Postal Address Detection from Web Documents

Lin Can, Zhang Qian, Meng Xiaofeng
School of Information
Renmin University
Beijing, 100872, PRC
{lincan, zhangqian15, xfmeng}@ruc.edu.cn

Liu Wenyin
Department of Computer Science
City University of Hong Kong
Tat Chee Avenue, Hong Kong SAR, PRC
csliuwy@cityu.edu.hk

Abstract

An approach to postal address detection from webpages is proposed. The webpages are first segmented into text blocks based on their visual similarity. The text content in each block undergoes the recognition process, which employs a syntactic approach. The grammars of almost all possible patterns of postal addresses are built for this purpose. The results of our preliminary experiments on 44 webpages with 56 true addresses show that our approach can detect the postal addresses with a high precision (89.3%) and a low false alarms rate (3.8%).

1. Introduction

Like named entities [1], postal address is a kind of useful information for document understanding and can be used in many applications. As the amount of online resources, especially webpages (in html format), is growing rapidly, a huge number of postal addresses can be collected based on postal address detection from these webpages. It is expected that these addresses can be extracted and used for automatic e-commerce applications.

Computers cannot easily understand the semantics of webpages and hence automatic detection of postal addresses is a non-trivial task for computers. The first reason is that computers cannot easily (correctly) group the related text in nearby sections into meaningful blocks, and the second reason is that it cannot easily recognize the meaning of text.

The problems involved in postal address detection from webpages include segmentation and recognition. First of all, the text blocks containing candidate postal addresses should be segmented from its context (or environment, which is actually the document or webpage where the address resides). Each text block is then recognized as an address or not.

There have been many research works about text segmentation, mainly in the information retrieval and machine learning community. Statistical methods [5] and similarity-based methods [6] are commonly employed for this purpose. These works mainly handle plain text, and these methods are suitable for relatively large topic boundary detection. In order to extract data from webpages, schema-based method [3] and wrappers [4] are used.

Existing research works on postal address detection are mainly handwritten postal address recognition in postal systems [7].

In this paper, we propose an approach to postal address detection from webpages. It consists of two steps. First, a vision based text segmentation is done on webpages, which uses only the layout formatting information, such as font-size, font-family, font-weight, border, fore-color, position, size, and so on, and does not use the text itself. Then, the second step applies a syntactic pattern recognition method to identify if the text block is a postal address or not. We have summarized a set of syntactic rules (grammars) of common postal address patterns for this purpose.

The rest of this paper is organized as follows. Section 2 presents our method for text segmentation on webpages. Section 3 presents our method for postal address recognition. Section 4 shows our preliminary experiments and results. Finally, we present our concluding remarks in Section 5.

2. Vision Based Text Segmentation

There have been many works about text segmentation, mainly about plain text segmentation, including statistical methods [5] and word similarity based methods [6]. But these methods are not suitable for text segmentation on webpages. In statistical methods, corpus is used and errors may occur due to the limitation of the corpus. In word similarity based methods, lexicons must be built in advance.

In order to segment text on a large amount of webpages, collecting a corpus or building lexicons is not an easy task. Besides, the text blocks on a webpage are relatively small and classical segmentation methods for plain text are no

longer effective. The major difference between plain text and webpages is that a webpage has both text content and layout formatting information. The layout formatting information can be used for more precise text segmentation.

In Liu et al's paper [2], they assume that when a webpage (especially, a news webpage) is composed or updated, if there are multiple topics on the page, the editor of the page usually puts content of the same topic in a visually distinguishable block. They proposed a method to divide a webpage into salient blocks (whose content are visually consistent within blocks and distinguishable among adjacent blocks) and then merge them into several big topic blocks according to their similarity in appearance, such as font-family, font-weight, font-size, fore-color, back-color, borders, position, size, and so on. The main purpose of their paper is to track the user's clicking actions in the same topic blocks in a webpage's consecutive days' editions in order to detect the user's interest. The resulting topic blocks are relatively big. A typical news webpage is usually divided into 5 - 20 topic blocks.

