Soft Computing (2023) 27:14631–14645
https://doi.org/10.1007/s00500-023-09076-x

MATHEMATICAL METHODS IN DATA SCIENCE

An efficient content extraction method for webpage based on tag-line-block analysis

Zeqiu Chen[1] · Jianghui Zhou[2] · Ruizhi Sun[1,3]

Accepted: 29 July 2023 / Published online: 24 August 2023

Abstract
World Wide Web is a vast information resource that can be used in a broad range of applications. Web content is an efficient way to derive valuable information from webpages, and many efforts have been made on this subject. However, due to the increasing complexity of webpage technology, the existing methods cannot match quite well the requirements for the content extraction of webpages. This paper proposed an improved content extraction method for webpage based on Cx-Extractor, which is capable of dealing with content extraction for different types of webpages. Several improvements have been made for the proposed method: (1) The hyperlink tags are not removed directly to avoid mistaking the dense hyperlink groups for the main content. (2) The starting point of the main content is taken as the line number of tag-line-block whose size exceeds the threshold and thus the first few short texts of the main content can be retained. (3) The threshold value of tag-line-block for the main content is calculated automatically instead of being set manually. The above can improve the accuracy of the extracted content. Moreover, (4) the blank spaces in the original text of webpage are retained, which can increase the readability of the extracted content by avoiding connecting English words into pieces. (5) The multimedia information (e.g., pictures and videos) can be selectively retained by users, allowing for maximum flexibility and usage in multiple industries. The experimental results conducted on real-world webpages show that the proposed content extraction method works well for both single-content and multi-content webpages. Furthermore, the performance of the proposed content extraction method was compared with the Chinese extraction method called Cx-Extractor and the English extraction method called Readability. It is found that the proposed method in this study outperforms these two methods in precision, recall, and readability. In addition, the extraction efficiency of the proposed method is superior to that of the Readability method.

Keywords Web content extraction · Tag-line-block distribution function · Tag semantic information · Automatic threshold setting

1 Introduction

With the continuous development of network technology and information technology, the Internet has gradually become a vital resource to provide rich information. The total amount of data created, captured, copied, and consumed worldwide was estimated to be 97 zettabytes in 2022, as shown in Fig. 1, which may exceed 180 zettabytes by 2025 (IDC and Statista 2022). The ever-increasing amount of data has led people to focus on collecting data from the Internet and using it to create greater value (Ramakrishna et al. 2010). For example, in natural language processing, since the establishment of a text corpus is very expensive and time-consuming, people often use the content on the Internet as the corpus to obtain useful knowledge.

However, extracting the information of interest is a complex work because the main content is usually surrounded by various unrelated content in the webpage. Such

✉ Ruizhi Sun
sunruizhi@cau.edu.cn

[1] College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China

[2] JD Tech, Beijing 100176, China

[3] Scientific Research Base for Integrated Technologies of Precision Agriculture (Animal Husbandry), The Ministry of Agriculture, Beijing 100083, China

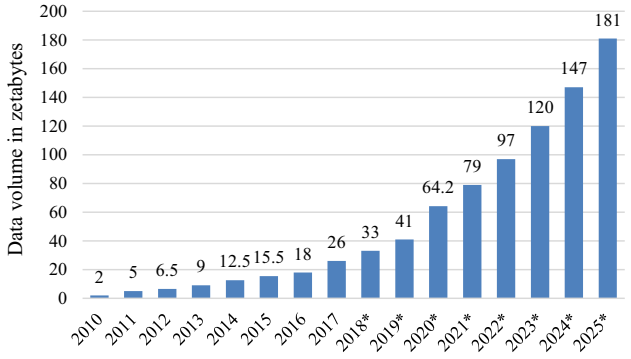


Fig. 1 Volume of data created, captured, copied, and consumed globally from 2010 to 2020, and forecasts from 2021 to 2025 (IDC and Statista 2022)

unrelated content commonly includes banners, navigation bars, commercial advertisements, user comments, and copyright information, which are usually called "noise" (Waldherr et al. 2017). It has been concluded that noises can account for almost half of the webpage content (Gibson et al. 2005). The presence of noise will not only reduce the efficiency of information retrieval, but also has a negative impact on the integrity and accuracy of the extracted information. When compiling web crawler programs, people often compile the locations of elements manually to get the required information, which will be time-consuming and needs to spend too much work on maintenance. Currently, how to update the dynamic content and extract efficient content without the noise block effect remains the most prominent challenge (Samuel et al. 2019). Therefore, it is of significance to conduct further studies on the information retrieval and webpage mining.

To extract valuable information with limited human efforts from a huge amount of information on the Internet, several content extraction methods have been developed, e.g., the Cx-Extractor method (Chen 2011) based on tag-line-block distribution function. The character distribution function of tag-line-block reflects the number of characters in each tag-line-block. Due to the continuity of the main content, the tag-line-block distribution function established in linear time can directly and accurately locate the main webpage content. However, the manual threshold setting in Cx-Extractor can greatly affect the accuracy of content extraction, and the automatic content extraction method for webpage still remains challenging. In addition, the multiple-source and heterogeneous features of new webpage will increase the extraction difficulty. Recently, plenty of studies have been conducted on this topic. However, the existing methods have not yet achieved the desired results. Moreover, most of the studies are focused on single-content webpages such as news websites and blogs (Wu 2016; Sandeep and Patil 2018), which cannot be well applied to multiple-content webpages like forums. Therefore, it is still necessary to conduct further investigations to extend the application of the existing webpage content extraction methods.

This study aims mainly to develop an improved content extraction method based on the Cx-Extractor method for different types of webpages (e.g., English webpage and Chinese webpage, single-content webpage and multiple-content webpage), which considers both the character distribution function of tag-line-block and the semantic information of HTML tag. In summary, the main contributions of the improved content extraction method are shown as followings:

(1) HTML tags are semantically analyzed, and the hyperlink tags will not be removed directly, which can avoid mistaking the dense hyperlink groups for the main content and improve the accuracy of the extracted content.
(2) The starting point of the main content is taken as the line number of tag-line-block whose size exceeds the threshold, and the size of the two adjacent tag-line-blocks after this tag-line-block should be nonzero. It can avoid missing the first few short texts of the main content and thus improve the accuracy of the extracted content.
(3) The threshold value of tag-line-block for the main content will be calculated automatically according to the characters of tag-line-block instead of being set and adjusted manually, which can enhance the stability and accuracy of content extraction.
(4) The blank spaces in the original text of webpage will be retained during the content extraction, which can avoid connecting English words into pieces and increase the readability of the extracted content.
(5) The multimedia information including the pictures and videos in the webpage can be retained according to the user's demand. Therefore, the improved content extraction method has great flexibility and scalability.

The rest of this paper is organized as follows: Sect. 2 presents a brief introduction of the existing web content extraction methods. In Sect. 3, the proposed content extraction method for webpage via the features of tag-line-block will be described. Section 4 will introduce the datasets, the evaluation metrics, the experimental settings, and the performance comparison of the proposed method with several popular methods. Finally, the findings, the limitations, and the further research directions are put forward in Sect. 5.

