VFX: A VISION-BASED APPROACH TO FORUM DATA EXTRACTION

Chen Hui Ng
Tunku Abdul Rahman University-College

Choon Jin Ng
Tunku Abdul Rahman University-College

Tong Ming Lim
Tunku Abdul Rahman University-College

ABSTRACT

Rapid development of the Internet has dramatically increased information available on the World Wide Web. Amongst these vast sources of information, discussion forums may be useful for businesses and organizations to get a glimpse of customer opinions or to extract product information. Little existing work reported in the literature has systemically investigated the problem of extracting user posts from forum sites. Extracting forum posts accurately raises a few challenges. First, forum comes in a variety of templates and this makes it hard to formalize general rules to extract forum posts. Second, each post record might appear relatively different from each other. This introduces inconsistency in the Document Object Model (DOM) for comparisons. Third, each post in the forum can consist of complicated subtrees rather than a single node in the DOM tree. To tackle these challenges, a vision-based approach was introduced to automatically extract posts from a web forum page based on its visual cues. In this paper, we propose a visual-based forum extraction (VFX) algorithm that can extract user posts in any types of forum without the need to inspect its template structure in advance.

KEYWORDS

Forum data extraction, Forum visual cues, Forum layout structure, Vision-based extraction

1. INTRODUCTION

The increasing accessibility of the Web has increased the proliferations of web based forums. This has enabled users to freely voice their views and promote discussions on a variety of topics ranging from politics, entertainment, to products and services reviews. As such, these silos of knowledge can conveniently become another important source of information. These extracted information can then be processed and used for monitoring and analysis. For instance, a company can monitor customer feedback on its products and services.

In this paper, we focus on the problem of automatically extracting posts in any forum site without the need to inspect its template structure in advance. In general, a web page contains not only useful data, but also other irrelevant data such as header, footer, navigational panels, advertisements, comments, and so on. These irrelevant data are known as noises. Reducing noises will reduce preprocessing complexities during the forum data analytics cycle. The goal of forum data extraction is to remove these noises from a web page and extract only the required data records from these pages. There are two main tasks in forum data extraction: (a) post record identification, and (b) post content extraction. Post record identification is to find out where the posts are located in a forum web page using visual cues. Once the position of post records is found, the next step is to extract individual post in the post records.

Obviously, it is impractical to manually inspect all forums' structures in order to extract their posts. Therefore, the goal of this project is to propose a general algorithm that can automatically extract user posts from any given forum page.

2. LITERATURE REVIEW

Liu et al. proposed an algorithm named Mining Data Records (MDR) (Bing, Grossman, & Zhai, 2004). MDR identifies data regions by searching for multiple generalized nodes using edit-distance similarity where generalized nodes are a fix combination of multiple child nodes and their corresponding subtrees. MDR does not identify the most relevant data records but rather reports each repetitive subregion contained in a Web page. Later, the author proposed an improvement of their system named DEPTA operating on a tag tree built according to visual rendering information (Liu & Zhai, 2005). Additionally, they mentioned that gap information is incorporated to eliminate false node combinations but nothing is said about the gap information. Finally, they proposed an approach for data record alignment by progressively growing a seed tag tree. The alignment is partial because only these nodes of a data record become aligned whose position for inserting into the seed tree can be uniquely determined. They tested their MDR system on a collection of handpicked sample pages with a precision of 100% and a recall of 99.8%.

Cai et al. proposed a vision-based page segmentation algorithm (VIPS) (Liu, Meng, & Meng, 2006). VIPS is proposed to extract the semantic structure for web page where semantic structure is a hierarchical structure where each node corresponds to a block. VIPS algorithm makes full use of a page layout feature. It first extracts all suitable blocks from the HTML DOM tree, then it tries to find the separators between these extracted blocks. Separators denote the horizontal or vertical lines in a web page that visually crosses those extracted blocks. These separators are also assigned weightage based on its visual cues. Finally, the separators with the lowest weight and the blocks beside these separators are merged to form new blocks. This merging process iterates until separators with maximum weights are met. VIPS algorithm takes advantage of visual cues to obtain the vision-based content structure of a web page and thus successfully bridges the gap between the DOM structure and the semantic structure. The page is partitioned based on visual separators and structured as a hierarchy. This semantic hierarchy is consistent with human perception to some extent.