2.1. Text Segmentation

Based on the idea in Liu et al's paper [2], we propose a vision based text segmentation method for postal address detection from webpages.

The basic idea is that we first obtain all smallest text snippets based on visible elements containing text on webpages (in DOM [8], an element means an object defined in html). These text snippets contain both text content and layout information, such as font, border, color, position, size, and so on. Then, we merge them into more integrate and meaning blocks based on their visual similarity and adjacency relationship.

Our method is similar to Liu et al's paper [2], except for the following aspects:

1. In the first step, we extract all text snippets from the webpage, which are actually the text nodes in the webpage's DOM tree. They are actually the smallest unit for merging in the next step. This is different from Liu et al's method, which yields salient blocks as the smallest unit for merging.
2. In merging, we obtain the text blocks by combining the text snippets instead of larger salient blocks [2]. A text block refers to a visual consistent block grouped by several text snippets.

2.1.1. Text Snippet Extraction

A segment of a webpage marked with html tags is shown in Figure 1. In Figure 1, there are 5 text nodes, each resides in an element. This is the simplest situation; we can easily obtain each text snippet's layout information from the corresponding DOM element without any further calculation.

Another segment of a webpage marked with html tags is shown in Figure 2. The difference between Figure 2 and Figure 1 is that in Figure 2, element <P> has several text nodes. We cannot easily see these text nodes as one big text node, because they belong to different text blocks. For example, text snippets "IBM Corporation", "1133 Westchester Avenue", "White Plains, New York 10604", "United States" belong to a single text block, and text snippet "1-800-IBM-4YOU" belongs to another text block, text snippets "Mailing Address:", "Phone numbers:" also belong to different text blocks (as you can see in Figure 4). In order to merge these text nodes into correct text blocks, each text node is considered as a single text snippet and its display attributes and layout attributes (including font, border, color, position, size, etc.) are calculated based on its parent and siblings (the display and layout attributes for text nodes cannot be obtained directly from DOM). For example, suppose P stands for element <P>, B stands for the first element <B> (it contains "Mailing Address:" as its text node), S1 stands for the text snippet "IBM Corporation". S1 inherits several display attributes from its parent, e.g. S1.font-family = P.font-family (where, S1.font-family stands for the font-family attribute of S1); the position of S1 is calculated from its sibling, e.g. S1.Top = B.Top + B.Height(where, S1.TOP means the y-coordinate of the upper-left corner of S1, B.Height means the geometry height of B). All layout attributes such as font, border, color, position, and size are calculated in similar way.

Figure 1. A segment of a webpage marked with html tags.

2.1.2. Merging

Figure 2. Another segment of a webpage marked with html tags.

After the text snippet extraction step is finished, we obtain many small text snippets with both text content and layout information. The next step is to merge these small text snippets into relatively large meaningful text blocks.

The merging step is a clustering process. The distance matrix is calculated based on the similarity and adjacency relationship of any two text blocks. The criterion we used is the same as in [2], as shown in Figure 3. However, we tune the weights to meet the postal address recognition situation.

Dis(A, B)	=	IsNeighbor(A, B)	×	Neighbor_Factor
	+	Offset(A, B)	×	Offset_Factor
	+	Background(A, B)	×	Background_Factor
	+	Foreground(A, B)	×	Foreground_Factor
	+	Font(A, B)	×	Font_Factor
	+	Size(A, B)	×	Size_Factor
	+	Alignment(A, B)	×	Alignment_Factor

Figure 3. Definition of Distance Function.

The full explanation of the distance function can be found in Liu et al's paper[2]. Background(A, B) is the background difference of text block A and B. If A and B have the same background, then Background(A, B) = 1, otherwise Background(A, B) = 0. Background_Factor is the weight of Background(A, B) in the calculation of Dis(A, B).

The merging result of Figure 2 is shown in Figure 4, which consists of four text blocks. This is actually the segmentation result of Figure 2.