2 Related work

The concept of content extraction was first proposed by Rahman et al. (2001). They extracted the most important content of HTML document by segmentation summary, which can reduce the amount of the transmitted data and provide a more intuitive and faster browsing experience for small-screen handheld device users. Webpage content extraction is different from ordinary text extraction because the webpage itself conforms to the DOM standard formulated by W3C. The introduction of HTML tags will increase the difficulty of information extraction. In addition, due to the diversity of CSS styles, the structures of webpages are not always the same and even change over time, which will make it difficult to satisfy the universality requirement of content extraction. Recently, many scientific tools including machine learning, ontology, and natural language processing have been employed to extract content from the webpage (Ferrara et al. 2014; Gan et al. 2023; Joe and Surendiran 2022; Wang et al. 2022; Yunis et al. 2016). Currently, the content extraction methods can be classified as wrapper-based, template-based, statistics-based, and visualization-based methods.

2.1 Webpage content extraction via wrapper-based method

One of the most important webpage content extraction methods was the wrapper-based method. Wrapper refers to any procedure that aims to extract structured data from semi-structured or unstructured data sources in the literature (Ferrara et al. 2014). Hammer et al. (1997) proposed a wrapper method for the first time in TSIMMIS, which locates content information by manually compiling extraction rules for each information source. Then a semi-automatic XWRAP system (Liu et al. 2000) and a fully automatic RoadRunner tool (Crescenzi et al. 2001) were developed to improve extraction efficiency and to avoid manual compiling. Although the wrapper-based methods possess high accuracy, the wrapper is generally only suitable for one information source due to the complexity and uncertainty of the webpage structure. Since there exists a huge amount of multi-source information in the Internet environment, it is a difficult task to build the corresponding wrapper for each information source. In addition, once the website is updated or revised, the wrapper may have to be rebuilt, which will undoubtedly require a lot of labor costs.

2.2 Webpage content extraction via template-based method

The template-based method has also been commonly used for the content extraction of the webpage. Since the template is widely used on the Internet, the webpage can be denoised based on the similarity of the webpage. The template-based method can perform well when the webpage is composed of two parts: the template and the content. In such webpages, a similar part of the webpage often belongs to the template, while the rest is the main content. Up to date, many researches have been conducted on the content extraction of webpage based on the template-based method. For example, Tan et al. (2018) proposed a title-based web content extracting model called TWCEM to extract the webpage contents, which can effectively filter the noises and accurately locate the content positions. Gu et al. (2014) developed an algorithm based on template and domain ontology to extract deep web information. It is reported that the average accuracy rate and recall rate can achieve above 95%. Sestito and Dillon (1993) used multilayer neural networks to form the extraction rule. It is found that the neural network has a good ability to deal with noise so that users can produce the correct rule in a noisy domain. In general, the template-based method has a high accuracy for webpages generated by the same template. However, it is very difficult to construct the template. As the styles of templates become more and more diverse, the maintenance cost of the template is very high.

2.3 Webpage content extraction via statistics-based method

For the versatility problem of multi-source webpages, the statistics-based method has also been used to analyze the design rule of the webpage and determine the location of the main content. Arc90 Labs developed the Arc90 Readability algorithm, which aimed at making webpage reading more comfortable (such as on mobile devices). The Arc90 Readability algorithm uses a scoring function that rewards elements related to the main content of a website. However, it may cause the partial main content to be lost and is not suitable for multiple-content webpages. Gupta et al. (2003) determined the location of the main content of a webpage based on the proportion of hyperlink text to normal text. Generally, the probability of normal text being the main body is higher, and the probability of hyperlink text being an advertisement is higher. Sun and Guan (2004) proposed a universal webpage content extraction method called CETD based on text density where the importance of nodes was measured. Liang and Yang (2018) extracted tag features of webpage with noise information and then

obtained content based on support vector machine. Zhang et al. (2023) first efficiently exploited the structural co-occurrences over the surface form and DOM tree for extracting structured web data. However, it should be noted that most of the statistics-based methods depend on the DOM tree structures of webpages and need to deal with complex HTML tags. Moreover, the previous webpages were designed according to the rules that the main content exists in the bottom table tag (Sun et al. 2011; Yu et al. 2005), which are no longer applicable to the current rules where div and CSS have become the mainstream of web design. Chen (2011) proposed a universal web content extraction method called Cx-Extractor, which is completely separated from HTML tags and converts the content extraction problem into calculating the tag-line-block distribution function. Cx-Extractor has a high processing efficiency since it only scans the untagged text once. However, most of the statistics-based methods require manual threshold setting. If the threshold is not set properly, the accuracy of content extraction will be significantly reduced.

2.4 Webpage content extraction via visualization-based method

The actual layout of a webpage is usually quite different from the structure of HTML. Therefore, it is impossible to accurately obtain the semantic information of a webpage simply by analyzing the structure of the HTML document. Cai et al. (2003) proposed the VIPS method, which divided pages at the semantic level by making full use of the features of page layout such as blank areas, text size, color, and other visual elements. Zhang et al. (2019) proposed a framework to model and visualize location-referenced web text information. In the case of a large gap between the HTML structure and the page layout, the visualization-based method has a good extraction result. However, it relies too much on the heuristic knowledge. The diversity of webpage designs and the complexity of visual features make it necessary to adjust the rules constantly. Therefore, the construction and maintenance of webpages are very difficult.

Table 1 summarizes the comparison of the four methods for content extraction of webpages. In general, the wrapper-based method has great advantages in accuracy and construction complexity (i.e., simplicity), while it has disadvantages in universality and scalability. Except for accuracy, other performances of the template-based method are nearly opposite to the wrapper-based. The statistics-based method has great advantages in universality, construction complexity, and scalability. However, the accuracy of the statistics-based method may be poorer than that of the other methods. The visualization-based method has higher accuracy than the statistics-based method due to the introduction of semantic information in HTML tags. However, the construction and maintenance of rule base are difficult in the visualization-based method. In this study, the improved method simultaneously considers the character distribution function of tag-line-block in HTML document (statistics-based) and the semantic information of HTML tags (visualization-based) to achieve more accurate and universal content extraction for different types of webpages.

3 Proposed methods for webpage content extraction

Although there exist lots of methods to extract content from the webpage (Cardoso et al. 2011; Karthikeyan et al. 2019; Laber et al. 2009), the content extraction of webpage has not yet been addressed well due to the complexity of webpage structure. In this work, we followed the traditional tag-line-block approach and proposed an efficient but simple algorithm for different types of webpages such as single-content and multiple-content webpages displayed in English and Chinese.

To extract the core contents of webpage as accurately as possible, our method is developed based on the following practical bases:

(1) The main content is commonly situated in the middle zone of an HTML document, which includes a large number of ordinary texts in long-text form but rarely includes hyperlink texts.
(2) The noises are usually located at the top and bottom zones of a webpage, which are mostly displayed in short-text form. In addition, different noises are usually distributed dispersedly within the webpage.
(3) HTML tags contain rich semantic information and can contribute to a better understanding of the webpage layout, which can provide an alternative method for the main content extraction.

