International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume 2, Issue 3 (March'2016)

REMOVAL OF MALICIOUS SIDE INFORMATION IN WEB DOCUMENTS

E. Silambarasan,
Department Of Information Technology
Saranathan College of Engineering,
Tiruchirapalli, Tamilnadu, India.

M.Sindhuja,
Department Of Information Technology
Saranathan College of Engineering,
Tiruchirapalli, Tamilnadu, India.

T.Pragathi,
Department Of Information Technology
Saranathan College of Engineering,
Tiruchirapalli, Tamilnadu, India.

S.Siva Abbirammi,
Department Of Information Technology
Saranathan College of Engineering,
Tiruchirapalli, Tamilnadu, India.

K.Gayathri,
Department Of Information Technology
Saranathan College of Engineering,
Tiruchirapalli, Tamilnadu, India.

Abstract: Web Page Noise Cleaning is one of the new research areas of study for removing the noise patterns of web pages for effective web mining. The World Wide Web contains large amount of web pages which are accessible to users. With conventional data or text, Web pages generally contain a large amount of noise information that is not part of the main contents of the web pages, e.g., advertisement banners, navigation bars, and disclaimer/copyright notices. The main objective of this area is removing such irrelevant information (i.e. Web Page Noise or Local Noise) in Web pages that can seriously harm Web mining task such as clustering and classification etc. For detection and removal of noises a new DOM tree structure is proposed. After DOM tree construction, we can implement DUSTER framework for crawling the document using normalized rules. The result shows the remarkable increase in F score and accuracy is obtained. In this work, we focus on detecting and eliminating local noises in Web pages to improve the performance of Web mining that is Web page clustering and classification. Then our experimental results show that improved performance at the time of classification and clustering.

Keywords: Noise cleaning, DOM tree, DUSTER , web mining, clustering, classification

I.INTRODUCTION

The overall goal of the data mining process is to extract information from a dataset and transform it into an understandable structure for further use. Classification is a classic data mining technique. Basically classification is used to classify each item in a set of data into one of predefined set of classes or groups. Classification method makes use of mathematical techniques such as decision trees, linear programming, neural network and statistics. Clustering is a data mining technique that makes meaningful or useful cluster of objects that have similar characteristic using automatic technique. Web mining - is the application of data mining techniques to discover patterns from the Web. According to analysis targets, web mining can be divided into three different types, which are Web usage mining, Web content mining and Web structure mining.

The main disadvantage of viewing webpages are its page loading efficiency because of unwanted noise(advertisement banners, navigation bars and disclaimer/copyright notices). This unwanted information leads to the loss of resources, page loading time, quality of webpage. The limitation of the existing webpages enhancement system detect and eliminates the unwanted noise from the sides of the webpage and it doesn't block the graphical images found in the side information.

Current research on DUST detection can be classified into two main families of methods: content-based and URL-based. In content based DUST detection, the similarity of two URLs is determined by comparing their contents using syntactic or semantic evidence as shingles, text signatures, pair-wise similarities, sentence-wise similarities, and semantic graphs [2]. Web page noise cleaning is one of the new research areas of study for removing the noise pattern of web pages for effective web mining. Web pages normally contains a large amount of noise information that is not part of the main contents of the web pages e.g., advertisement banners, navigation bars, and disclaimer/copy write notices. Web noise can be categorized into two types: global noise and local noise. The main purpose of the proposed system is to remove such irrelevant information (i.e. web page noise or local noise). In web pages that can seriously harm web mining task such as clustering and

classification etc. For detection and removal of noises a new DOM tree structure is proposed. After DOM tree construction we implement DUSTER framework for crawling the document using normalized rules.

II. RELATED WORK

A large number of URLs collected by web crawlers correspond to pages with duplicate or near-duplicate contents. To crawl, store, and use such duplicate data implies a waste of resources, the building of low quality rankings and poor user experiences. To deal with this problem, the existing method used normalizations rule to transform all duplicate URLs into the same canonical form. A challenging aspect of this strategy is deriving a set of general and precise rules.

