Extracting Informative Content from Web Pages Using Content Extraction Algorithm

Yu Wai Hlaing
University of Computer Studies, Yangon
Yuwaihlaing.1987@gmail.com

Abstract

Apart from the main content blocks, almost all web pages on the Internet contain such blocks as navigation, copyright information, privacy notices, and advertisements, which are not related to the topic of the web page. These blocks are called noisy blocks, and the main content blocks are called informative blocks. The information contained in the noisy blocks can seriously harm Web mining and searching. So discriminating informative blocks from the noisy blocks and then extracting the information contained in the informative blocks is an important task. In this paper, the problem of automatically extracting the web information (unsupervised IE) without any learning examples or other similar human input is studied. Firstly, web pages are segmented into several raw chunks. Then removed the noisy blocks based on product features. Content extraction is based on the relation among punctuation mark density, length of information text and anchor text density. This approach requires no human intervention, no prior knowledge of the input HTML page and no training set are required.
Keywords: Web Mining, Information Extraction (IE), Unsupervised IE, Informative Blocks

1. Introduction

The explosive growth and popularity of the world-wide web has resulted in a huge number of information sources on the Internet. As web sites are getting more complicated, the construction of web information extraction systems becomes more difficult and time-consuming. Therefore, Web information extraction is an important task for information integration. However, due to the heterogeneity and the lack of structure of Web information sources, access to this huge collection of information has been limited to browsing and searching. Sophisticated Web mining applications, such as comparison shopping robots, required expensive maintenance to deal with different data formats. To automate the translation of input pages into structured data, a lot of efforts have been devoted in the area of information extraction (IE). Unlike information retrieval (IR), which concerns how to identify relevant documents from a document collection, IE produces structured data ready for post-processing, which is crucial to many applications of web mining and searching tools. A typical web page consists of many blocks or areas, e.g., main content areas, navigation areas, advertisements, copyright information etc. For a particular application, only part of the information is useful, and the rest are noises. In this approach, removing noise is based on common web page features. Hence, it is useful to separate these areas automatically.

Web information extraction (WIE) is concerned with the extraction of relevant information from web pages and transforming it into a form suitable for computerized data-processing applications. Example applications include: price monitoring, market analysis and portal integration. Mining these data records in web pages is useful because they typically present their host pages' essential information, such as lists of products and services. Many researchers research on extraction of information from web pages in different domains (travelling, products and business intelligence) but this paper deal with product domain on mobile phone store web sites. It is well known that web pages are used to publish information for humans to browse, and not designed for computers to extract information automatically. Especially web pages from shopping sites, the underlying structure of current web pages is more complicated than ever and is far different from their layouts on web browsers. This makes it more difficult for existing solutions to infer the regularity of the structure of web pages by only analyzing the tag structures.

Meanwhile, to ease human users' consumption of the information extraction, good template designers of web pages always arrange the data records and data items with visual regularity to meet the reading habits of human beings. For example, all the data records in Figure. 1 is clearly separated, and the data items of the same semantic in different data records are similar on layout and font.

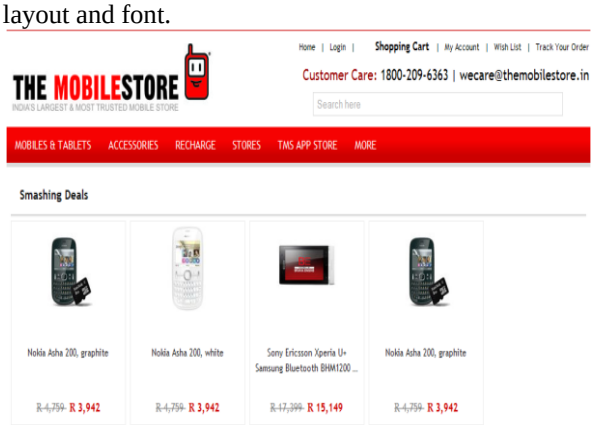

Figure1. An example of web page form Yahoo mobile store web site

In this approach, a three-step strategy is employed to extract information content on web document. First, given a web page, transform it into DOM tree and parse several initial raw chunks or blocks; second, filtering

noisy block based on product features; and third, extracting the main content based on their weights. In the latest case, an algorithm is used to extract informative content.

