www.turkishstudies.net/turkishstudies

Turkish Studies

eISSN: 1308-2140

INTERNATIONAL BALKAN UNIVERSITY
Sponsored by IBU

Research Article / Araştırma Makalesi

Mevcut Yöntemlerin Sentezi ile Web Sayfalarının Belli Yapılardaki Bölgelerinin Tespiti

Localization of Structured Regions of a Web Page Through a Synthesis of Existing Methods

Malika Sirat[*] - Adem Özyavaş[**]

Abstract: Web content mining is a branch of web mining where the content usually consists of html tag-data and more importantly structured data such as lists and tables with html tag-data and possibly other nested structured data. Usually, useful pieces of information reside in these structured data regions following some pattern. But due to noises that break this pattern, detection and extraction of useful data records turn out to be a challenging task. For data record detection and extraction applications, any non-regular html tag in the structured data region as well as any data outside of the structed data regions of interest are considered to be noise. The proposed work is a synthesis of existing methods combining their successful components with minor modifications to further improve the accuracy performance of the extraction systems. Most of the existing methods utilize the Document Object Model (DOM) tree to cluster similar nodes to detect the data record regions or convert DOM tree into a Tag Path Sequence (TPS) to represent a web page before detection. The proposed work first generates the unique sequence that encodes a webpage and utilizes the Suffix Tree-based Extraction Method (STEM) to extract data record patterns. It then clusters subsequences based on the assumption that data records that are related are similar in structure. The similarity between sequences are computed using the longest common subsequence (LCS) algorithm. In the last step, the noisy subsequences are removed by pruning the corresponding nodes from the DOM tree. Some publicly available web pages and Testbed dataset are used for testing the proposed work. The obtained results are compared with those of the TPS filter method. Experiments show that the average number of noise removed using the proposed work is 43% of the DOM tree, which can match previous reports and it is higher than that of TPS filter.

Structured Abstract: Web mining is an area of data mining on extracting informative data embedded in web pages. It is divided into three categories: web structure mining, web usage mining and web content mining (Jokar et al., 2016, pp. 321-328). Web content mining is further divided into multimedia mining, text mining, and structured record mining (Mughal, 2018). A structured record (data region) of a web page contains data with some pattern repeated in which is different from the rest of the web page with no visible pattern, that is, no regularity in the structure of the content. Often times, the patterns in structured data is interrupted by occasional noise(Footer, Privacy policy and etc) and requires the removal of it.

* Sorumlu Yazar: Yüksek Lisans Öğrencisi, İstanbul Aydın Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
Corresponding Author: MSc in Computer Engineering, Istanbul Aydin University, Engineering Faculty, Computer Engineering
ORCİD 0000-0002-5998-0846
malikasirat@stu.aydin.edu.tr
[**]Dr. Öğr. Üyesi, İstanbul Atlas Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Yazılım Mühendisliği
Asst. Prof. Dr., Istanbul Atlas University, Engineering and Natural Sciences Faculty, Software Engineering
ORCİD 0000-0001-5375-1826
adem.ozyavas@atlas.edu.tr

Cite as/ Atıf: Sirat, M. & Özyavaş, A. (2022). Mevcut yöntemlerin sentezi ile web sayfalarının belli yapılardaki bölgelerinin tespiti. *Turkish Studies, 17*(1), 163-175. https://dx.doi.org/10.7827/TurkishStudies.52730
Received/Geliş: 24 September/Eylül 2021
Accepted/Kabul: 25 February/Şubat 2022
Published/Yayın: 28 February/Şubat 2022
Checked by plagiarism software
© Yazar(lar)/Author(s) | CC BY-NC 4.0

Figure 1 shows a segment of a web page taken from amazon.com with some omissions, which contain three regions: a header, a left navigation bar and the main content of the page. The main content has three data records placed on their respective rows with similar structure that can easily be captured by human eyes (Miao et al., 2009, pp. 981-990). According to (Velloso and Dorneles, 2013, p. 173), data records in the same region have similar HTML tag path sequence. Data records are parts of a web page that contain useful information (Thamviset& Wongthanavasu ,2014:284-289). Therefore, based on the aforementioned observations data of interested should be mined within such regions. Recently the number of tree based approaches have grown substantially (Gupta et al., 2003, pp. 207-214; Zhai et al., 2005, pp. 76-85; Liu et al., 2003, pp. 601-606). In tree-based approaches, Mining Data Record (MDR) (Liu et al., 2003, pp. 601-606) is the most commonly used method that relies on the idea that information in web documents usually occur in contiguous regions of the page, called data record regions. MDR localizes data regions in two steps: Firstly, it finds nodes that contain sibling nodes called generalized node. Secondly, it groups similar generalized nodes using an edit distance algorithm.

