JQCM
Journal Of Al-Qadisiyah For Computer Science And Mathematics
University Of AL-Qadisiyah

Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

MINISTRY OF HIGHER EDUCATION
& SCIENTIFIC RESEARCH
AL-QADISIYA UNIVERSITY
COLLEGE OF COMPUTER SCIENCE & INFORMATION
TECHNOLOGY
AL-Qadisiya Journal
for computer science & mathematics
PERIODICAL SCIENCE COMPACT
ISSUED BY COLLEGE OF COMPUTER SCIENCE &
INFORMATION TECHNOLOGY
THE DEPOSITING NUMBER OF THE LIBRARY &
DOCUMENTS 1386
ISSN: 2074 – 0204
VOL. 8 NO. 2 2016

# Dom Tree as the base for webpage content extraction: Review

***Hind Sabah Rahim[a], Aliea Salman sabir [b], Nahla Abbas Flayh [c]***

[a] *Department of Computer information systems, College of Computer Science and Information Technology, University of Basrah, Iraq, itpg.hind.sabah@uobasrah.edu.iq*

[b] *Department of Computer information systems, College of Computer Science and Information Technology, University of Basrah, Iraq, aliea.sabir@uobasrah.edu.iq*

[c] *Department of Computer information systems, College of Computer Science and Information Technology, University of Basrah, Iraq,nahla.flayh@uobasrah.edu.iq*

ARTICLE INFO

*Article history:*
Received: 14 /06/2022
Rrevised form: 18 /07/2022
Accepted : 24 /07/2022
Available online: 12 /08/2022

*Keywords:*

Webpages

Structured information

Information extraction

DOM Tree

HTML tags

ABSTRACT

Because of the fast advancement of internet technology in the last twenty years, which leads to a huge number of web pages that contain a massive amount of information in every domain, the volume of available information has been steadily expanding every minute, so the analyzing and extracting information from web pages is becoming increasingly crucial, add to that information in webpages in an unstructured or semi-structured format need to transform in a structured format.

Since it is hard to collect the information manually, scientists have devised a variety of methods to help extract information from different domains in an automatic way. the main information in web pages is mixed with a significant amount of unrelated information (noise) like advertisements, boxes with links to relevant material, boxes with photos or other media, top and/or side navigation bars, animated commercials, etc., effect on the performance of information extraction and web content analysis technologies. to eliminate the noise by using the Document Object Model (DOM) that can easily reach every tag in the structure of the webpages to extract the information or delete the noise.

This article explores in-depth DOM tree-based approaches, such as HTML tags and the DOM tree, by reviewing works from 2011 to 2021 and comparing numerous elements comprehensively, including classifier methods, contribution, limitation, and evaluation metrics.

MSC.

https://doi.org/10.29304/jqcm.2022.14.3.985

## 1. INTRODUCTION

In the last decades, the internet has become necessary in our lives. It enters every corner of our life. Health has seen an explosion in digital information stored in Electronic Health Records (EHRs) that need to process automatically [1].

∗Corresponding author

Email addresses:

Communicated by ‘sub etitor’

In business, education, and social media, there is an opinion for people to know more about the product's quality, allowing them to decide whether to buy, manufacture, and market it[2]. That may cause a massive amount of data in an unstructured format that is complex to process manually into a structured format. Human beings transformed from the industrial to the information age through computers and information technology [3].

So that opens the field to extracting information from webpages and building methods to have structured data extracting information differs from information retrieval.
An information retrieval (IR) system determines information in unstructured or semi-structured document collections pertinent to the user query[4].
Information extraction (IE) automatically gathers knowledge and relations from a set of documents or raw text[5].
In another definition, information extraction is an information extraction from unstructured data to find structured data, find entities, and classify and store them in a database[6].

Web information extraction (WIE) Identifies what parts of a webpage comprise the main textual content, thus discarding additional contexts such as menus, status bars, advertisements, and sponsored information [7]. These are called noise to the informative information that the user needs. the main content extraction from web pages has become more complex and essential [8].

The noise affects the efficiency and accuracy of algorithms to classify, cluster, and extract the information on the web page, so the need to filter the DOM tree can do this job. Most of the content of the web pages is noise, about 40% - 50% [9].

Most methods rely on the web source code parsed into Document Object Model, and the DOM tree is the programming interface application of HTML and XML for web pages.

The DOM model explains the behavior of the node object in addition to the document's structure. quickly access, alter, add, and delete the nodes and contents of the DOM Tree using the methods and properties of objects [10].

This article is a review of research from 2011 to 2021 that explains techniques of IE based on the Dom tree.

.