The proposed method includes four main steps: the first step refers to the removal the replacement of HTML tags. The second step is to calculate the distribution function of characters in tag-line-block of webpage. Then the threshold value of tag-line-block for the main content will be determined in the third step. In the final step, the main content can be extracted based on the distribution function of characters in tag-line-block as well as the threshold value. Detailed descriptions of all the steps and the related algorithms are shown in the following sections.

Table 1 Comparison of the four webpage content extraction methods

Method	Accuracy	Universality	Simplicity	Scalability	References
Wrapper-based	★★★	★☆☆	★★★	★☆☆	Crescenzi et al. (2001), Hammer et al. (1997) and Liu et al. (2000)
Template-based	★★☆	★★☆	★☆☆	★★☆	Gu et al. (2014), Sestito and Dillon (1993) and Tan et al. (2018)
Statistics-based	★☆☆	★★★	★★★	★★★	Chen (2011), Gupta et al. (2003), Liang and Yang (2018), Sun and Guan (2004), Sun et al. (2011), Yu et al. (2005) and Zhang et al. (2023)
Visual-based	★★☆	★★★	★☆☆	★☆☆	Cai et al. (2003) and Zhang et al. (2019)

3.1 Removal and replacement of HTML tags

To improve extraction efficiency, preprocessing should be done on the HTML tags. It is well known that the backbone of an HTML document is tag. Each tag is an object of the DOM which is the data representation of the object comprising the structure and content of a webpage document.

Figure 2 shows the DOM tree structure of a typical HTML document. In general, an HTML document is constituted of a <head> element and a <body> element. All the header tag elements are contained in the <head> element, which include scripts, style files, meta information, etc. Tags, e.g., <script> , <title> , <style> , <meta> , <link> , <noscript> , and < base> , can be added to the header area. In the webpage, the content of the header tag element is not visible. The <body> element contains the visible page contents. Therefore, we only need to analyze the content in the <body> element to obtain the main content. And the content in the <head> element should be removed.

In HTML5, the newly defined elements include < header> and <footer> . The header or part of a document is defined by the <header> element, which is usually used to introduce content or to navigate the link bar. The footer or part of a document is defined by the <footer> element, which is usually used to describe the name of the document creator, the copyright information of the document, the terms of use, and the contact information, etc. Since these two elements are usually unrelated to the main content, they should be removed during the first step of our proposed method.

In addition, there are some tag elements where the contents are not visible in the webpage, which are summarized in Table 2. These tags should also be removed from the webpage in the first step to ease the content extraction in the following steps.

During the process of HTML coding, line breaks are often added by programmers to make the code interface more intuitive. However, it should keep in mind that these line breaks will not be displayed in the front-end interface of webpage according to the HTML rule conventions. Therefore, such invalid line breaks used in the HTML coding should be removed from the HTML document. Similarly, some invalid blanks used for intuitive code interfaces should also be removed.

It should be noted that there are also several types of tags to realize the line break in the HTML document, which are summarized in Table 3. These tags with line break function will be replaced with "\n". The hyperlink tags (e.g., <a>) and multimedia tags

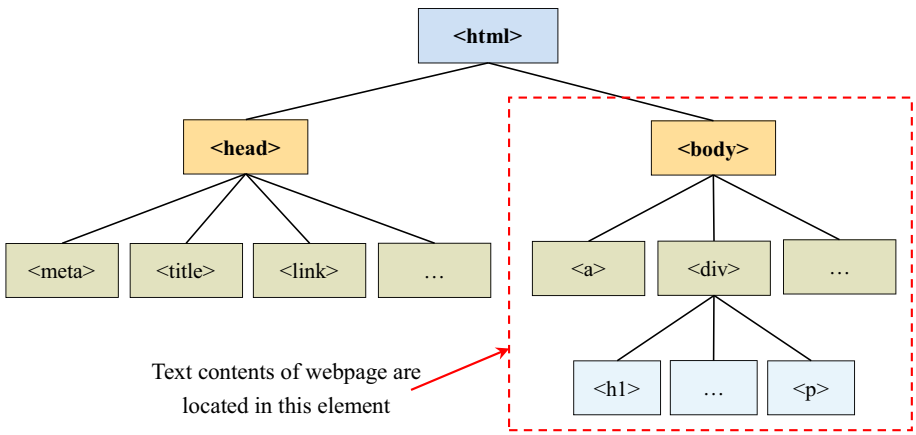


Fig. 2 Document object model tree structure of a typical HTML document

Table 2 Summary of tags with invisible content in HTML document

Type	Tags
Basics	<!–…–>|<title>
Format	<meter>|<progress>
Frame	<frameset>|<iframe>
Form	<datalist>|<select> … <option>
Program	<script>|<noscript>|<object>
Style/Section	<style>|<details> … <summary>

(e.g., <img> , <embedded> , <video> , <audio>) will be preserved in this step. As for the escape characters (e.g., <, >, &, ", , etc.), they should be replaced by their true form displayed in the front-end interface of webpage. For example, "<" should be replaced by ">". After the above preprocessing, we can obtain the main content of webpage without noise, which is saved as *Ctext* for the next step.

3.2 Calculation of distribution function of characters in tag-line-block

Due to the mix of main content and HTML tags, it is a tough task to develop a universal content extraction method for all types of HTML documents. In this study, we develop an efficient content extraction method for webpage based on the distribution features of characters in HTML document, which is capable of dealing with the main content extraction for both single-content and multiple-content webpages in English and Chinses. The distribution feature of characters in HTML document is characterized by the distribution function of characters in tag-line-block. Before showing the calculation steps of distribution function of characters in tag-line-block, four important widely accepted definitions are given as followings:

Definition 1 Tag-line (TL) segment.

A tag-line segment is a sequence of characters that contains valid tags in the preliminary denoised *Ctext*. When there is no character in a tag-line, it is considered an empty tag-line. The size of the j th TL can be expressed as $size\ (TL_j)$.

Definition 2 Tag-line-block (TLB).

The tag-line-block segment is composed of k consecutive tag-lines. k represents the thickness of TLB, which is set as 3 in this study.

Definition 3 Size of tag-line-block (STLB).

The total number of characters in a TLB. The size of the jth TLB can be expressed as $size\ (TLB_j)$.

Definition 4 Distribution function of characters in tag-line-block (DFTLB).

The distribution function of characters in tag-line-block denotes the relation between the number of tag-line-block and the size of tag-line-block, which can be expressed as:

$$f_{\mathrm{TLB}}(i) = \mathrm{size}(\mathrm{TLB}_i) = \sum_{j=i}^{i+k-1} \mathrm{size}(\mathrm{TL}_j) \tag{1}$$

where $i \in [1, m–k + 1]$, m is the total number of tag-line in a webpage.

Next, we will show how to calculate the distribution function of characters in tag-line-block based on the *Ctext* obtained from Sect. 3.1. The preliminary-denoised *Ctext* is first divided into several lines by line breaks. It is noted that the multimedia tags should also be removed when the users don't choose to retain multimedia information. Then, the procedures can be done to obtain the distribution function of characters in tag-line-block:

(1) Remove the hyperlink tags and the related contents within the *Ctext*.
(2) If there exist stop words defined by the users in the terminator lexicon, replace the row with a null character.