Some approaches convert the DOM tree into visual information and sequence of identifiers to make the extraction process more efficient. In the visual information model, Tag Path Clustering (TPC) is proposed by (Miao et al., 2009, pp. 981-990). It represent a web page as a visual signal vector and extract data record patterns from visual information. But, if an extra or a missing item occur in some record sequences, the TPC will easily be affected and produce inaccurate results. To boost the data extraction, methods like Tag Path Sequence (TPS) filter (Velloso and Dorneles, 2013, p. 173) and Suffix Tree-Based Data Extraction (STEM) (Fang et al, 2018, pp. 305-331) are introduced that represent the web page as sequence of identifiers. Based on the observation and assumption that the main region (data region) of a web page is denser and sequences in data region and non-data regions (noisy regions) have no intersections, TPS can filter out noisy regions. But, the idea may fail for web pages with large size of noisy areas.

The repeated and continuous occurence of data record sequences is a well accepted assumption also mentioned by (Fang et al, 2018, pp. 305-331). The proposed work also followed it, and involves four major steps to locate data regions: (1) Parsing and standardizing a web page (2) Extracting data record sequences (3) Clustering data record sequences (4) Removing noisy regions. The first step utilizes the Jsoup library (Java open-source webpage renderer library) to parse the web page given the URL of desired web page. For cleaning some faulty and ill formatted HTML tags and hence to standardize a web page, the Jtidy library is used. In second step, data record sequence extraction is performed through the use of filters suggested in STEM which does it without removing noise from the data region. Third step involves clustering all sequences of the web page similar to the sequence identified as the target data record sequences using LCS (Longest Common Subsequence) algorithm. Finally, DOM tree is traversed to prune the leaf nodes that do not exist in the data region cluster.

The propsed work locates data regions in a web page and performs denoising based on some observations of the properties that exist in most web documents while keeping the proper DOM structure for further processing. Experiment results show that the current work can segment successfully the main region of 195 web pages out of 255 from Testbed dataset and some publicly available web pages. Some comparisons between the propsed work and TPS filter method are shown in table 2. As it can be seen the number of removed noisy nodes and number of successfully founded data regions by the current work is higher than those of TPS filtering method.

Keywords: Web mining, Data region, Noise Removal, Document Object Model tree, HTML

Öz: Web içerik madenciliği, içeriğin genellikle html etiket verilerinden ve daha da önemlisi, html etiket verili listeler ve tablolar ve muhtemelen diğer iç içe yapılandırılmış verilerden oluştuğu bir web madenciliği dalıdır. Genellikle, yararlı bilgi parçaları, bazı kalıplar halinde bu yapılandırılmış veri bölgelerinde bulunur. Ancak bu kalıbı bozan gürültüler nedeniyle, faydalı veri kayıtlarının tespiti ve çıkarılması zorlu bir görev haline gelir. Veri kaydının yerinin tespiti ve çıkarılmasında, yapılandırılmış veri bölgesindeki düzenli olmayan html etiketinin yanı sıra yapılandırılmış veri bölgelerinin dışındaki herhangi bir veri, gürültü olarak kabul edilir. Önerilen çalışma, ilgili veriyi çıkarma sistemlerinin doğruluk performansını daha da iyileştirmek için mevcut yöntemlerin başarılı bileşenlerini küçük değişikliklerle birleştiren bir sentezidir. Mevcut yöntemlerin çoğu, veri kaydı bölgelerini tespit için benzer düğümleri kümelemek veya tespitten önce bir web sayfasını temsil etmek için DOM ağacını bir Etiket Yolu Sırasına (TPS) dönüştürmek için Belge Nesne

Modeli (DOM) ağacını kullanır. Önerilen çalışma ilk olarak bir web sayfasını kodlayan ve veri kayıt modellerini çıkarmak için Sonek Ağacı Tabanlı Çıkarım Yöntemini (STEM) kullanan benzersiz diziyi oluşturur. Daha sonra, ilişkili veri kayıtlarının yapı olarak benzer olduğu varsayımına dayalı olarak alt dizileri kümeler. Diziler arasındaki benzerlik En Uzun Ortak Alt Dizi (LCS) algoritması kullanılarak hesaplanır. Son adımda, gürültülü alt dizilerinin DOM ağacının düğümlerine karşılık gelenleri ağaçtan budanır. Önerilen çalışmayı test etmek için bazı herkese açık web sayfaları ve Testbed veri seti kullanılmıştır. Elde edilen sonuçlar, TPS filtre yönteminin sonuçları ile karşılaştırılmıştır. Deneyler, önerilen çalışma kullanılarak kaldırılan ortalama gürültü sayısının, önceki raporlarla eşleşebilen DOM ağacının %43'ü olduğunu ve TPS filtresinden daha yüksek olduğunu göstermektedir.

