Classification of News Web Documents Based on Structural Features

Shisanu Tongchim, Virach Sornlertlamvanich, and Hitoshi Isahara

Thai Computational Linguistics Laboratory
National Institute of Information and Communications Technology
112 Paholyothin Road, Klong 1, Klong Luang, Pathumthani 12120, Thailand
shisanu@tcllab.org, virach@tcllab.org, isahara@nict.go.jp

Abstract. The motivation of this work comes from the need of a Thai web corpus for testing our information retrieval algorithm. Two collections of news web documents are gathered from two different Thai newspaper web sites. Our goal is to find a simple yet effective method to extract news articles from these web collections. We explore the use of machine learning methods to distinguish article pages from non-article pages, e.g. table of contents, advertisements. Then, the selected web articles are compared in a fine-grained manner in order to find informative structures. Both steps of information extraction utilize the structural features of web documents rather than the extracted keywords or terms. Thus, the inherent errors of word segmentation, one of the major problems in Thai text processing, do not affect to this method.

1 Introduction

The web has been proved to be a valuable source of information for computational linguistics studies. With the growing number of web documents and online information, *web mining* plays an important role in extracting useful information from the World Wide Web. According to a taxonomy proposed by Cooley *et al.* [1], the term 'web mining' has been used in two distinct ways, namely *web content mining* and *web usage mining*. The first one refers to information discovery on the World Wide Web, whereas the second one describes the research in analyzing the user access patterns from web servers. Our study can be classified to the web content mining category. This study is motivated by the need of a Thai web corpus for testing our information retrieval algorithm. We use Thai newspaper web sites as sources of information. In general, a newspaper web site consists of thousands of web pages. The desired pages are the article pages. Thus, the first goal is to identify which pages are the article pages. The non-article pages, e.g. table of contents, advertisements, opinion or query submission forms, should be screened out. In the second step, the selected pages are analyzed to eliminate non-informative parts of pages, e.g. the navigation bar.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 provides the description of a technique for identifying the article pages. Section 4 presents a technique for analyzing web pages in a fine-grained manner

T. Salakoski et al. (Eds.): FinTAL 2006, LNAI 4139, pp. 153–160, 2006.

in order to eliminate non-informative structures in web documents. Section 5 presents the experimental results and discussion. Finally, Section 6 concludes our work and discusses future research.

2 Related Work

Similarity measurement between web documents is the central idea for web classification. Similarity measurement and Classification can be done on features drawn from web documents. In general, the research in this area can be classified into three groups according to the type of features. The first group utilizes extracted terms and textual information of web documents. The second group is based solely on the structural information of documents. The last group uses a combination of textual information and structural information.

A number of techniques have been proposed based on extracted terms and textual information [2,3,4]. A set of words extracted from web documents are used as features for classification algorithms. In general, the plain text is obtained by removing all HTML tags. Then, the stop words are usually removed from the extracted word list. After this phase, some studies also transfer each word into its stem by using some stemming algorithms. The extracted terms are used to represent web documents and their classes. Typically, only some extracted words are selected as features or attributes for classification algorithms since the extracted word list is usually large and it is impractical for classification algorithms. For the languages which have no explicit word boundary, some word segmentation algorithms are applied to web documents. An example of using a word segmentation algorithm with Chinese web documents before constructing a word list was presented by He *et al.* [5]. In general, the errors from word segmentation are unavoidable. This case also applies to Thai language which perfect word segmentation is hardly achieved. Thus, we intend not to use textual information as features for classification.

Some researchers utilize structural features of web documents for classification [6,7,8]. Joshi *et al.* [6] converted a tree representation of web documents to a simpler representation and measured structural similarity. Cruz *et al.* [7] measured similarity between web pages based on the frequencies of HTML tags. Wong and Fu [8] generalized some knowledge from a hierarchical structure of web documents and used this knowledge to classify web pages.

The last category is to use a combination of textual information and structural information. An example is the article by Tombros and Ali [9]. They experimented the use of three different types of features, namely the textual content from different parts of documents, HTML tag frequencies, and the query terms found in pages.

3 Web Page Classification

We choose the frequencies of HTML tags as structural features for web classification. We adopt the frequencies of tags in percentage from the article by

Cruz *et al.* [7]. Let $T = \{t_1, t_2, ..., t_n\}$ be the collection of n frequent used HTML tags found in a document collection $\mathcal{C}$. Let $m_j(t_i)$ be the number of occurrences of the tag t_i in the document j. The frequency $f_j(t_i)$ (in %) of the tag t_i in the document j can be calculated as follows:

$$f_j(t_i) = \frac{m_j(t_i)}{\sum_{k=1}^{n} m_j(t_k)} \times 100 \tag{1}$$

We define a feature vector $F = \{f_j(t_1), f_j(t_2), ..., f_j(t_n)\}$ as a representation of the document j.

