Product Centric Web Page Segmentation and Localization

John Cuzzola
Ryerson University
350 Victoria St
Toronto, ON M5B 2K3
Canada
jcuzzola@ryerson.ca

Dragan Gašević
Athabasca University
1 University Drive
Athabasca, AB T9S 3A3
Canada
dgasevic@acm.org

Ebrahim Bagheri
Ryerson University
350 Victoria St
Toronto, ON M5B 2K3
Canada
bagheri@ryerson.ca

ABSTRACT
The Internet is home to an ever increasing array of goods and services available to the general consumer. These products are often discovered through search engines whose focus is on document retrieval rather than product procurement. The demand for details of specific products as opposed to just documents containing such information has resulted in an influx of product collection databases, deal aggregation services, mobile apps, twitter feeds and other just-in-time methods for rapid finding, indexing, and notifying shoppers to sale events. This has led to our development of intelligent Web crawler technology aimed towards this specific category of information retrieval. In this paper, we demonstrate our solution for Web page categorization, segmentation and localization for identifying Web pages with shopping deals and automatically extracting specifics from the identified Web pages. Our work is supported with empirical data of its effectiveness. A screencast demonstration is also available online at http://youtu.be/HHPme6AJuCk.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval - *Information filtering, retrieval models, search process, selection process*. I.2.7 [**Artificial Intelligence**]: Natural Language Processing - *text analysis*.

General Terms
Algorithms, Experimentation.

Keywords
Natural language processing, search, classification, segmentation, localization, deals, products, web crawling

1. INTRODUCTION
The World Wide Web has given rise to a digital marketplace where goods and services of all varieties are sold. Retailers, wholesalers, and private individuals are using this communication medium to advertise their products directly to the consumer. Conversely, consumers are looking for these products and are using the traditional search engine as the method for discovery. However, these engines are document-centric rather than product-centric; hence they are optimized for the former rather than the latter. A successful search engine relies on its web crawlers to intelligently process visited Web pages for useful information while discarding data that does not contribute to retrieval. Geared specifically to this domain of product search, we have created technology that can identify product Web pages, segment Web pages into logical regions, and discard those regions that do not contain information regarding a specific goods or service. The remainder of this paper explains our Web page classification, segmentation and deal localization technology.

2. BACKGROUND
Our work reported in this paper was inspired by the needs of our industrial partner, SideBuy Technologies, which is a daily deal aggregator; a service which collects for-purchase goods and services from various deal sites such as Groupon, PriceGrabber and others. The process of collecting and aggregating these deal information is performed manually where large numbers of staff are employed as deal seekers [5]. Deal aggregators commonly deploy web scraping tools targeted at deal sites to harvest these deals. However, the collection process usually is dependent on pre-programmed recognized patterns specific to the site being scraped, e.g., using specific sequence of HTML tags. Consequently, even small modifications in such Websites will require programming changes in scraping tools to accommodate these changes. Furthermore, this targeted pattern matching approach does not scale to the unstructured and ever-changing content of the Web where many products are being sold but remain unnoticed and out-of-reach from the scrapers. Finally, the time sensitive nature of these deals further fuels the desire to leverage a more automated solution to the deal discovery dilemma.

To this end, we have developed algorithms to allow Web crawlers to identify unstructured, previously unseen, Web pages as containing information regarding relevant online deals. Once a page is classified as containing relevant information, our algorithms can segment and localize the regions of the Web page that contain product information, while discarding those areas that are not of interest.

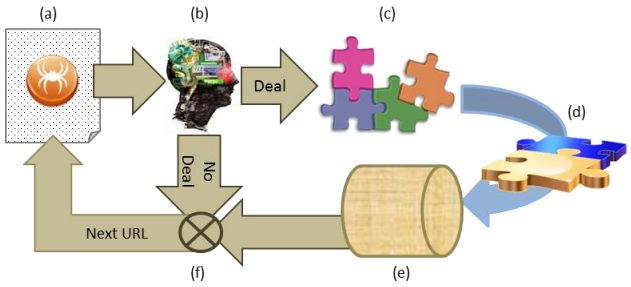

Figure 1: Technology pipeline: (a) Web crawler (b) Deal classifier (c) Page Segmentation (d) Localization (e) Storage