Anahtar kelimeler: Web madenciliği, Veri bölgesi, Gürültü Giderme, Document Object Model ağacı, HTML

Introduction

We are in big data era. Digital data are recorded mostly without not much regularity or format and they come from different resources. Web is one of the largest data sources that contain data with several types, include hyperlink, text, video, audio, image, and structured data. Web is also defined as a multimedia database (Aggarwal, 2014) with data of different modality. Mining such data is divided into three categories: Web Structure Mining, Web Usage Mining and Web Content Mining (Liu, 2011, p. 7; Jokar et al., 2016, pp. 321-328). Web Structure Mining is the process of extracting useful data from webpages with structural relationships. In this category hyperlinks or shortly links that shown by the <a> tag plays a significant role for connecting/linking related pages. Web Usage Mining is the process of collecting users/customers activities and interaction over different web pages to improve the Web sites in the users view. Association rules, Sequential patterns, and Clustering are the three main methods in this category (Jokar et al., 2016, pp. 321-328).

Web Content Mining is the process of extracting useful information from web documents. Contents of a web page can be text, image, audio, video or structured. Text documents are mined using text mining, machine learning and natural language processing methods. Extracting information of interest from images, audios and vidoes is performed through multimedia mining techniques some of which are Multimedia Miner, Shot Boundary Detection, SKICAT, Color Histogram Matching (Mughal, 2018).

Structured data in web pages are typically generated from underlying databases. The structured data generally follow a pattern of html tags. To extract these pattern, there are some obstacle as irrelevant parts (navigation sidebars, headers, copyright information, etc) or sometimes this pattern is distrupted by extra html tags considered noise that are not repeated in all sequence of the structured data. Noise forms the main obstacle before extraction and integration of data from web resources and still stays as open problem (Schulz et al., 2016, pp. 562-567).

The main purpose of this work is the localization of structured data which usually contains data of intereste. Detecting such data regions can be a preliminary step that can help extracting and transforming them into expected format for further processing. In the literature, an applicatin performing this task is known as a wrapper. There are three main wrapper approaches: manual approach, wrapper induction, and automatic extraction (Liu, 2011, p. 363). Each of them consists of the same three steps: wrapper generation, wrapper execution and wrapper maintenance (Ferrara et al., 2014, pp. 301-323).

In the manual approach, a programmer observes a web page and its source code and writes a program to extract the target data. The manual approach is only suitable for small-size problems fails to meet the requirements of large volumes of web data.

Partially automatic or wrapper induction is based on supervised learning and involves user labeling, like Repetitive Subject Pattern (RSP), proposed by (Thamviset and Wongthanavasu, 2012, p. 1). The automatic extraction as its name implies is full automatic based on unsupervised learning methods. Because of growing size of data, interest in full automatic extraction methods has grown substantially as opposed to less interest in manual and semi-automatic solutions.

The wrapper methods are categorized into three main types: regular expressions, wrapper programming language and tree-based (Ferrara et al., 2014, pp. 301-323). Tools based on regular expressions identify strings or patterns in unstructured text, withour taking into consideration of the DOM tree structure of web page contents. It defines some rules it uses to extract data from the structured content, for example WYSIWYG Web Wrapper Factory (W4F) (Sahuguet and Azavant, 1999, pp. 738-741). The wrapper programming language, instead of considering web pages as unstructured simple text strings, it identifies a web page as semi-structured tree document or Data Object Model (DOM) tree. In tree based approach, wrappers are based on structured representation of documents and utilize a matching algorithm for string matching and tree matching operations. An instance of a tree-based approach is MDR, which extracts data using partial tree alignment and edit distance techniques (Zhai and Liu, 2005, pp. 76-85).