The use of structural information like the tag frequencies for the classification is motivated by the following reasons.

- The construction of feature vectors is simple, fast and straightforward. Thus, it is suitable for a web site with thousands of web pages like a newspaper web site.
- The use of structural information avoids the inherent errors of word segmentation. Unlike the use of textual information, the proposed method does not use the keyword extraction. Thus, word segmentation is unnecessary.
- In general, a newspaper web site contains several categories, e.g. Sport, Politics, Entertainment. The preferred pages are the article pages, no matter what categories they belong to. The structural information should be a better representation than the textual information.

4 Selection of Informative Structures

After selecting article pages from the collected collections, the next task is to eliminate non-informative structures existing in the selected article pages. In general, a web page may contain many information blocks. Some blocks contain information which does not relate to the main content, for example, navigation bars, copyright notices, advertisements, etc. Such information blocks can be regarded as the noisy blocks [10]. If the noisy information blocks have not been eliminated from the collection of web pages, they may affect the evaluation of information retrieval algorithms later. Yi *et al.* [10] pointed out that the elimination of noisy information improves the performance of two data mining tasks.

In this section, we perform a fine-grained analysis on the article pages to estimate the importance of a particular information block. Our intuition is that the noisy blocks tend to appear in many other pages in the same web site, while the main contents are quite unique. To identify which information blocks are likely to be noisy information, the comparisons among pages are performed. The frequency of a particular information block will indicate whether that information block is informative or not.

An HTML document can be modeled as a tree. Figure 1 shows an example of HTML document. By using the relations among tags, a hierarchical structure of this HTML document can be constructed as a tree presented in Figure 2.

```
<html>
<head>
<title>Contact-TCL</title>
</head>
<body>
<table width="800" border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td colspan="2"><h1>Contact Address</h1></td>
  </tr>
  <tr>
    <td width="128"><h2>Address</h2></td>
    <td width="672">Room 224, NECTEC Building,
                    Thailand Science Park 112 Paholyothin Road,
                    Klong 1, Klong Luang, Pathumthani, 12120,
                     Thailand </td>
  </tr>
  <tr>
    <td><h2>Telephone</h2></td>
    <td>(+66)-2564-7990 </td>
  </tr>
  <tr>
    <td><h2>Fax</h2></td>
    <td>(+66)-2564-7992 </td>
  </tr>
  <tr>
    <td><h2>E-mail</h2></td>
    <td><a href="mailto:info.tcllab.org">info@tcllab.org</a></td>
  </tr>
</table>
</body>
</html>
```

Fig. 1. An example of HTML code

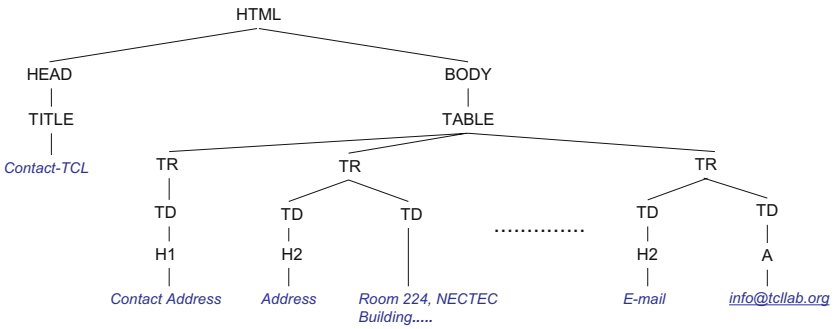

Fig. 2. An example of HTML tag tree

Although the tree representation completely contains the structural information, it is not simple to manipulate. The comparisons among trees represented HTML documents are computational intensive.

To make the tree representation easier to manipulate, the tree structure is converted into a simpler representation. We use the set of paths from the root node to the leaf nodes or the terminal nodes to represent a document. Let m be the number of leaf nodes in the document i, and $L_i = \{l_1, l_2, ..., l_m\}$ be the collection of leaf nodes of the document i. Thus, the number of paths from the root node to the leaf

nodes is equal to m. Let $P_i = \{p_1, p_2, ..., p_m\}$ be the collection of paths of the document i. A path p_j contains the root node, the leaf node (l_j) and all the intermediate nodes between the root node and the the leaf node (l_j). From the figure 2, the first path p_1 can be defined as $\{HTML, HEAD, TITLE, Contact - TCL\}$. Let n be the number of documents. By comparing all path collections $P_i, 1 \leq i \leq n$, we can create the collection of distinct paths, $P_{dist} = \{p'_1, p'_2, ..., p'_N\}$. Let $m(p'_j)$ be the number of occurrences of the path p'_j in all path collections $P_i, 1 \leq i \leq n$. We use the ratio of the number of occurrences of a particular path to the number of documents, $m(p'_j)/n$, to identify whether this path is informative or not. If $m(p'_j)/n$ is less than a predefined threshold value, this path is informative. In this study, the threshold is 0.1.