3. SYSTEM PIPELINE
Our process of information extraction from unstructured Web content is summarized in Figure 1. A Web crawler scrapes a given page for its HTML content (a). A binary classifier then determines whether the text of the page contains products for purchase (deal)

or no such offerings exists on that page (no-deal). Those pages classified as not containing products (no-deal) are discarded (f) while those pages categorized as deal undergo segmentation resulting in several segments per page (c). Each of the extracted segments will in turn be recursively classified as either containing deal or no-deal information in their own respect in an effort to localize individual products (d). Further processing on the deal segments involve semantic annotation, pattern matching, and image recognition that would extract property/value pairs, which are ultimately stored in a central repository (e).

3.1 Binary Classifier

We have developed a binary classifier capable of classifying a text/html fragment as either containing relevant products (deals) information or being void of such information (no-deal). The classifier is a hybrid Naive Bayes/Expectation-Maximization model trained using the WEKA machine learning framework [4]. We use the OpenNLP toolkit to incorporate named entity recognition for dates, organizational entities, time, location, percentages, money, and people. Part of speech tagging is combined with the WordNet lexical database to disambiguate word sense forms [3]. This information is used as features within our training dataset. The classifier is trained on information already manually extracted using SideBuy Technologies' deal scrapers. The detail of our classifier is available in [1].

```
<div class>
|_____<div style>
        |_____<p>
                The X7 Smartphone features a/b/g/n WiFi.
<div class>
|_____<div style>
        |_____<blockquote>
                The model S2 tablet comes with 4-GB RAM.
```

Listing 1: A sample recurring pattern in HTML.

3.2 Segmentation

Web page segmentation is the process of partitioning a Web page into logically grouped sections either visually, structurally, or semantically to form cohesive subsets of the Web page. As already reported by various researchers [6,7,8], ecommerce Websites often use a recurring pattern to represent product information. Therefore, each of the product information sets is represented under its own Web segment within the page. Besides the product segments on the page, there may be other segments such as banners, Web page footers, and others that are not relevant to product retrieval and search and can hence be discarded for our purpose (see Figure 2). We base our work on this observation and develop a Web page partitioning algorithm that processes Web page HTML contents and extracts all possible Web segments from that page.

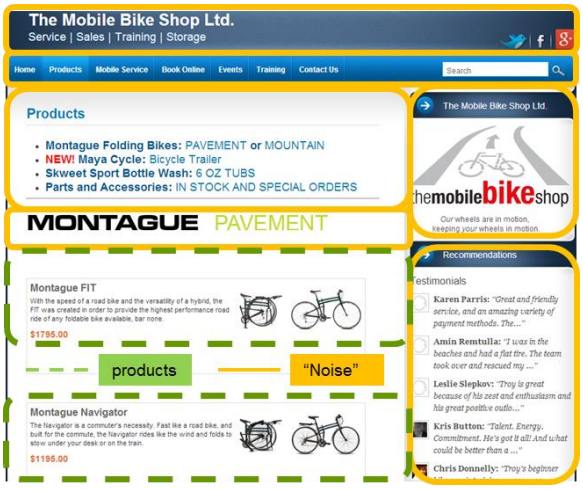

Figure 2: A segmented page. Product blocks in green dashed. "Noisy" blocks include header/footer, navigation bar, company logo, customer tweets in yellow solid.

Our system segments web pages based on HTML structure and textual clues obtained from natural language processing. Segmentation of a Web page is accomplished by finding the Longest Frequent Pattern (LFP) [2] of HTML tags at the topmost (outermost) block level. The identified LFP becomes the boundary of division for each partition in the Web page. For example, consider the sequence of nested HTML tags and textual content in Listing 1.

The topmost longest frequent pattern occurs twice with <div class>,<div style> resulting in two segments with fragments of "<p> *the X7 Smartphone feature a/b/g/n WiFi*" and "<blockquote> *the model S2 tablet comes with 4-GB RAM*". The result of this segmentation process is the localization of individual product offerings within each page in such a way that each individual segment will either contain individual product specifications such as name, description, and price or will represent non-product information in which case the segment is of no interest to us.

