page into either words or tags; the page is then sectioned into 3 contiguous regions, placing boundaries to partition the document such that most tags are placed into outside regions and word tokens into the center region. This approach works well for single-body documents, but destroys the structure of the HTML and doesn’t produce good results for multi-body documents, i.e., where content is segmented into multiple smaller pieces like we find on WEB Blogs.

McKeon [32] in the NLP (Natural Language Processing) group at Columbia University detects the largest body of text on a webpage (by counting the number of words) and classifies that as content. This method works well with simple pages. However, this algorithm produces noisy or inaccurate results handling multi-body documents, especially with random advertisement and image placement.

Rahman [13], [17] in first International workshop on “Web Document Analysis” propose another technique that uses structural analysis, contextual analysis, and summarization. The structure of an HTML document is first analyzed and then properly decomposed into smaller subsections. The content of the individual sections is then
extracted and summarized. However, this proposal has yet to be implemented. Furthermore, while the paper lays out prerequisites for content extraction, it doesn’t actually propose methods to do so. Thus it again proves ineffective in actual content extraction. A variety of approaches have been suggested for formatting web pages to fit on the small screens of cellular phones and PDAs however, they basically end up only reorganizing the content of the webpage to fit on a constrained device and require a user to scroll and hunt for content. The main aim is however to device a method for the generic Web documents accessible on any device.

Buyukkokten [14], [15] defines “accordion summarization” as a strategy where a page can be shrunk or expanded much like the instrument. They also discuss a method to transform a web page into a hierarchy of individual content units called Semantic Textual Units, or STUs. First, STUs are built by analyzing syntactic features of an HTML document, such as text contained within paragraph (<P>), table cell (<TD>), and frame component (<FRAME>) tags. These features are then arranged into a hierarchy based on the HTML formatting of each STU. STUs that contain HTML header tags (<H1>, <H2>, and <H3>) or bold text (<B>) are given a higher level in the hierarchy than plain text. This hierarchical structure is finally displayed on PDAs and cellular phones.

While Buyukkokten’s hierarchy is similar to our DOM tree-based model, DOM trees remain highly editable and can easily be reconstructed back into a complete webpage. DOM trees are also a widely-adopted W3C standard, easing support and integration of our technology. The main problem with the STU approach is that once the STU has been identified, Buyukkokten, perform summarization on the STUs to produce the content that is then displayed on PDAs and cell phones. However, this requires editing the original content and displaying information that is different from the original work.

Kaasinen [33] discusses methods to divide a web page into individual units likened to cards in a deck. Like STUs, a web page is divided into a series of hierarchical “cards” that are placed into a “deck”. This deck of cards is presented to the user one card at a time for easy browsing. The paper also suggests a simple conversion of HTML content to WML (Wireless Markup Language), resulting in the removal of simple information such as images and bitmaps from the web page so that scrolling is minimized for small displays. While this reduction has advantages, the method proposed in that paper shares problems with STUs. The problem with the deck-of-cards model is that it relies on splitting a page into tiny sections that can then be browsed as windows. But this means that it is up to the user to determine on which cards the actual contents are located.

Several approaches have been defined for single document extraction. Below are some of the most popular such algorithmic approaches:

- **Crunch framework** is a content extraction program developed by Suhit Gupta. Instead of using raw HTML text; it uses the DOM tree representation of a web document. Receiving input of a HTML page, Crunch will parse the HTML string, construct the DOM, and traverse the nodes recursively and filter out. One of the advantages of Crunch is the genre detection. It uses a classifiers based on the distribution of keywords to classify a web site to one of the pre-defined categories before doing the classification such as news websites, portal websites, shopping websites etc.

- **The Feature Extractor (FE) algorithm** by Debnath is a content extraction algorithm which based on DOM block structures. The algorithm segments a web document into blocks and selects certain blocks to be extracted. A block here corresponds to the DOM sub tree nodes. The algorithm will start working from the root node and recursively splitting the document into blocks. They defined a set of HTML tags which denotes a block namely table, tr, hr, and url. FE uses the feature such as the presence of nested blocks, texts, images, applets, or contained JavaScript code. FE will extract the blocks which is dominant in certain features.