Akpinar et al. then extended the VIPS algorithm (Akpinar, Elgin, & Yesilada, 2012). The original VIPS algorithm has a limited coverage on HTML tags. Any tags not covered are then referred to a single generic rule. This becomes an issue as most websites are now written in HTML5 tags. To tackle such problems, Akpinar et al. modified the heuristic rules of the algorithm by adding more visual features such as margin, float, and images. Besides, some rules required a threshold to be defined. It is also not possible to find a cue on how to calculate this threshold.

Song et al. proposed a VIPS-based algorithm to rank block importance for web pages through machine learning algorithms using spatial features (Song, Liu, Wen, & Ma, 2004). They extracted query-independent rankings for the fragments for the purpose of improving the performance of web search and also to facilitate web mining and accessibility. They partitioned a web page into semantic blocks with a hierarchical structure, constructed the spatial and content features, and used neural network and Support Vector Machine (SVM) to build the block importance model for ranking the importance of each block in the web page.

Simon et al. presented a fully automatic information extraction tools called ViPER (Simon & Lausen, 2005). ViPER provides a better sub-tree comparing method that allows consecutive data records with various lengths. The technique is based on the assumption that a web page contains at least two consecutive data records which exhibits some kind of structural and visible similarities. For example, a search engine's result page. ViPER is able to extract relevant data with respect to user's visual perception of the web page. It also introduces a multiple sequence alignment algorithm (MSA) that aligns maximal unique matches at different levels to extract the template of data records. With respect to the 676 correctly extracted data records by ViPER, only 11 data items could not be aligned correctly.

Shuyi et al. proposed a novel template independent news extraction approach to easily identify news articles based on visual consistency (Zheng, Song, & Wen, 2007). They first represent a page as a visual block tree. Then, by extracting a series of visual features, they can derive a composite visual feature set that is stable in the news domain. The visual features used included position, size, rich format features (font size, bold, and italic), and statistical features (number of images and hyperlinks, text length, number of paragraphs, number of paragraph with italic, number of table, number of paragraph with bold). Finally, they used a machine learning approach to generate a template-independent wrapper. Experimental results indicated that their approach is effective in extracting news across websites, even from unseen websites. The F1 performance was around 95%.

Zhao et al. presented the ViNTs algorithm for automatically producing wrappers that can be used to extract search result records from dynamically generated result pages returned by search engines (Zhao, Meng, Wu,

Raghavan, & Yu, 2005). ViNTs automatically generates SRR (Search Result Record) extraction rules using visual context features and tag structure information. Therefore, it first utilizes the visual content without considering the tag structure to identify content regularities denoted as content lines and then combines them with the HTML tag structure regularities to generate wrappers. To weight the relevance of different extraction rules, visual and non-visual features have been used. ViNTs builds a wrapper for a search engine using several result pages and one no-result page. The resulting wrapper is represented by a regular expressions of alternative horizontal separators tags such as <HR> or

, which segment descendants into SRRs. Due to the fact that the ViNTs system only supplies horizontal separators, which is sufficient when considering document result pages, it could not separate horizontally arranged data records which will require vertical separators. Additionally, at least four data records must be present in a web page for building the wrapper. In case the data records are distributed over multiple sections, only the major section is reported.

3. SYSTEM OVERVIEW

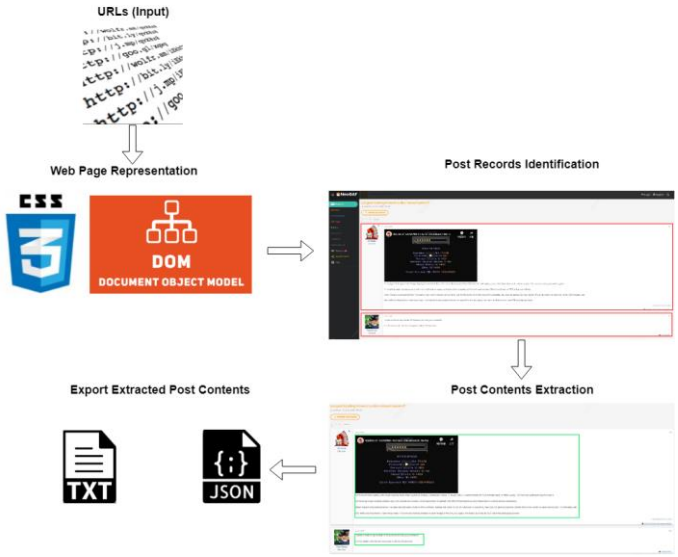


Figure 1. Overview of the stages in the Visual-based Forum Extraction (VFX) algorithm.

