Automatic Web-Page Classification by Using Machine Learning Methods

Makoto Tsukada, Takashi Washio, Hiroshi Motoda

Institute of Scientific and Industrial Research,
Osaka University
Mihogaoka, Ibaraki, Osaka 567-0047, JAPAN
{tsukada,washio,motoda}@sanken.osaka-u.ac.jp

Abstract. This paper describes automatic Web-page classification by using machine learning methods. Recently, the importance of portal site services is increasing including the search engine function on World Wide Web. Especially, the portal site such as for Yahoo! service which hierarchically classifies Web-pages into many categories is becoming popular. However, the classification of Web-page into each category exclusively relies on man power which costs much time and care. To alleviate this problem, we propose techniques to generate attributes by using co-occurrence analysis and to classify Web-page automatically based on machine learning. We apply these techniques to Web-pages on Yahoo! JAPAN and construct decision trees which determine appropriate category for each Web-page. The performance of this proposed method is evaluated in terms of error rate, recall, and precision. The experimental evaluation demonstrates that this method provides high accuracy with the classification of Web-page into top level categories on Yahoo! JAPAN.

1 Introduction

At present, the number of Web-pages on World Wide Web are increasing significantly. The task to find Web-pages which present information satisfying our requirements by traversing hyperlinks is difficult. Therefore, we use search engines frequently on the portal site. There are two kinds of search engines. i.e., directory-style search engines such as Yahoo! JAPAN[1] and ISIZE[2], and robot-style ones such as goo[3], excite[4] and altavista[5]. The latter displays the lists of Web-pages which contain input keywords without checking themes characterizing respective Web-pages. For this reason these search engines are likely to provide misdirected Web-pages. On the other hand, in directory-style search engines, Web-pages stored in a database are classified with hierarchical categories compatible with their themes in order. This enables us to obtain Web-pages including information that meets our purpose by not only following input keywords but also traversing hyperlinks classifying Web-pages into categories in systematic order.

However, directory-style search engines at present require that man power classifies a large number of Web-pages into each appropriate category with their

themes. Therefore, this task costs much time and care. This indicates that the task to classify ever increasing number of Web-pages becomes increasingly difficult. For example, Yahoo! JAPAN, a typical directory-style search engine, receives tremendous amount of requests to enter Web-pages into the database daily. It then occasionally takes several weeks to determine an appropriate category for each theme of Web-page, and confirms this entry in the database. We deem that automatic Web-page classification affords much easier construction of the database, and contributes to reductions in costs and man power successfully.

In the past, a considerable number of studies has been made on text classification of newspaper articles, based on k-nearest neighbor, support vector machine and so on[7][8][9][10][11][12]. In addition, many comparative studies of these methods have so far been made[7][8][9]. However, no studies in the above aim to classify Web-pages and to apply supervised learning in terms of the classification based on man power. In addition, although there is some research of supervised learning for Web-page classification in Yahoo! U.S.A.[6][13], only a few attempts at this kind of research have been made. Furthermore, no one has ever tried to classify Web-page in Japanese search engines automatically by supervised learning. Under these circumstances, our study aims at developing a technique by which to classify Web-page automatically by supervised machine learning using a man-made class attribute. In addition, we develop a method for attribute generation by using co-occurrence analysis. We then apply these techniques to classify Web-pages into top-level categories included in the index of Yahoo! JAPAN. By constructing decision trees, we evaluate them in terms of three criteria, i.e., error rate, recall and precision.

This paper is organized as follows. Our developed technique is explained in section 2. We apply these methods to classification of Web-pages in Yahoo! JAPAN and assess the accuracy of decision trees in section 3. Finally the paper ends with the concluding remarks in section 4.

2 The Proposed Methods

2.1 Extraction of Nouns from Web-pages

Fig. 1. The structure of hyperlinks in Yahoo! JAPAN

In the directory-style search engine, a great deal of Web-pages classified and registered in the database are interconnected with hyperlinks and make a hierarchical tree structure to improve usability. A node in this structure indicates a

category whose name accords with themes represented by Web-pages in it. As Fig. 1 illustrates, Yahoo! JAPAN has some subcategories such as "*Gambling*" below the parent category of "*Recreation and Sports*". Furthermore, there can be hyperlinks from one category to some other categories or to top-level categories. Therefore a Web-page may be classified into multiple categories simultaneously.

