Integrated-Block: A New Combination Model to Improve Web Page Segmentation

Saeedeh Sadat Sajjadi Ghaemmaghami* and James Miller

University of Alberta, Canada
E-mail: sajjadig@ualberta.ca; jimm@ualberta.ca
**Corresponding Author*

Received 19 April 2021; Accepted 31 January 2022;
Publication 15 April 2022

Abstract

Context: Web page segmentation methods have been used for different purposes such as web page classification and content analysis. These methods categorize a web page into different blocks, where each block contains similar components.

Objective: The goal of this paper is to propose a new segmentation approach that semantically segments web pages into integrated blocks and obtains high segmentation accuracy.

Method: In this paper, we propose a new segmentation model that semantically segments web pages into integrated blocks, where (1) it merges web page content into basic-blocks by simulating human perception using Gestalt laws of grouping; and, (2) it utilizes semantic text similarity to identify similar blocks and regroup these similar basic-blocks as integrated blocks.

Results: To verify the accuracy of our approach, we (1) applied it to three datasets, (2) compared it with the five existing state-of-the-art algorithms. The results show that our approach outperforms all the five comparison methods in terms of precision, recall, F-1 score, and ARI.

Journal of Web Engineering, Vol. 21_4, 1103–1144.
doi: 10.13052/jwe1540-9589.2146

Conclusion: In this paper, we propose a new segmentation model and apply it to three datasets to (1) generate basic-blocks by simulating human perception to segment a web page, (2) identify semantically related blocks and regroup them as an integrated block, and (3) address limitations found in existing approaches.

Keywords: Web page analysis, web page segmentation, semantic text similarity, Gestalt law of grouping.

1 Introduction

Web page segmentation is the process of segmenting a web page into different blocks, where each block contains similar components in terms of structural, visual, or contextual similarity. Web page analysis methods utilize web page segmentation for different purposes such as web page classification, detecting malicious web pages, and content analysis [1]. Most of these methods use the Document Object Model (DOM) structure of a web page to extract information. A DOM tree is a cross-platform structure that represents the HTML document of a web page as a tree, where each fragment (HTML tag) of the document is related to a particular node of the tree. Although a DOM tree represents the structure of a web page, limited information is available from the DOM tree, and it cannot represent the semantic concepts of a page accurately.

To overcome this limitation, segmentation methods have been carried out using vision-based techniques. The visual layout of web pages is analyzed to segment a page into different blocks that are visually similar [2]. Although these segmentation methods identify visually similar blocks, they cannot identify semantically similar blocks. For example, some blocks can have different visual layouts with similar concepts, so they need to be groped in a single block.

Some research proposes segmentation methods based on both DOM-based and text-based approaches [3–5]. They use textual properties such as text density to contextually segment a page and identify the blocks. Although they use textual properties, they do not semantically segment web pages. Semantic analysis includes extracting text from segmented blocks, computing textual similarity, and regrouping blocks. A fusion segmentation approach that simulates human perception and utilizes structure, vision, and text-based methods is required to identify similar blocks and obtain higher segmentation accuracy.

Human tends to classify ambiguous objects based on their understanding. So, they group visually similar elements in a category. As an example, assume a page of a newspaper. Our mind automatically categorizes this page into separate groups without reading the text according to different features of the page such as the size of each column, font size, aligned lines, images, etc. This idea is proposed by Koffka et al. [6] and is known as Gestalt laws. According to Gestalt laws, humans group visually similar objects together based on several rules known as Gestalt laws of grouping [6–9].

We propose a new combination model of web page segmentation by dividing the content of a web page into blocks by initially considering human perception (inspired by Gestalt laws of grouping) and subsequentially re-segmenting initial similar blocks using semantic text similarity. This paper contributes to current research in web page analysis in the following ways:

- To improve the segmentation accuracy, this paper provides a new semantic method of web page segmentation by merging the DOM structure, vision-based similarity features, and text-based similarity metrics of web pages.
- Specifically, can a low-level visual-based segmentation be augmented with a high-level segmentation process that provides a semantic analysis of textual features?
- Further, we demonstrate the utility of transformer technology as a vehicle for this text-based process.
- By evaluating the system on three public datasets and by comparing it with state-of-the-art studies, the results represent that our proposed approach outperforms five other existing web page segmentation methods, in terms of higher accuracy.

The rest of the paper is organized as follows. Section 2 explains the challenges and our research motivations, while Section 3 reviews related work. A detailed description of our approach is provided in Section 4, whereas Section 5 presents an evaluation of the proposed approach. Section 6 concludes the paper and provides some future work directions.

2 Problem Statement and Research Motivation

Humans can easily segment web pages into related blocks based on their understanding. To segment web pages based on human perception, it is essential to employ laws that simulate human understanding. Also, it is required to utilize semantic text similarity to segment web pages. This paper

segments web pages by simulating human perception and utilizing structure, vision, and text-based methods.

The DOM structure of a web page is used in most of the existing research on web page segmentation. This segmentation model can only gain limited information from web pages. To improve the segmentation method, some research has been carried out using visual features of web pages. To simulate human perception in the segmentation process, a series of laws are presented, the Gestalt laws of grouping. According to these laws, humans group visually similar objects together. However, Gestalt laws do not consider the text-similarity of web pages. To consider the text similarity, we use semantic text similarity metrics in addition to these laws to segment web pages into blocks. Thus, each block has related content in terms of both visual and textual features. In this paper, semantic segmentation includes dividing the content of a web page into blocks by initially considering human perception (inspired by the Gestalt laws of grouping), and subsequentially re-segmenting these initial similar blocks using semantic text similarity. Further details of Gestalt laws and text similarity metrics are provided in Section 4.

The shortcomings of the DOM structure lead to performance limitations of the structural-based segmentation methods; typically, the (long) text of a web page results in several short or scatter text segments [10]. However, these scattered text sections have related content; and hence, need to be grouped into a single block. It is hypothesized that to regroup small blocks and obtain longer text segments in larger merged blocks, a semantic analysis that uses Natural Language Processing (NLP) techniques is required.

Some segmentation methods have been carried out using NLP techniques [3–5]. These methods consider text density metrics such as text formats and words' frequency of a document but do not consider the semantic text-similarity of blocks. Two sentences can have the same meaning regardless of the choice of word and hence should be grouped in a single block; for example, consider the following two sentences that are grouped in two different blocks using a segmentation method.

"I read more books than Sarah", and "She reads fewer books than me".

Although these sentences have different words they have the same meaning. The segmentation method that segments these two blocks does not consider the semantic similarity of the sentences. Most of the segmentation methods focus on the structural and visual features of a web page. The semantic similarity of documents determines the probability of the relatedness of the documents [11]. Generally, documents are semantically similar if they convey the same meaning.

Figure 1 A Part of the Page "www.journalregister.com".

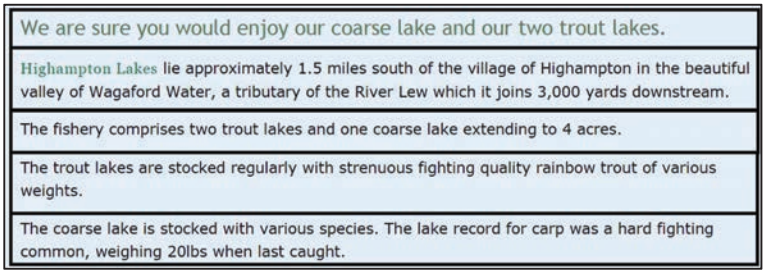


Figure 2 A Part of the Page "www.fishdevon.co.uk".

As another example, consider some parts of the web pages "www.jour nalregister.com" and "www.fishdevon.co.uk" as shown in Figures 1 and 2, respectively. The existing segmentation methods that we used in this paper to compare with our method, segment the paragraphs shown in Figures 1 and 2 into four and five separated blocks, respectively, while these blocks are semantically related and need to be grouped in a single block. These methods use DOM structure and visual properties of web pages to segment pages. Most of the segmentation methods do not use semantic text-similarity of blocks. Thus, some paragraphs with similar subjects are segmented into different blocks based on their text formats, instead of employing the semantic text-similarity of blocks.

Some research use textual features of web pages, for instance, Jiang et al. [10] propose a segmentation method that uses logical and visual features of content. Also, this method uses text density (the number of words in a block of text) to segment web pages. As mention in [10], this method does not consider semantic text similarity metrics of blocks. Also, we have proposed

a segmentation method [37] that uses structural, visual, and textual features of web pages. This method uses the Doc2Vec technique to compare the text-similarity of web pages. Appendix-A provides a quick background of the text representation methods.

Doc2Vec is an NLP technique for representing documents as a vector and is a generalization of the Word2Vec method. The Word2Vec technique assigns a single word embedding vector to each word in a text corpus based on the frequency of words. For example, the word "goal" in "last-minute goal" and "life goal" has the same word embedding representation vector. However, Doc2Vec does not use the relation of the word to the other words in a document by considering the meaning of sentences. Thus, it does not deeply compare the text-similarity of web pages. A segmentation approach that deeply computes the semantic text-similarity of blocks is required to regroup similar blocks.

Google has proposed the Bidirectional Encoder Representations from Transformers (BERT) method for NLP techniques such as text classification, summarization, generation, and similarity [12]. BERT is a neural network-based technique to represent text. It uses the transformer encoder to learn a language model [13]. The transformer uses a sequence of tokens as an input and returns a sequence of vectors as an output, where each vector corresponds to an input token. A token is an instance of a sequence of characters in a document that is grouped as a semantic unit for processing [14]. First, the transformer randomly masks some of the input tokens to train a deep bidirectional representation and then predicts the masked tokens [12]. For example, consider this sentence "Sarah reads more books than me". BERT represents "reads" using both its right and left words by masking the word in the input as "Sarah xxx more books than me". Then, it runs the entire sequence through a bidirectional transformer encoder and predicts the masked word. More information can be found in [12]. (This paper investigates the application of BERT as a mechanism to improve semantic web-based segmentation). To overcome the shortcomings of the structural-based and the NLP-based segmentation methods, this paper uses BERT to deeply compute the semantic textual similarity between the basic-blocks and regroup the related blocks in merged blocks. This similarity regrouping model leads to more stable semantic features. Further details are provided in Section 5; Figures 15(e) and 16(e) show the results of web page segmentation using our approach on some parts of the web pages represented in Figures 1 and 2.