2.2. Cue Block and Body Block

We classify the text blocks into two categories, cue blocks and body blocks. Cue blocks are usually for the purpose of indications, annotation, and explanation. Body blocks contain main text body content, such as postal address, telephone number, price, product description, etc. In Figure 4, the first block is a cue block, and the second block is a body block.

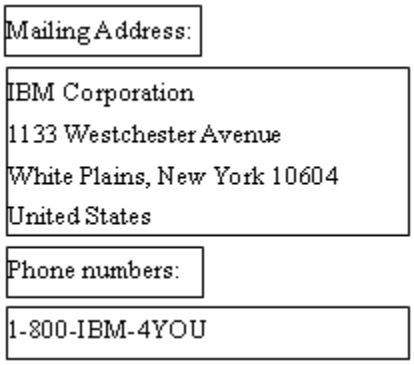

Figure 4. Segmentation result of Figure 2.

In the recognition step, we focus on body blocks, but cue block can still provide with useful information for recognition. Hence, if we can find a cue block, which is associated with a body block, the body block can be recognized as a postal address with a higher certainty.

Based on observation of many real cases, we find a very simple rule that can be used for recognition of cue blocks: if a block's occupation space is less than a threshold, it is considered as a cue block. This rule works very well in our experiments.

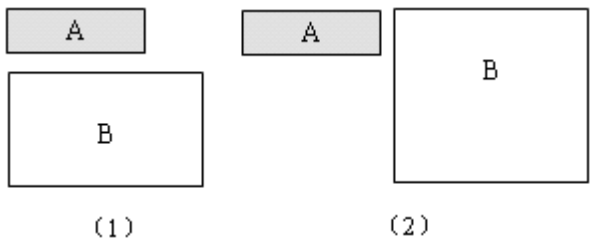

Figure 5. Illustration of spatial relationships of cue blocks and body blocks.

In order to use the help of cue blocks for recognition of postal address, we should know which cue block is associated with the candidate body block. We calculate the position relationship between a given body block and all cue blocks. As illustrated in Figure 5, A stands for a cue block and B stands for a body block. For the vertical situation (1), if the cue block and the body block satisfy the following three conditions, the cue block is considered as a candidate cue block for the body block.

Condition 1: $A.X > B.X || (B.X - A.X) < \alpha$
Condition 2: $(A.X + A.Width < B.X + B.Width) || (B.X + B.Width) - (A.X + A.Width) < \alpha$
Condition 3: $A.Y + A.Height < B.Y$

Here, A.X means the x-coordinate of the upper-left corner of A, A.Y means the y-coordinate of the upper-left corner of A.

The Condition 1 And Condition 2 mean that the horizontal range of A should be with the range of B extended to a small extent α. Condition 3 means that A must above B.

The candidate cue block that has the least value of $B.Y - (A.Y + A.Height)$ is considered as the vertical cue block for the body block. This block is the closest block above B.

3. Recognition of Postal Address

After text segmentation, we obtain several body blocks and their associative cue blocks. The next step is to decide whether a body block is a postal address or not. In this paper we propose a syntactic method for this purpose.

We use both the body block content and its vertical/horizontal cue block. Our recognition result is the confidence that a text block is a postal address other than a true/false determination. If the confidence value is greater than a threshold, the content of the body block is considered as postal address. The overall certainty that a text block is a postal address is calculated as follows:

$$C = C(cue) * \alpha + C(body) * \beta \quad (1)$$

where, α and β are weights, and $\alpha + \beta = 1$, C(cue) is the confidence that the cue block is an address indicator, C(body) is the confidence that the body block's content is a postal address, and C is the overall confidence that the body block is postal address based on both the cue block's hint and text block's content.

Based on investigation of many real cases, we build a lexicon of indication text of postal address(such as "mailing address", "office address"). If the text in a block is in the lexicon, it is considered as an address indicator.