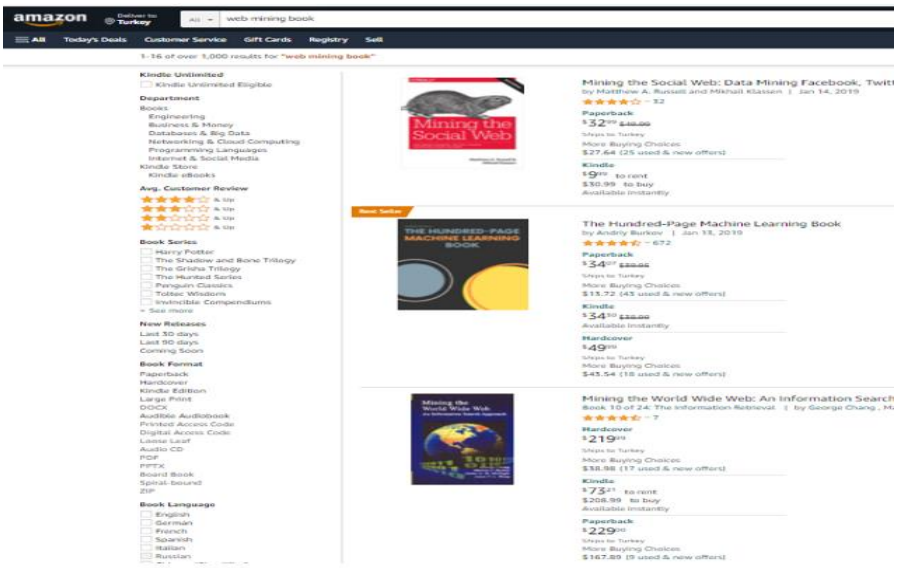


Figure 1. A Segment of a Web Page From: Https://Www.Amazon.Com/S?K=Web+Mining+Book&Ref=Nb_Sb_Noss

Due to nested structure of HTML documents tree-based approaches provides a natural represention of a web page. Recently tree-based appraoches became more popular and most of them analyze the DOM tree of an input page. One such work is proposed by (Liu et al., 2003, pp. 601-606; Gupta et al., 2003, pp. 207-214; Thamviset and Wongthanavasu, 2014, pp. 284-289). Others convert the DOM tree structure of web page into a different form before they can extract the data of interest (Miao et al., 2009, pp. 981-990; Velloso and Dorneles, 2020, pp. 109-120). This work represents a web page as a visual signal vector. (Kayed and Chang, 2009, pp. 249-263; Velloso and Dorneles, 2013, p. 173) translate an HTML document into a sequence of identifiers and try to find some regularities in these sequences to segment or extract data from web pages. The proposed work also follows this encoding in its detection of structured regions without extracting data records. There are other techniques addressing the problem of extraction and alignment of

attributes in records (Zai and Liu, 2005, pp. 76-85). The proposed method combines those techniques to perform the entire data extraction process.

The structure of this paper is as follow: Section 1 provides brief introduction to the tree-based approach. Section 2 is about localizing regions of structured data and segmention of web pages after removing noise. Section 3 shows experimental results. Section 4 includes conclusion and future work**.** Section 5 finalizes with references.

1. Tree-Based Approach

The document object model (DOM) tree is the backbone of HTML documents which is tree-form arrangement of the web page contents. The DOM tree will be referred to as just tree in the remainder of the discussion. In the tree, each tag forms a node as well as the contents of the tags. The nested tags and contents within a tag are the children of the node. Figure 2. illustrates an example that demonstrates how an HTML document is represented as a DOM tree. In this figure, the node "ul"is the parent node of "li" and therefore "li" is a descendant of "ul". There "li" nodes are siblings under the same parent "ul". In this tree "html" is the unique root node and "p, img, em" are the leaf nodes.