Figure 1 shows the stages of the Visual-based Forum Extraction (VFX) algorithm. The first stage is to prepare DOM node information, render it in memory, and acquire its CSS properties. The second stage is to identify which block in a forum web page holds the post records. The third stage is to extract the post contents from the identified records. Finally, the last stage is to export those contents into a readable format such as the JavaScript Object Notation (JSON) or a comma separated value (CSV) file for further analysis.

3.1 Web page representation

The input to the VFX is the forum page's URL link. VFX will fetch the page and parse its HTML, simulate any JavaScript events, and render them into a DOM tree representation.

As HTML tags do not accurately reflect a semantic representation of a web page, visual information needs to be acquired from the post-rendered Cascading Style Sheets (CSS) in order to improve the extraction performance. Visual information in web page has proven to be a very useful feature for web data search and extraction, and several vision-based approaches have been proposed, such as those presented in Liu et al. (2006) and Akpinar et al. (2012). During the parsing process, useful visual information is obtained and attached to the nodes of the DOM tree. The visual information used in this paper is listed as the following:

- *Position*: The coordinate of the left-top corner of a node.
- *Size*: The width and height of the rectangle that a node occupies in the web page.

- *Font*: The text properties including its font size, font style, font colour and font weight.
- *Margin*: The margin applied to the rectangle area occupied by a node.
- *Colour*: The background colour of the node's rectangular area.

3.2 Post records identification

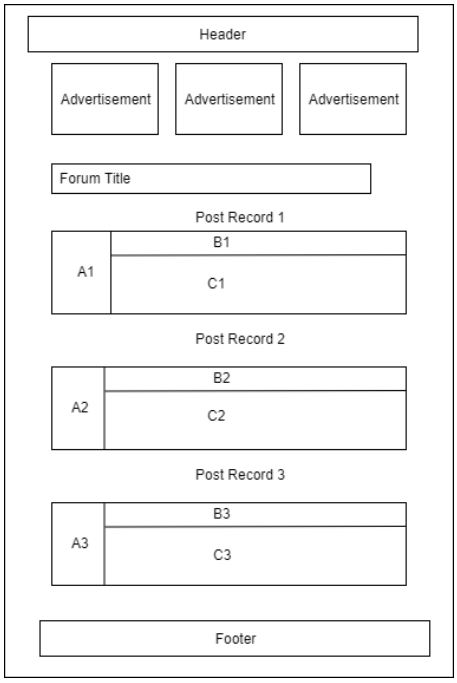


Figure 2. A typical web forum layout with the post title and the user posts.

To ease reader's convenience, forum pages are often designed to have regular layout of post records. Based on the observations of a large number of forum pages, a group of useful features and visual information is generalized to create a set of heuristic rules for post records identification. To help understand the features, Figure 2 shows a general layout of the post contents in a typical forum page. Based on Figure 2, the useful features are identified as below:

- *Feature 1*: Post records are arranged vertically with the same width and flushed left.
- *Feature 2*: The templates of the post records in one page are identical.
- *Feature 3*: The items with the same semantics in different post records are similar in appearance, including position and font, but the user-generated contents (post contents) can be different.
- *Feature 4*: Title is always located at top of the page and is bold.
- *Feature 5*: The title font size is larger than other text.

The components to be extracted are the forum thread title and the user generated posts, known as the post records. The first post record (post record 1) shown in Figure 2 is formed by blocks A1, B1 and C1. Amongst these blocks, C1 is the post content containing the user's post content.

On the other hand, blocks that are outside of the forum title and the post records are known as noises. In Figure 2, the noises are the header, advertisement and footer.

