Table Recognition Using Attribute Likelihood Computed from Words

Tsubasa Kitayama, Kazutaka Shimada and Tsutomu Endo

Department of Artificial Intelligence,
Kyushu Institute of Technology,
Iizuka, Fukuoka 820-8502 Japan
{t_kitayama, shimada, endo}@pluto.ai.kyutech.ac.jp

Abstract. Tables are an efficient way to express relational information. Understanding tables is an important task for various applications. In this paper, we describe a method of recognition of table structures. Here the table structures are the relations of attributes and values in a table. For this process, we employ attribute likelihood which is computed from words in tables. First we compute weights of words from training data. Next our method identifies an attribute row (or column) using the weights. Experimental results show the effectiveness of our method.

1 Introduction

As the World Wide Web rapidly grows, a huge number of online documents are easily accessible on the Web. Finding information relevant to user needs has become increasingly important. Information on the WWW is in the form of not only text but also images and tables. Although tables are structured information, i.e., attribute-value pairs, most of conventional information retrieval systems treat tables as text. Since tables are an efficient way to express relational information, table extraction is a significant task for web mining, QA systems, summarization and so on [2, 3, 4, 7].

In general, tables on the WWW are written in a <TABLE> tag. However, the presence of the <TABLE> tag in an HTML document does not necessarily indicate the presence of tables. Less than 30% of HTML <TABLE> tags are real tables in one particular domain [1]. There are several approaches to treat HTML-based tables. Although Chen et al. have reported a method for extracting tables from HTML documents, they employed heuristic rules for table extraction [1]. Constructing rules by handwork is costly. Wang et al. have evaluated a table extraction task with machine learning based approaches: decision tree learning and SVMs [10]. We have reported a method for table extraction based on Bayes' rule [8] and Transductive SVMs [9].

The purpose of most of the previous work is to detect real tables in HTML documents. For the development of systems handling tables appropriately we need to identify relations in tables. In this paper we define the relation as attribute-value pairs. Masuda et al. have reported a table structure recognition method based on content features and layout features [5]. For example, one of the content features is unit features such as “feet” and “inch”. The appropriate selection of these features by handwork is costly. Yoshida et al. have proposed a method for table structure recognition using an unsupervised method that was achieved by utilizing the EM algorithm [11]. However, their method can not recognize all types of the location of attributes and values because the number of table structures they defined was only 9.

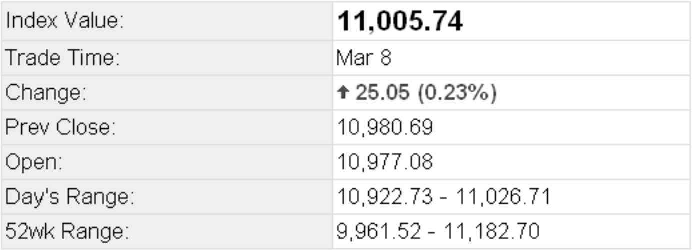
In this paper, we propose a method of recognition of table structures. For this method, we employ attribute likelihood which is computed from words in tables. First we compute weights of words from training data. The weight denotes the polarization of a word: a word is an attribute or a value. Next our method identifies an attribute row (or column) using the weights. This method classifies <TABLE> tags into real tables and <TABLE> tags used for layout and also can recognize all kinds of table structures.

2 Table Recognition

In this paper, real tables denote that <TABLE> tags consist of attributes and values. Figure 1 shows an example of a real table. In Figure 1, the 1st column is the attribute area and the 2nd column is the value area.

2.1 Weight of words

We use the weights of words appearing in tables for table structure recognition. The weights are computed from training data. The weight denotes the polarization of a word. If the weight of a word is large, the



Index Value:	**11,005.74**
Trade Time:	Mar 8
Change:	**↑ 25.05 (0.23%)**
Prev Close:	10,980.69
Open:	10,977.08
Day's Range:	10,922.73 - 11,026.71
52wk Range:	9,961.52 - 11,182.70

Fig. 1. An example of a real table.

word appears frequently in attribute rows and columns. For this process, we need a corpus containing tagged cells (“attribute” or “value”).

The algorithm consists of the following steps:

1. Extract word strings in table cells from training data.
2. Divide the word strings into words. In order to obtain the words, we use the Japanese morphological analyzer ChaSen[1] .
3. For each word, the weight is computed as follows:
$$P_{word}(w) = \frac{freq^{w}_{attr}}{freq^{w}_{table}} \tag{1}$$
where $freq^{w}_{attr}$ is the frequency of a word w in attribute rows or columns. $freq^{w}_{table}$ is the frequency of the word w in training data.

$P_{word}(w)$ is a probability that the word w occurs in attribute rows or columns.

2.2 Attribute likelihood

We estimate attribute likelihood of each row and column by using the word weights computed by Eq. (1). First we compute the weight of each cell and then the weight of all cells. Finally we compute the influence of each row and column. If the attribute likelihood computed from the influence is more than or equal to a threshold, the <TABLE> tag is a real table and the row or column is an attribute. If not, our method judges that the <TABLE> tag is used for layout.