## 2. RELATED WORK

Numerous approaches have been suggested to extract the information from the web in the last years, differing in techniques, degrees of automation, and human supervision. This section presents a survey of previously proposed approaches for web information extraction based on the Dom tree made by the researcher.

In [16], the authors presented (CETD) Content Extraction via Text Density, a speed, accurate and general method for extracting content from varied web pages. By determining the nodes if content text or noise depends on the threshold, if the text density of the node is greater than or equal to the threshold, it is the content node less than the threshold; it is noise. The problem faced by the authors is that some nodes containing pictures, hyperlinks, etc., have unnatural text density values that are not like the nodes around them. To solve this problem, they propose a technique called Density-Sum. In this approach, the body is a content node because of all the text in the body branches. Density-sum: a technique that notices a web page content block belongs to an ancestor node in the structure of DOM, the content block gets a high value when adding the children's text densities.

In [12]method that extracts important content to the user from the webpage based on DOM tree cleared in several steps: Cleaning to ensure that the Html tags are correct, preprocessing to deleting all the tags that do not have text, determining the place of the optimum node to be extracted, Extract the node's content using tools such as Html parser and then select the nodes with satisfying content.

In [14], the authors explained that the document consists of HTML tags and elements, text nodes, and comment nodes. The Dom tree parses these documents to the tree with objects and manipulates the object through the Dom tree (delete, add, edit, etc.). Authors proposed and redesigned an approach to solve the problem that can extract the information from web pages with different domains like news websites, Blogs, and Forums, each with a different structure. By introducing two different approaches: Extraction of content using a stored URL list which extracts the content from the stored list like a list containing just webpages for news or any domain. And the extraction of content using runtime generated URL list. This approach is helpful if the user does not know the specific URL. In this situation, information will be extracted from the runtime generated lists, i.e., through search engines like GOOGLE and others.

The authors in [20]suggested the PAREX method. The proposed method focused on paragraph tags to get the content, which they figured out in steps: in the preprocessing step, they used the JSoup API to read the HTM code from the websites to make the filter, and in this step, they figured out which tags were the parent tags. Then Clustering <p> tag step for text-to-tag ratio, then Parent Header step to finish optimization of method PAREX that met the following criteria: first, <p> tags must be the main content that webpages use, and second, use webpages with limited user comment sections, like news webpages.

In [13], the authors proposed Heuristic Algorithm based on the DOM tree. They worked on scientific journals to extract their information, surmised as steps, input to the system is URL of the Publisher's web site. After that heuristic algorithm crawl in the base URL applies to all the journals linked to the Publisher's website is retrieved. When the DOM tree is constructed to represent the journal's home page, then the works focused only on the leaf nodes of the DOM tree are filtered and contrasted with domain terms like Impact Factor, SJR, and SNIP to determine where the target information is located, which journals features. Finally, location is represented as XPATH expression, and in RDBMS information is gathered and saved in a structured form.

The authors of [17] suggested a method based on the DOM tree with statistical information to extract the content with high performance despite the difficulty given by the continuously changing structure of web pages. The method called(CEDS) content extraction based on the DOM structure and statistical information gets higher results; the domain is the news webpage, which summarizes in many steps, beginning by parsing the web page into the DOM tree after that made to the blocks the technique de-noising to the content block which cleared from any noise.

The authors of [11]developed a method for extracting information from the DOM tree using machine learning and characteristics from the tree. They concentrate on the document's HTML tags and the properties that separate tags into two categories: tags that contain text and tags that do not. Textual features are derived from a node's class and id properties and numeric features. Researchers examined the performance of numerous models in their article, including SVM, decision trees, random forests, and basic multilayer perceptron (MLP). In their tests, they employ the Cleaneval and Dragnet datasets.

In [15], researchers developed a D-rank method to extract keywords from a web page, which deepened on information and the web feature after parsing to the DOM tree. In their research, they used language-independent features so that multiple languages might be used with the technique, not by NLP. The method is summarized as preprocessing the HTML content to extract nominee keywords. various scores go to the words based on the information from HTML tags, which specify the positions of words. then chooses the top 10 possibilities as the webpage’s representative keywords. In the D-rank method: Word Position and Frequency, Keyword Extraction Process, the main steps in the method. By constructing the DOM tree of the webpage, extract text from HTML content.

Cleaning the text from symbols, tokenizing the text to have individual words, set each kind of word in lists. an essential step in keyword extraction methods is removing stop words from the list of words (stop words: words constantly recur in the language)

In many cases, the web page often contains more than one content block; if the text density is equal to or greater than the threshold, the same method is used to extract content. This block is lost if a content block under the body tag is less than the threshold (text density for body tag). They find a solution with the technique Density-Sum.