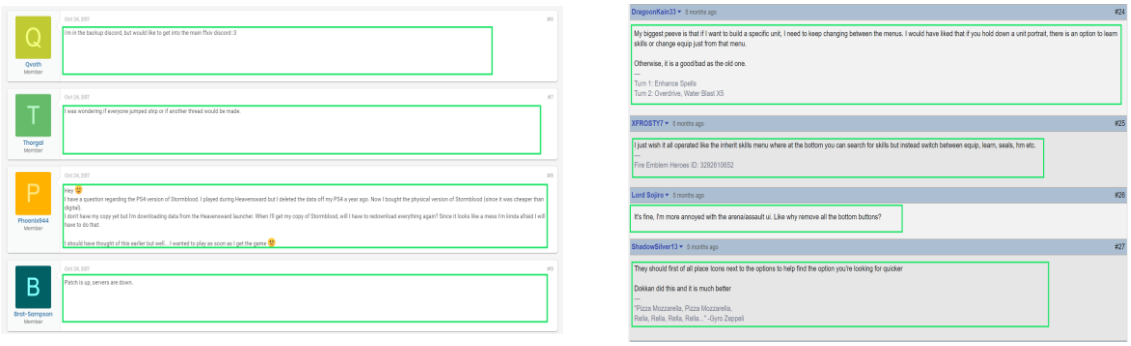
To be able to extract the title and user generated posts accurately, we need to identify which blocks are important. Every node in the DOM tree is checked to determine whether it forms a single block or not. If not, its children will be processed in the same way. Score will be assigned to each extracted block based on the block's visual cues. These scores are known as the TitleValue and the ContentValue. TitleValue is the score assigned to the block based on the heuristic rules in Table 1. The blocks with the highest TitleValue will be identified as the forum thread title. ContentValue is the score assigned based on heuristic rules to identify whether it is a post record. Blocks with equal ContentValue will be identified as the post records.

Then, the block with the highest TitleValue or equal ContentValue will be appended to a list for later extraction. A set of heuristic rules is produced to justify whether the node can form a single block and to assign the TitleValue and ContentValue. Tag cues and visual information acquired in the web page representation stage are utilized to produce the heuristic rules. Tags such as <H1>, <H2>, <B> and <STRONG> are often used to highlight a reader's attention such as the forum thread title. Thus, when a block has such tags, especially <H1> tag, higher TitleValue will be assigned to that particular block. The heuristic rules are listed in Table 1. The rules precedence are processed in the order of the rule set.

Table 1. Heuristic rules for post records identification

No.	Rules
1.	If the DOM node is a valid node (visible element) and has the following HTML tags: • If it has <h1> HTML tag, add 5 points to TitleValue • If it has <h2> HTML tag, add 3 points to TitleValue • If it has <b> or HTML tag, add 2 points to TitleValue
2.	If the top and bottom margin of the DOM node are nonzero: Add 2 points to TitleValue
3	If the font size of the DOM node is greater than the average font size: Add 3 points to TitleValue
4.	If the font weight of the DOM node is bold or the value is greater than 550: Add 3 points to TitleValue
5.	If the DOM node has a text node or a virtual text node as its child nodes, and their font colors are different: Add 2 points to TitleValue
6.	If the background color of the DOM node is different from its child nodes: Add 2 points to TitleValue
7.	If the text-align style of the DOM node is center: Add 1 points to TitleValue
8.	If the height of the DOM node occupied at least 70% of the page height and it is a visible element: Add 5 points to ContentValue
9.	If the DOM node's width is the same with all its sibling nodes: Add 5 points to ContentValue
10.	If the DOM node class name is same with all its sibling nodes' class name: Add 5 points to ContentValue

3.3 Post contents extraction



(a) NeoGAF forum sample. (b) GameFAQ forum sample.

Figure 3. Sample post record from different source of forums

Based on the observation on samples of forum pages, we found out that region containing post contents usually have greater width or greater height if compared to its sibling nodes. The blocks highlighted in Figure 3 show the regions containing the post contents. In Figure 3(a), the post contents have greater width when compared to their siblings (regions with the user profile details). In Figure 3(b), the post contents have greater height when compared to their siblings (regions with the user name and post number).