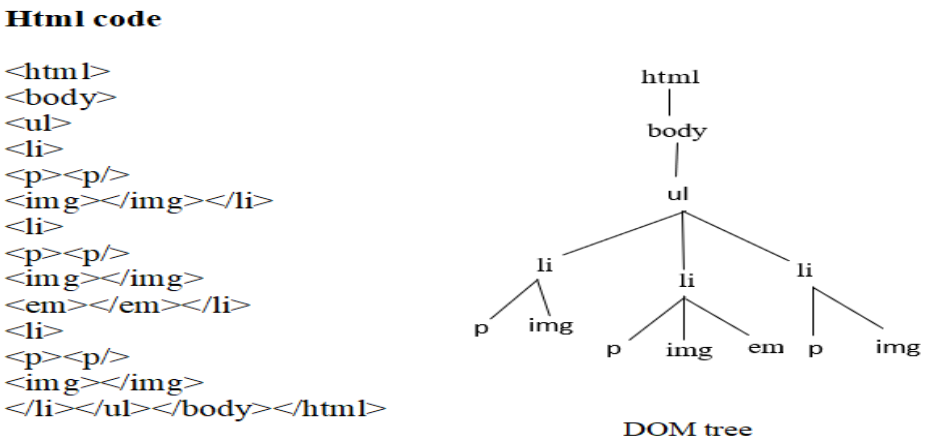


Figure 2. A Sample of Html Code and Corresponding DOM Tree

Some tree-based methods, such as MDR, utilize edit distance algorithms (Zhang and Shasha, 1989, pp. 1245-1262) to measure the similarity of subtrees for extracting data records. Bottom-Up wrapper (BUW) is another wrapper method which traverses the DOM tree from leaves to the root proposed by (Thamviset and Wongthanavasu, 2014, pp. 284-289). It measures the similarity of data items and data records to detect main regions. It aims at extracting data record items that display some form of regularity from web page content that is in unordered (ul) or ordered lists (ol). Like this, there is an area work that focus on analysis of the DOM tree as DEPTA, DOM-based content extraction of HTML documents, proposed by (Gupta et al., 2003, pp. 207-214).

Recent research has modeled DOM trees of Web pages as visual signal vectors and sequence of identifiers, like (Simon and Lausen, 2005, pp. 381-388; Miao et al., 2009, pp. 981-990; Velloso and Dorneles, 2020, pp. 109-120) and (Kayed and Chang, 2009, pp. 249-263; Fang et al., 2018, pp. 305-331; Velloso and Dorneles, 2013, p. 173) respectively. Extracting data from these

encoding is performed using signal processing and sequence mining techniques. The remainder of this section will be the discussion of these two models.

1.1. Representation of Web Pages as Visual signals

Visual signal representation of a web page enocodes the DOM tree as a string of tag paths and converts them into a visual signal vector (Simon and Lausen, 2005, pp. 381-388; Miao et al., 2009, pp. 981-990; Velloso and Dorneles, 2020, pp. 109-120). The TPC proposed by (Miao et al., 2009, pp. 981-990) is based on the observation that a data region is a segment of a web page that consists of visually aligned data records. To identify this region, a pair-wise similarity metrics is devised between tag paths. Figure 3 shows an example that represents a web page as a visual signal. The complete process of converting a web page to its corresponding visual signal vectors is described in (Miao et al., 2009, pp. 981-990). Figure 3 shows the traversal of the DOM tree given in Figure 2 in the pre-oreder fashion and extracts all the tag paths. The number associated with a tag path is equal to the sequence of numbers of nodes in DOM tree from the root to that tag. The extracted tag paths are shown in Table 1. An inverted index data structure is used for mapping from HTML tag paths to their locations as shown in Figure 3. For visual representaion of this docummet the existence of row entities in the column path is flagged by 1 (white pixel) and non-existence with "0" (black pixel).

Table 1. Extracted Tag Path and Corresponding Identifier

All the tag paths	HTML identifier
Html	1
html/body	2
html/body/ul	3
html/body/ul/li	4
html/body/ul/li/p	5
html/body/ul/li/img	6
html/body/ul/li	4
html/body/ul/li/p	5
html/body/ul/li/img	6
html/body/ul/li/em	7
html/body/ul/li	4
html/body/ul/li/p	5
html/body/ul/li/img	6

Representing Webpage as Signal Vector														Visual Representation of DOM tree that given in Figure 2
	1	2	3	4	5	6	4	5	6	7	4	5	6	
1	1	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	1	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	1	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	1	0	0	1	0	0	0	1	0	0	
5	0	0	0	0	1	0	0	1	0	0	0	1	0	
6	0	0	0	0	0	1	0	0	1	0	0	0	1	
7	0	0	0	0	0	0	0	0	0	1	0	0	0	

Figure 3. Representation Of Web Page as a Visual Signal

1.2. Representation of Web Page as Sequence of İdentifier

Beside visual representation of a web page, there are some approaches that translate a DOM tree into a sequence of identifiers (Kayed and Chang, 2009, pp. 249-263; Fang et al., 2018, pp. 305-331; Velloso and Dorneles, 2013, p. 173). They represent a web document as a sequence of identifiers by mapping each tag path of the DOM tree to numerical identifiers. The illustrated sample in Table 1 shows an example representation. For example, the assigned identifier for tag path "html/body/ul/li/p" is 5. If the same tag path appers multiple distinct paths in a document, all will be assigned the same identifier. The sequence of identifiers for the DOM tree given in Figure 2 is shown in Table 1. There are two sequence-based methods partially adopted in the proposed work, namely STEM and TPS filter. The former is a data record extractor method, and the latter is used for segmenting data record by removing noisy nodes.