Table 3 Summary of tags with line break function in HTML document

Type	Tags
Basics	<h1>|<h2>|<h3>|<h4>|<h5>|<p>| |<hr>
Format	<address>|<blockquote>| |<pre>
Form	<form>|<fieldset>|<legend>
Image	<figcaption>|<figure>
Link	<nav>
Table	<caption>|<th>|<td>
List	<url>| | |<dir>|<dl>|<dt>|<dd>|<menu>
Style/Section	|<header>|<footer>|<section>|<article>|<aside>|<dialog>

(3) When the multimedia needs to be retained during the content extraction, it would be converted to the number of equivalent characters based on its size attributes. The number of equivalent characters for the multimedia ($f_{\text{multimedia}}$) can be expressed as:

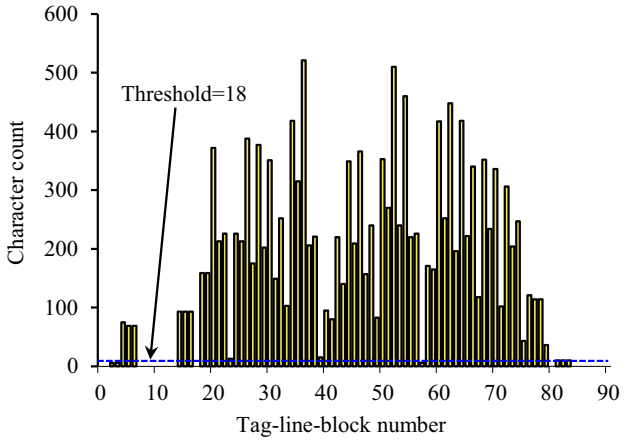
$$f_{\text{multimedia}} = \frac{L \times W}{300} \tag{2}$$

where L and W are the length and width of the multimedia, respectively.

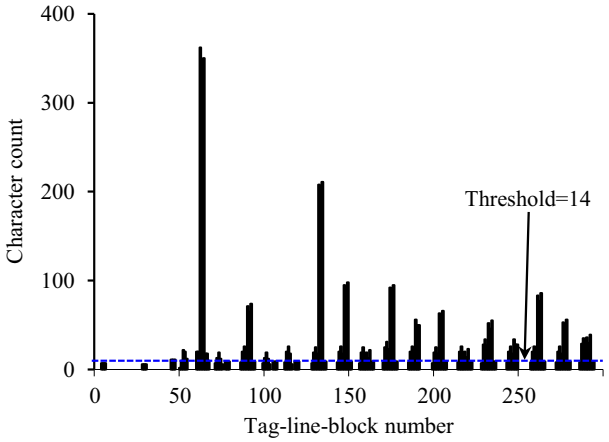
Figure 3 illustrates the distribution functions of characters in tag-line-block for two specific webpages, i.e., single-content webpage and multiple-content webpage, respectively. The x-axis denotes the number of tag-line-block and the y-axis denotes the number of characters in the corresponding tag-line-block. It can be seen that the distribution function of characters in tag-line-block for the single-content webpage is different from that for the multiple-content webpage. In general, the main content in the single-content webpage is distributed more densely than that in the multiple-content webpage.

3.3 Determination of threshold value of tag-line-block for main content

In this study, the main content was extracted from the webpage based on the feature of tag-line-block, i.e., the distribution function of characters in tag-line-block. Since the number of characters distributed in the noises is usually less than that in the main content, the distribution function of characters in tag-line-block through the whole webpage will vary with the tag-line-block. The size of tag-line-block located in the noise zone will usually be smaller than that located in the main text zone. Therefore, there would exist a threshold for the size of tag-line-block to distinguish the noise and the main content, which can be used for webpage content extraction based on the distribution function of characters in tag-line-block. This section will show how to



(a) Single-content webpage: http://www.globaltimes.cn/content/1135986.shtml



(b) Multiple-content webpage: http://www.mylot.com/post/3257034/out-for-lunch

Fig. 3 Distribution functions of characters in tag-line-block for **a** single-content webpage and **b** multiple-content webpage

Table 4 Proposed algorithm for web content extraction

Algorithm 1. Content extraction.

```
INPUT: k, threshold, stopWords, Lblock, Cline, Oline, endWords  // endWords ← [':', ' : ', '(', ' ( ', '[', ' 【', '{', ' , ', ',',...]
OUTPUT: text  // Extracted text of webpage
text ← "", start ← -1, flag ← True  // Predefinition of variables
for i ← 0 to Lblock.length-1 do
    if flag == True then  // Find the start position
        if i < Lblock.length-k and Lblock[i] > threshold and Lblock[i+1] > 0 and Lblock[i+2] > 0 then
            start ← i, flag ← False
            while start > 0 and Cline[start-1].length > 0 do  // Find the first non-blank line
                start ← start - 1
            end while
        end if
    else if i == Lblock.length-2 or (Lblock[i] < 3 and Lblock[i+1] < 3) then  // Find the end position
        tmp ← "", end ← i
        if i == Lblock.length-2 then
            end ← i + k
        end if
        for j ← start to end do  // Extract valid text between start and end
            if j == end - 1 and Cline[j][-1] in endWords then  // Remove lines containing non-terminators
                continue
            else if Oline[j].strip().length≠0 and Oline[j] not include stopWords then  // Line is not empty or with stopWords
                tmp ← tmp + Oline[j].strip() + "\n"  // Add to temp
            end if
        end for
        text ← text + tmp, flag ← True  // Add temp to text and find the next start position
    end if
end for
return text
```

determine the threshold value of tag-line-block for the main content.

First, we will introduce an important terminology that will be used during the determination of the threshold value of tag-line-block for the main content, which is shown as followings:

Definition 5 Maximum Global Consecutive Tag-Line-Block Group (MGCTLBG).

The global tag-line-block group is composed of the maximum number of consecutive tag-line-blocks with nonzero size. It should be noted that the size of the two tag-line-blocks adjacent to MGCTLBG is zero.

Based on Fig. 3, it can be seen that the size of tag-line-block within the main content zone is usually larger than that within the noise zone in the webpage. The maximum global consecutive tag-line-block group usually belongs to the main content part. However, it should be noted that the size of tag-line-block within the main content zone is not necessarily nonzero, which occurs commonly in the multiple-content webpage. In order to extract the main content of webpage as exactly as possible, it is of importance to determine the threshold value of the size of tag-line-block for main content and noises. Unlike some previous studies where the threshold value was set manually (Chen 2011), an automatic setting method was proposed in this study to determine the threshold value of the tag-line block for main content. The main procedures are shown as follows:

(1) The starting and ending line numbers of the maximum global consecutive tag-line-block group are

determined based on the distribution function of characters in tag-line-block, which can be denoted as s (starting point) and e (ending point), respectively.