To achieve a high-performance segmentation model, it is required to merge the visual and textual content of a web page into a single model. By

using the latest semantic NLP techniques, such as BERT, we can produce a superior model. We demonstrate the validity of this conjecture by an empirical comparison against the current state-of-the-art techniques.

3 Related Work

In this section, we briefly review the relevant literature on web page segmentation, which can be divided into three categories: DOM-based, vision-based, and fusion approaches. The fusion approaches combine the DOM-based, vision-based, and/or text-based approaches to obtain higher segmentation accuracy or for specific applications. The related work of each category is explained in the following paragraphs.

Most segmentation methods follow a DOM-based approach that divides a page into blocks. For example, the DOM tree representation of a web page is created by using an HTML parser to analyze and extract content, as described by Gupta et al. [15]. The content extractor uses filtering techniques to navigate the DOM tree and modify specific nodes, eliminating non-content nodes in the process. This method, it should be noted, does not perform well on rich format web pages in comparison with textual pages.

Chen et al. [16] propose a method that identifies content blocks using a partitioning algorithm that divides a single content block into several smaller ones. This approach considers the whole page as a single block that it then partitions into constituent subcomponents such as a left sidebar, a right sidebar, a header, a footer, and the main content. Fan et al. [17], meanwhile, propose a Site Style Tree (SST) to capture web page content. In this method, information-rich content is extracted from each node of the SST using entropy thresholding.

Some methods segment web pages using vision-based properties. For example, Kong et al. [18] propose a segmentation approach by using the Spatial Graph Grammar (SGG) without relying on DOM structures. This method directly interprets a web page from its image, instead of DOM structures. Image-processing techniques are used to divide an image into different regions and recognize and classify objects, such as texts, buttons, etc., in each region.

Other methods segment web pages using fusion approaches. In this paper, we divide them into two categories of (1) DOM-based and vision-based, and (2) a combination of DOM-based, vision-based, and text-based approaches.

The first category of fusion methods segments web pages using structural features and visual properties. For example, Sanoja et al. [19] propose the

Block-O-Matic strategy, inspired by visual-based content segmentation techniques and automated document processing methods. This method includes three phases—analysis, understanding, and reconstruction of the web page. It combines the logical, visual, and structural features of web pages to understand and analyze the content. Manabe et al. [20], meanwhile, propose a method—called HEPS (HEading-based Page Segmentation)—to extract logical structures of web pages. This method uses the HTML tags, computed style calculated by Web browsers based on several factors, and the image height to determine the visual style of the DOM tree nodes of a web page.

A Vision-based Page Segmentation (VIPS) algorithm is proposed by Cai et al. [21, 22]. This algorithm divides a page into fragments based on the visual properties and logical structure (i.e., DOM) of the page. Once again, although this method performs well on traditional pages, it does not perform well on modern web pages. The box clustering segmentation model, introduced by Zeleny et al. [2], uses visual properties of web pages, the distance between elements, and their visual similarity, and follows a three-step procedure: box extraction, computation of distances between boxes, and clustering of boxes. Cormer et al. [23] propose a hierarchal segmentation method that similarly uses the visual properties of the web page to achieve segmentation. The method presented by Mehta et al. [24] uses the VIPS algorithm to divide pages into small fragments based on visual properties, it also uses a pre-trained Naive Bayes classifier to create bigger blocks.

Liu et al. [25] propose the ViDE approach that primarily utilizes the visual features human users can capture on the web pages to perform deep web data extraction, including data record extraction and data item extraction. By using visual features for data extraction, ViDE avoids the limitations of those solutions that need to analyze complex web page source files.

Kumar et al. [26] propose a web page segmentation algorithm that re-DOMs the input page to produce clean and consistent segmentations. The algorithm includes four stages. First, each inlined element is identified and wrapped inside a tag. Next, the hierarchy is reshuffled. Third, redundant nodes that do not contribute to the visual layout of the page are removed. Finally, the hierarchy is supplemented to introduce missing structure which is accomplished by computing a set of VIPS-style separators across each page region and inserting enclosing DOM nodes accordingly.

To segment blocks accurately in a manner that simulates human perception in identifying related content, Xu and Miller [27–30] propose the "Gestalt Layer Merging" (GLM) model, premised on the Gestalt laws of grouping. This method can be used to segment blocks in complex modern

web pages. In the research presented in this paper, we use this method to generate the basic blocks.

The second category of fusion methods combines the structural, visual properties, and textual content of a web page to achieve segmentation. For example, Jiang et al. [10] propose a web page segmentation method that uses both visual and logical features of content. Their method uses text density (the number of words in a block of text) as its segmentation algorithm. The densitometric approach proposed by Kohlschütter et al. [31] uses the text density metric to identify blocks of a web page. These approaches, it should be noted, do not consider semantic text similarity metrics, but instead, focus on the structural and visual features of the web page. S-Ghaemmaghami and Miller [32] propose a fusion segmentation approach that uses DOM structure and visual features of web pages to construct basic-blocks. It uses Doc2Vec to compare the text-similarity of basic-blocks and regroup the similar blocks as fusion blocks. This method does not use deep semantic text similarity. Table 1 represents a comparison of the related work. It specifies the categories of each method. Deep semantic specifies whether or not a method uses the relation

Table 1 Comparison of properties of different segmentation methods

			Fusion Approaches			
				DOM & Vision & Text-based		
					Semantic Text Similarity	
Method	DOM Based	Vision Based	DOM & Vision-Based	No Semantic Text Similarity	Non-Deep Semantic	Deep Semantic
Gupta et al. [15]	✓					
Chen et al. [16]	✓					
Fan et al. [17]	✓					
Kong et al. [18]		✓				
Sanoja et al. [19]			✓			
Manabe et al. [20]			✓			
Cai et al. [21, 22]			✓			
Zeleny et al. [2]			✓			
Cormer et al. [23]			✓			
Mehta et al. [24]			✓			
Liu et al. [25]			✓			
Kumar et al. [26]			✓			
Xu and Miller [27–30]			✓			
Jiang et al. [10]				✓		
Kohlschütter et al. [31]				✓		
S-Ghaemmaghami and Miller [32]					✓	
Current Approach						✓

of the word to the other words in a document by considering the meaning of sentences.

There is a lack of standard procedures to compare the accuracy of web page segmentation methods [2]. However, to compare the accuracy of segmentation techniques, some research has been carried out. For example, Blustein et al. [33] design experiments to compare web page segmenters by proposing some questions that a segmentation method should answer. The questions are about the purpose of web page segmentation and the dataset used in an experiment. Some studies utilize precision, recall, and F-measure metrics to evaluate the performance of their segmentation approach. For example, Kovacevic et al. [34], Xu and Miller [27], and S-Ghaemmaghami and Miller [32] evaluate the performance of their segmentation approach using these metrics.

Our proposed approach combines the DOM-based structure, vision, and text-based segmentation techniques. As mentioned above, it generates its basic blocks using the Gestalt laws of grouping, and then employs a deep semantic text similarity method to regroup these related basic blocks into larger "integrated-blocks". In other words, each integrated-block is composed of a group of basic blocks with similar text content. It is expected that these enhanced semantically-based, visually-initiated blocks will deliver superior performance across a wide array of tasks on modern, multi-media web pages.

4 The Proposed Combination Web page Segmentation Approach

Web page segmentation keeps related content together as blocks, where each block contains distinctive content. The goal of web page segmentation is to construct a content structure from web page features that groups the elements of a web page using metrics such as distances, locations, and semantic context. An integrated model that includes DOM, visual, and semantic text-based segmentation approach is required to achieve a superior segmentation result. Our proposed approach is explained in the following paragraphs.

4.1 Proposed Approach

We have designed and implemented a framework to generate an integrated web page segmentation model that can be applied to different web pages. Our approach combines the DOM structure, visual, and textual features of

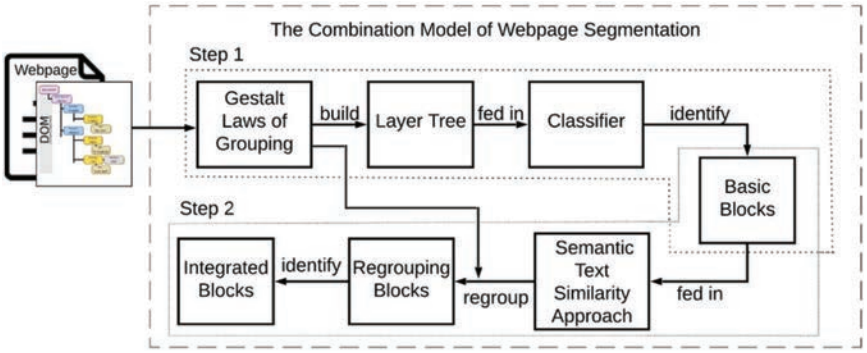


Figure 3 The framework of the combination model of web page segmentation.

web pages and overcomes the limitations of former methods. Figure 3 shows the main steps of our proposed framework.

According to Figure 3, our integrated model is mainly categorized in two steps; (1) it identifies basic blocks in web pages using the Gestalt laws of grouping technique, following the work of Xu and Miller [27]; and (2) the model employs a semantic text similarity method to regroup these related basic blocks identified in step 1 into larger integrated-blocks. These two steps are explained in the following paragraphs.

4.1.1 Step 1: Basic-Block

In step 1, our model segments web page content into basic blocks by simulating human understanding. A "layer tree" is designed to represent a web page [27]. Screenshots (images) and DOM trees are two major ways to represent a web page for visual similarity evaluation. A web page representation method that merges the advantages of these two methods is proposed as a layer tree [30]. The hierarchy of a layer tree is constructed from the DOM tree of a web page.