In [6] they have used a approach called DUSTER to derive quality rules that take advantage of multi-sequence alignment strategy. The DUSTER method takes the advantage of multiple sequence alignment in order to obtain a smaller and more general set of normalization rules by applying multiple sequence alignment they identified similarities and differences among strings.Existing method is able to generate rules involving multiple DNS names and has an acceptable computational cost even when crawling in large scale scenario. In [1]this paper URL are aligned using algorithms namely Principal coordinates Analysis and Multi-dimensional scaling(MDS). In this method it examines the URL only upto two substrings. Duplicate URLs will be eliminated .The repeated URLs of same webpage will be represented by a single URL in the web history by special characters such as *.It compares URLs only up to two substrings.

In[2] the URL alignment is applicable only to the URL and not to the substring i.e., the substring will not be taken into considerations. This paper eliminates the URL based DUST removal only in particular field e.g., science, politics. Diverse source can not be integrated using [2]. For detecting duplicate URLs two algorithm has been implemented namely DUST BUSTER and fuzzy semantic based string similarity approach. This method is difficult to remove miscellaneous information in the pages and extract the texts. In [3] repeated duplicate URLs are detected using DUST BUSTER and URL canonization algorithms. Canonical terms are predicted based on similar text matching. In this the crawling overhead is difficult to reduce. Problem raises on mining sight-specific DUST rules. In [4] generic tokenization and host specific tokenization techniques are used. It synchronize each and every URL and predict duplicate web pages. The computational complexity is high and discovery of substring substitution rules is difficult. In [5] pair-wise rule generation technique is used. In this rules from URLs are mined and utilize these rules for de-duplication using URL strings. This method provides a large number of false positives and coverage threshold is not sufficient.

III.CONTENT BASED DUST DETECTION

The large amount of noise information that is not part of the main contents of the web pages, e.g., advertisement banners, navigation bars, and disclaimer/copyright notices. The main objective of this area is removing such irrelevant information. This is done using content based DUST detection by comparing the content of the two URLs.

A.Web document acquisition

In this framework the input is fetched. The data sets are fetched as web documents. Web documents contain HTML and URL codes and so on. Then from the uploaded datasets the codes will be read. In this module the concept of URL alignment has been implemented. In URL alignment the URLs are aligned in separate clusters and consensus sequence will be generated as a result of the alignment. Before the URL alignment, URL tokenization will be carried out. Similar to URL alignment the content of the web documents will be aligned and stored for the further use.

B.Dom implementation

DOM tree construction provides a hierarchical structure of the contents. The Document Object Models are used to build document, navigate their structure, and add, modify, or delete elements and content. In the DOM, documents have a logical structure which is very much like a tree; to be more precise which is like a "forest" or "grove", which can contain more than one tree. In this framework the Document object tree will be generated. It is commonly used for representing the structure of the web page. Each HTML page corresponds to a DOM tree where tags are internal nodes and the detailed texts, images are the leaf nodes. The DOM tree is implemented in a hierarchical structure. The main purpose of DOM tree implementation is that it will be easy to predict the alignment of the codes. The HTML code separation will be carried out in the module.

C.Duster framework construction

In this module candidate rule generation algorithm has been implemented. In candidate rule generation algorithm for URL based DUST detection it consist of two stages. i) generation of candidate rules, where a multi-sequence alignment algorithm generates candidate rules from dup-cluster and rules validation, where DUSTER filter out candidate rules according to their performance in a validation set.