(2) The average size of tag-line-block within the maximum global consecutive tag-line-block group is calculated, which can be expressed as:

$$\mathrm{STLB}_{\mathrm{ave}} = \frac{\sum_{i=s}^{e} f_{\mathrm{TLB}}(i)}{e - s + 1} \tag{3}$$

(3) Then, the average size of other tag-lines except those within the maximum global consecutive tag-line-block group can be obtained by:

$$\mathrm{OSTLB}_{\mathrm{ave}} = \frac{\sum_{i=1}^{s-1} f_{\mathrm{TLB}}(i) + \sum_{j=e+1}^{m-k} f_{\mathrm{TLB}}(j)}{m - k - e + s - 1} \tag{4}$$

(4) The main contents were identified from the distribution function of characters in tag-line-block by using the threshold value that depends on the character distribution of the whole webpage. In this study, the threshold can be expressed as:

$$\mathrm{threshold} = \gamma \times \mathrm{STLB}_{\mathrm{ave}} + (1 - \gamma) \times \mathrm{OSTLB}_{\mathrm{ave}} \tag{5}$$

where γ is the weight for the average size of tag-line-block within the maximum global consecutive tag-line-block group, which depends on the webpage type. For single-content webpages, $\gamma \in [0.5, 1]$. A greater γ indicates a stronger contribution of the main content to the threshold. The value of γ was set as 0.7 for single-content webpages in this study based on the results of experimental trials. However, $\gamma = 0$ for the multiple-content webpages where the maximum global consecutive tag-line-block group usually accounts for a small proportion of the webpage. Therefore, $\gamma = 0$ can reduce the influence of the maximum global consecutive tag-line-block group on the threshold value.

3.4 Content extraction of webpage

After the distribution function of characters in tag-line-block and the threshold value are obtained, the extraction of the main content can be conducted. Table 4 displays the pseudo-code of the proposed algorithm. The main procedures are shown as follows:

(1) Perform the search from the beginning of the webpage for the first tag-line-block whose size is greater than the threshold value. In addition, the size of two adjacent tag-line-blocks after this tag-line-block should be nonzero. The line number of this tag-line-block is set as the starting point of the webpage main content.

(2) Continue the search from the starting point part for the tag-line-block that contains less than 3 characters. The information that contains less than 3 characters may be the noise. For example, the noise texts in a Chinese webpage such as the share and comment symbols are usually less than 3 characters. For an English webpage, a complete sentence with less than 3 characters in the main content part is also apparently rare. Besides, the next tag-line-block should also contain less than 3 characters. Then, the line number of former tag-line-block containing less than 3 characters is set as the ending point of the main content. It should be noted that the ending point of the main content that satisfies the above requirements may not exist for some types of webpages. For this case, the last line number of the webpage would be set as the ending point.

(3) The webpage contents from the starting and ending points are stored in the content set *Oline*[*starting*, *ending*]. Then, the empty lines and the lines containing stop words in *Oline*[*starting*, *ending*] will be removed. It should be noted that the ending line

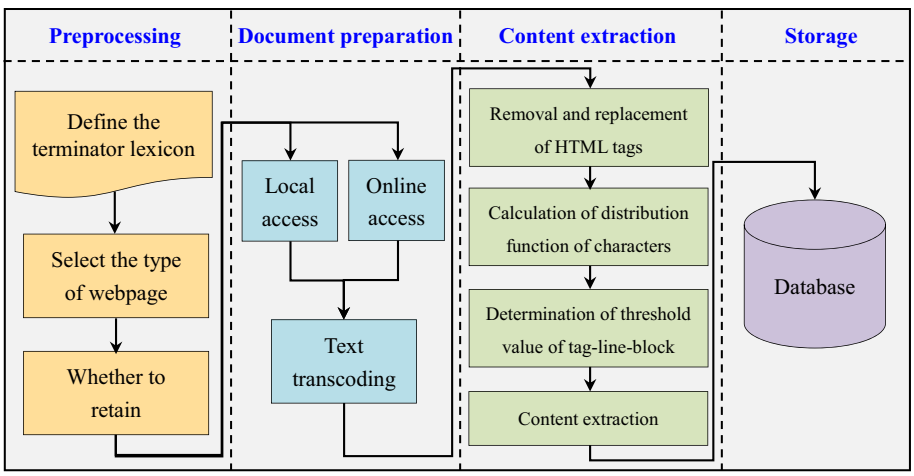


Fig. 4 Framework of the tag-line-block-based content extraction system

needs to be removed if the null character is the non-terminator, such as colon, comma, and left bracket. And the rest of *Oline*[*starting*, *ending*] will be preserved in the new set *text*.

(4) Repeat the above three steps to search the other main content until the last tag-line-block of the webpage.

4 Experiments

The quantitative evaluations of our method will be conducted using real webpages. In the following, we will introduce the framework of tag-line-block-based content extraction system, the datasets, the evaluation metrics, and the experimental settings. Finally, the performance of our proposed method will be compared with those of other content extraction methods based on the experimental results.

4.1 Framework of tag-line-block-based content extraction system

The performance of our method will be evaluated by using the different types of webpages. The proposed method was integrated into the tag-line-block-based content extraction system whose framework is shown in Fig. 4. The tag-line-block-based content extraction system is composed of four modules, i.e., the preprocessing module, document preparation module, content extraction module, and storage module.

The preprocessing module is used for personalization. In this module, users can define the terminator lexicon. Besides, users can select the webpage type (e.g., single-content webpage or multiple-content webpage) and determine whether to retain multimedia in the webpage.

The document preparation module will help users to get access to the HTML document. In this module, users can choose local or online access. Then, the document will be converted into an appropriate encoding format.

The content extraction module is the core component of the system where our proposed method for webpage content extraction is implemented. The main body of the HTML document can be finally obtained after conducting the four steps shown in Sect. 3.

The storage module is used to save the extracted main contents of webpages in the database, which can be used to evaluate the performance of the proposed method.

4.2 Datasets

In this study, two datasets (i.e., CleanEval dataset and CETRB dataset) were used for the performance evaluation of the proposed web content extraction method.

The CleanEval dataset, whose topic is cleaning arbitrary webpages, was established to prepare web data as a corpus for research and development of linguistic and language technology (Baroni et al. 2008). The webpages in the CleanEval dataset come from different websites (English and Chinese) with various page structures and design styles, which can be divided into development sets and test sets. The English dataset, i.e., Cleaneval-EN, contains 57 development samples and 684 test samples. And the Chinese dataset, i.e., Cleaneval-ZH, contains 60 development samples and 653 test samples. During the experimental evaluation, the development sets of the CleanEval dataset in both English and Chinese were used for the performance evaluation. Some webpages from the test sets were also selected to examine the performance. Finally, we collected 200 English webpages and 200 Chinese webpages from the CleanEval dataset for the performance evaluation.