The TPS filter is a sequence-based approach used to clean up a web page before extracting the records from it. By considering the frequency of the web page sequences and the size of the page, it recursively divide the entire web page into data record region and non-data record region until the web page is no longer partitionable. The procedure stops when nodes with higher frequencies are less than %20 of the size of the whole page and returns the region as data region. The proposed work deviates slightly from the original implementation of the TPS filtering method. The following line in the original TPS filter algorithm

$$\text{if (currentAlphabet! = } \emptyset \text{ and } (n - 2 \times i) / n > 0.20)$$

is replaced with the following line

$$\text{if((Math.abs(n-2.0} \times \text{i)/(n} \times \text{1.0f)) > .20)}$$

Because when i is in right half part of TPS, $n-2 \times i$ is negative ($n-2 \times i<0$) but should always be positive. The above algorithm segments the web page if the above condition is true and mostly fails at:

- Heterogeneous web pages (main content divided into multiple regions),
- Homogeneous web pages (main content and other no-interest regions are very similar), and
- Web pages with the smaller main content with respect to the rest of the page.

To extend the algorithm and overcome these problems, there is a need for stronger assumptions. For this purpose, observations proposed by (Fang et al., 2018, pp. 305-331) are utilized in the proposed work which is explained in the Methodology section.

Methodology

Noise is the main reason that can distrupt the web data mining process. Therefore, it is crutial to clean up a web page before extracting the records from it. Thus the current work aims at removing noise and then identify data regions. It involves the following five steps:

1. Mapping the tag path to a sequence of identifiers.
2. Constructing Suffix trie (Ukkonen, 1995, pp. 249-260), from sequences of identifier that acquired from the previous step.
3. Extracting true data record sequences using four refining filters as: Frequency filter, Gap filter, Tree filter, and Repetition filters.
4. Identifying data regions.
5. Removing noise by pruning the DOM tree.

The first step, representation of web page as sequence of identifier, has been introduced in the previous section. The second step is implemented using the Ukkonen algorithm; a well known algorithm for constructing the Suffix trie of a given string in linear time. The third step extracts true sequence of data records from the constructed Suffix trie in Step 2; which is refined through use of filters from the STEM algorithm. The last two steps will be introduced in the following sections.

Determining Data Region

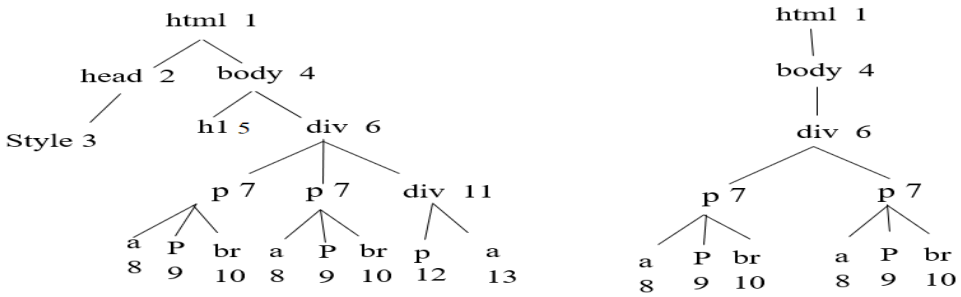
Localization of data regions is based on the insight that data records in the same region are more structurally similar to each other. These similar web page sequences are putted together in a cluster named Data Record Cluster (DRC). For example, let a web page sequence identifier be a list of sequences S where S = $(s_{1\dots K}, s_{2\dots K}, s_{3\dots K}, s_{n\dots K})$ and a data record sequence be a set R, R= $(r_1, r_2, r_3 \dots r_k)$. Elements of R of length k are aligned with those of list S from left to right. Then, the similarity of two sequences are measured using the formula of Equation 1 where lcs is the longest common subsequence algorithm. If μ>0.60, the matched part of S is placed inside DRC cluster. When a match is not found, the next element is selected and the procedure continues with this selection.

$$\mu(s_{n\dots k,r}) = \frac{length(lcs(s_n, r_k))}{length(r_k)} \quad \dots\dots\dots\dots\dots (1)$$

Finally, the DRC will be formed as $(s_1, s_2, \dots, s_i)$, where the subscript i is the size of the cluster, n is the size of web page sequence, k is the length of data record sequence and r is the record pattern. For example, the web page sequence identifier of DOM tree given in Figure 2 is ”1,2,3,4,5,6,4,5,6,7,4,5,6” where n = 13 and data record sequence is “4,5,6” with k = 3, the DRC will contian sequences 3,4,5,6,4,5,6,7,4,5,6 with i = 11. The unmatched elements are noises which can be eliminated with the noise removal step.