CandidateRules generation algorithm

Input: Training Set T S = {c_1, ...,c_n} with n duplicate clusters
Output: Set of m candidate rules CR = {r_1, ...,r_m}
1: Create table RT (context, transformation, domains, support)
2:Create table CRT (context, transformation, domains, support)
3: for all clusters $c_i \in$ T S do
4: T = selectKRandomlyURLsFrom(ci)
5: π = MultipleURLAligment(T)
6: r = generateRule (π.consensus)
7: add (r.context, r.transformation, π.domains, π.support) to RT
8: end for
9: group tuples in RT into buckets by (context, transformation)
10: for all buckets B do
11: if (|B| >= min_{freq}) then
12: D_{omains} = ∅; S_{upport} = ∅;
13: for all tuples t ∈ B do
14: $D_{omains} = D_{omains} \cup t.domains$
15: $S_{upport} = S_{upport} \cup t.support$
16: end for
17: α = the first tuple in B
18: add (α.context, α.transformation, D_{omains}, S_{upport}) to CRT
19: end if
20: end for
21: return a set CR of rules created from CRT

In this module the codes are clustered and saved. In this there are two types of codes namely basic codes, web design code. The basic code consist of basic tags such as div, ul, li, p, h1, span. The web design code consist of main phase such as imgsrc, table, hyperlink.

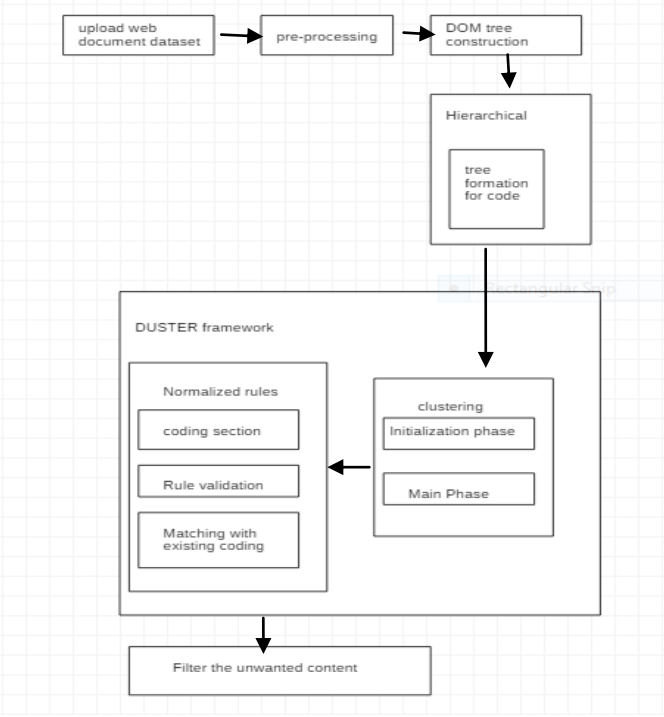
D. Content Prediction

In this framework DUSTER can be used to find and validate rules, by splitting it in training and validating sets. The resulting rules are then used to normalize the known Contents yielding a new (and reduced) set of Contents to be crawled. By using this set and the set of DUST rules, the crawler can gather new Contents, closing the cycle. Candidate table is given as input, the candidate table consist of the coding of the web page(Advertisements, Invoices, alert messages) that is added to the coding of our own webpage.

E. Evaluation criteria

Clustering and Classification are performed to find the accuracy of elimination of web noise from the webpage. Clustering is a popular strategy for implementing parallel processing applications .It is the process of grouping a set of objects into classes of similar objects.Classification is a classic data mining technique based on machine learning. Basically classification is used to classify each item in a set of data into one of predefined set of classes or groups. Classification method makes use of mathematical techniques such as decision trees, linear programming, neural network and statistics. In classification, we make the software that can learn how to classify the data items into groups. Clustering is a process which consist of two phases namely the initialization phase and the main phase. The initialization phase consist of the basic coding of the HTML page whereas the Main phase consist of the coding's like code for images, hyperlinks etc.,It ensures that all the noise are eliminated from the webpage. This is represented with the help of graphical representation. This framework helps to ensure that noises are eliminated from the web page.

IV. SYSTEM OVERVIEW



Web pages generally contain a large amount of noise information that is not part of the main contents of the web pages, e.g., advertisement banners, navigation bars, and disclaimer/copyright notices.