- **Link Quota Filter** is a simple heuristic method to remove link lists and navigational elements. There are many variations of LQF implementations, however generally it measures the ratio between hyperlinked contents to the non-hyperlinked contents in a DOM node. If the ratio is greater than certain threshold, the DOM node should be removed. LQF may work effectively to remove additional contents which consist of
high frequency of hyperlinked text but most likely will fail to deal with additional contents such as headers, footers.

Document Slope Curve (DSC) is another content extraction method which only uses the HTML source code to extract main contents. DSC tokenizes the HTML source code and counts the cumulative tags. After that it will analyze the cumulative tag distribution to identify the main content in a web document. Suppose we have an arbitrary HTML, DSC will tokenize the document and saves the frequency of tag token in every token position in the document. The result of this process can be seen as the graph of tag tokens vs. token. They assumes that the long regions with low slope (contain less HTML tags) are the main contents. In order to check the average slope, they applied window-ing technique to define the document section. If the average slope of the document section within the current window is less than 50% of the average slope of the entire document, the document section is marked as low slope section. One of the advantages of DSC is, it can retrieve more than one main content in a web page.

As low slope section may occur in a scattered manner in the document. However, there is a possibility for DSC to fail when deal with noisy contents such as user comments as most likely user comments will have low slope value. Apart from the traditional algorithmic approaches there are various software/tools employed for the extraction of relevant content for the end users. Most of them were based on pre-existing techniques, thus are limited in one or the other aspects. The common tools include Ad Eater, Internet Junk Buster, Muffin and Web Washer.

III. OUR APPROACH

Our solution employs multiple extensible techniques that incorporate the advantages of the previous work on content extraction. In order to analyze a web page for content extraction, the page is first passed through an HTML parser that creates a DOM tree representation of the web page. According to W3C specification, DOM is an application programming interface (API) for valid HTML and well-formed XML documents. It defines the logical structure of documents and the way how a document is accessed and manipulated. By utilizing DOM, one can construct document, navigate through the structures of the document, and perform operations such as add, update, and delete the properties of the elements. As a standard programming interface, DOM is designed to be programming language independent. There are numerous languages binding for DOM such as Java and ECMA Script (an industry-standard scripting language based on JavaScript and JScript).

In the DOM, the logical structure of a document is represented as a tree structure. Once processed, the resulting DOM document can be seamlessly shown as a webpage to the end-user as if it were HTML. The DOM tree is hierarchically arranged and can be analyzed in sections or as a whole, providing a wide range of flexibility for our extraction algorithm.

Our Web content extractor navigates the DOM tree recursively, using a series of different filtering techniques to remove and modify specific nodes and leave only the content behind. Each of the filters can be easily turned on and off and customized to a certain degree. The filters utilize several attributes and properties of a HTML element to filter out them. For example, the innerHTML and outerHTML properties of a HTML tag are modified programmatically to carry out the task of content extraction. Filtering unwanted or irrelevant data we can easily find out the differences between the sizes of WebPages. A huge number of internet consumers are there in the internetwork environment that consumes and accesses the internet and also download a plenty of data.

The main differences between the traditional approaches and the approached we followed is that we can improve the bandwidth requirement of a Corporate intranet by using the Application on a Proxy server. Since we are actually removing the heavy loaded elements (images, flash advertisements etc) before rendering the output to the end user, the clients within the intranetworking can easily browse; download the documents of their interest using lesser bandwidth.

The main advantages of the proposed approach are:

• Denoising enhances the efficiency of web mining
Users can get relevant or specific data easily

Web pages would be more efficient and effective

After the entire DOM tree is parsed and modified appropriately, it can be output in either HTML or as plain text. The plain text output removes all the tags and retains only the text of the site, while eliminating most white space. The result is a text document that contains the main content of the page in a format suitable for summarization, speech rendering or storage.