Since the CleanEval dataset was developed quite early, which may not represent the features of the webpages with new structures and design styles, a dataset called CETRB was developed in this study. The webpages in the CETRB dataset come from four representative websites of different types, i.e., single-content English news website (Global Times), multi-content English blog website (myLot), single-content Chinese blog website (Sina blog), and multi-

Table 5 An example of single-content webpage markup

No	Message
1	<i> With a computer and a big smile, Mike Alfi starts his work as early as 6 am. The … for three years
2	<i> Filipino English teachers started gaining popularity in China after several Chinese … and First Future
3	<i> These internet platforms mainly target kindergarteners to 12th graders (K-12) … before school hours
4	<i> “I usually work 6–7 h a day. I get up at 5 am and start work at 6. The busiest … day,” said Alfi
5	<i> Like Alfi, there are more than 10,000 Filipinos working for internet-based teaching … is growing fast
…	
25	<i> “I’m learning Chinese now. I would love to work in China. If there is an opportunity, why not?”

Table 6 An example of multi-content webpage markup

No	Message
1	<i> By snowy @snowy22315 (61,836) United States MARCH 2, 2019 3:29PM CST
	So we went to one of the local restaurants for lunch today. It is a pretty nice … lunch on occassion?
2	<i> Nevena Zivkovic @Nevena83 (16,753) · Serbia
	I never go to lunch because I do not have money for restaurants
3	<i> Judy Evans @JudyEv (155,709) · Bunbury, Australia
	I quite like going to lunch. You usually have quite a different choice rather than going out for dinner
4	<i> Belle Starr (Iz) @BelleStarr (41,887) · United States
	We usually go out to lunch not dinner, much less expensive for lunch
5	<i> Jose Juan @quantum2020 (10,341) · Ciudad De Mexico, Mexico
	I usually go out for luncj at about 2 or 3 in the afternoon
…	
9	<i> SS @moonandstars (12,745) · Zagreb, Croatia (Hrvatska)
	Not so much. A crabcake sandwich sounds fabolous!

Table 7 Calculated values of the three metrics by the proposed content extraction method on the CleanEval-ZH dataset

No	t_e	t_i	f_i	P	R	F_1
1	24	0	0	100.00%	100.00%	100.00%
2	11	1	1	91.67%	91.67%	91.67%
3	33	0	2	100.00%	94.29%	97.06%
4	37	1	0	97.37%	100.00%	98.67%
5	12	1	0	92.31%	100.00%	96.00%
				…		
200	37	2	0	94.88%	100.00%	97.37%
Average				95.16%	93.63%	94.39%

content Chinese forum website (Tianya bbs). For each website, 200 webpages were collected to evaluate the performance of the proposed method.

During the experiments, the two datasets provide up to 1200 webpages for performance evaluation. For all the webpages, we marked the main contents manually, which will be used to verify the performance. In the single-content webpage, a paragraph was treated as a message, and the symbol <i> was added at the beginning of each message, as shown in Table 5, while in the multiple-content webpage, a reply was treated as a message, and the symbol <i> was added at the beginning of each message, as shown in Table 6.

Table 8 Comparison of extraction accuracy of different webpage content extraction methods

Data set	Readability			Cx-extractor			Proposed method		
	P (%)	R (%)	F_1 (%)	P (%)	R (%)	F_1 (%)	P (%)	R (%)	F_1 (%)
CleanEval-EN	96.38	82.17	88.71	87.69	75.44	81.11	95.64	97.40	96.51
CleanEval-ZH	94.43	87.80	90.99	91.85	77.95	84.33	95.16	93.63	94.39
Global Times	96.09	99.99	98.00	99.38	13.00	22.99	99.99	99.77	99.88
myLot	99.90	3.68	7.10	96.32	9.07	16.58	98.83	98.02	98.42
Sina blog	97.28	96.47	96.87	98.73	74.33	84.81	99.19	95.20	97.15
Tianya bbs	99.76	22.37	36.55	99.88	17.89	30.34	97.56	90.96	93.61

4.3 Evaluation metrics

We used three metrics, i.e., precision (P), recall (R), and F_1-measure (F_1), to evaluate the performance of the webpage content extraction methods, which have been widely used in the previous literature (Gottron 2008; Sun et al. 2011; Weninger et al. 2010; Wu 2016).

The metric precision can be used to characterize the accuracy of content extraction method, which can be expressed as Eq. 6. The value of P lies between 0 and 1. The greater P is, the more accurate the content extraction method is.

$$P = \frac{t_e}{t_e + t_i} \times 100\% \tag{6}$$

where t_e denotes the number of valid messages that have been extracted from the webpage, which can be determined by counting the number of symbols $<$ i $>$ in the result of the extraction; t_i denotes the number of invalid messages that have been extracted from the webpage.

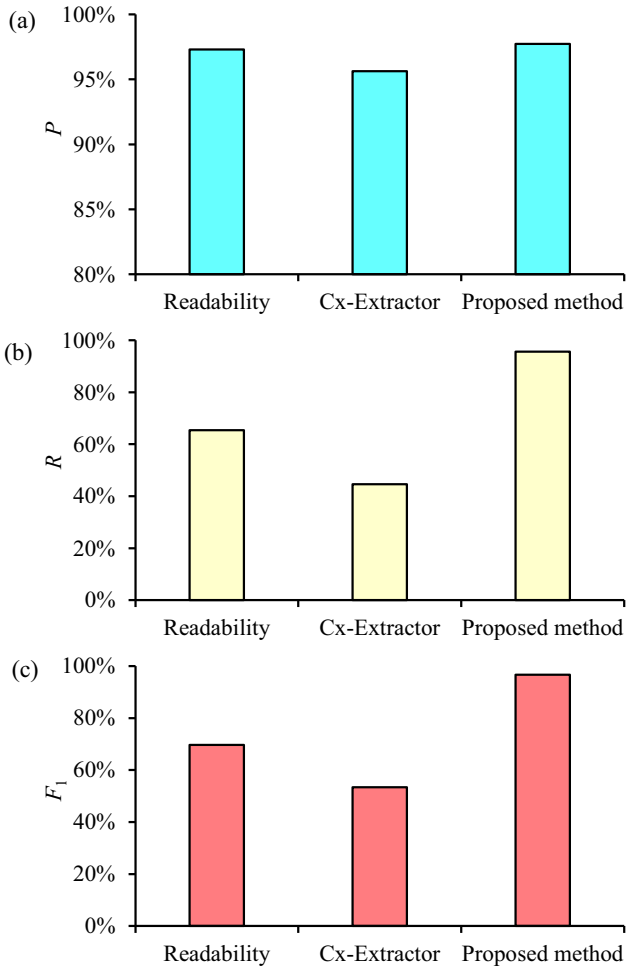


Fig. 5 Comparison of **a** precision, **b** recall, and **c** F_1-measure of different webpage content extraction methods

The metric recall can be used to characterize the completeness of content extraction method, which can be expressed as Eq. 7. The value of R lies between 0 and 1. The greater R is, the more valid content can be extracted by the evaluated method.

$$R = \frac{t_e}{t_e + f_i} \times 100\% \tag{7}$$

where f_i denotes the number of valid messages that have not been extracted from the webpage; the sum of t_e and f_i is equal to the total amount of symbol <i> in each webpage.