Nodes of a semantic block tree (layer tree) indicated as basic blocks in Figure 3 are constructed by merging correlated blocks with the Gestalt laws of grouping. The Gestalt Layer Merging (GLM) model includes three components: (1) the layer tree constructor, (2) the Gestalt laws translator, and (3) the web page block identifier [27]. The DOM tree of a web page is taken as a prototype by the layer tree constructor to build up its layer tree. Constructing layer tree nodes is done simultaneously with building up the layer tree and starts from adding the root node to the layer tree, and

then executes recursively until all visible DOM elements are extracted and added to the layer tree. Further details of constructing a layer tree can be found in [27].

The layer tree is built by removing hierarchical inconsistencies between the DOM tree representation and the visual layout of the web page. In the DOM tree, child elements are located inside their parent elements by default; however, some CSS rules can manipulate locations, such as "position", "float", etc. These rules sometimes cause the DOM hierarchy to be misaligned against the visual hierarchy. Therefore, such an inconsistency must be eliminated in the layer tree construction. Also, invisible elements existing in the DOM tree are removed when constructing the layer tree. An invisible DOM element is either an element with an area of 0 (including the borders and shadows), an element without any actual content (text, image, background, etc.), or an element that is completely covered by its visible child elements. A modification is necessary to be done on a layer tree according to [27]. Therefore, a layer tree only extracts visible DOM elements and merges these elements into separate groups according to their semantic meanings. The details of this modification can be found in [27].

The Gestalt laws explain the mechanisms of how humans perceive and understand things. To construct each block of the layer tree, the Gestalt laws translator interprets the Gestalt laws of grouping into machine compatible rules. The Gestalt laws of grouping contain 6 laws described in the following paragraphs [27].

1. Gestalt Law of Simplicity
   This law indicates that humans prefer to organize objects into the simplest units on a web page. Figure 4 shows the logo of the University of Alberta, "https://www.ualberta.ca". The logo contains multiple parts: the figure, the phrase "UNIVERSITY OF", and the big bold "ALBERTA". However, these elements have different types and styles, they are considered as a single group according to the Gestalt law of simplicity. This law helps to make the process of reading and understanding a page more straightforward.

Figure 4 Gestalt law of simplicity ("https://www.ualberta.ca").

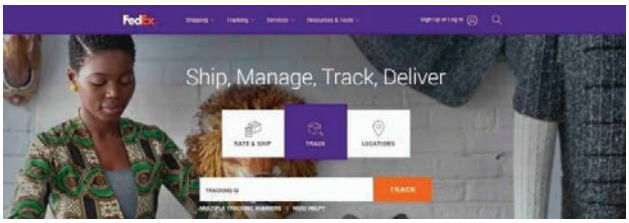


Figure 5 Gestalt law of closure ("https://www.fedex.com/").

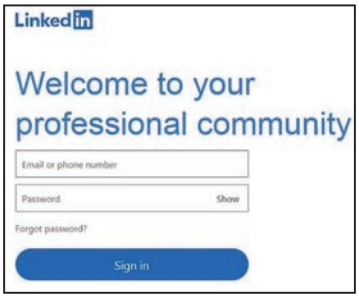


Figure 6 Gestalt law of proximity.

2. Gestalt Law of Closure
   This law states that humans tend to perceive incomplete shapes as complete ones. As an example, Figure 5 represents the homepage of FedEx, "https://www.fedex.com/". However, the middle part of the background image is covered by a search and three other boxes, it is believed that the background image is complete [27].
3. Gestalt Law of Proximity
   According to this law, humans tend to group close objects. This law groups elements based on their distances. To determine proximity, the distance in the GLM model is defined as the Normalized Hausdorff Distance (NHD) between layers, which provides the best performance as a proximity estimation [28]. NHD aims to group elements if their distances with adjacent elements are similar. Further details can be found in [28].

$$NHD(L_1, L_2) = \max\left\{\left(\frac{hd_{1,2}}{Re_{L_1}}, \frac{hd_{2,1}}{Re_{L_2}}\right)\right\} \tag{1}$$

Re_{L_1} and Re_{L_2} are the relevant lengths of layers L_1 and L_2, and $hd_{1,2}$ and $hd_{2,1}$ are the Hausdorff distance from L_1 to L_2 and L_2 to L_1,

respectively. Using the sign-in page of LinkedIn (https://www.linkedin.com/) as an example shown in Figure 6, the two boxes regarding sign-in ("Email or phone number" and "Password") are regarded as a group.

4. Gestalt Law of Similarity
   The Gestalt law of similarity indicates that humans perceive similar elements as a single group. To compare elements, this law considers their visual features such as background similarity, foreground similarity, and size similarity. Background similarity includes both the color and the image; foreground similarity compares textual and paragraph styles; and size similarity checks if the two bocks share the same width or height. A more precise set of definitions can be found in [29].
   To calculate digital image comparison to imitate human perception, Structural Similarity Index (SSIM) [6] is used. SSIM is capable of distinguishing between similar pages and dissimilar pages [28]. Figure 7 shows six objects grouped into three groups in terms of styles. The left two objects are in one group, the middle two objects belong to a second group, and the right two objects are included in a third group.
5. Gestalt Law of Continuity
   This law expresses that humans tend to judge the elements on a web page as related in a situation where they are aligned, and as dissimilar when they are not aligned. Using the homepage of the University of Alberta (https://www.ualberta.ca/) as an example shown in Figure 8, the paragraphs in the black rectangle ("Campus Info", "Locations", "Libraries", "Dining Services", etc.) are left-aligned, indicating they are related content. To evaluate continuity, we compare the left, top, right, and bottom coordinates of two elements. If any of the four coordinates of two elements are the same, we conclude that they are related and that they are dissimilar, otherwise [28].

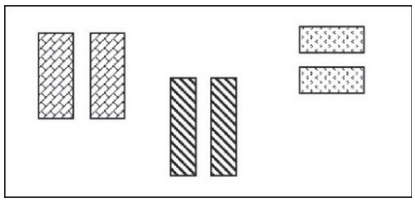


Figure 7 Gestalt law of similarity.

Figure 8 Gestalt law of continuity.

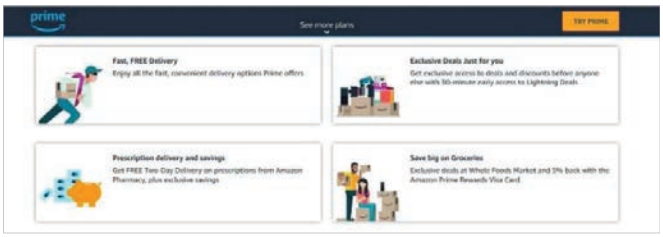
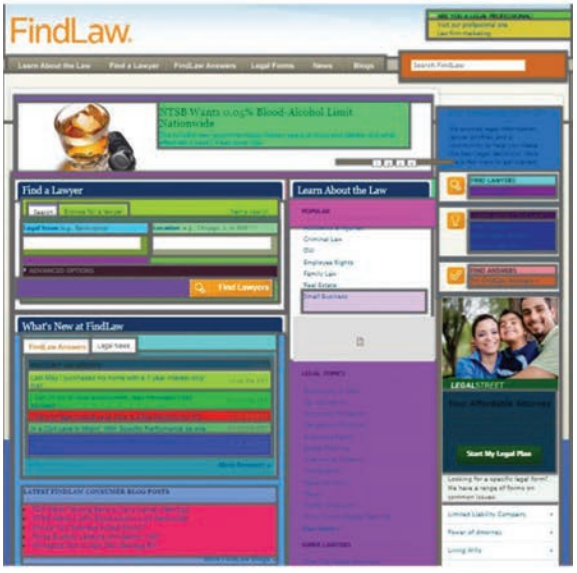


Figure 9 Gestalt law of common fate.

6. Gestalt Law of Common Fate
   This law describes that people tend to regard elements with the same motion trend as related. For example, the upper ribbon with the dark blue background color in Figure 9 (the homepage of Amazon Prime, "www.amazon.com/amazonprime") hangs at the top and does not move with scrolling the page, but other content moves accordingly.

According to these six laws, a model can determine elements to be segmented in a group or not. Each group (basic block) includes the results of six laws merged. Our model uses a naïve Bayes classifier same as [30] to merge these laws and explore the hidden connection between them. In this classifier, the category variable C of the classifier is set as "0" and "1", representing "not merge" and "merge", respectively, while the feature vector consists of six variables, each representing the corresponding Gestalt law ($F = (F_1, F_2, F_3, F_4, F_5, F_6)$). Figure 10 shows the basic blocks of two web pages (the homepage of the United States Consumer Product Safety Commission "https://www.cpsc.gov" and FindLaw "www.findlaw.com"), where for each basic block, a different background color is assigned. Step 2 of our model is presented to complete the segmentation process by considering semantic textual analysis explained in the following paragraphs.

(a)



(b)

Figure 10 An Example of the Basic Blocks of the Two Web pages, (a) "www.cpsc.gov", (b) "www.findlaw.com".

4.1.2 Step 2: Integrated-block

As represented in step 1, block features such as textual styles, width, height, and color are compared regardless of the semantic analysis. Hence, objects may be segmented in different blocks, even though they have semantically related text. It is required to employ semantic analysis to address this problem. This paper utilizes semantic text similarity to identify semantically related blocks and regroup these blocks as an integrated-block shown in Figure 3. A description of text representation and textual semantic analysis are explained in the following paragraphs.

4.1.2.1 *Text representation*

Text analysis has been a common research topic in tasks such as text representation, text similarity, etc. [35]. It plays an important role in NLP and aims to numerically represent unstructured text documents into a structured form that can be processed and analyzed by computers [36]. Many methods have been proposed to transform raw data (a series of symbols and words) into the form of a vector at character, word, sentence, or document level to express the similarity and dissimilarity between textual elements [37]. Text representation methods are mainly divided into two types: non-contextual (for example, Word2Vec [38, 39], GloVe [40], and Doc2Vec [41]) and contextual representation (for example, ELMo [42], GPT [43], and BERT [12]). A quick background of the text representation methods is provided in Appendix-A.