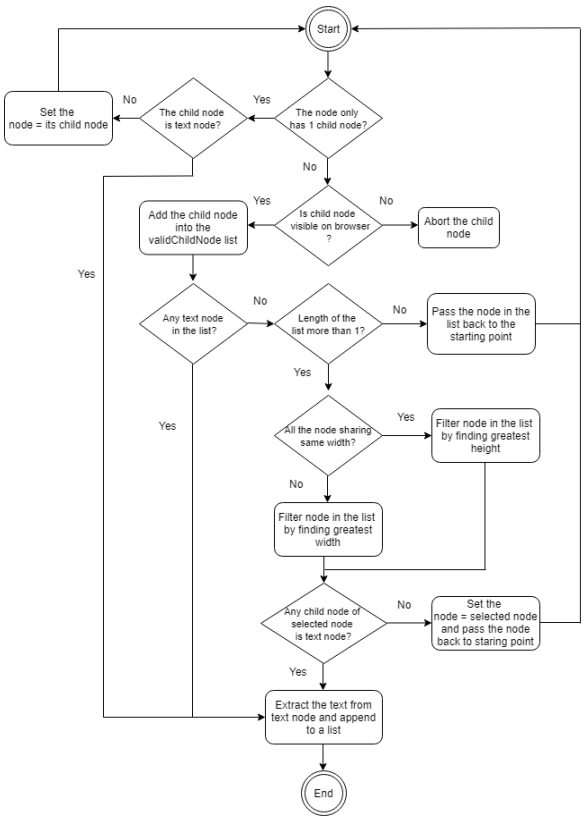
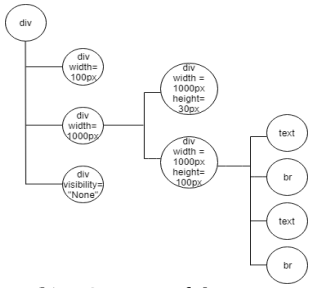


Figure 4. The process flow of the Visual-Based Forum Extraction (VFX) algorithm.

Hence, we conclude that a DOM node with greater width or greater height in a post records is the node that holds the post contents. Then, an algorithm is created based on these observations. Figure 4 illustrates the process of the VFX algorithm. Post records will be passed to this algorithm in order to extract post contents inside it. The post record is divided from the root of its DOM tree. This process is recursively repeated until all the post contents are extracted or all the DOM nodes in the post records are processed. However, not every forum page employs such layout. Some of the forum designed its post contents with lesser width and lesser height when compared to its sibling nodes. Thus, this algorithm only work on those forum pages where their layouts are consistent with the assumptions made in this paper.

In record extraction, all visual blocks with the same ContentValue are marked as the candidate blocks. These candidate blocks can potentially hold the forum posts and will be extracted. The visual blocks are always processed recursively starting from the root of the DOM tree. In this stage, the nodes in the post records with the highest width or highest height are marked as the post contents. However, the width measurement takes a higher priority over the height measurement. This process is illustrated in the example shown in Figure 5.

(a) Sample post record from the NeoGAF forum.



(b) DOM tree of the Figure 5(a)

Figure 5. An illustration of the rendered DOMs in the NeoGAF forum

Figure 5(a) shows a sample post record from the NeoGAF forum while Figure 5(b) shows the DOM tree of that post record. In this example, the parent node is the first DIV node processed by the VFX algorithm. From there, there are three DIV child nodes. These nodes' visibility on the web browser is determined by inspecting some of their CSS properties such as the visibility property, size and coordinate. Out of these three child nodes, only two have been determined to be visible on the web browser while the third child node is not visible since its visibility property is "None". The two visible child nodes are then appended into a validChildNode list.

Among these two nodes in the validChildNode list, the node with the greatest width is selected. In Figure 5(b), the DIV node with the width of 1000px is selected. Its child nodes will be inspected to determine if there is a text node. If no text node is found, then the node will be passed to the start of the algorithm again. In this case, the selected node only has two DIV node and so it is passed to the start of the algorithm.

With this selected node (DIV node with 1000px width), as all its child nodes are not a text node, the visibility of the children are validated. Only visible child nodes are appended to the validChildNode list. In this case, all the children are visible so they are appended to the list. Then, two nodes in the list are compared in order to select the node with greater width. As both nodes in the validChildNode list share the same width, 1000px, the nodes are selected based on their height instead. Hence the DIV node with 100px height is selected.

For this selected node, two of its child nodes are text nodes, hence the texts are extracted from the text nodes and the process ends here. The extracted text is appended into a global list.