The transformation of the tree structure to parent-child relations is close to the *bag of tree paths model* presented by Joshi *et al.* [6]. However, they used the path information to compute the similarity of documents rather than identifying informative structures. Another difference is that they discarded the textual information. Thus, every text node is ignored. In contrast, we use both textual information and structural information. We consider paths that have text nodes as the leaf nodes.

5 Experimental Results

In this section, we report the results of the proposed method on two collections of news documents. The first collection of 4497 documents is gathered from the Manager web site[1]. The number of article pages in the Manager collection is 1263. The second collection of 623 documents is harvested from the BangkokBiz web site[2]. In the BangkokBiz collection, 416 pages are article documents.

The first experiment is to explore the use of tag frequencies for page classification. By analyzing web pages in both collections, there are 51 commonly used tags. We create feature vectors of these 51 tags as in the section 3. Three machine learning algorithms are compared, namely Support Vector Machine (SVM), C4.5 and Naive Bayes. The experiment is done on the Weka workbench [11]. The parameters of all learning algorithms are set to their default values. For SVM, the linear kernel is used with the complexity parameter of 1.0. For C4.5, the confidence factor is set to 2.5. The labeled collections of pages are split into two parts, namely 5% are considered as training data and 95% are testing data. Therefore, 225 pages from the Manager collection are used as training data, while 31 pages from the BangkokBiz collection are used as training data.

Tables 1 and 2 show the classification results. Table 1 presents the performance of correctly identifying web pages as article pages, whereas table 2 shows the performance of correctly identifying web pages that are not article pages. On two collections, SVM performs better than the other two algorithms. However, the difference among the algorithms is marginal. Overall, the performance of algorithms on the Manager collection is slightly better than that of the BangkokBiz

[1] http://www.manager.co.th
[2] http://www.bangkokbiznews.com

Table 1. Classification results of different algorithms for article pages

Method	Manager			BangkokBiz		
	Pr	Re	F_1	Pr	Re	F_1
SVM	0.976	0.997	0.986	0.968	0.99	0.979
C4.5	0.946	0.967	0.957	0.951	0.913	0.931
Naive Bayes	0.892	0.973	0.931	0.985	0.988	0.986

Table 2. Classification results of different algorithms for non-article pages

Method	Manager			BangkokBiz		
	Pr	Re	F_1	Pr	Re	F_1
SVM	0.999	0.991	0.995	0.978	0.932	0.954
C4.5	0.987	0.978	0.983	0.931	0.901	0.864
Naive Bayes	0.989	0.954	0.971	0.974	0.969	0.971

collection. The results suggest that the use of tag frequencies is feasible and sufficient for our classification problem. Even using a small number of training examples (like 31 pages for the BangkokBiz collection), only a small number of pages are wrongly classified. Moreover, we have illustrated the feasibility of this technique for the classification task by using default parameter values. The algorithms achieve high precision without the need of parameter tuning.

After selecting which pages are article pages, the article pages are compared by using the proposed idea presented in the section 4. We randomly select two sets of 100 article pages from both collections. The first set is obtained from the Manager collection, whereas the second set is acquired from the BangkokBiz collection. The algorithm is implemented in Java. We use HTMLParser[3] to parse the HTML documents. Each document is converted to a tree. Then, the tree representation is transformed to a set of paths. There are 2642 and 2409 distinct paths for the first set and the second set respectively. It is interesting to note that the same path may occur in two or more times even in a single document. Therefore, the number of occurrences of a particular path may be greater than the number of documents. In the set of 100 Manager articles, the most frequent used path is found 2784 times. In contrast, the most frequent used path in the set from the BangkokBiz is found 97 times. Another observation is that the vast majority of paths are unique. They appear only one time in the whole collection. There are 2234 paths that are unique for the first set and 2100 paths for the second set. According to our intuition, these paths are likely to be informative structures.