Removing Noise

To remove noise and preserve the DOM structure of a web page, DOM tree pruning algorithm is utilized proposed by (Velloso and Dorneles, 2013, p. 173). The algorithm takes two inputs: A DOM tree of the desired web page as a map and node sequences which will be removed. A priliminary step is taken first to make a list of all nodes that do not belong to DRC which is called Exclusion (E) sequences. Then, leaf nodes of E are removed from the DOM. Consider Figure 4 as an example.



(A). DOM tree (B). Pruned DOM tree

Figure 4.A Sample of DOM Tree and Pruned DOM After Removing Noise

Representation of DOM tree shown in Figure 4 as a sequence of identifiers is "1,2,3,4,5,6,7,8,9,10,7,8,9,10,11,12,13" and the true data record sequence is "7,8,9,10". The cluster contains the sequence "6,7,8,9,10,7,8,9". The sequence "1,2,3,4,5,11,12,13" is an element of the list E. To remove a sequence from E, its DOM tree is referred to check whether this sequence belong to a leaf node. If so, the sequence is removed from its parent node recursively. For example, 12 (<p>) is a leaf node and is an element of E. Therefore, it is removed from the DOM tree which results in the DOM tree shown in Figure 4(B).

Data Set and Experimental Results

Both algorithms are run on Testbed data set collected by (Yamada et al., 2004) and some publicly availabe webpages shown in table 2. Testbed data set contains 51 web sites each with five web pages totalling 255 web pages. The systems are tested on randomly selected pages and the results are shown in Table 2. In this table the first column from left shows the website number or name, column two shows the total number of nodes in original web page. Column three and four from left show the number of noisy nodes pruned from DOM and its percentage respectively by the proposed work. It also shows that the average number of noisy nodes per page is 42.90 %. This percentage is in agreement with the results that around 40–50% of a web page is template content put forward by (Gibson et al., 2005, pp. 830-839). Whereas the TPS reports 21.52% removal of noisy nodes. Actual data regions detected are shown in the fifth and eight columns as 85% and 77% by the proposed work and the TPS algorithm respectively. Thus, proposed work can reach a detection rate of true positives of almost 85% while removing 43% of (noisy) nodes from DOM tree.

Table 2. The Result of Proposed Work (PW) and TPS Filter Method

Web site	Total # Of nodes	#Removed nodes by PW.	Reduction	Actual data region found by PW.	#Removed nodes by TPS	Reduction	Actual data region found by TPS
1	512	238	46.48%	Positive	174	33.99%	Positive
6	308	95	30.85%	Positive	62	20.12%	Positive
10	385	86	22.34%	Positive	79	20.52%	Positive
17	535	137	32.34%	Positive	52	9.72%	Positive
24	1045	790	75.6%	Positive	294	28.14%	Positive
28	256	129	50.4%	Positive	111	43.33%	Negative
30	103	42	40.78%	Positive	17	16.5%	Positive
31	425	183	43.06%	Positive	4	0.95%	Negative
50	501	338	67.47%	Positive	17	3.4%	Positive
aydin.edu.tr	1656	802	48.43%	Positive	780	47.11%	Positive
amazon.com	2444	622	25.46%	Positive	575	23.53%	Positive
acm.org	783	203	25.93%	Negative	161	20.57%	Positive
alibaba.com	982	477	48.58%	Negative	117	11.92%	Negative
Average			42.9%	84.61%		21.52%	76.92%

An example of a web page from Testbed data set shown in Figure 5 (A). It shows that the original page's DOM tree has 535 nodes. Figure 5 (B) shows the same web page after removing noise from the original using the proposed method. As it can be seen 137 noisy nodes are removed. Whereas applying the TPS filter on the same page, only 52 noisy nodes are removed leaving 85 noisy nodes in the DOM tree. Also, the number of true positives fort he data regions usign the propsed method is higher than that of TPS filter.



(A) Original web page (B) Web page after noise removal

Figure 5. Noise removal using proposed work from web site number 17 of Testbed data set

Overall testing on the whole website of aforementioned data set shows that the proposed work can detect data regions correctly 39 out of 51 web sites. Out of 12 web sites the proposed method missed 7 regions and data regions of 5 web site are still detected with some missing records.