Google proposes an improved method, BERT (Bidirectional Encoder Representations from Transformers) [12], which can effectively exploit the deep semantic information of a sentence. Deep bidirectional means that it is conditioned on every word in the left and right contexts at the same time [44]. It works by masking some percentage of the input tokens at random and then predicting those masked tokens. Several studies reported that contextualized embeddings such as BERT better encode semantic information of a text [14]. Also, according to [14], BERT obtains state-of-the-art results in numerous benchmarks. BERT has set an advanced performance on sentence-pair regression tasks like semantic textual similarity [45]. A shortcoming of the BERT network structure is that it maps sentences to a vector space that is rather unsuitable to be used with common similarity measures like cosine-similarity. To address this limitation, Reimers et al. [45] modify the pre-trained BERT model and propose Sentence-BERT (SBERT), which uses Siamese and triplet network structures to derive semantically meaningful sentence embeddings [45]. It uses cosine-similarity to compare the similarity between two sentence embeddings. SBERT is trained on the SNLI [46],

Multi-Genre NLI [47], and STS benchmark dataset. It is fine-tuned with a 3-way softmax-classifier objective function for one epoch. More details can be found in [45]. To identify semantically related blocks and regroup them as an integrated block, our model compares the semantic similarity of nearby basic blocks using the SBERT technique in the same way as [45]. Therefore, integrated blocks not only form based on visual features by simulating human perception but also utilize semantic analysis to improve web page segmentation.

4.1.2.2 *Textual semantic analysis*

As it is mentioned earlier, the performance of structural-based segmentation methods can be restricted by the shortcomings of the DOM structure itself. The complex DOM structure leads to shortening or scattering of long text of a web page content that is hard to extract useful features. Therefore, it is difficult to extract useful features from short text content, which challenges semantic analysis. Also, paragraphs with similar subjects are separated into different blocks because they contain text with different formats such as different font sizes, font colors, etc. Our proposed approach addresses these problems using semantic analysis and regrouping the related scattered blocks into an integrated block that contains longer text. In the first step of our method, all the six Gestalt laws are already translated and considered to identify the basic blocks. In the second step, our method compares the text-similarity of blocks containing textual content. Our semantic regrouping method is presented in Algorithm 1. In this algorithm, there are two inputs, neighboring (adjacent) basic blocks, and text difference limit t. (Text difference limit threshold can be set as $t = 0.4$, which is based on the empirical results presented in Section 5.) This algorithm semantically regroups basic blocks based on the text-similarity and the Gestalt laws of grouping (proximity and continuity). We used SBERT to evaluate the text-similarity of the basic blocks. Our model employs Gestalt laws of grouping (proximity and continuity) same as [30] to regroup the basic blocks. The following highlights represent the reasons for using these two laws in the second step of our approach.

- The Gestalt laws of simplicity, closure and common fate are already translated and employed in the first step. They are not changed in the second step.
- The Gestalt law of similarity considers the visual features such as background similarity, foreground similarity, and size similarity of blocks.

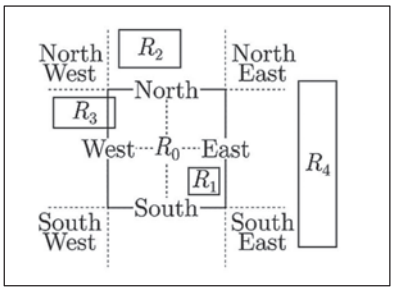


Figure 11 The NHD normalizing factor [27].

Since our algorithm regroups the related basic blocks regardless of the text format, the Gestalt law of similarity is not used in this step.

- To group related neighboring blocks, the distance of these blocks needs to be considered. Thus, the Gestalt law of proximity needs to be used in the second step.
- To group related neighboring blocks in terms of considering human perception in segmenting the content of web pages, the Gestalt law of continuity is used in this step.

Our model employs Gestalt laws of proximity and continuity same as Xu and Miller [27, 28] to regroup basic blocks. According to the Gestalt law of Proximity, humans tend to put close elements of a web page into the same group and assign distant elements into different groups. We used the Normalized Hausdorff Distance (NHD), same as [27, 28] to determine the proximity metric between blocks (objects). NHD aims to group elements if their distances with adjacent elements are similar.

The normalized Hausdorff distance (NHD) is calculated by adding a normalizing factor f to the Hausdorff distance (HD). The normalizing factor f can be the width, height, or diagonal distance of the render-block, depending on their relative position. As shown in Figure 11, the surrounding region of $R0$ is split by the dashed lines. The normalizing factor f is calculated as the height of $R2$ ($R2$ locates in the north/south region of $R0$); the width of $R3$ ($R3$ locates in the west/east region of $R0$); or the diagonal of $R4$ ($R4$ covers corner regions of $R0$).

Gestalt law of Continuity expresses that humans tend to judge the elements on a webpage as related in a situation where they are aligned, and as dissimilar when they are not aligned. To evaluate continuity, we compare the left, top, right, and bottom coordinates of the two elements. If any of the four

coordinates of two elements are the same, we conclude that they are related and that they are dissimilar, otherwise.

ALGORITHM 1: Semantic Regrouping Algorithm

Input: Two Neighboring Basic Blocks (B_1, B_2), Text Difference Limit t

Output: Integrated Block

Begin

If *Text_Similarity* (B_1, B_2) > t*

*Integrated_Block ← Regroup** (B_1, B_2);*

else

Break;

End

Return *Integrated_Block;*

End

* We used SBERT algorithm

** The blocks are regrouped according to Gestalt laws

We will continue with an example to explain our proposed approach. Figure 12 shows a part of the homepage of Highampton Lakes-Trout and Coarse Fishery (www.fishdevon.co.uk). According to part (a) of this figure, the paragraphs with similar subjects are separated into different blocks by current segmentation methods since they contain the text in different formats. Using the semantic analysis method, our model considers these similar paragraphs as a single group regardless of the different text formats as shown in Figure 12(b).

To semantically segment the part of the web page (www.fishdevon.co.uk), firstly, our model identifies basic-blocks according to the Gestalt laws of grouping. These paragraphs have related content; however, they are separated into different basic-blocks. In the second step of our model, the semantic similarity of the extracted basic-bocks are calculated and the semantically textual related blocks can be grouped according to Gestalt laws of proximity and continuity as shown in Algorithm 1. Our model regroups these basic-blocks as a single integrated block since the text difference limit of these basic-blocks is greater than t. We utilized the Gestalt laws in addition to the text-similarity of blocks to segment a web page according to human perception. After regrouping, the blocks are transformed into bigger integrated blocks that contain much more stable semantic features than before. These features can be extracted more accurately due to the bigger blocks and longer text sentences and can thus be used to achieve better performance on web

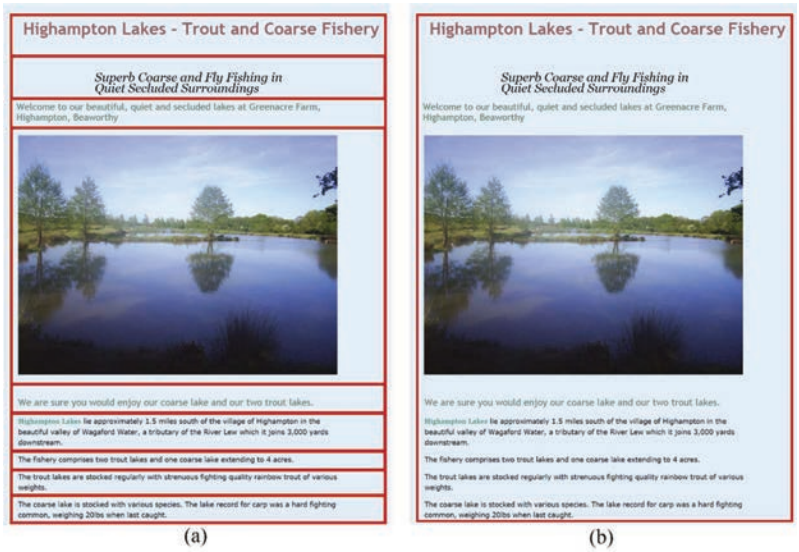


Figure 12 A part of the homepage of www.fishdevon.co.uk, (a) segmented blocks using current segmentation methods (b) segmented block using our approach.

page segmentation. Thus, our integrated approach combines DOM structure, visual and textual features of web pages to improve the segmentation accuracy.

5 Evaluation

To verify the performance of our proposed approach, we apply it to three datasets. Also, we compare our method with five existing state-of-the-art algorithms and use four evaluation metrics: precision, recall, F-1 score, and the Adjusted Rand Index (ARI). More information about our experiments and their results is presented in this section.

5.1 Research Goal

The goal of this paper is to improve new web page segmentation methods by combining the DOM structure, visual features, and semantic text similarity metrics to achieve better segmentation performance. Our method utilizes human perception to merge web page content into basic-blocks using the Gestalt laws. It subsequentially utilizes semantic text similarity to regroup similar basic-blocks as integrated blocks.

Table 2 The statistics of the datasets

Dataset	Number of Web pages	Average Number of Blocks	Median Number of Blocks
$DS_{popular}$	70	12.59	16
DS_{random}	82	8.46	13
DS_{new}	50	18.33	12.5

5.2 Dataset

To investigate the performance of our approach, we evaluate it against the following three datasets. These datasets are utilized to segment the content of web pages according to human judges (semantic blocks are manually specified).

1. DSpopular [48], a public dataset of 70 homepages of popular Websites such as "www.foxnews.com", "www.gnu.org", "www.google.com", etc., with manually labeled ground truths for segmentation collected in 2014. This dataset includes three versions of each page: (1) the basic HTML, (2) a serialized version of the DOM after all external resources are loaded, and (3) a DOM page with manually labeled semantic blocks.
2. DSrandom [49], a public dataset of 82 homepages of random Websites such as "www.honda.dk", "www.aiact.org", etc. with manually labeled ground truths for segmentation collected in 2014. Each page contains three versions as per DSpopular.
3. DSnew [50], a dataset of 50 homepages of Websites from Alexa Topsites 50 collected in 2017 such as "www.wikipedia.org", "www.facebo ok.com", etc. These pages are viewed and labeled according to human judges. This dataset is not publicly available and is created for this work.