This paper is divided into several sections. Section 2 describes the related work on web information extraction. In section 3, the proposed methodology is discussed in detail. Section 4 shows the result of experimental tests while section 5 concludes this paper.

2. Related Work

Currently, there are already a lot of content extraction algorithms based on different features of main text.

1) Lin and Ho [3] proposed an extraction method based on information entropy, the web page is divided into content block according table tag, each block has entropy, and then information blocks are obtained by comparing with threshold value. But this method just applies to web pages which contain table tags, while increases the complexity of the algorithm.

2) Yi and Liu [4] put forward an extraction approach based on template, this method assumes that the same part of two web pages having same format, so it is simple and effective to remove noises by comparison of two web pages coming from one source. But it is difficult to identify so many templates for a variety of web pages.

3) In the extraction approach based on framework web pages and rules [5], supposes it is reasonable the noise blocks generally locate in the secondary positions in the page. This method compares the ratio of width and height attributes of every table tag, and removes the tags of bigger ratio. It is hard to work well on table tags with less height and width attributes. The table tag is the only Processing Objects to this approach.

4) In web content information extraction method based on tag window [6], which could cope with some special circumstances that web pages content text locate in table and div tags, all page content information is put into one td or several tds, and the length of body text is short as that of the other information such as navigation bars, and the copyright, etc. But during the process of judging body text, it involves word segmentation and computing similarity of string, which has enhanced the complexity of information content extraction.

5) Pan and Qiu [7] put forward a web page content extraction method based on link density and statistic, which recognizes main content according to the different properties between content nodes and non content nodes of web page represented as a tree. But the threshold values do not always adapt to some news pages, so it is still hard to find a set of universal parameters.

6) ROADRUNNER [10] extracts a template by analyzing a pair of web pages of the same class at a time. It uses one page to drive an initial template and then tries to match the second page with the template. Deriving of the initial template has to be again done manually, which is a major limitation of this approach.

These methods, based on removing noise is suitable to delete a lot of nodes without any web content information, they can contribute to filter the unrelated part using the layout properties of noises in the web pages. The most current ways are lack of enough considering on removing noise in the preprocessing. Furthermore, large numbers of web sites have so many hyperlinks in the information text that those methods which over-reliance on links have poor results. However, proposed method focuses on list pages of same presentation template. Although some aspects and pieces of web information extraction may be around in various techniques, the important of this paper focus on the some interesting features of web page and the relation among punctuation mark density, length of information text and anchor text density is considered enough in the extraction stage.

3. System Overview

This section describes architecture of proposed system. This paper proposes an automatic information extraction from product web pages. Automatic methods aim to find patterns/grammars from the web pages and then use them to extract data. Examples of automatic systems are MDR [1] and VIPS [11]. The proposed system architecture is shown in figure 2. First of all, input HTML page is changing DOM tree and cleaning useless node as preprocessing step.

Secondly, ticking out several raw chunks as a first round and then filter the noisy block based on noisy features of web pages is carried out. The third step is extraction main content blocks. It involves calculating node link density, non anchor text density, punctuation mark density and rough weight, then adjusting the weights to make the result is precise. Finally, extract information from input web page.