Conclusions and Future Work

The existence of noise in web documents negatively affects extraction of informative data from web pages. To overcome this problem, a synthesis is created combining the most successful components of STEM and TPS methods with some slight modifications to further improve accuracy of detection. In contrast to BUW and RSP, the proposed method is a general method that can deal with HTML tag dependency (ul/ol dependency) and user labeling limitations. To simplify the Web page representation, a sequence-based representation has been used instead of the traditional DOM tree. Then, based on the insight that structured regions of a web page consist of sequences which appear repeatedly and continuously, extraction such sequences is performed using the STEM algorithm. Collection of all similar data record sequences in the cluster named DRC is done through some similarity measure. As the last step in the process, noise in the structred regions are removed by pruning nodes that break the pattern in the sequence. The results obtained using the proposed methods show that its accuracy is superior to TPS in terms of noise removal and true positives. The proposed work still fails to find all data regions for some sites which leaves room for improvement as future work.

References

Aggarwal, C. C. (2014). An Introduction to Data Classification. *Data classification: algorithms and applications*, *1*.

Azir, M. A. B. M., & Ahmad, K. B. (2017, November). Wrapper approaches for web data extraction: A review. In *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)* (pp. 1-6). IEEE.

Fang, Y., Xie, X., Zhang, X., Cheng, R., & Zhang, Z. (2018). STEM: a suffix tree-based method for web data records extraction. *Knowledge and Information Systems*, *55*(2), 305-331.

Ferrara, E., De Meo, P., Fiumara, G., & Baumgartner, R. (2014). Web data extraction, applications and techniques: A survey. *Knowledge-based systems*, *70*, 301-323.

Gupta, S., Kaiser, G., Neistadt, D., & Grimm, P. (2003, May). DOM-based content extraction of HTML documents. In *Proceedings of the 12th international conference on World Wide Web* (pp. 207-214).

Jokar, N., Honarvar, A. R., AgHAMIRZADEH, S., & Esfandiari, K. (2016). Web mining and Web usage mining techniques. *Bulletin de la Société des Sciences de Liège*, *85*(1), 321-328.

Kayed, M., & Chang, C. H. (2009). FiVaTech: Page-level web data extraction from template pages. *IEEE transactions on knowledge and data engineering*, *22*(2), 249-263.

Liu, B. (2011). Web data mining: exploring hyperlinks, contents, and usage data. *Berlin: springer*, *1*.

Liu, B., Grossman, R., & Zhai, Y. (2003, August). Mining data records in web pages. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 601-606).

Miao, G., Tatemura, J., Hsiung, W. P., Sawires, A., & Moser, L. E. (2009, April). Extracting data records from the web using tag path clustering. In *Proceedings of the 18th international conference on World wide web* (pp. 981-990).

Mughal, M. J. H. (2018). Data mining: Web data mining techniques, tools and algorithms: An overview. *Information Retrieval*, *9*(6).

Sahuguet, A., & Azavant, F. (1999, September). Building light-weight wrappers for legacy web

Schulz, A., Lässig, J., & Gaedke, M. (2016, October). Practical Web data extraction: are we there yet?-a short survey. In *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)* (pp. 562-567). IEEE.

Simon, K., & Lausen, G. (2005, October). ViPER: augmenting automatic information extraction with visual perceptions. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 381-388).

Thamviset, W., & Wongthanavasu, S. (2014, July). Bottom-up region extractor for semi-structured web pages. In *2014 International Computer Science and Engineering Conference (ICSEC)* (pp. 284-289). IEEE.

Thamviset, W., & Wongthanavasu, S. (2012, May). Structured web information extraction using repetitive subject pattern. In *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology* (pp. 1) IEEE.

Ukkonen, E. (1995). On-line construction of suffix trees. *Algorithmica*, *14*(3), 249-260.

Velloso, R. P., & Dorneles, C. F. (2013). Automatic web page segmentation and noise removal for structured extraction using tag path sequences. *Journal of Information and Data Management*, *4*(3), 173-173.

Velloso, R. P., & Dorneles, C. F. (2020, September). Optimized Extraction of Records from the Web Using Signal Processing and Machine Learning. In *Anais do XXXV Simpósio Brasileiro de Bancos de Dados* (pp. 109-120). SBC.

Yamada, Y., Craswell, N., Nakatoh, T., & Hirokawa, S. (2004, May). Testbed for information extraction from deep web. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters* (pp. 346-347).

Zhai, Y., & Liu, B. (2005, May). Web data extraction based on partial tree alignment. In *Proceedings of the 14th international conference on World Wide Web* (pp. 76-85).

Zhang, K., & Shasha, D. (1989). Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, *18*(6), 1245-1262.

Beyan ve Açıklamalar (Disclosure Statements)

1. Araştırmacıların katkı oranı beyanı / Contribution rate statement of researchers:

    1. Yazar/First author %70,

    2. Yazar/Second author %30.

2. Yazarlar tarafından herhangi bir çıkar çatışması beyan edilmemiştir (No potential conflict of interest was reported by the authors).