1. Let C be a set of candidate blocks of a web page.
   1.1 Initialize C with the outermost block.
   (Typically C←<HTML>…</HTML>)
2. For each block in C, classify block as either deal or no-deal using the binary classifier. Separate blocks into a deal set (η) or non-deal set.
   2.1 for each block $f \in \eta$
      2.1.1 Find the longest frequent HTML pattern
      (LFP) of sentence block f.
      2.1.2 If (LFP) exists:
         2.1.2.1 Split f in blocks on (LFP) $\rightarrow \beta$
         2.1.2.2 Add split blocks to C: $C \leftarrow C + \beta$
3. Goto Step 2 if C is non-empty

Algorithm 1: The Segmentation-Localization algorithm

3.3 Localization

Once Web segments have been extracted from a Web page, we perform localization on each of these segments. Localization is the process of determining which of these extracted segments contain useful and relevant product information such as the green dashed boxes in Figure 2 and also identifying those segments that contain non-relevant information and can be discarded such as the solid yellow boxes in Figure 2. In order to be able to efficiently perform the location process, we employ the same classifier that was introduced in Section 3.1. The classifier will now be used to determine whether each segment on their own would be classified as containing product-specific information or not. Therefore, the difference between the first step and the localization step would be that in the first step the classifier is used to determine whether the whole page contains product information, while in the localization step an individual segment within an already positively classified page is tested for containing product-specific information. Here, rather than evaluating the text of the entire page, only the text within this candidate segment is considered. If this block is positively classified, it is split recursively into smaller segments using the segmentation approach of Section 3.2. This process repeats iteratively for each newly segmented block until either the new block is negatively labeled, or a frequent pattern of HTML tags cannot be found. This process is illustrated in Figure 3 and can be visually summarized in a *segmentation parse tree* which is constructed by our implementation shown in Figure 4. The leaves of the segmentation parse tree represent the final outcome where each leaf node is either a segment of non-interest (negatively classified) or a segment containing a single product offering (positively classified localized segment). The localization algorithm is formally defined in Algorithm 1.

4. EVALUATION

Initial testing of our segmentation and deal localization algorithm involved 42 individual Web pages each from different Web sites. This set gave us a total of 1,402 individual products. The criteria used in the determination whether the final outcome was successful were as follows.

Criteria 1: A block is correctly classified if and only if the block makes reference to exactly one product offering. If the block contains information for more than a single product then it was under-partitioned and should have undergone further segmentation in order to split its contents into individual products.

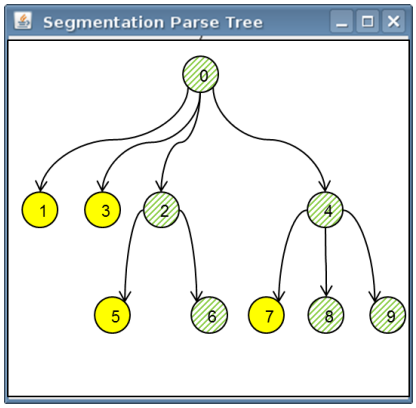

Figure 4: Tree representation of Figure 3. Segments 6, 8, and 9 contain individual product offerings (relevant).

Criteria 2: Because the descriptiveness of a product will vary significantly between websites; the minimum amount of information necessary is the name of the product and its price. Blocks that do not meet this minimum were considered to be over-partitioned.

Criteria 3: A leaf node that satisfies Criteria 1 and 2 but makes reference to the same product will only get credit for correctly classifying the product once.

With the above criteria in place, our system performed favorably with an average F-score of 0.903. The algorithm correctly identified 1,282 products with 154 misclassifications (false positives). A summary of the results is given in Table 1 sorted by best F-score. The relatively poor F-score's of the bottom 5 web pages appeared to be related to either the structure of the web page in which frequent patterns were difficult to find or the content of the page itself where the classifier mislabeled the segmented region as a non-deal area.