We use a syntactic method to calculate C (body). Although compared with more structural information, such as telephone number, fax number, etc, postal address is less structural; it has its own loose format(patterns). For example, a postal address in USA usually consists of organization name, street, city, zip code, state, Nation in this order. However, some of them are optional and the order may not always be from small to big.

The syntactic method used in this paper includes two steps, tokenization and parsing.

3.1. Tokenization

We tokenize the content in a body block into a sequence of tokens. Five lexicons are used, Nation lexicon, state lexicon, city lexicon, street suffix lexicon, and organization suffix lexicon.

First, the text content is spitted into words, white spaces are ignored but other special characters are kept, such as commas, semicolons, and newlines, because they are useful to separate the whole sentence into meaningful segments. For example, the organization name and the street part may be written in different lines and commas can be used for separation. Then, the words in the text content which are found in the lexicons are replaced by their corresponding lexicon type, such as CITY, STATE, and so on, based on maximal matching.

We define the token types to include the following:
CAPS
LOWER
NUMBER
DELIMITER
ORGANIZATION_SUFFIX
AVENUE_SUFFIX
CITY
STATE
NATION
UNKNOW.

For example, the body block in the second block of Figure 4 is tokenized to Figure 6. Notice that commas and newlines are kept and assigned a type of delimiter. "White Plains" is recognized as CITY, "New York" is recognized as STATE, "United States" is recognized as NATION based on the lexicons.

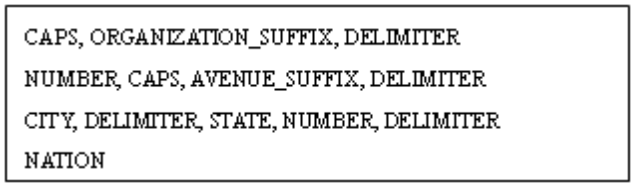

Figure 6. Tokenization of the postal address in Figure 4.

In the process of tokenization, there might be ambiguities in some situations. For example, "America" can be recognized as STATE or CITY. In this situation, we simply cover both situations and produce several tokenization sequences. Then, in the parsing step, if any sequence matches

the grammar, the body block is considered as a postal address.

3.2. Parsing

Because the structure format of postal address is not strict, we use regular grammars with confidence to describe the patterns of postal addresses.

Each rule has a confidence. A rule will be defined in the following format:

$$N \rightarrow \xi, Pr(N \rightarrow \xi) = \alpha \quad (2)$$

where α is the confidence. During the processing of a body block, if a sequence of rules r_1, $r_2 \ldots r_n$ is used, their corresponding confidences are P_1, $P_2 \ldots P_n$, then, $C(body) = \prod P_i$.This means that if all rules used during the parsing step have high confidence, the overall confidence is high.

A sample set of rules we used is shown as follows.

1. ADDRESS := ORGANIZATION STREET CITY STATE NUMBER NATION (100%)
2. ADDRESS := STREET CITY STATE NUMBER (90%)
3. ORGANIZATION := CAPS [CAPS|LOWER]* ORGANIZATION_SUFFIX (90%)
4. AVENUE := CAPS [CAPS|LOWER]* AVENUE_SUFFIX (90%)
5. AVENUE := NUMBER CAPS [CAPS|LOWER]* AVENUE_SUFFIX (100%)

For example, if rules 1, 3, 4 are used during the parsing process, then, $Pr(body) = 100\% \times 90\% \times 90\% = 81\%$.

4. Experiments and Results

Our approach to postal address detection from web documents consist of 2 major steps: text segmentation on web pages and postal address recognition. Hence, we do experiments to evaluate the precision of the 2 steps.