Table 2 represents the number of web pages, the average, and the median number of blocks in each dataset. To analyze an algorithm's correctness, it is essential to have a ground truth, validated by human assessors. Thus, we evaluate the accuracy of our proposed method using the manually labeled ground truths provided for each dataset. These datasets are suitable for evaluating our method of web page segmentation since they are collected from a real-world environment and contain type-rich content.

5.3 Comparison Methods

We compare our proposed method (Integrated-Block) with the following five well-known existing web page segmentation algorithms. According to the

results, our method is superior to all these algorithms in terms of semantic web page segmentation based on human judgment.

1. VIPS [22], a well-known approach for segmenting a web page content structure based on its visual representation. This paper is used the open-source implementation of VIPS [51] which was utilized in other papers [2, 52]. As the default setting of the tool permitted Degree of Coherence parameter in VIPS is set to 8 the same as [10].
2. BoM [19], a hybrid web page segmentation method that combines structural, visual, and logical features of web pages. This method includes three phases: analysis, understanding, and reconstruction of a web page.
3. A web page segmentation method [10] combines visual, logic, and features of the content on a web page. For simplicity, we name this segmentation method as SegBlock in this paper.
4. Semantic-Block [27], a web page block identification algorithm utilizing the Gestalt laws of grouping to simulate human perception.
5. Fusion-Block [32], a fusion web page segmentation that combines structural, visual, and textual features of web pages. It uses Doc2Vec [41] technique to compare the text-similarity of web pages.

5.4 Segmentation Accuracy

It is required to define the metrics to measure the correctness of a method. Some research questions are proposed by Blustein et al. [33] that segmentation methods should answer. One of the questions is about the purpose of web page segmentation. This paper focuses on a new technique of web page segmentation that can be used in various fields such as recreating a web page in ways that can better fit the needs of users, improving the usability of web pages on mobile devices, etc. The other questions proposed in [33] are about quantifying the accuracy of the algorithm and the dataset that a segmentation method used. There is a lack of a standard procedure to compare the accuracy of web page segmentation methods [33]. However, precision, recall, and F-score are common metrics of accuracy evaluation in statistical analysis [2]. This paper uses datasets provided by ground truths to segment the content of web pages. Thus, we focus on evaluation approaches based on a ground truth such as precision, recall, and F-score. The segmentation result generated by our approach groups the elements of a web page into cohesive regions both visually and semantically. Similar to the previous works [31, 53], we regard each generated segment (block) as a cluster and employ cluster correlation metrics to conduct the evaluation. In data clustering, the Adjusted Rand Index

(ARI) is commonly used to measure the similarity between two clusters [2]. In this paper, to verify the accuracy of the segmentation method computed by our approach and the other five comparison algorithms, the following four evaluation metrics (precision, recall, F-1 score, and Adjusted Rand Index (ARI)) are used.

1. Precision represents the ratio of correctly segmented blocks over the blocks segmented by the algorithm as Equation (2).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

   TP denotes two similar blocks that are correctly identified as similar; while FP indicates that two different blocks are identified as similar, incorrectly.
2. Recall represents the ratio of correctly segmented blocks over the blocks that are manually obtained by humans (ground truth) as Equation (3).

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

   FN indicates that two similar blocks are identified as different, incorrectly.
3. F-1 score which combines the precision and recall metrics and is computed as follows.

$$\text{F-1 score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

4. Adjusted Rand Index (ARI) [54], which identifies the agreement between two clusters (segmented blocks and ground truth clustering) on a particular dataset shown in Equation (5). Value of the Rand Index is between 0 and 1; clusters' agreement on any pair of elements leads to value 1 shows these clusters are the same, and 0 states that the clusters do not agree on any elements. A version of the Rand Index is called ARI which has a value between 0 and 1. 1 indicates that two blocks are identical; and for random blocks, the value is 0 on average. ARI can be calculated as follow.

Consider a set of n objects $S = \{O_1, O_2, \ldots, O_n\}$, and suppose that $X = \{x_1, x_2, \ldots, x_r\}$ and $Y = \{y_1, y_2, \ldots, y_s\}$ represent two different partitions (blocks) of the objects in S. Given two partitions, X and Y, with r and s subsets, respectively, the contingency Table 2 can be formed to indicate the

Table 3 Contingency table for comparing partitions X and Y

		Y				
Partition	Group	y_1	y_2	...	y_s	Sums
X	x_1	n_{11}	n_{12}	...	n_{1s}	a_1
	x_2	n_{21}	n_{22}	...	n_{2s}	a_2
	...	...	...	...	...	...
	x_r	n_{r1}	n_{r2}	...	n_{rs}	a_r
	Sums	b_1	b_2	...	b_s	

group overlap between X and Y as n_{ij}, where $n_{ij} = |x_i \cap y_j|$. Let a_i and b_j be the number of objects in partitions x_i and y_j, respectively. In Table 3, a generic entry, n_{rs}, represents the number of objects that were partitioned in the rth subset of partition r and the sth subset of partition s [55]. Thus, the ARI can be expressed as:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}} \tag{5}$$

5.5 Evaluation and Results

All the web pages in the datasets have been segmented with our approach and the other five methods. Table 4 represents the results of the evaluation metrics in each dataset for the six comparison methods. "Total" contains the results of applying the methods on the combined three datasets (DSpopular, DSrandom, and DSnew). "Correct" includes the average number of correctly segmented blocks of web pages for each dataset; a block is correctly segmented if its geometry and location are equal to a labeled block in the ground truth. According to this table, BoM, VIPS, SegBlock, Semantic-Block, Fusion-Block, and Integrated-Block methods achieve 25.74%, 24.14%, 38.16%, 41.67%, 48.06%, and 53.55% of the average number of correctly labeled blocks in the Total dataset, respectively. Thus, it indicates that our approach (Integrated-Block) achieves an improvement in terms of the average number of correctly labeled blocks.

The Fusion-Block method reaches the second-highest value of the average number of correctly segmented blocks. This method segments web pages by simulating human perception using the Gestalt laws of grouping. It also compares the textual similarity of blocks using the Doc2Vec algorithm. The

third-highest value of the average number of correctly segmented blocks belongs to the Semantic-Block algorithm. This method segments web pages by simulating human perception regardless of textual analysis. We believe that simulating human perception allows this method to achieve the third-highest average number of correctly labeled blocks. The next highest amount of the average number of correctly labeled blocks belongs to the SegBlock method that segments web pages using visual and logical features of content. The number of characters within a particular document is used by this method. We believe that SegBlock uses more features comparing to BoM and VIPS, and thus, it reaches the fourth maximum average number of blocks after our approach, Fusion-Block, and Semantic-Block. The VIPS method identifies blocks using visual separators of web pages. The BoM algorithm relies on visual and logical features to segment a web page. Since the layouts of modern web pages are more complicated than before and the visual separators are much less obvious, we believe that BoM achieves a slightly better performance than VIPS in the amount of correctly segmented blocks over the whole dataset.

As shown in Table 4, our approach (Integrated-Block) outperforms all five comparison methods in terms of precision, recall, F-1 score, and ARI. It obtains 53.4%, 63.2%, 57.8%, and 0.660 in precision, recall, F-1 score, and ARI, respectively which shows a noticeable improvement in the segmentation's quality. According to this table, Integrated-Block achieves 15.3%, 15.1%, 15.1%, and 13.2% improvements against Fusion-Block (the second-highest amount of the evaluation metrics in the Total dataset) on the precision, recall, F-1 score, and ARI, respectively. The following highlights represent the results of comparing our approach (Integrated-Block) with the other comparison methods.

- On precision, Integrated-Block reaches 70.1%, 116.2%, 34.8%, 29.6%, and 15.3% improvements against BoM, VIPS, SegBlock, Semantic-Block, and Fusion-Block, respectively.
- On recall, Integrated-Block reaches 126.5%, 144.9%, 49.0%, 35.3%, and 15.1% improvements against BoM, VIPS, SegBlock, Semantic-Block, and Fusion-Block, respectively.
- On F-1 score, Integrated-Block reaches 95.9%, 129.3%, 41.3%, 31.9%, and 15.1% improvements against BoM, VIPS, SegBlock, Semantic-Block, and Fusion-Block, respectively.
- On ARI, Integrated-Block reaches 46.6%, 62.9%, 28.4%, 25.5%, and 13.2% improvements against BoM, VIPS, SegBlock, Semantic-Block, and Fusion-Block, respectively.

Table 4 Evaluation results

			$DS_{popular}$		
	Correct	Precision	Recall	F-1 Score	ARI
BoM	2.78	30.5%	26.1%	28.1%	0.452
VIPS	2.78	23.7%	26.2%	24.9%	0.420
SegBlock	5.62	38.1%	40.2%	39.1%	0.530
Semantic-Block	6.75	40.3%	43.4%	41.8%	0.532
Fusion-Block	8.70	44.7%	54.1%	48.9%	0.598
Integrated-Block	9.50	51.7%	61.7%	52.6%	0.632
			DS_{random}		
	Correct	Precision	Recall	F-1 Score	ARI
BoM	2.55	30.8%	33.0%	31.8%	0.473
VIPS	1.97	27.8%	26.4%	27.1%	0.371
SegBlock	3.74	41.9%	44.8%	43.3%	0.531
Semantic-Block	3.82	43.3%	53.6%	48.0%	0.549
Fusion-Block	4.54	49.3%	61.2%	54.6%	0.610
Integrated-Block	5.55	56.6%	67.5%	61.5%	0.718
			DS_{new}		
	Correct	Precision	Recall	F-1 Score	ARI
BoM	5.54	30.5%	21.7%	25.3%	0.426
VIPS	6.38	20.7%	24.4%	22.4%	0.415
SegBlock	5.59	39.2%	38.9%	39.0%	0.464
Semantic-Block	5.89	40.6 %	41.7%	41.1%	0.483
Fusion-Block	6.02	45.8%	48.2%	47.0%	0.533
Integrated-Block	6.75	52.9%	58.7%	55.4%	0.619
			Total		
	Correct	Precision	Recall	F-1 Score	ARI
BoM	3.38	31.4%	27.9%	29.5%	0.450
VIPS	3.17	24.7%	25.8%	25.2%	0.405
SegBlock	5.01	39.6%	42.4%	40.9%	0.514
Semantic-Block	5.47	41.2 %	46.7%	43.8%	0.526
Fusion-Block	6.31	46.3%	54.9%	50.2%	0.583
Integrated-Block	7.03	53.4%	63.2%	57.8%	0.660

As shown in Table 4, for all the methods, DSrandom dataset has the maximum value of precision, recall, and F-1 score compared to the DSpopular, and the DSnew datasets. It shows that web pages in DSrandom tend to be less complicated (with less content) rather than DSpopular and DSnew.