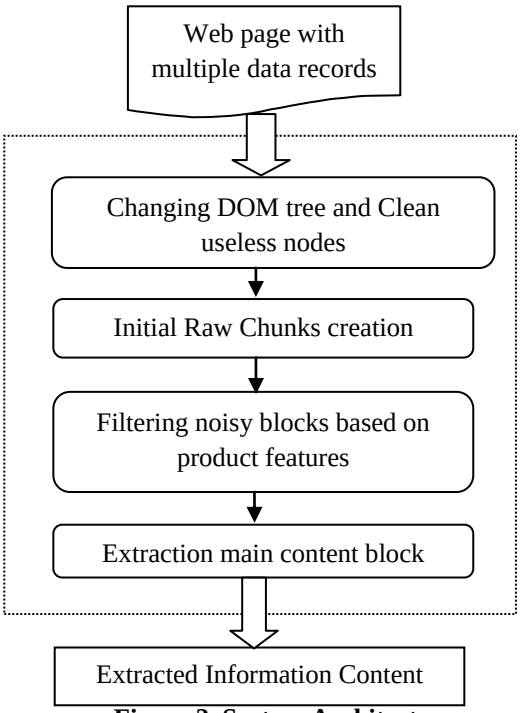

Figure 2. System Architecture

3.1. Feature in web pages

Web pages are used to publish information to users, similar to other kinds of media, such as newspaper and TV. The designer often associate different types of information with distinct virtual characteristics to make the information on web pages easy to understand. As a result, virtual features are important for identifying special information on web pages.

Position Features (PFs). These features indicate the location of data region on a web page.
PF1: Data regions are always centered horizontally.
PF2: The size of the data region is usually large relative to the area size of the whole page.

Layout Features (LFs). These features indicate how the data records in the data region are typically arranged.
LF1: The data records are usually aligned flush left in the data region.
LF2: All data records are adjoining.
LF3: Adjoining data records do not overlap, and the space between any two adjoining records is the same.

Appearance Features (AFs). These features capture the visual features within data records.
AF1: Data records are very similar in their appearances and the similarity includes the number of images they contain and the fonts they use.
AF2: The data items of same semantic in different data records have similar presentations with respect to position, size (image data item) and font (text data item).
AF3: The neighboring text data items of different semantics often use distinguishable fonts.

3.2. DOM Tree Generation and Clean useless node (Preprocessing)

To begin with, a DOM tree should be generated from html tags of the page. The Document Object Model (DOM) is a standard for creating and manipulating in-memory representations of HTML (and XML) content. By parsing a webpage's HTML into a DOM tree, we can not only extract information from large logical units but can also manipulate smaller units such as specific links within the structure of the DOM tree. In addition, DOM trees are highly editable and can be easily used to reconstruct a complete webpage. At the same time, features/attributes about the node should be included in these tree nodes, respectively; described in next section.

Preprocessing is necessary in order to clean HTML pages, e.g., to remove header details, scripts, styles, comments, hidden tags, space, tag properties, empty tags, etc. In this step, the white relax functions of the jsoup parsing tool [12] for removing cleaning HTML tag is used. First of all, we need to eliminate these nodes to get clean HTML page for further processing.

3.3. Initial Raw Chunks Creation

After the generation of DOM tree, following processes are based on it. The root of this tree corresponds to the whole document. The intermediate nodes represent HTML tags (e.g., <table>, <ul>, <tr>, <p>, etc) that determine the layout of the page. At first, those HTML tags, such as <table>, , <ul>, <tr> and <p> naturally form chunks; then we could extract those chunks when encountering them for the first time. That is to say initially, partition a web page into several raw chunks, just like B-1, B-1-1, B-1-2, B-1-3 and so on is illustrated in figure 3.

As described in previous section, each node contains some attributes. Here we list these attributes as follows:

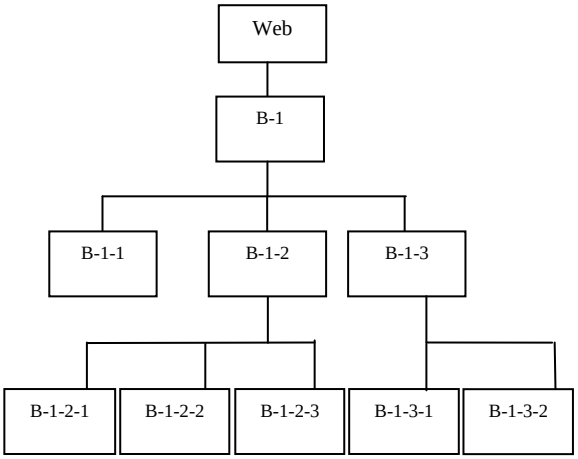

Figure 3. Example of partitioning web page

{ImgNum, ImgSize, LinkNum, LinkTextLen, InteractionNum, InteractionSize, FormNum, FormSize, EmailNum**}**