We submit a query "allinurl:com/contact" (all URLs that contain "com/contact", e.g., "www.johnkerry.com/contact/") to www.google.com. Among the first 50 results returned by google, 6 cannot be accessed. We select the other 44 webpages as the test data set. These pages are mostly the contact pages of certain companies, and may have postal addresses on them. Figure 7 shows a typical context of a postal address on a webpage, in which a cue block and a body block containing the postal address are marked. In total, we have found 56 postal addresses on these 44 webpages by manually ground-truthing. Among the 44 webpages, 16 contain at least one postal address. The other 28 contain no address at all. However, we use all of them as the test data to evaluate both the detection accuracy and the false alarm rate. The test data set and the ground truth data can be found at http://idke.ruc.edu.cn/wdml/address-truth.zip.

Figure 7. A typical context of a postal address on a webpage.

We first evaluate our approach in terms of the segmentation precision of postal address blocks. Our approach has segmented in total 513 text blocks from the 44 test webpages. The segmentation results on postal address blocks in various situations are listed in Table 1. Among the 56 addresses, 1 has not been segmented, 7 have been segmented as larger blocks containing extra text, 5 have been segmented as smaller blocks than the actual blocks, and 43 have been exactly segmented as the ground truths. Since the address blocks segmented as larger the ground truths usually contains just telephone numbers below the addresses. They do not affect the recognition accuracy. Hence, we also consider this case as correct segmentation. In this sense, our correct segmentation rate is (7+43)/56=89.3%.

Situation	Number
address blocks not extracted	1
address blocks segmented as larger blocks	7
address blocks segmented as smaller blocks	5
address blocks exactly segmented	43

Table 1. Setmentation result of postal address blocks.

All the 513 text blocks segmented have undergone the recognition process and reported 52 postal addresses, among which 50 are the address blocks segmented as exactly equal to or larger than the ground truths,

and 2 are false alarms. Hence, the total address detection accuracy is 50/56=89.3% and the false alarm rate is 2/52=3.8%.

From the experiments, we can see that our approach to postal address detection from webpages can achieve a high precision with a low false alarm rate.

5. Summary and Future Works

In this paper, we propose an approach to postal address detection from webpages, which consists of two steps. The first step uses a vision-based method to split a webpage into text blocks, which can be cue blocks or body blocks. We mainly use the layout formatting information, such as font-size, and so on for this purpose. The second step applies a syntactic pattern recognition method to identify if the body block is a postal address or not. We have summarized a set of syntactic rules (grammars) of common postal address patterns for this purpose. Preliminary experiments show that our approach can achieve a high detection rate and a low false alarm rate.

In future works, we will refine our approach and test it with bigger test data set. We also consider applying this approach for detection of other types of information, such as telephone number, product price, and product description, from webpage.

References

[1] Chen Z., Liu W., and Zhang F. A New Statistical Approach to Personal Name Extraction. In *Proc. International Conference on Machine Learning*, pp. 67-74, Sydney, July, 2002.

[2] Liu Y., Liu W., and Jiang C. User Interest Detection on Webpages for Building Personalized Information Agent. In *Proc. International Conference on Web-Age Information Management*(LNCS, Vol. 3129), pp. 280-287, Dalian, China, 2004.

[3] Meng X., Lu H., et al. Data Extraction from the Web based on Pre-defined Schema. In *JCST, Vol.17 (4)*, pp. 377-388, 2002, 7

[4] Meng X., Hu D., Li C. Schema-Guided Wrapper Maintenance for Web-Data Extraction. In *ACM Fifth International Workshop on Web Information and Data Management (WIDM 2003)*, November 7-8, 2003, New Orleans, Lousiana, USA.

[5] Beeferman D., Berger A., and Lafferty J. Statistical Models for Text Segmentation. *Machine Learning 34*: 177-210, 1999.

[6] An Automatic Method of Finding Topic Boundaries, In *Proc. Annual Meeting of the ACL*, pp. 331- 333, 1994.

[7] Blumenstein M., and Verma B. A Segmentation Algorithm used in Conjunction with Artificial Neural Networks for the Recognition of Real-World Postal Addresses. In *Proc. International Conference*.

[8] Microsoft. About the W3C Document Object Model. 2002. http://msdn.microsoft.com/library/default.asp?url=/workshop/author/dom/domoverview.asp