Fig. 2. Download of Web-pages according to the categories in the top page

We focus on the top-level page on Yahoo! JAPAN [13] and assign distinct class labels to some top-categories in this page. Web-pages corresponding to each class are downloaded separately as Fig. 2 illustrates,

$$class_c \leftrightarrow \{Page_1^c, \ldots, Page_i^c, \ldots\},$$

where $Page_i^c$ indicates i-th Web-page labeled $class_c$.

We then delete all of tags such as <A HREF > and <IMG > from documents of Web-pages described by Hyper Text Markup Language and extract all nouns by morphological analysis. Morphological analysis is a technique to divide a sentence into parts of speech such as nouns, adjective and adverb. We suppose that some nouns in the sentence are typical to the theme of Web-page among others. For this reason we extract some nouns from documents of Web-pages. We refer to each noun as an *item* and form a *transaction* $page_i^c$ which consists of some items $word_{ij}^c$ as follows.

$$Page_i^c =< word_{i1}^c, \ldots, word_{ij}^c, \ldots >,$$

where $word_{ij}^c$ indicates the j-th item extracted from $Page_i^c$ labeled as $class_c$. In addition, we integrate them into a set of transactions for each class label.

Fig. 3. The set of transactions each consisting of some items

However, the system of Japanese morphological analysis abstracts as nouns to characterize Web-pages respectively not only the stems of nouns but also the desinences to obscure the meaning of nouns. Definitely their items don't imply distinct meanings. Therefore we eliminate these desinences from all transactions.

In addition, the technical terms specialized in WWW's field are insensitive to a theme of Web-page, and worthless as items which reflect their themes. For example, "*hyperlink*" is exceptionally popular in WWW's field and used frequently regardless of the Web page class. We call a list containing such a meaningless term as *stoplist*. The words "*hyperlink*", "*tag*", "*page*", "$form$" and "$frame$" are the members of the stoplist. Additionally, insignificant terms such as "*thing*" and "*something*" that may appear in the sentence can not represent the theme of Web-page in isolation. We select this kind of nouns as objectively as possible and add these nouns to the stoplist. Finally we eliminate items in the stoplist from all transactions and construct more refined transactions $Page_i^c$.

2.2 Generation of Attributes

We generate attributes to design tabular data from Web-pages by applying *basket analysis* typical of association analysis and well-known in the field of data mining.

Fig. 4. Derivation of frequent itemsets common to all classes above a *support* level

Basket analysis targets a set of transactions consisting of a set of items. The first step of basket analysis is to derive itemsets having *support* greater than a user specified threshold. The support of an itemset I means how frequently I appears, and it is defined as the ratio of the number of transactions including the itemset to the total number of transactions. Itemsets having support greater than its threshold "*minimum support*" are called "$frequent\ itemsets$", and the basket analysis generates all frequent itemsets. It is known that Apriori algorithm[15] efficiently extracts all frequent itemsets from the massive transaction data. The second step of the basket analysis is to generate association rules having $confidence$ greater than a user specified threshold "$minimum\ confidence$". An association rule $B \Rightarrow H (B \cap H = \emptyset)$ is characterized by support and confidence. They are defined follows :

$$sup(B \Rightarrow H) = sup(B \cup H), \quad conf(B \Rightarrow H) = \frac{sup(B \cup H)}{sup(B)}$$

We apply the first step of basket analysis and regard frequent itemsets extracted from Web-pages as the attributes which reflect the features of Web-pages for each class label. This is based on the simple assumption that the set of nouns characterizing the Web-pages occurs very frequently. We specify the minimum support which is common to the transaction data of all classes in advance. A concrete instance is presented in Fig. 4. Next, we merge frequent itemsets for respective classes into a set of attributes as follows.

$$\{ \underbrace{Itemset_1^1, Itemset_2^1}_{frequent\ itemset\ :\ class_1} , \ldots, \quad \underbrace{Itemset_l^c, \ldots}_{frequent\ itemsets\ :\ class_c} \quad , \ldots\},$$

where $Itemset_l^c$ indicates the l-th-$frequent\ Itemset$ extracted from the sets of transactions labeled $class_c$. These attributes are numbered as $Attribute_1$,$Attribute_2$, ...,$Attribute_l$,... in a sequential order. Then, the sub-data D_c composed of transactions labeled $class_c$ is constructed as depicted in Table 1 where every $flag_{mn}^c$ is represented as

$$flag_{mn}^c = \begin{cases} 1 : \forall Itemset_n^c \subset page_m^c \\ 0 : \text{the others} \end{cases}$$

This serves to predict the class of new examples by verifying whether specific nouns exist in the document of the Web-page or not. We repeat this procedure across all classes, and integrate D_c of every class into a whole data $Data = \{D_1, \ldots, D_C\}$ where C is the number of classes.