ImgNum and ImgSize are the number and size of images contained in the block. LinkNum and LinkTextLen are the number of hyperlinks and anchor text length of the block. InteractionNum and InteractionSize are the number and size of elements with the tags of <input> and <select>. FormNum and FormSize are the number and size of element with the tag of <form>. EmailNum is the number of email contained in this block.

Thus, a DOM tree can be parsed into several initial blocks containing relevant features. However, some blocks, such as navigation and content bar are unrelated to the topic at all; are removed for reducing the complexity of information extraction.

3.4. Filtering Noisy Blocks

Web page designers tend to organize their content in a reasonable way: giving prominence to important things and deemphasizing the important parts with proper features such as position, size, color, word, image, link, etc. All of product page features are related to the importance. For example, an advertisement may contain only images but no texts, a contact information

bar may contain email, and a navigation bar may contain quite a few hyperlinks. However, these features have to be normalized by the features values of the whole page to reflect the image of the whole page. For example, the LinkNum of a block should be normalized by the link numbers of the whole page. Then all these features are formulated with equation (1).

$$f_i(a) = \frac{\text{number of attributes in block i}}{\text{Number of these attributes in whole page}} \quad (1)$$

Firstly, some conclusions are given on product features. All of those conclusions are according to the observation of product list page on web site.

1) If a block contains email elements, then it is entirely possible a contact block.
2) If TextLen/LinkTextLen<threshold, then it is quite possible a hub block [8].
3) If <p> is included in a block, then this block is possibly authority block [8].
4) If the normalized LinkNum > threshold, then it is quite possible a hub block.

Accordingly, these rules are calculated into equation (2) and then F indicates the possibility of noisy block.

$$F=\sum \alpha_i.f_i(b)= \alpha_1.f_{email}(b)+ \alpha_2.f_{textlen/linktextlen}(b)+\ldots+\alpha_4.f_{links}(b),\sum \alpha_i=1 \quad (2)$$

Where α_i is coefficient, we can set different weights on block importance respectively. In this paper, product domain is mobile phone store web sites so the value of coefficient α_i is used as 0.2. But it may be changed depending on web pages in different domains. Additionally, all these parameters can be adjusted to adapt to different conditions.

Also, if F is beyond a predefined threshold, the value of 0.5 is used in here, and then this chunk has to be ruled out and remove it. Finally, regarding product features, an important block is extracted from further processing. Consequently, filtering noisy block can decrease the complexity of web information extraction through narrowing down the processing scope.

3.5. Extraction Main Content Block

The main text often contains the comparatively many characters; in addition, the amount of punctuation marks is proportional to length of information text. Most of the noise nodes are composed almost exclusively of anchor text, almost no texts, and few punctuation marks. The extraction algorithm is described by the following parameters.

1) Node-Link Text Density (NLTD): the ratio of the anchor text length to all text length in a given node. To a certain extent, this value plays a good role in identifying main content of general web pages, Comparing noise blocks, the smaller this value, the greater the possibility of main content. But it is not uncertainty in some special circumstances that the greater value always represents the higher possibility of noise blocks. This indicator can work well when cooperates with other indices.

2) Non-Anchor Text Density (NATD): the ratio of the non-link text length in a given node to the total non link text length in all nodes of web pages. This indicator denotes the absolute proportion of useful information in the whole web page. So the possibility of information content blocks is in direct proportion to this value.