The author of [19]presented a text extraction technique that merges many features. The algorithm contains five steps: The input HTML page is cleaned by eliminating elements that do not influence the information, such as style and script tags, and then turned into a DOM tree object. The next step is node feature representation and extraction, which divides its task into four subtasks: label semantic, Node Statistical Characteristics, **-** Heuristic Higher-Order Features, and finally Neural Network Classifier, which uses three features (semantic, statistical, and heuristic) as input to determine whether a node is a text or noise. The technique was evaluated on news, policy, and blog sites. The outcome demonstrates that the neural network that includes various variables is more adaptable to varied page kinds; learning the training set eliminates human threshold selection and yields the best performance.

In [18]the authors proposed a (Cova) Context-aware Visual Attention-based method. They try to imitate the ability of humans to recognize visual objects, for example, where are images and price advertisements. So, extracting the information with all the challenges of natural language processing must be solved, which makes obtaining optimal results difficult. In addition, the nature of a webpage has many types of noise and unrelated information. The steps of the method begin with the screenshots of a webpage, and they made lists of bounding boxes of the web element and neighborhood information for each element obtained from DOM. The (Cova) process step completes its task in four stages, the graph representation extraction for the webpage, the Representation Network(RN), and the Graph Attention Network (GAT). Finally, the fully connected layer (FC) is the last step, the dataset generation, and they generate their large-scale dataset for object detection on products webpage screenshots.

Table 1 **-** summarizes all the related work from different aspects like the classifier technique, contributions, limitations, and evaluations measures to be easier for the reader to extract information

**Table 1- related work summary**

| Work | Research year | Author name | Method name | Classifier<br>Technique | Contribution | Limitation | Measures |
|---|---|---|---|---|---|---|---|
| [16] | 2011 | Fei Sun, Dandan Song_ and Lejian ,Liao | CETD | DOM tree with Text Density | Practical, straightforwa rd in concept and impletion | Some pages are not parsed. Invisible elements on the page | Precision, Recall, F1 |
| [12] | 2014 | Pranjali G. Gondse1 Anjali B. Raut2 | informat ive content<br>extract | DOM tree features | used in information<br>retrieval | need to determine the optimum nodes | Precision, Recall, F1 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| [14] | 2015 | Bhavdeep Mehta, Meera ,Narvekar | | DOM tree with node and threshold concept | significant advantages in both<br>precision and .recall | use more features from the webpage | Precision, Recall, False Alarm,<br>Detection |
| [20] | 2016 | Howard J. Carey, Milos Manic | Parex | DOM tree with paragraph tags | Work with different website styles and designs | need to improve the clustering algorithm | Precision, Recall, F1 |
| [13] | 2017 | Umamageswari Kumaresan, Kalpana Ramanujam | | DOM tree with a heuristic approach | Extract hidden data from a webpage | extracted data must be enclosed with HTML tags, heterogeneous formats, | Precision, Recall, F1 |
| [17] | 2017 | Xin Yu, Zhengping Jin | | DOM Tree and Statistical<br>Information | Improving the accuracy and<br>The versatility of the extraction | use more features from the webpage | Precision, Recall, F1 |
| [11] | 2018 | Nichita Ut,iu, , Vlad-Sebastian ,Ionescu | | DOM tree features with machine learning | Obtain good performance without preprocessing steps | prone model instability, use less statistical techniques | F1 |
| [15] | 2019 | Himat Shah, Mohammad Rezaei , Pasi ,Fränti | D-rank | DOM tree features of a webpage | Fast, practical for different languages in webpages | Use more features from the webpage | Precision<br>Recall<br>F-score |
| [19] | 2019 | Bowen Yu, Junping Du, Yingxia Shao | | DOM tree with multi-features fusion, semantic, statistical | avoids manually determining the threshold | Lack of open data sets for training | accuracy, recall, F1 |
| [18] | 2021 | Anurendra Kumar, Keval Morabia, Jingjin Wang, Kevin Chen-,ChuanChang<br>Alexander Schwing | Cova | appearance features with the syntactical structure of the DOM tree | Able to deal with important context | Use more features from the webpage | Precision, Recall, F1 |

## 3. Conclusion

This study provided a summary of related works dependent on the DOM tree. And seeing how successful the DOM tree is in extracting information makes it easy and quick to access the information with great precision.

Most of these works begin by parsing web pages and removing page noise by utilizing the DOM tree's characteristics, followed by ways to extract information.

Using the feature of DOM tree combined with multi-features fusion, semantic, and statistical to extract webpage information achieved high performance and accuracy.