The algorithm consists of the following steps:

a) Compute the weight of each cell. It is computed as follows:
$$P_{cell}(x) = \frac{\sum_{w \in x} P_{word}(w)}{N_x} \tag{2}$$
where x and N_x is a cell and the number of words in the cell x respectively. If a cell does not contain any words extracted in Section 2.1, the weight of the cell ($P_{cell}(x)$) is 0.

b) Compute the weight of a table as follows:
$$P_{table}(t) = \frac{\sum_{x \in t} P_{cell}(x)}{N_t} \tag{3}$$
where t and N_t is a <TABLE> tag and the number of cells in the <TABLE> tag t respectively.

c) Compute the influence of each row and column. We compute two values as follows:
$$P_{row}(i) = \frac{\sum_{x \in t, x \notin i} P_{cell}(x)}{N^{i}_{t}}, \quad P_{col}(j) = \frac{\sum_{x \in t, x \notin j} P_{cell}(x)}{N^{j}_{t}} \tag{4}$$
where i and j are a row and a column respectively. N^{i}_{t} and N^{j}_{t} are the total number of cells from which the row i and the column j were removed. If i (or j) is an attribute in t, $P_{row}(i)$ (or $P_{col}(j)$) is small value.

d) Detect the minimum value P_{min} from all P_{row} and P_{col}. Finally we compute the maximum value of attribute likelihood.
$$P_{max} = P_{table} - P_{min} \tag{5}$$
If $P_{max} > Th$, the row or the column is an attribute of the table t. Th is a threshold value.

Figure 2 shows an example of the process. The values in the cells in Fig. 2 denote the values of $P_{cell}(x)$.

[1] http://chasen.naist.jp/hiki/ChaSen/

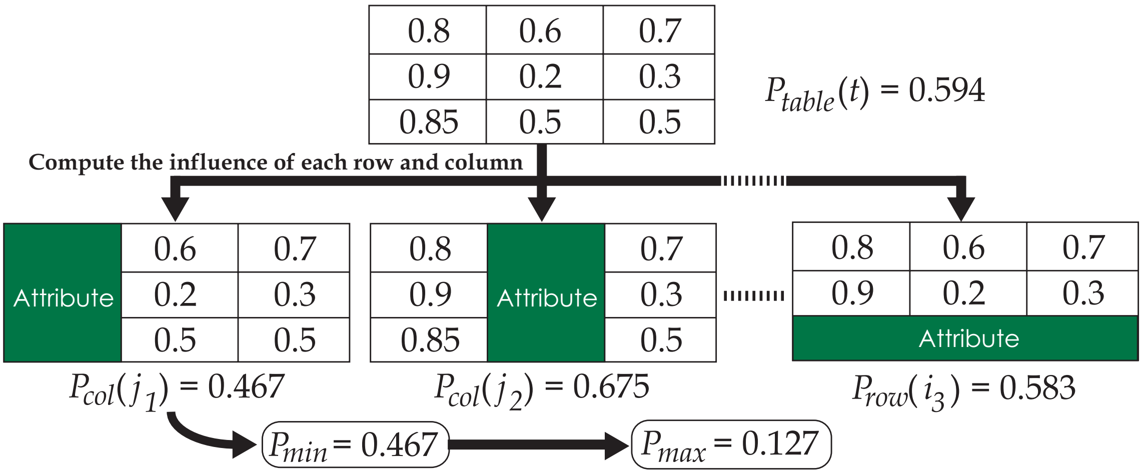


Fig. 2. An example of the process.

3 Experiment

3.1 Dataset

We evaluated the appropriateness for the proposed method with a dataset. The data was extracted from several manufacturer's sites by a file-downloading software. The total number of pages is 10935. We extracted 3229 <TABLE> tags from the pages. These <TABLE> tags did not contain nested tables, <A> tags, <IMG> tags, COLSPAN and ROWSPAN attributes[2] . The tables in the data contained 2 or more rows and columns at least. The dataset for this evaluation was constructed by 1000 <TABLE> tags extracted from the <TABLE> tags randomly. The dataset contains real tables and <TABLE> tags for layout. We evaluated our method with 5-fold cross validation.

Our method needs a threshold Th in the detection process of the maximum value of attribute likelihood. We compared several threshold by handwork with a threshold calculated from training data automatically. The threshold by mechanical work was computed as follows:

$$Th = \frac{\text{\# of words that belong to attributes in training data}}{\text{\# of words in training data}} \tag{6}$$

3.2 Experimental results

Table 1 shows the experimental result. The "simple method" in Table 1 was a simplified method. This method used the weights computed in the process **a)** in the Section 2.2. Next the method computed the following scores of each row and column.

$$S_{row}(i) = \frac{\sum_{x \in i} P_{cell}(x)}{\text{\# of cells in a row } i}, \quad S_{col}(j) = \frac{\sum_{x \in j} P_{cell}(x)}{\text{\# of cells in a column } j} \tag{7}$$

Finally the method computed $P_{max} = \max(S_{row}, S_{col})$.