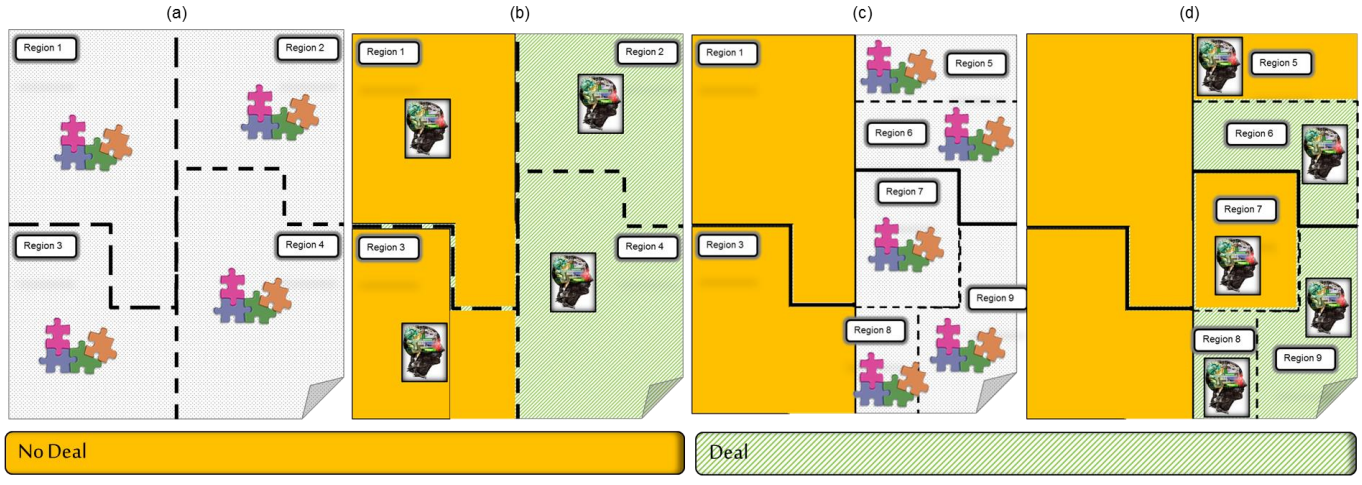

Figure 3: Segmentation/Localization illustrated. (a) Segmentation is performed on the entire web page using Longest Frequent Pattern (LFP). (b) Binary classifier labels each segment as either relevant (dashed green) or non-relevant (solid yellow). (c) Relevant segments are further partitioned using LFP. (d) The classifier labels partitions

Table 1: segmentation/localization evaluation results.

SITE	ACTUAL DEALS	FOUND DEALS	RIGHT	WRONG	PREC.	RE-CALL	F-SCORE
dailynews.com	8	8	8	0	1	1	1
trackdailydeals.com	13	13	13	0	1	1	1
deals.com	25	25	25	0	1	1	1
dealextreme.com	52	53	52	1	0.981	1	0.99
mydealbag.com	246	238	238	0	1	0.967	0.983
↖ TOP 5 ↗							
(32 sites aggregated)	950	970	869	101	0.896	0.915	0.905
↙ BOTTOM 5 ↘							
sidebuy.com	9	17	9	8	0.529	1	0.692
music123.com	20	13	11	2	0.846	0.55	0.667
dealfrenzy.com	5	10	5	5	0.5	1	0.667
elivedeals.com	53	64	39	25	0.609	0.736	0.667
rubywallet.com	21	25	13	12	0.52	0.619	0.565
TOTALS:	1402	1436	1282	154	0.893	0.914	0.903

5. DEMONSTRATION

Our segmentation/localization system was tested on a Web page from a deal aggregator's website: *pushadeal.com*. The output of the analysis is shown in Figure 5. Our intelligent crawler correctly identified the HTML pattern that encompasses individual products on this Web page.