Figure 13 presents the distributions of the precision of our proposed approach and the five comparing methods on the total datasets. It shows the outperformance of our method.

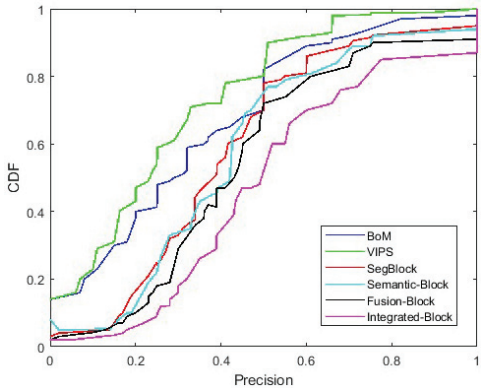


Figure 13 The performance results on the total dataset.

Table 5 ARI values of different threshold

Threshold	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
ARI (Total Dataset)	0	0.056	0.102	0.482	**0.660**	0.575	0.546	0.541	0.530	0.521	0.514

Our model segments web pages by combining the logic, visual and textual content of a web page using Gestalt laws of grouping to simulate human understandings. It merges web page content into blocks and compares the text-similarity of the blocks to regroup these similar blocks as integrated-blocks. Among the other comparing methods, Fusion-Block uses the text-similarity method (Doc2Vec) to compare the text-similarity of blocks and the other methods do not use the textual-similarity method to segment web pages. They only focus on the page structure and the visual features without considering the semantic text similarity metrics of blocks. As shown in Table 4, the minimum amount of precision, recall, F-1 score, and ARI belong to VIPS since we believe that it uses only the visual features of web pages, which makes it perform less accurately on the evaluation metrics. Table 4 represents that Integrated-Block, Fusion-Block, Semantic-Block, and SegBloak were achieved F-1 score values of more than 40% in the Total dataset, which are 57.8%, 50.2%, 43.8%, and 40.9%, respectively. Additionally, these four methods achieved ARI values greater than 0.5 in the Total dataset that shows their segmentation is not close to randomness. The ARI values of Integrated-Block, Fusion-Block, Semantic-Block, and SegBlock are 0.660, 0.583, 0.526, and 0.514, respectively.

Our model uses the semantic text similarity method (SBERT) to identify semantically related blocks and regroup them as integrated blocks. Table 5

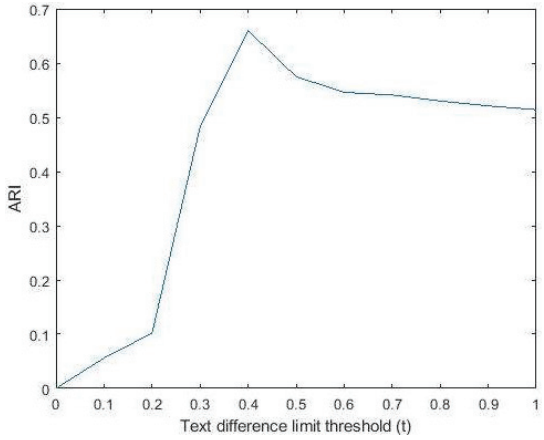


Figure 14 ARI distribution of different threshold.

shows different text difference limit thresholds (t in Algorithm 1) from 0 to 1. It represents that $t = 0.4$ results in the highest amount of ARI (comparing to the ground truth) in the Total dataset which is 0.660. The ARI distribution over the different text difference limit is shown in Figure 14. Thus, according to this result, we set $t = 0.4$.

Figures 15, 16, and 17 represent the result of web page segmentation using four different methods. The segmentation methods are applied on a part of the homepage of "www.koreanconsulate.on.ca", "www.fishdevon.co.uk", and "www.journalregister.com" from DSrandom, DSrandom, and DSpopular dataset, respectively. Subfigures (a) of these figures represent the manually labeled ground truth of these web pages. Subfigures (b) and (c) of these figures show the segmented blocks caused by the SegBlock and Semantic-Block, respectively. SegBlock and Semantic-Block methods do not segment blocks using semantic text-similarity of blocks; this limitation is indicated and can be found in [10].

The result of segmentation using the SegBlock and the Semantic-Block methods are identical as shown in Figures 16(b), 16(c), 17(b), and 17(c), while they are different in Figures 15(b), and 15(c). As represented in subfigures (b) and (c) of Figures 15–17, the SegBlock and Semantic-Block methods did not group the whole paragraph; we believe that this is because the different font styles were used in the paragraph. Subfigures (d) and (e) of these figures represent the result of web page segmentation using the Fusion-Block and Integrated-Block methods. As shown in subfigures (d) and (e) of Figures 15 and 16, the segmented blocks using the Fusion-Block and

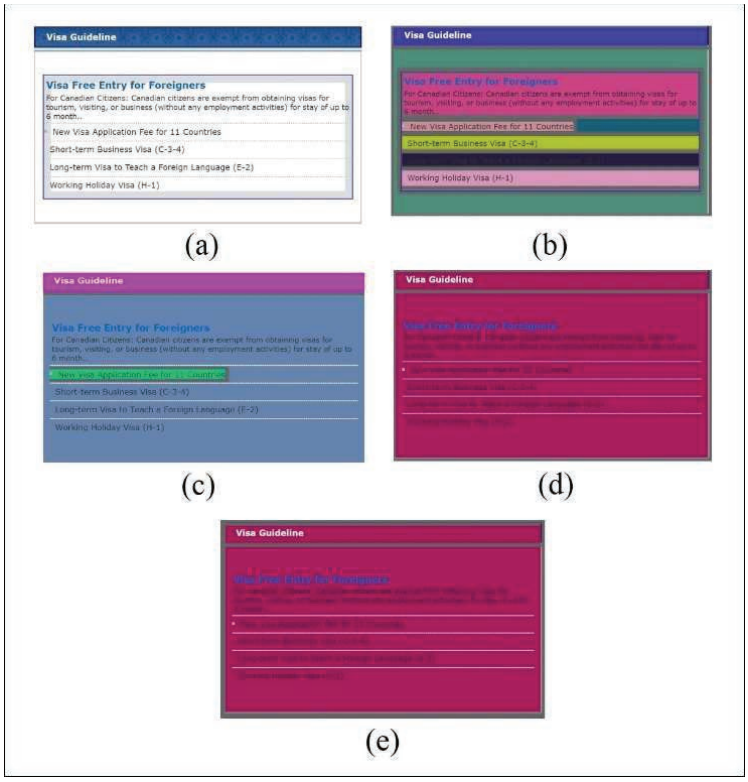


Figure 15 A part of the web page of "www.koreanconsulate.on.ca" from DSrandom, (a) manually labeled ground truth, (b) segmented blocks using segblock, (c) segmented blocks using semantic-block, (d) segmented blocks using fusion-block, (e) (d) segmented blocks using integrated-block.

Integrated-Block methods are identical. This shows that the result of the text-similarity using Doc2Vec (Fusion-Block) and SBERT (Integrated-Block) are identical; the paragraphs are related and grouped into a single block. The Fusion-Block method segments the paragraph shown in Figure 17(d) into four separated blocks while these blocks are semantically related and need to be grouped in a single block. However, as represented in Figure 17(e), our approach (Integrated-Block) groups these related blocks as a single block.

As shown in Figures 15(b) and 15(c), the Semantic-Block method segmented the blocks better than SegBlock, we believe that it is because Semantic-Block simulates human perception using the Gestalt laws of grouping. Also, as shown in subfigures (d) of Figures 15 and 16, the Fusion-Block

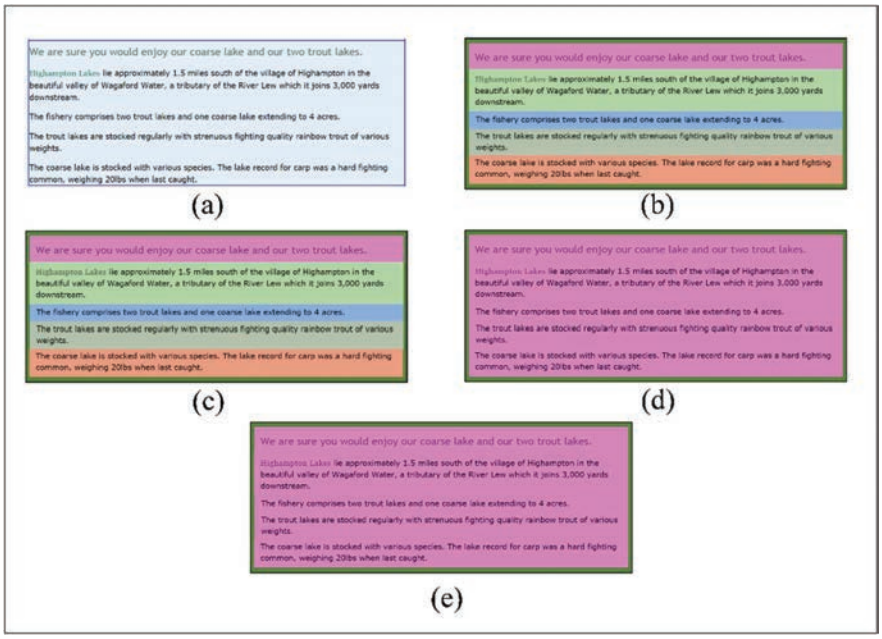


Figure 16 A part of the web page of "www.fishdevon.co.uk" from DSrandom, (a) manually labeled ground truth, (b) segmented blocks using segblock, (c) segmented blocks using semantic-block, (d) segmented blocks using fusion-block, (e) (d) segmented blocks using integrated-block.

method segments the blocks better than the SegBlock and Semantic-Block methods; it uses the text-similarity of blocks using the Doc2Vec technique and regroups similar blocks as a single block. Our approach used the semantic analysis method and grouped the whole paragraph as a single block regardless of the different font styles. Thus, our approach groups similar semantic text contents as an integrated block and overcomes the scattering or shortening of the long text of web page content mentioned in Sections 1 and 2. Our implementation has been released at https://github.com/Saeedeh-SGH/Integrated_Block.