Proposed Solution Implementation Architecture:

In order to make our extractor easy to use, we implemented it as a web proxy Application. This allows an administrator to set up the extractor and provide content extraction services for a group. The proxy Application is coupled with a graphical user interface (GUI) to customize its behavior. This allows the network administrator to have full control on the data sent to the clients within its intranet work. The current implementation is using C# as the programming language and is developed on the .NET platform due to the centralized approach and powerful object oriented features. The application can handle most web pages, including those with badly formatted HTML, the page is parsed down into individual elements using DOM tree approach. Depending on the type and complexity of the web page, the content extraction suite can produce a wide variety of output. The approach performs well on pages with large blocks of text such as news articles and mid-size to long informational passages. Most navigational bars and extraneous elements of web pages such as advertisements and side panels are removed effectively.

Application Architecture Diagram

![Application Architecture Diagram](image)

Figure1. Application Architecture Diagram

IV.EXPERIMENTATION AND RESULT

The web page generated because of the user query is transferred to HTML Parser to generate a hierarchical tag tree with nested object nodes. According to our analysis, group of data records which are of similar type are being placed under one parent node in the tag tree. All the internal nodes of the tag tree are marked as HTML or XML tag nodes and all the leaf nodes of the tag tree are marked as data or content nodes (Content nodes can be any of text, number or MIME data). HTML parser also deals with any type i.e. HTML tag error resiliency. The HTML parser used will
generate independent tag trees for every web page linked to the web site. So we will generate a procedure to integrate all the tag trees into a single tree, having common features of all the web pages.

This integrated tree will help us in analyzing the content and structure of the web page. WEB Content Extractor is tested with different categories of 50 web sites. In these web sites total 142 Image advertisements are found in ordinary web browser and 139 image advertisements are removed from proposed tool WEB content Extractor. There isn’t any sort of preservation of objects that may be lost after the HTML is passed through our parser. The user would have to change the settings of the proxy and reload the page to see changes.
These examples depict the Contents of a web document before and after the extraction settings were applied. The filtering method is based on examining the HTML DOM tags which we obtain as a result of the IHTML interface document from the response. In order to analyze a web page for content extraction, the page is passed through an HTML parser that creates a Document Object Model tree. Microsoft’s HTML parser (which is used in Internet Explorer) is much more robust and support a wider variety of content. Integration will be accomplished by porting the existing proxy to C#/NET, which will allow for easy integration with COM components (of which the MS HTML parser is one). The application begins by starting at the root node of the DOM tree (the <HTML> tag), and proceeds by parsing through its children. Based on the content extractor settings by the network administrator, the application is removing/filtering the elements with those particular tag name and other attributes.

IV. CONCLUSION

We summarize in brief the contributions and formulate the conclusions that can be drawn from the results. In terms of feature used for the classification, we found that by using several strong features may produce satisfying result. We are able to improve the network bandwidth requirements using this application at an intranet level. As the results from the feature selection for the segment training, the features such as InnerHTML, OuterHTML and Tag ID’s provides an effective way to implement highly efficient Web Content Extraction. The web content extraction by applying plain classification namely segment classification and content classification turned out to yield relatively good result in terms of recall. However, we realized that by only performing classification is not sufficient, in particular if we want to get high score of precision. The reason is, often inside a DOM node which is classified as main content there can be many embedded noisy contents. This case is common for modern web documents as advertisements, link to related articles, comments etc often placed together with the main content in the same DOM node. Therefore, we think the classification approach is relatively good to identify which DOM node contains the main content however for extracting the exact Information from the main content we have to add something else namely heuristic rules. Another aspect of the current approach is that it identifies and filter-out the elements based on some general properties, which sometimes result in filtering of some information relevant for a particular end user. Thus we need to identify more sophisticated and efficient heuristic approach algorithms to extract noisy data, due to the complications of structure of modern WEB documents.

V. FUTURE WORK

In our current work, we focused on the general structure of different web documents by using the DOM model of page segmentation and using element classification techniques to filter out and determine the noisy data from the web pages. In this approach the filtering rules applies to a number of users within the corporate Intranet. Thus we are working on approach which can extract the web elements based on an individual end user. It is also interesting to complement the web content extraction module with a tracking module that can monitor the structure changes of the web page and the extraction result.

VI. REFERENCE


Web Content Extraction to Facilitate Web Mining