The accuracy of the proposed method outperformed the simple method and the related work. The best accuracy was 84.5 % in the case that the $Th = 0.07$. The threshold by handwork outperformed that computed from the training data automatically. However, the best threshold was close to the mechanical threshold. Hence the value computed from training data is useful to determine the best threshold by handwork.

One of the reasons that the accuracy of related work was low is the problem of the features they employed. The accuracy of their method depends on the features that they decided by hand. Their features were not suitable for our dataset. Moreover their method is not appropriate to recognize small tables, especially 2 × 2 tables. These results show the effectiveness of our method.

[2] The <A> tag is the anchor tag for hypertext documents. The <IMG> tag is used to insert images within text. The ROWSPAN and COLSPAN attributes indicate how many rows or columns this cell overlaps.

Table 1. The result of the experiment

Th	by handwork							from training data	simple method	Related work [5]
	0.04	0.05	0.06	0.07	0.08	0.09	0.1	0.0786	0.15	
Accuracy	67.2	76.9	82	**84.5**	82.5	81.7	80.7	82.6	78.4	56.6

One of the problems of our method is the number of training data. For constructing a strong table recognizer, our method requires a large amount of training data. However, gathering the training data by handwork is costly. To solve this problem, Yoshida et al. employed EM algorithm [11]. Ohmae et al. have reported a method of automated extraction of attribute words in tables [6]. The method is based on a supposition that attribute words frequently appear in the 1st column and row of the tables. With regard to their experiment, they obtained 77.4 points in F-measure without training data. We evaluated this method with our experimental dataset. However, the accuracy of the table recognition was very low (less than 30%). The reason for the low accuracy in our dataset was that they treated only data that was retrieved with some queries in their experiment. In other words, their method is an effective solution for domain-specific data. This result shows that we need to consider some modifications for applying the approach to our method.

4 Conclusion

In this paper, we proposed a method of recognition of table structures. For this process, we used attribute likelihood which is computed from words in tables. This method classified <TABLE> tags into real tables and <TABLE> tags used for layout and also recognized table structures, i.e., attribute-value pairs. The experimental results show the effectiveness of the proposed method.

Our future work includes (1) the expansion of our method for tables that contain 2 or more attribute rows or columns and (2) the reduction of training data with machine learning methods and heuristic models of tables.

References

1. H. H. Chen, S. C. Tsai and J. H. Tsai: Mining tables from large scale HTML texts, Proc. of COLING2000, pp. 166-172, 2000.
2. Gen Hattori, Kazunori Matsumoto, Fumiaki Sugaya, "Understandable and Discriminative Object Label Extraction Scheme for Summarizing Table-Type Information", Transactions of IEICE, D-I, Vol. J88-D-I, No. 9, pp. 1467-1476, 2005 (in Japanese).
3. Matthew Hurst, "Layout and language: Challenges for table understanding on the web", Proceedings of Workshop on Web Document Analysis, WDA01, pp. 27-30, 2001.
4. Kumi Itai, Atsuhiro Takasu and Jun Adachi, Information extraction from HTML pages and its integration, Proceedings of the 2003 Symposium on Application and the Internet Workshops (SAINT03), pp. 276-281, 2003.
5. Hidetaka Masuda, Shuichi Tsukamoto and Hiroshi Nakagawa, "Recognition of HTML Table Structure", The First International Joint Conference on Natural Language Processing (IJCNLP-04), pp.183-188, 2004.
6. Nobuhiro Ohmae and Koichi Kise, "Automatic recognition of attributes from tables in web pages", Technical Report of IPSJ, NL-171, pp. 43-48, 2006 (in Japanese).
7. Kazutaka Shimada, Atsushi Fukumoto and Tsutomu Endo, Information extraction from personal computer specifications on the Web using a user's request, IEICE Transactions on Information and Systems, E86-D, No. 8, pp. 1386-1395, 2003.
8. Kazutaka Shimada, Koji Hayashi and Tsutomu Endo, "Keyword and Weighting for Product Specifications Extraction", Proceedings of PACLING 2003, pp. 285-293, 2003.
9. Kazutaka Shimada, Koji Hayashi and Tsutomu Endo, "Product Specification Extraction Using SVM and Transductive SVM", Journal of Natural Language Processing, Vol.12, No.3, pp. 43-66, 2005 (in Japanese).
10. Yalin Wang, Jianying Hu, "A Machine Learning Based Approach for Table Detection on The Web", The Eleventh International World Wide Web Conference, 2002.
11. Minoru Yoshida, Kentaro Torisawa, Jun'ichi Tsujii, "Integrating Tables on the World Wide Web", Transactions of the Japanese Society for Artificial Intelligence, 19(6). pp. 548-560, 2004.