6 Conclusions

6.1 Limitations and Future Work

According to the results of Section 5, our model outperforms the existing methods. However, there are also limitations in this segmentation technique

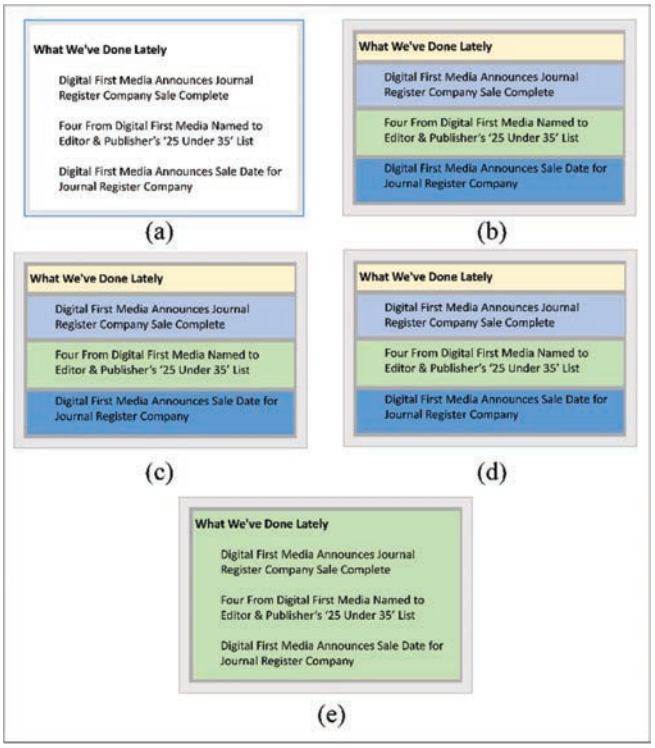


Figure 17 A part of the web page of "www.journalregister.com" from DSpopular, (a) manually labeled ground truth, (b) segmented blocks using segblock, (c) segmented blocks using semantic-block, (d) segmented blocks using fusion-block, (e) (d) segmented blocks using integrated-block.

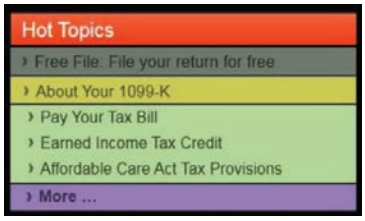


Figure 18 A part of the web page of "www.irs.gov" from DSpopular.

that we plan to address in future work. We used SBERT to compare the textual similarity of basic-blocks to regroup them as integrated blocks by simulating human perception using Gestalt laws of grouping. Figure 18 represents several segmented blocks using our approach on a part of the

web page of "www.irs.gov" from the DSpopular dataset. These blocks are considered as a single group in the ground truth which means that they have a similar concept. However, our proposed approach segments this part of the web page into five different blocks. It shows that the text representation model that we used (SBERT) does not group these blocks as a single block. Using fusion semantic analysis methods may yield better results when paragraphs have the same concept during the regrouping stage.

Reimers et al. [45] experimented with SBERT within two setups: (1) Only training on STSb (STS benchmark), and (2) first training on NLI, then training on STSb. They compare the results of the experiment of these two setups with BERT. The results show that the spearman values of BERT, first strategy (BERT-STSb-base), and the second strategy (Trained on NLI data + STS-b data) are 84.30, 84.67, and 88.33, respectively.

Sun et al. [62] presented ERNIE 2.0 and compare the results of their approach with BERT [12] on STS-b. As shown in the BASE model [62], BERT-base and ERNIE obtain the Spearman correlation scores of 85.8 and 86.5, respectively. However, Sun et al. [62] do not compare their approach with SBERT.

In this paper, we have not implemented ERNIE 2.0, but according to the papers (Reimers et al. [45] and Sun et al. [62]), SBERT and ERNIE 2.0 obtain close results in the Semantic Text Similarity (STS) task. In future work, we plan to employ models such as ERNIE 2.0 in our approach and evaluate the results.

The representativeness of the material used for evaluation questions the external validity of the study. The presented evaluation results on the non-representative datasets may not be generalized to web pages in general. We intend to test our model on additional datasets and utilize fusion semantic analysis methods to extend our model in future work.

6.2 Conclusions

We proposed an approach to improve web page segmentation methods and applied it to three datasets. Our approach semantically segments web pages into integrated blocks and has two main steps; in the first step, it merges web page content into basic-blocks by simulating human perception using Gestalt laws of grouping. In the second step, it utilizes semantic text similarity to identify similar blocks and regroup these similar basic-blocks as integrated blocks. To verify the accuracy of our approach, we (1) apply it to the three datasets, and (2) compare it to five existing state-of-the-art algorithms. The

results show that our approach outperforms all the five comparison methods in terms of precision, recall, F-1 score, and ARI.

Appendix-A

One of the main challenges in computing the semantic similarity of documents is the shortage of training data [56, 57]. To overcome this limitation, different approaches are using a large amount of unannotated text (such as Wikipedia) for training general-purpose language representation models known as "pre-training" methods [12]. The pre-trained model can then be fine-tuned on a variety of specific NLP tasks. Pre-trained representation techniques are divided into two categories: context-free and contextual representation models.

The main step of non-contextual text representation is to map discrete language symbols into a distributed embedding space [56]. Each word of the document is mapped into a vector. Vector representations of text can be constructed in many ways. For example, Mikolov et al. [38, 39] propose Word2Vec, an effective tool for learning word representations from a corpus, which implements two models: Continuous Bag-Of-Words (CBOW) and Skip-gram [58]. The CBOW model scans the text with a context window and learns to predict the target word [44]. The Skip-gram model predicts the words in the context of the target word [11]. Word2Vec uses local neighboring words as context [44]. Pennington et al. [40] propose Global Vectors for word representation (GloVe) that directly captures global corpus statistics. Comparing to GloVe, the word embeddings trained from Word2Vec can better capture the semantics of words and exploit the relatedness of words. Le et al. [41] propose Doc2Vec which is an extension of Word2Vec that can learn representations for documents. FastText is an extension of Word2Vec proposed by Facebook [59]. Instead of using individual words, FastText breaks words into several n-grams (sub-words) [37]. These approaches have two major limitations: (1) However, the ordering of words in a text is meaningful, these representation models are insensitive to word order and only capture the relations between words [44], and (2) they only obtain a single global representation for each word and ignore their context [13]. Thus, they fail to capture higher-level concepts in context. To address these issues, contextual representation is proposed.

Contextual representation models assign each word a representation based on its context [60]. These models are divided into two

categories of unidirectional and bidirectional representations. Unidirectional representation of a word is generated based on the left or right surrounding words in a document [12]. For example, the unidirectional representation of the word "goals" in the sentence "I have three goals in my life", is based on "I have three" or "in my life", not both of them. Bidirectional representation of a word considers the left and right surrounding words in a text corpus [12]. For example, a bidirectional representation of the word "goals" in the sentence is generated based on "I have three" and "in my life". The bidirectional representation model has a deeper sense of context than unidirectional models since it considers the context of a word based on all of its surroundings [44].

Different from non-contextual word representations, contextual representations move beyond word-level semantics where each token is associated with a representation that is a function of the entire input sequence [13]. These representations can capture many syntactic and semantic properties of words under diverse linguistic contexts [44]. Some studies have shown that contextual embeddings, pre-trained on a large-scale unlabelled corpus, can achieve high performance on a wide range of NLP tasks and can avoid training a new model from scratch [13]. Some other studies express those contextual embeddings can learn useful and transferable representations across languages [13]. The contextual representations are better suited to capture the semantics of text [44]. They have some model architectures such as transformer [61]; a neural network architecture based exclusively on attention mechanisms [44]. The neural attention mechanism aims to capture long-range dependencies and is inspired by how humans read and understand longer texts. The neural representation learning can be treated as a pre-training step or language modeling step for NLP downstream tasks [44]. Many methods focus on learning contextual word embeddings such as ELMo [42], GPT [43], and BERT [12]. ELMo and GPT do not consider the left and right surrounding words in a text corpus [13]. Google proposes an improved method, BERT (Bidirectional Encoder Representations from Transformers) [12], which can effectively exploit the deep semantic information of a sentence. Deep bidirectional means that it is conditioned on every word in the left and right contexts at the same time [44]. It works by masking some percentage of the input tokens at random and then predicting those masked tokens. Several studies reported that contextualized embeddings such as BERT better encode semantic information of a text [14]. Also, according to [12], BERT obtains state-of-the-art results in numerous benchmarks. Figure 19 shows the explained representation models in a timeline since 2013.

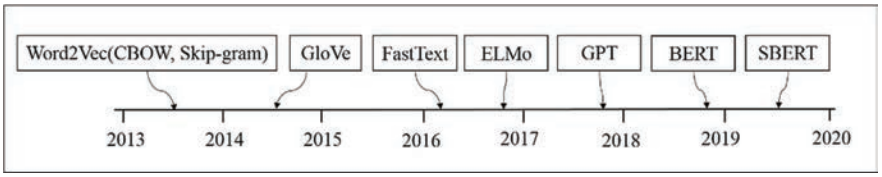


Figure 19 A timeline of the recent text representation models since 2013.

References

[1] P. Malhotra and S. K. Malik, "Web Page Segmentation Towards Information Extraction for Web Semantics," in *International Conference on Innovative Computing and Communications*, 2019, pp. 431–442: Springer.

[2] J. Zeleny, R. Burget, and J. Zendulka, "Box clustering segmentation: A new method for vision-based web page preprocessing," *Information Processing & Management*, vol. 53, no. 3, pp. 735–750, 2017.