In both news collections, most article pages have small discussion boards at the end of articles. The discussion boards allow readers to submit their opinions about

[3] http://htmlparser.sourceforge.net/

news articles. The textual information in these discussion boards is problematic for our analysis. Although the textual information of opinion boards loosely relates to the articles, they are not parts of the main contents. They can be considered as noisy information. It is not trivial to detect these parts by using the proposed method since they are quite unique. We will leave this problem to the future work. In this study, these parts are not considered in our experiment.

The 2642 extracted paths of the first set and the 2409 paths of the second set are classified by using the threshold value of 0.1. Then, they are manually checked by hand. Note that the numbers of non-informative blocks are 8.59% and 8.26% for the first set and the second set respectively. The results are shown in Table 3. From the results, the recalls for classifying non-informative structures are 0.423 and 0.839 for the first set and the second set respectively. The recalls for identifying informative structures for both sets are all 1.0. This means that a number of false positives occur, but no false negative. All structures classified as non-informative structures are correct. However, some non-informative structures are still ambiguous. The number of occurrences is not significant enough for the algorithm to detect them as non-informative structures.

Table 3. Classification results for informative and non-informative structures

	Manager			BangkokBiz		
	Pr	Re	F_1	Pr	Re	F_1
Informative Structure	0.948	1.000	0.973	0.986	1.000	0.993
Non-informative Structure	1.000	0.423	0.594	1.000	0.839	0.913

6 Conclusions and Future Work

We have proposed a method to extract some useful textual information from the newspaper web sites. The goal is to construct a Thai web corpus from news articles. The proposed method works in two steps. The first step is to select the article pages from the collections of web documents. To avoid the problem of word segmentation, our proposed method uses only the structural information, namely the tag frequencies. SVM performs better than the other two classifiers. However, the difference among the algorithms is not significant. The second step is to eliminate noisy information in the selected web articles. The results show that the majority of structures are correctly classified. However, there is still a problem with discussion boards. We leave this problem to the future work.

There are several possible extensions to this study. We are currently using the Thai web corpus from this study to examine our information retrieval algorithm. The results will be compared with the use of news web pages without any preprocessing. The second one is to explore a method to detect and eliminate the noisy information of discussion boards. The third one is to use other information for detecting noisy information.

References

1. Cooley, R., Mobasher, B., Srivastava, J.: Web mining: Information and pattern discovery on the world wide web. In: ICTAI. (1997) 558–567
2. Sun, A., Lim, E.P., Ng, W.K.: Web classification using support vector machine. In Chiang, R.H.L., Lim, E.P., eds.: WIDM, ACM (2002) 96–99
3. Holden, N., Freitas, A.A.: Web page classification with an ant colony algorithm. In Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Guervós, J.J.M., Bullinaria, J.A., Rowe, J.E., Tiño, P., Kabán, A., Schwefel, H.P., eds.: PPSN. Volume 3242 of Lecture Notes in Computer Science., Springer (2004) 1092–1102
4. An, A., Huang, Y., Huang, X., Cercone, N.: Feature selection with rough sets for web page classification. In Peters, J.F., Skowron, A., Dubois, D., Grzymala-Busse, J.W., Inuiguchi, M., Polkowski, L., eds.: T. Rough Sets. Volume 3135 of Lecture Notes in Computer Science., Springer (2004) 1–13
5. He, J., Tan, A.H., Tan, C.L.: Machine learning methods for chinese web page categorization. In: ACL'2000 2nd Workshop on Chinese Language Processing, Hongkong, China (2000) 93–100
6. Joshi, S., Agrawal, N., Krishnapuram, R., Negi, S.: A bag of paths model for measuring structural similarity in web documents. [12] 577–582
7. Cruz, I.F., Borisov, S., Marks, M.A., Webb, T.R.: Measuring structural similarity among web documents: Preliminary results. In Hersch, R.D., André, J., Brown, H., eds.: EP. Volume 1375 of Lecture Notes in Computer Science., Springer (1998) 513–524
8. Wong, W.C., Fu, A.W.C.: Finding structure and characteristics of web documents for classification. In: ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. (2000) 96–105
9. Tombros, A., Ali, Z.: Factors affecting web page similarity. In Losada, D.E., Fernández-Luna, J.M., eds.: ECIR. Volume 3408 of Lecture Notes in Computer Science., Springer (2005) 487–501
10. Yi, L., Liu, B., Li, X.: Eliminating noisy information in web pages for data mining. [12] 296–305
11. Witten, I., Frank, E.: Data Mining: Practical machine learning tools and techniques. 2nd edn. Morgan Kaufmann, San Francisco (2005)
12. Getoor, L., Senator, T.E., Domingos, P., Faloutsos, C., eds.: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003. In Getoor, L., Senator, T.E., Domingos, P., Faloutsos, C., eds.: KDD, ACM (2003)