The metric F_1-measure represents the harmonic mean of the precision and recall, which can be expressed as Eq. 8. When F_1 is closer to 1.0, the content extraction method will show a better performance.

$$F_1 = \frac{2 \times P \times R}{P + R} \tag{8}$$

Table 7 summarizes the experimental results of the three metrics by the proposed content extraction method in this study. The evaluation was conducted on the CleanEval-ZH dataset. It can be seen that all the values of the three metrics are greater than 90%, which indicates that the proposed content extraction in this study performs well for Chinese webpages in the CleanEval dataset.

4.4 Settings

The proposed method was compared with the other two content extraction algorithms: Readability and Cx-Extractor. Readability is a DOM-tree-based method for English webpages, and the coding can be found in website: https://github.com/buriy/python-readability. While Cx-Extractor is an HTML-tag-based method for Chinese webpages, the coding can be found in website: https://code.google.com/p/cx-extractor.

All of the compared methods and our method were complemented by Python3.7 programming language. Pycharm 2018.3 was used as the compiler.

Table 9 Comparison of time complexity of different webpage content extraction algorithms

Method	Readability	Cx-extractor	Proposed method
Time complexity	$O(n^2)$	$O(n)$	$O(mn)$

n is the size of the webpage; m is the size of HTML tags

Table 10 Comparison of running time of different webpage content extraction algorithms

Data set	Run time (ms)		
	Readability	Cx-Extractor	Proposed method
CleanEval-EN	18.27	5.11	16.22
CleanEval-ZH	8.53	1.59	5.07
Global Times	5.62	1.20	4.09
myLot	30.20	7.18	20.63
Sina blog	86.23	8.79	34.88
Tianya bbs	65.67	7.10	16.42
Average	35.75	5.16	16.22

4.5 Experimental results

4.5.1 Comparison of extraction accuracy

The extraction accuracy of the three methods is assessed in precision, recall, and F_1-measure, which are listed in Table 8. It can be seen that the Readability method performs well in terms of precision, which can give a precision of more than 90% for both single-content and multiple-content webpages. However, the Readability method has a poor recall rate when it is used for multiple-content webpages. For example, the recall values of the Readability method calculated based on the webpages from myLot and Tianya bbs are 7.10% and 36.55%, respectively. In other words, the Readability method cannot extract the whole valid main content for multiple-content webpages. The experimental results of the Cx-Extractor method are similar to those of the Readability method. Although the Cx-Extractor method works well in terms of precision for both single-content and multiple-content webpages, the recall value of the Cx-Extractor method is relatively low (9.07–77.95%), i.e., a non-negligible amount of valid main content of the webpage will be missed when using the Cx-Extractor method. The main reason lies in that the threshold set manually has a significant influence on the extracted results of the Cx-Extractor method. As shown in Table 8, both the precision and recall of our method are up to 90%, which indicates that the method can extract most of the main contents of webpages and only a small amount of invalid information is included in the extracted contents.

In order to further compare the three methods, Fig. 5 shows the estimated precision, recall, and F_1-measure by these different webpage content extraction methods. As shown in Fig. 5a, the three methods exhibit a negligible difference in the precisions, which are between 95.6 and 97.75%. Among the three methods, our proposed method has the highest precision. It can be seen from Fig. 5b that the recall rate values of the CX-Extraction method, the Readability method, and our proposed method increase in turn, indicating that our proposed method will extract the least amount of invalid information from the webpages. Figure 5c reveals that our proposed method possesses the highest F_1-measure value, which verifies again that our proposed method outperforms these two methods. The proposed content extraction method has a good performance in extraction accuracy when dealing with different types of webpages (e.g., English webpage and Chinese webpage, single-content webpage and multiple-content webpage). It is mainly attributed to the combination of the character distribution function of tag-line-block and the semantic information of HTML tag as well as the automatic determination of threshold value of tag-line-block for main content.

4.5.2 Comparison of extraction efficiency

The extraction efficiency of the three methods is evaluated in this section. First, the runtime complexity was analyzed. Table 9 summarizes the comparison of time complexity of the three webpage content extraction methods. Since the Readability method is a DOM-tree-based method, it will take $O(n^2)$ to scan and count the number of words for each tag-line. In the Cx-Extractor method, the input webpage does not require constructing a DOM tree, and all tag lines will be traversed only once. Therefore, the Cx-Extractor method has a time complexity of $O(n)$. Our proposed method also does not need to construct the DOM tree, but it needs to deal with tag-line-block. Consequently, the proposed method has a time complexity of $O(mn)$. In summary, the runtime complexity of the Cx-Extractor method, our proposed method, and the Readability method will decrease in turn.

To further illustrate the extraction efficiency of the three methods, we compare the running time of different webpage content extraction algorithms on the two datasets shown in Sect. 4.2. The runtime for each dataset in Table 10 is the average of the results by running 10,000 extraction experiments repeatedly. The average runtimes for the Readability method, the Cx-Extractor method, and our proposed method are 35.75 ms, 5.16 ms, and 16.22 ms, respectively, which is consistent with the results shown in Table 9. The runtime of our method is more efficient than the Readability method based on the DOM tree, as can be seen in Table 10. The reason is that our method is based on the Cx-Extractor method that traverses all tag lines in content extraction. However, it should be noted that it is not as efficient as the Cx-Extractor method since our method needs to deal with the HTML tags of the webpage.

4.5.3 Comparison of display of extracted results

In this section, the display of the extracted results by the three methods is analyzed. The Readability method does not process the tags of HTML document during the extraction of main content, and thus, the extracted results are displayed in HTML document format. The Cx-Extractor method displays the extracted results according to the HTML coding rules. Therefore, a complete statement may be divided into several separate lines. Besides, the Cx-Extractor method is mainly developed for Chinese webpages. During the content extraction process, all the blank spaces will be removed. When it is used for English webpages, words are linked together, which increases the difficulty of reading. However, our proposed method has an obvious advantage in terms of the display of the extracted content. Each paragraph is displayed as a single component, which can ensure semantic coherence. Besides, the blank spaces in the original text of webpage are retained during the content extraction, which can avoid the problem of connecting English words into pieces and improve the readability of the extracted content.

5 Conclusions

This paper is intended to develop an efficient content extraction method for webpage based on the character distribution function of tag-line-block in HTML document and the semantic information of HTML tags. Several improvements have been made in our proposed method, which are shown as followings: (1) The hyperlink tags are not removed directly to avoid mistaking the dense hyperlink groups for the main content. (2) The starting point of the web main content is taken as the line number of tag-line-block whose size exceeds the threshold to avoid missing the first few short texts of the main content. (3) The threshold value of tag-line-block for main content is calculated automatically to avoid setting the threshold value manually. (4) The blank spaces in the original text of webpage are retained to avoid connecting English words into pieces. (5) The multimedia information can be selectively retained by users to allow for maximum flexibility and usage in multiple industries. As a result, our proposed method can increase the precision, recall rate, and readability of the extracted webpage contents. Moreover, it can be well used in different types of webpages.

To illustrate the performance of the proposed method, it was compared with several popular content extraction methods, i.e., the Chinese extraction method called Cx-Extractor and the English extraction method called Readability. The experiments were conducted on the CleanEval and CETRB datasets. It is found that the proposed method outperforms these compared methods in precision and recall. In addition, the extraction efficiency of the proposed method is superior to that of the Readability method. Moreover, compared with the other two methods, our method can ensure semantic coherence and enhance the readability of the extracted content.