[3] Z. Bu, C. Zhang, Z. Xia, and J. Wang, "An FAR-SW based approach for webpage information extraction," *Information Systems Frontiers*, vol. 16, no. 5, pp. 771–785, 2014.

[4] H. F. Eldirdiery and A. Ahmed, "Web document segmentation for better extraction of information: a review," *International Journal of Computer Applications*, vol. 110, no. 3, 2015.

[5] C. Kohlschütter, P. Fankhauser, and W. Nejdl, "Boilerplate detection using shallow text features," in *Proceedings of the third ACM international conference on Web search and data mining*, 2010, pp. 441–450.

[6] K. Koffka, *Principles of Gestalt psychology*. Routledge, 2013.

[7] S. E. Palmer, "Modern theories of Gestalt perception," 1992.

[8] R. J. Sternberg and K. Sternberg, *Cognitive psychology*. Nelson Education, 2016.

[9] G. Wen, X. Pan, L. Jiang, and J. Wen, "Modeling Gestalt laws for classification," in *9th IEEE International Conference on Cognitive Informatics (ICCI'10)*, 2010, pp. 914–918: IEEE.

[10] Z. Jiang, H. Yin, Y. Wu, Y. Lyu, G. Min, and X. Zhang, "Constructing Novel Block Layouts for Webpage Analysis," *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 3, pp. 1–18, 2019.

[11] S. Wang, W. Zhou, and C. Jiang, "A survey of word embeddings based on deep learning," *Computing*, vol. 102, no. 3, pp. 717–740, 2020.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[13] Q. Liu, M. J. Kusner, and P. Blunsom, "A survey on contextual embeddings," *arXiv preprint arXiv:2003.07278*, 2020.

[14] A. Rogers, O. Kovaleva, and A. Rumshisky, "A primer in bertology: What we know about how bert works," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 842–866, 2020.

[15] S. Gupta, G. Kaiser, D. Neistadt, and P. Grimm, "DOM-based content extraction of HTML documents," in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 207–214.

[16] Y. Chen, W.-Y. Ma, and H.-J. Zhang, "Detecting web page structure for adaptive viewing on small form factor devices," in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 225–233.

[17] Q. Fan, C. Yan, and L. Huang, "Discovering Informative Contents of Web Pages," in *International Conference on Web-Age Information Management*, 2014, pp. 180–191: Springer.

[18] J. Kong et al., "Web interface interpretation using graph grammars," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 590–602, 2011.

[19] A. Sanoja and S. Gançarski, "Block-o-matic: A web page segmentation framework," in *2014 international conference on multimedia computing and systems (ICMCS)*, 2014, pp. 595–600: IEEE.

[20] T. Manabe and K. Tajima, "Extracting logical hierarchical structure of HTML documents based on headings," *Proceedings of the VLDB Endowment*, vol. 8, no. 12, pp. 1606–1617, 2015.

[21] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, "Extracting Content Structure for Web Pages Based on Visual Representation," Berlin, Heidelberg, 2003, pp. 406–417: Springer Berlin Heidelberg.

[22] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma, "Vips: a vision-based page segmentation algorithm," 2003.

[23] M. Cormer, R. Mann, K. Moffatt, and R. Cohen, "Towards an improved vision-based web page segmentation algorithm," in *2017 14th Conference on Computer and Robot Vision (CRV)*, 2017, pp. 345–352: IEEE.

[24] R. R. Mehta, P. Mitra, and H. Karnick, "Extracting semantic structure of web documents using content and visual information," in *Special interest tracks and posters of the 14th international conference on World Wide Web*, 2005, pp. 928–929.

[25] W. Liu, X. Meng, and W. Meng, "Vide: A vision-based approach for deep web data extraction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 3, pp. 447–460, 2009.

[26] R. Kumar, J. O. Talton, S. Ahmad, and S. R. Klemmer, "Bricolage: A Structured-Prediction Algorithm for Example-Based Web Design," *Proc. CHI 2011*, 2011.

[27] Z. Xu and J. Miller, "Identifying semantic blocks in Web pages using Gestalt laws of grouping," *World Wide Web*, vol. 19, no. 5, pp. 957–978, 2016.

[28] Z. Xu and J. Miller, "Cross-browser differences detection based on an empirical metric for web page visual similarity," *ACM Transactions on Internet Technology (TOIT)*, vol. 18, no. 3, pp. 1–23, 2018.

[29] Z. Xu and J. Miller, "A new webpage classification model based on visual information using gestalt laws of grouping," in *International Conference on Web Information Systems Engineering*, 2015, pp. 225–232: Springer.

[30] Z. Xu and J. Miller, "Estimating similarity of rich internet pages using visual information," *International Journal of Web Engineering and Technology*, vol. 12, no. 2, pp. 97–119, 2017.

[31] C. Kohlschütter and W. Nejdl, "A densitometric approach to web page segmentation," in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 1173–1182.

[32] S. S. Sajjadi-Ghaemmaghami and J. Miller, "A New Semantic Approach to Improve Web page Segmentation", *Journal of Web Engineering*, vol. 20, no. 4, pp. 963992, June 2021, DOI: 10.13052/jwe1540-9589. 2042.

[33] J. Blustein, N. R. D. Matteo, and D. Macrini, "Designing Experiments to Compare Web Page Segmenters," presented at the Proceedings of the 2nd International Workshop on Human Factors in Hypertext, Hof, Germany, 2019. Available: https://doi-org.login.ezproxy.library.ualberta.ca/10.1145/3345509.3349280

[34] M. Kovacevic, M. Diligenti, M. Gori, and V. Milutinovic, "Recognition of Common Areas in a Web Page Using Visual Information: a possible application in a page classification," in *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, 2002, pp. 250–257: IEEE.

[35] W. H. Gomaa and A. Fahmy, "A Survey of Text Similarity Approaches," *International Journal of Computer Applications*, vol. 68, pp. 13–18, 2013.

[36] G. Liu, C. Guo, L. Xie, W. Liu, N. Xiong, and G. Chen, "An intelligent CNN-VAE text representation technology based on text semantics for comprehensive big data," *arXiv preprint arXiv:2008.12522*, 2020.

[37] J. Yan, "Text Representation," in *Encyclopedia of Database Systems*, L. Liu and M. T. ÖZsu, Eds. Boston, MA: Springer US, 2009, pp. 3069–3072.

[38] T. Mikolov, Q. V. Le, and I. Sutskever, "Exploiting similarities among languages for machine translation," *arXiv preprint arXiv:1309.4168*, 2013.

[39] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[40] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[41] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, 2014, pp. 1188–1196.

[42] M. E. Peters et al., "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[43] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[44] K. Babić, S. Martinčić-Ipšić, and A. Meštrović, "Survey of Neural Text Representation Models," *Information*, vol. 11, no. 11, p. 511, 2020.

[45] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[46] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *arXiv preprint arXiv:1508.05326*, 2015.

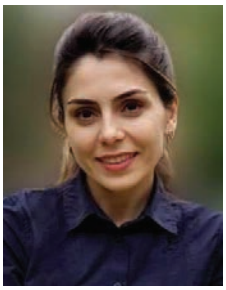
[47] A. Williams, N. Nangia, and S. R. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," *arXiv preprint arXiv:1704.05426*, 2017.

[48] "dataset-popular 2014. A dataset of popular pages (taken from dir.yahoo.com) with manually marked up semantic blocks. Retrieved from https://github.com/rkrzr/dataset-popular.," ed.

[49] "dataset-random 2014. A dataset of random pages with manually marked up semantic blocks. Retrieved from https://github.com/rkrzr /dataset-random.," ed.

[50] "Alexa. 2016. The top 500 sites on the web. Retrieved from http://www.alexa.com/topsites.," ed.
[51] VIPS-JAVA [n.d.]. Implementation of Vision Based Page Segmentation Algorithm in Java. Retrieved from https://github.com/tpopela/vips-java.
[52] A. S. Bozkir and E. A. Sezer, "Layout-based computation of web page similarity ranks," *International Journal of Human-Computer Studies*, vol. 110, pp. 95–114, 2018.
[53] D. Chakrabarti, R. Kumar, and K. Punera, "A graph-theoretic approach to webpage segmentation," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 377–386.
[54] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, no. 1, pp. 193–218, 1985.
[55] K. Y. Yeung and W. L. Ruzzo, "Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data," *Bioinformatics*, vol. 17, no. 9, pp. 763–774, 2001.
[56] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, pp. 1–26, 2020.
[57] U. Naseem, I. Razzak, S. K. Khan, and M. Prasad, "A Comprehensive Survey on Word Representation Models: From Classical to State-Of-The-Art Word Representation Language Models," *arXiv preprint arXiv:2010.15036*, 2020.
[58] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *arXiv preprint arXiv:1310.4546*, 2013.
[59] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*, 2016.
[60] A. Miaschi and F. Dell'Orletta, "Contextual and Non-Contextual Word Embeddings: an in-depth Linguistic Investigation," in *Proceedings of the 5th Workshop on Representation Learning for NLP*, 2020, pp. 110–119.
[61] A. Vaswani et al., "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017, Available online: https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf (accessed on 29 October 2020).
[62] Sun, Yu, et al. "Ernie 2.0: A continual pre-training framework for language understanding," Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, No. 05, 2020.

Biographies



Saeedeh Sadat Sajjadi Ghaemmaghami received the Ph.D. degree in computer engineering from the University of Alberta in 2021. Her research interests include web page analysis, machine learning, natural language processing, pattern recognition, and data mining.

James Miller, P.Eng (Alberta) has been a full professor with the Dept. Electrical and Computer Engineering at The University of Alberta since 2000. Previously, he was a professor at the University of Strathclyde (U.K.) and a principal research scientist at the National Electronics Research Initiative (U.K.). He has been an active researcher for more than thirty years across a wide range of topics, ranging from Computer Vision, Pattern Recognition, Embedded System Design, Software Engineering, Web Engineering and Proactive Analytics. He has published more than 100 articles in peer-reviewed journals including many IEEE and ACM venues.