3) Punctuation Mark Density (PMD): the ratio of the number of punctuators in a given node to the number of punctuation marks in the news page. Because of the lack of punctuation marks in the noise blocks, it is reasonable to consider punctuation density as an indicator of judging information content blocks.

These three parameters are used to calculate the weight of each information content block. None of them is decisive factor in judging main content, so we obtain a good result when they coordinate with each other according a certain proportion. The aim of this process is to select the optimum node containing the content. According to the experimental results and statistics [2], select NLTD = 0.4, NATD=0.5, PMD = 0.7, as the basic content node judgment parameters. The tag window contains all the information of the tag, such as original text and code, length of text, amount of punctuation marks, etc. Each tag window has a weight which indicates the possibility of main content. They are calculated according to the following equation (3).

$$\text{Weight} = \text{NLTD} + \text{NATD} + \beta \text{ X PMD} \quad (3)$$

In this equation β is used as a predefined coefficient for calculating the weight value. In this paper, product domain is mobile phone store web sites so 0.7 is used as coefficient's value. But it may be changed depending on web pages in different domains. The bigger the tag window, the more information content it contains. After this measure, rough weights of tag windows are calculated. It is impossible to completely eliminate the noise from web pages in the front work; most pages still have some noise, such as "contact", "company profiles", "copyright ", etc. Though they may appear in the main content, the possibility of appearance together is low. Accordingly, we stipulate that if five or more of these words appear in some tag window. The value of five is used as a predefined threshold to adjust the weight value in next step. And then subtract 0.6 from weight of tag window. After sorting the weights, it is easy to find the optimum tag window which has the greatest weight. If less than 5 of these words appear in some tag window, it goes to next stage of extraction informative content.

Finally, extract information from input web page. The flowchart of overall informative content extraction is illustrated in figure 4.

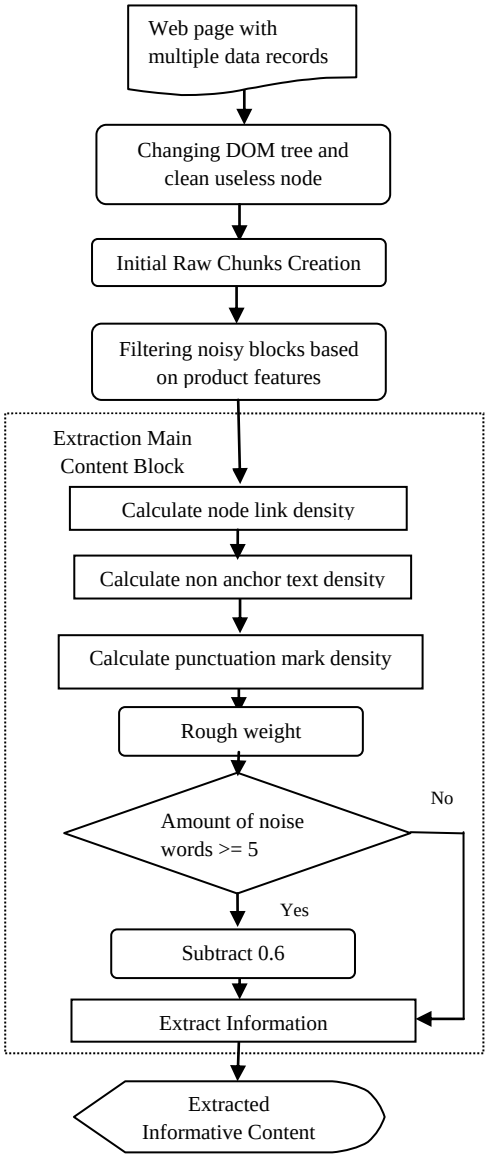

Figure4. Flowchart of content extraction

The goal of the system is to offer the extracted data records (shown in table 1) from web pages to the information integration system such as price comparison system and recommendation system. The content is extracted by tools such as htmlparser [13]. Then the extracted informative content is shown in table 1 as follows.