After all the visual blocks in the list are processed by VFX, a list that holds all the comments are generated. As extracted text may contain some special characters such as '\n', '\t' and '\a', these characters are cleaned from the final result before exporting.

4. RESULTS

In order to test the robustness of this algorithm, we have randomly chosen 50 forum pages for the implementation test. Throughout this test, the algorithm went through a few iterations of fixes to improve its accuracy.

After a few rounds of test-and-debug, 46 out of 50 forum web pages were successfully identified and extracted. However, to ensure this algorithm is template-independent we carried out a validation test with another 50 forum pages. The result of the implementation test and data validation test are shown in Table 2.

Table 2. Implementation test and validation test results

Implementation test	Validation test
No. of forum tested:	**No. of forum tested:**
50 forum web pages	50 forum web pages
No. of successful extraction:	**No. of successful extraction:**
46/50 (Approximate 92%)	34/50 (Approximate 68%)
1 forum failed in identification stage	8 forums failed in identification stage
3 forums failed in extraction stage	8 forums failed in extraction stage

(a) Number of success and failed cases in implementation test and validation test

Testing	Precision	Recall	F-measure
Implementation Test	92%	90%	91%
Validation Test	68%	89%	77%

(b) Overall performance between implementation test and data validation test.

After going through data validation, the results were lower when compared to the implementation test. This situation occurs because improvements to the algorithm were made based on the failed cases in the implementation test. Hence, this might cause the algorithm to overfit the fifty forums in implementation test. By referring to Table 2, there is one failed case in implementation test and eight failed cases in validation test for the post record identification. These failures occurred because they did not comply with Rule 9 of Table 1.

Our algorithm will only rate the nodes as potential post records if all the nodes under same parent are sharing the exact same width. Hence, our system will not classify the nodes as potential post records even their width differences with their siblings are drifted in a small decimal amount. Thus, this is the main reason that our algorithm failed to identify post records on some of the forum websites.

There are some limitations in the heuristic rules used in the post records identification stage. Rule 9 states that the node and all its siblings must share the same width. So, when there is the case that the node has different width with its siblings, the heuristic rules would fail to identify the content records. Besides that, as Rule 9 and Rule 10 need to compare sibling nodes in order to identify the content records, a forum page must have at least two post records. Hence, the algorithm would have failed to identify the post record if it has only one post record. Rule 10 states that if the node shares the same class name with its sibling nodes, then points are awarded to the node. However, not every post record can have the same class name. For some websites, the post records do not even have a class name. In order to solve these issues, more visual features should be added into the heuristic rules in order to identify the post records accurately.

5. CONCLUSION

In this paper, we introduced the VFX algorithm to automatically extract forum data without needing a user to explicitly identify the area of interest. This paper tackled the challenges posed by user-generated post contents where they can be highly variable. The solution mainly consists of two steps. First, to identify the post records in a forum page by comparing the similarities of post records. Then second, to find the post contents from the extracted post records. The experimental results on 100 forum pages indicated that there are still room to improve the accuracy of the algorithm.

REFERENCES

Akpinar, E., & Yesilada, Y. (2012). Vision based page segmentation: Extended and improved algorithm.

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

Liu, W., Meng, X., & Meng, W. (2006, June). Vision-based web data records extraction. In *Proc. 9th international workshop on the web and databases* (pp. 20-25).

Liu, W., Yan, H., & Xiao, J. (2011). Automatically extracting user reviews from forum sites. *Computers & Mathematics with Applications*, *62*(7), 2779-2792.

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). ACM.

Song, R., Liu, H., Wen, J. R., & Ma, W. Y. (2004, May). Learning block importance models for web pages. In *Proceedings of the 13th international conference on World Wide Web* (pp. 203-211). ACM.

Zeleny, J., Burget, R., & Zendulka, J. (2017). Box clustering segmentation: A new method for vision-based web page preprocessing. *Information Processing & Management*, *53*(3), 735-750.

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). ACM.

Zhao, H., Meng, W., Wu, Z., Raghavan, V., & Yu, C. (2005, May). Fully automatic wrapper generation for search engines. In *Proceedings of the 14th international conference on World Wide Web* (pp. 66-75). ACM.

Zheng, S., Song, R., & Wen, J. R. (2007, July). Template-independent news extraction based on visual consistency. In *AAAI* (Vol. 7, pp. 1507-1513).