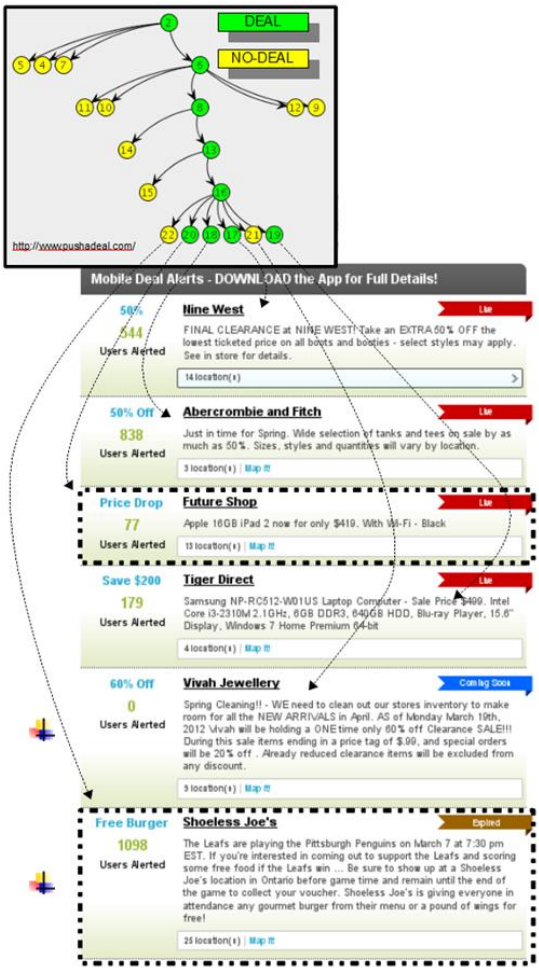

Figure 5: A segmented and deal-localized Web page.

The leaves of the generated segmentation parse tree reveal two potential product offerings that were classified as non-relevant (). By looking closely at the content of the page, one can see that this was correct since one product offer had "expired" while the other was "coming soon" and therefore not yet available. A further illustration of our system is available as a screencast at: http://youtu.be/HHPme6AJuCk. Also, visit the inextweb showcase section at http://inextweb.com which demonstrates how a database of localized segments are being utilized to provide an object-centered search engine over the familiar document centric engines of Google, Bing, and others.

6. CONCLUSION

This paper demonstrates our approach to Web page classification, segmentation and localization specific to the domain of goods and services procurement. We describe an intelligent Web crawler implementation that sees Web pages as containing product information. Our technology can be used to build a collection of properly annotated product objects, which can be leveraged for smarter search in the domain of e-commerce. In our demonstration we will showcase the described technology as follows 1) We will demonstrate how our machine learning and page segmentation techniques were trained and built; 2) We will introduce and provide open access to the wrapper API of our technology that is able to extract product information segments from Web pages; 3) We will show how to use our API to quickly write an application that would crawl a given website and extract product segments. An online demo is available at:

http://ls3.rnet.ryerson.ca:8086/DealExtractorSampleJavaClient/sampleform.html

7. ACKNOWLEDGMENTS

The authors would like to thank The National Science and Research Council of Canada (NSERC) and SideBuy Technologies Inc. for their funding support.

8. REFERENCES

[1] Cuzzola, J., Gašević, D., Bagheri, E., "What's the Deal? – Identifying Online Bargains," In Proceedings of the 2013 Australasian Web Conference (AWC 2013), Adelaide, Australia, 2013.

[2] J. Kang, J. Yang, J. Choi, "Repetition-based Web Page Segmentation by Detecting Tag Patterns for Small-Screen Devices", IEEE Transactions on Consumer Electronics, vol. 56, no. 2, pp. 980-986, 2010.

[3] Miller, G. "WordNet: A Lexical Database for English". Communications of the ACM 38(11): 39-41, 1995.

[4] Hall, M. Eibe, F., Holmes, G. Pfahringer, B., Reutemann, P. Witten, I. The WEKA Data Mining Software: An Update. SIGKDD Explorations, 11(1): 2009.

[5] Ghigliotty, D. "Do You Really Want a Job at Groupon?" Retrieved from http://salesjobs.fins.com/Articles/SBB00014240529702045282045770120734724148 32/Do-You-ReallyWant-a-Job-at-Groupon, 2011.

[6] Chakrabarti, D.,Kumar, R., Punera,K. Page-level template detection via isotonic smoothing. In Proceedings of the 16th international conference on World Wide Web (WWW '07). ACM, New York, NY, USA, 61-70, 2007.

[7] Kao, H., Ho, J., Chen, M. WISDOM: Web Intrapage Informative Structure Mining Based on Document Object Model. IEEE TKDE 17 (5): 614-627, 2005.

[8] Chakrabarti, D., Kumar, R., Punera, K. A graph-theoretic approach to webpage segmentation, International conference on World Wide Web, pp 377-386., 2008.