In practical engineering, extracting high-quality content from the webpage is critical. For example, for information retrieval, the accuracy of retrieval will be directly affected by the quality of the extracted content. Since our method has excellent performance in accuracy, universality, simplicity, and scalability, it can be easily implemented and well used in Internet services and applications, e.g., information search, automatic text classification, topic tracking, machine translation, automatic summarization, etc. In these applications, our proposed content extraction method, as a necessary step, provides the basic data for subsequent analysis and data mining. Therefore, it can provide some technical references for governments, companies, and other researchers to extract content more accurately and more generally.

However, many scanning operations are required during the replacement of HTML tags and escape characters in the current version algorithm, which will reduce the efficiency of the content extraction. Considering this limitation, an improvement needs to be made for the proposed method in the future so that it can complete all the replacing operations after scanning the HTML document once by matching the angle brackets. In addition, the precision of the main content extracted from multiple-content webpages can be further improved in the future. For example, the noises frequently occur in the form of short sentences around the comment messages of forum websites, which may usually be mistaken for the main content. If these types of noises can be identified and removed, the precision of our proposed method can be further improved, which is the direction of further research.

Funding This research was funded by National Key Research and Development Program of China, Grant Number 2021YFD1300101.

Data availability Enquiries about data availability should be directed to the authors.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Baroni M, Chantree F, Kilgarriff A et al (2008) Cleaneval: a competition for cleaning web pages. In: Proceedings of the 6th international conference on language resources and evaluation, pp 638–643

Cai D, Yu S, Wen J R, et al (2003) Extracting content structure for web pages based on visual representation. In: Proceedings of the 5th Asia-pacific web conference on web technologies and applications, pp 406–417

Cardoso E, Jabour I, Laber E, et al (2011) An efficient language-independent method to extract content from news webpages. In: Proceedings of the 11th ACM symposium on document engineering, pp 121–128

Chen X (2011) Universal web content extraction based on row block distribution function. https://code.google.com/p/cx-extractor

Crescenzi V, Mecca G, Merialdo P (2001) Roadrunner: towards automatic data extraction from large web sites. In: Proceedings of the 27th international conference on very large data bases, vol. 1, pp 109–118

Ferrara E, De Meo P, Fiumara G et al (2014) Web data extraction, applications and techniques: a survey. Knowl-Based Syst 70:301–323

Gan L, Ye B, Huang Z et al (2023) Knowledge graph construction based on ship collision accident reports to improve maritime traffic safety. Ocean Coast Manag 240:106660

Gibson D, Punera K, Tomkins A (2005) The volume and evolution of web page templates. In: Special interest tracks and posters of the 14th international conference on World Wide Web, pp 830–839

Gottron T (2008) Combining content extraction heuristics: the CombinE system. In: Proceedings of the 10th international conference on information integration and web-based applications and services, pp 591–595

Gu Y, Gao Y, Gao B et al (2014) Research on deep web information extraction based on template and domain ontology. Comput Eng Des 35:327–332

Gupta S, Kaiser G, Neistadt D et al (2003) DOM-based content extraction of html documents. In: Proceedings of the 12th international conference on World Wide Web, pp 207–214

Hammer J, McHugh J, Garcia-Molina H (1997) Semistructured data: the TSIMMIS experience. In: Proceedings of the 1th East-European symposium on advances in databases and information systems, vol. 1, pp 1–13

IDC, Statista (2022) Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2020, with forecasts from 2021 to 2025 (in zettabytes). https://www.statista.com/statistics/871513/worldwide-data-created/

Joe Dhanith PR, Surendiran B (2022) An ontology learning based approach for focused web crawling using combined normalized pointwise mutual information and Resnik algorithm. Int J Comput Appl 44(12):1123–1129

Karthikeyan T, Sekaran K, Ranjith D et al (2019) Personalized content extraction and text classification using effective web scraping techniques. Int J Web Port 11(2):41–52

Laber ES, de Souza CP, Jabour IV et al (2009) A fast and simple method for extracting relevant content from news webpages. In: Proceedings of the 18th ACM conference on information and knowledge management, pp 1685–1688

Liang D, Yang Y, Wei Z (2018) Information extraction of web pages based on support vector machine. Comput Mod 9:21–26

Liu L, Pu C, Han W (2000) XWRAP: an XML-enabled wrapper construction system for web information sources. In: Proceedings of the 16th international conference on data engineering, pp 611–621

Rahman A, Alam H, Hartono R (2001) Content extraction from html documents. In: Proceedings of the 1st international workshop on web document analysis, pp 1–4

Ramakrishna M, Gowdar L, Havanur MS et al (2010) Web mining: key accomplishments, applications and future directions. In: Proceedings of the 2010 international conference on data storage and data engineering, pp 187–191

Samuel MO, Tolulope AI, Oyejoke OO (2019) A systematic review of current trends in web content mining. In: Proceedings of the 3th international conference on science and sustainable development, vol. 1299, p 012040

Sandeep KS, Patil N (2018) A multidimensional approach to blog mining. progress in intelligent computing techniques: theory, practice, and applications. Adv Intell Syst Comput 719:51–58

Sestito S, Dillon T (1993) Knowledge acquisition of conjunctive rules using multilayered neural networks. Int J Intell Syst 8(7):779–805

Sun F, Song D, Liao L (2011) Dom based content extraction via text density. In: Proceedings of the 34th international ACM SIGIR conference on research and development in information retrieval, pp 245–254

Sun C, Guan Y (2004) A statistical approach for content extraction from web page. J Chin Inf Process 18(5):17–22

Tan Z, He C, Fang Y et al (2018) Title-based extraction of news contents for text mining. IEEE Access 6:64085–64095

Waldherr A, Maier D, Miltner P et al (2017) Big data, big noise: the challenge of finding issue networks on the web. Soc Sci Comput Rev 35(4):427–443

Wang Q, Fang Y, Ravula A, et al (2022) Webformer: the web-page transformer for structure information extraction. In: Proceedings of the 2022 ACM web conference, pp 3124–3133

Weninger T, Hsu WH, Han J (2010) CETR: content extraction via tag ratios. In: Proceedings of the 19th international conference on World Wide Web, pp 971–980

Wu Y (2016) Language independent web news extraction system based on text detection framework. Inf Sci 342:132–149

Yu M, Chen T, Xu H (2005) Research and design of HTML parser based on page segmentation. J Comput Appl 25(4):974–976

Yunis H, Stein B, Kiesel J et al (2016) Content extraction from webpages using machine learning. Bauhaus-Universitaet Weimar

Zhang H, Li L, Hu W et al (2019) Visualization of location-referenced web textual information based on map mashups. IEEE Access 7:40475–40487

Zhang Z, Yu B, Liu T, et al. (2023) Learning structural co-occurrences for structured web data extraction in low-resource settings. In: Proceedings of the 2023 ACM web conference, pp 1683–1692