Table 1. Extracted Informative Content

Serial No.	Brand and model	Price
1.	Nokia Asha 200, graphite	Rs 3,942
2.	Nokia Asha 200, white	Rs 3,942
3.	Sony Ericsson Xperia U+, Samsung Bluetooth BHM 1200…	Rs 12,149
4.	Samsung Champ Deluxe Duos+, 4GB memory card, Black	Rs 3,890

4. Experimental Results

The experiments were testing using commercial mobile store web sites collected from different web sites. The system takes as input raw HTML pages containing multiple data records. The measure of proposed method is based on three factors, true positive (TP), false positive (FP) and false negative (FN). Based on these three values, precision and recall are calculated according to the formulas:

Recall = (TP/ (TP+FN))*100
Precision = (TP/ (TP+FP))*100

where
FN = number of relevant data records that are not extracted
FP = number of irrelevant data records that are extracted
TP = number of relevant data records that are extracted

According to the above measurement, we tested web pages from various mobile store web sites and check each page by manually. The results are shown in table 2.

Table2. Results for selected web site

URL	Precision	Recall
yahoo.com	98	97
amazon.com	92.4	87.5
wirefly.com	100	98
bestbuy.com	98	90
Average	97.1	93.1

5. Conclusion

Web page content extraction is more and more essential for mining the main content of pages. In this paper, we have presented extraction of information content from semantic structured HTML documents. Firstly, the web page is segmented into several raw chunks. Second, filter the noisy block. Then the main content blocks are extracted based on the relation among punctuation mark density, length of information text and anchor text density. In this case, neither prior knowledge of the input HTML page nor any training set is required. The proposed method is experimented on multiple web sites to evaluate proposed method and the results prove the approach to be promising.

References

[1] B Liu, R.Grossman and Y. Zhai, "Mining data records in Web Pages", ACM SIGKDD Conference, 2003.
[2] R.Gunasundari, S.Karthikeyan, "A New Approach for Web Information Extraction", Int.J.Computer Techology & Applications, Vol 3 (1), Jan-Feb 2012, pp. 211-215.
[3] L.S.Hua, H.J.Ming, "Discovering informative content blocks from Web documents", Proceeding of the 8th ACM SIG KDD International Conference on Knowledge Discovery and Data Mining. Edmonton: ACM Press, 2002, pp. 588-593.
[4] Y.Lan, L.Bing, L.X.Li, "Eliminating Noisy Information in Web Pages for Data Mining", Proceeding of the 8th ACM SIG KDD International Conference on Knowledge Discovery

and Data Mining, Washington, DC: ACM Press,2003, pp. 296 – 305.
[5] D.Shi, H.Lin, Z.Yang, “An Approach to Eliminate Noise Based on Framework of Web Pages and Rules”, Computer Engineering, 2007, pp. 276-278.
[6] X.X.Zhao, H.G.Suo, Y.S.Liu, “Web Content Information Extraction Method Based on Tag Window”, 2007, pp. 144-145,180.
[7] P.Donghua, Q.Shaogang, “Web Page Content Extraction Method Based on Link Density and Statistic”, The 4Th International Conference on Wireless Communications, Networking and Mobile Computing, Oct. 2008, pp. 1-4.
[8] D.Cai, H.Xiaofei, W.Ji-Rong and M.Wei-Ying, “Block Level Link Analysis”, IGIR’04,July 25-29, 2004.
[9] N.N.Hlaing, T.T.S.Nyunt, “Extracting Informative Content from Web Page Using Block Clustering Method”, Proceedings of 10th International Conference on Computer Applications ICCA 2012, Feb 28-29 2012,pp. 78-81.
[10] Crescenzi, V. and Mecca, G. “Automatic Information extraction from large web sites”, Journal of the ACM, 2004, pp.731-779.
[11] Cai, D., Yu, S.,Wen, J.-R and Ma, W.-Y., VIPS: a vision-based page segmentation algorithm, Microsoft Technical Report.
[12] http://jsoup.org/
[13] http://sourceforge.net/projects/binhgiang/