A. Upload web documents ,pre- processing

In this the web documents are uploaded, that is the website coding of our webpage is uploaded. This consist of HTML coding, URLs. The uploaded web documents are than pre-processed i.e., the web documents are tokenized, the coding of the web documents are tokenized into small tokens which are further processed. This content tokenization separates the larger content into smaller contents for processing.

B.DOM tree construction

DOM tree construction provides a hierarchical structure of the contents. The Document Object Models are used to build document, navigate their structure, and add, modify, or delete elements and content. In the DOM, documents have a logical structure which is very much like a tree; to be more precise which is like a "forest" or "grove", which can contain more than one tree. Each document contains zero or one document type nodes, one root element node, and zero or more comment processing instructions; the root element serves as the root of the element tree for the document. One important property of DOM structure models is structural isomorphism- If any two Documents object Model implementations are used to create a representation of the same document, they will create the same structural model, in accordance with the XML information set. Thus the DOM structure can be used to represent the hierarchical representation of the HTML code which can be used to compare with the DOM structure of the added noise i.e. the adds, bannersetc.,

C.Duster framework

Clustering is a popular strategy for implementing parallel processing applications .It is the process of grouping a set of objects into classes of similar objects. Clustering is a process which consist of two phases namely the initialization phase and the main phase. The initialization phase consist of the basic coding of the HTML page whereas the Main phase consist of the coding's like code for images, hyperlinks etc. Then in the coding selection process the coding of the added noise like ads , disclaimers/copy. The rule validation compares the candidate table value with rule table. The clustered code then will be compared with the actual code, if the code matches it shows that noise is not added to our webpage otherwise if code doesn't matches it shows that the noise has not been added to our webpage and the web page is said to be noise free. Upon detection of the noise an intimation will be send to the admin about the added noise, the added noise code will be then blocked.

V.CONCLUSION

The information that webpage contains noises and non-essential information. These noises can be the pop-up advertisements,non-essential images and navigation links around the frame of the main content that a user does not need. So, inorder to make the webpages efficient, the noises has to be removed. The noise has been removed by implementing DUSTER framework , DOM implementation, clustering, comparing and eliminating the code from the web page. Images in the noise will also be eliminated. The addition of unwanted content will be send as an intimation to the admin of the webpage.

VI. REFERENCES

[1] A. Agarwal, H. S. Koppula, K. P. Leela, K. P. Chitrapura, S. Garg, P. K. GM, C. Haty, A. Roy, and A. Sasturkar. Url normalization for de-duplication of web pages. In Proceedings of the 18th ACM conference on Information and knowledge management, CIKM '09, pages 1987–1990, New York, NY, USA, 2009. ACM.

[2] B. S. Alsulami, M. F. Abulkhair, and F. E. Eassa. Near duplicate document detection survey. the Proceedings of International Journal of Computer Science and Communications Networks, 2(2):147–151, 2012.

[3] Z. Bar-Yossef, I. Keidar, and U. Schonfeld. Do not crawl in the dust: Different urls with similar text. ACM Trans. Web, 3(1):3:1–3:31, jan 2009.

[4] G. Blackshields, F. Sievers, W. Shi, A. Wilm, and D. G. Higgins. Sequence embedding for fast construction of guide trees for multiple sequence alignment. Algorithms MolBiol, 5:21, 2010.

[5] Kaio Rodrigues, Marco Cristo, Edleno S. De Moura, Altigran da silva. Removing DUST using Multiple Alignment of Sequences. August 2015.

[6] H.S .Koppula, K.P. Leela, A. Agarwal,K..P. Chitrapura, S.Garg, and A.Sasturkar. Learning url patterns for webpage de-duplication. In Proceedings of the third ACM international conference on web search and data mining, WSDM '10,pages 381-390,Newyork,NY,USA,2010. ACM.

[7]http://www.google.com