## References

[1] N. Noori and A. Yassin, "Towards for Designing Intelligent Health Care System Based on Machine Learning," *Iraqi J. Electr. Electron. Eng.*, vol. 17, no. 2, pp. 120–128, 2021, doi: 10.37917/ijeee.17.2.14.

[2] M. M. Almosawi and S. A. Mahmood, "Lexicon-Based Approach For Sentiment Analysis To Student Feedback," vol. 19, no. 1, pp. 6971–6989, 2022.

[3] Z. Shu and X. Li, "Automatic Extraction of Web Page Text Information Based on Network Topology Coincidence Degree," *Wirel. Commun. Mob. Comput.*, vol. 2022, 2022, DOI: 10.1155/2022/9220661.

[4] Z. A. Khalaf and I. A. Sheet, "News retrieval based on short queries expansion and best matching," *J. Theor. Appl. Inf. Technol.*, vol. 97, no. 2, pp. 490–500, 2019.

[5] J. B. Agbogun and V. A. Akpan, "On the Development of Machine Learning Algorithms for Information Extraction of Structured Academic Data from Unstructured Web Documents," no. October 2021.

[6] "What is Information Extraction? | Ontotext Fundamentals." https://www.ontotext.com/knowledgehub/fundamentals/information-extraction/ (accessed Jun. 22, 2022).

[7] S. López, J. Silva, and D. Insa, "Using the DOM tree for content extraction," *Electron. Proc. Theor. Comput. Sci. EPTCS*, vol. 98, no. Www, pp. 46–59, 2012, DOI: 10.4204/EPTCS.98.6.

[8] D. Song, F. Sun, and L. Liao, "A hybrid approach for content extraction with text density and visual importance of DOM nodes," *Knowl. Inf. Syst.*, vol. 42, no. 1, pp. 75–96, 2015, DOI: 10.1007/s10115-013-0687-x.

[9] D. Gibson, K. Punera, and A. Tomkins, "The volume and evolution of web page templates," *14th Int. World Wide Web Conf. WWW2005*, pp. 830–839, 2005, DOI: 10.1145/1062745.1062763.

[10] Y. F. Lou, Y. C. Zhang, and Z. J. Yuan, "Website information extraction based on DOM-model," *Appl. Mech. Mater.*, vol. 347–350, pp. 2889–2893, 2013, DOI: 10.4028/www.scientific.net/AMM.347-350.2889.

[11] N. Utiu and V. S. Ionescu, "Learning web content extraction with DOM features," *Proc. - 2018 IEEE 14th Int. Conf. Intell. Comput. Commun. Process. ICCP 2018*, no. February, pp. 5–11, 2018, DOI: 10.1109/ICCP.2018.8516632.

[12] A. B. Raut, "Main Content Extraction From Web Page Using," vol. 3, no. 3, pp. 5302–5304, 2014.

[13] K. Umamageswari and R. Kalpana, "Web data extraction from scientific publishers' website using a heuristic algorithm," *Int. J. Intell. Syst. Appl.*, vol. 9, no. 10, pp. 31–39, 2017, DOI: 10.5815/ijisa.2017.10.04.

[14] B. Mehta, "Extraction," 2015.

[15] H. Shah, M. Rezaei, and P. Fränti, "DOM-based keyword extraction from Web pages," *ACM Int. Conf. Proceeding Ser.*, 2019, DOI: 10.1145/3371425.3371495.

[16] F. Sun, D. Song, and L. Liao, "DOM-based content extraction via text density," *SIGIR'11 - Proc. 34th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, vol. 1, pp. 245–254, 2011, DOI: 10.1145/2009916.2009952.

[17] X. Yu and Z. Jin, "Web content information extraction based on DOM tree and statistical information," *Int. Conf. Commun. Technol. Proceedings, ICCT*, vol. 2017-October, pp. 1308–1311, 2018, DOI: 10.1109/ICCT.2017.8359846.

[18] A. Kumar, K. Morabia, J. Wang, K. C.-C. Chang, and A. Schwing, "CoVA: Context-aware Visual Attention for Webpage Information Extraction," pp. 1–11, 2021, DOI: 10.18653/v1/2022.ecnlp-1.11.

[19] B. Yu, J. Du, and Y. Shao, "Web Page Content Extraction Based on Multi-feature Fusion," no. 61772083, 2022, DOI: 10.7544/issn1000-1239.201.

[20] H. J. Carey and M. Manic, "HTML web content extraction using paragraph tags," *IEEE Int. Symp. Ind. Electron.*, vol. 2016-Novem, pp. 1099–1105, 2016, DOI: 10.1109/ISIE.2016.7745047.