Table 1. A sub-data set D_c for $class_c$

	$Attribute_1$	...	$Attribute_n$	...	class
$Page_1^c$	$flag_{11}^c$	...	...	...	$class_c$
...	...	...	...	...	$class_c$
$Page_m^c$	...	...	$flag_{mn}^c$	...	$class_c$
...	...	...	...	...	$class_c$

2.3 Binary Class

We divide the whole data $Data$ into a set of local data $Data_c$, each having examples of binary classes: positive examples of $class_c$ and its negative examples. The main reason is that decision tree learning for data having binary classes provides higher accuracy than multiple classes.

2.4 Application of Decision Tree Learning

Once $Data_c$ is obtained for each class, a decision tree learning technique C4.5[14] is applied for the classification of Web-pages. Decision tree algorithms begin with a set of examples and create a tree data structure that can be used to classify new examples. Each node of a decision tree contains a test, the result of which is used to decide which branch to follow from that node. The leaf nodes contain class

Fig. 5. Division into data having binary class

labels instead of tests. When a test example reach a leaf node, the decision tree classifies it using the label stored there. A decision tree is inferred by growing it from the root downward and greedily selecting the next best attribute for each new branch added to the tree. C4.5[14] uses a statistical criterion called *gain ratio* to evaluate the "goodness" of a test.

Once the tree is obtained, C4.5 algorithm applies *n-fold cross-validation* to evaluate the error rate of the tree. This method divides all examples into n subsets of approximately equal size. Each time one of the n subsets is used as a set of testing examples and the other $n-1$ subsets are put together to form a set of training examples. The same trial is repeated n times. These n trials present the average error rate properly, and robust evaluation of the learned decision trees comparatively, although we specifically need to generate attributes and evaluate the accuracy of decision trees respectively after dividing all examples into n subsets.

Decision trees constructed by C4.5 can provide a set of comprehensive rules to classify new examples as described later. These rules are clearly described in the form of tests and the results derived from them. This is an advantage of our approach which uses rule-based inductive classification.

3 Experimental Evaluation

3.1 Experimental Settings

We performed the experiments on the classification of Web-pages on 5 domains of 14 top-categories in Yahoo! JAPAN: "$Arts$ & $Humanities$", "$Business$ & $Economy$", "$Education$", "$Goverment$" and "$Health$". We randomly downloaded 200 Web-pages per category, and thus the total number of the Web-pages for the experiments is 1000.

Next, a morphological analysis was applied to the data of each Web-page by using the system "*chasen*"[16] developed at Nara Institute of Science and Technology, and its set of noun keywords $Page_i^c$ is derived. We generated meaningful attributes based on the 1000 $Page_i^c$ data by applying basket analysis under three support levels, 10%, 20% and 30%, and obtained three data sets of $Data$: Sup10, Sup20 and Sup30 respectively. Finally we constructed a decision tree of each class

using each data set by C4.5[14]. The performance of each tree is evaluated by 4-fold cross-validation

3.2 Performance Measures

The performance of the induced classifier is evaluated in terms of *Error rate*, *Recall*, and *Precision*. First, we define *Error rate* as follows.

$$Error\ rate = \frac{the\ number\ of\ all\ testing\ examples\ classified\ erroneously}{the\ number\ of\ all\ testing\ examples}$$

Error rate denotes the rate of both positive and negative testing example classified erroneously by a decision tree. The lower rate represents the higher accuracy of the decision tree.

In addition, we define *Recall* and *Precision* used in the evaluation of a information retrieval system frequently.

$$Recall = \frac{the\ number\ of\ testing\ examples\ classified\ correctly\ as\ positive}{the\ number\ of\ positive\ testing\ examples}$$

$$Precision = \frac{the\ number\ of\ testing\ examples\ classified\ correctly\ as\ positive}{the\ number\ of\ testing\ examples\ classified\ as\ positive}$$

The less the leak of classification from positive examples is, the larger *Recall* is, and the smaller the classification error of positive examples is, the larger *Precision* is. Both *Recall* and *Precision* are the indicators taking only positive examples into consideration. Therefore they qualify themselves for the reasonable evaluation of how correctly the decision trees can provide testing examples with the positive decision of the class labels. In the experiment of 4-fold cross-validation, the mean value of each measure over the 4 times validations is used for the evaluation instead of the value of individual validation.

3.3 Results

Table 2 shows the results of the evaluation: means of *Error rate*, *Recall*, and *Precision* for the decision trees. The first column shows the data set labels, the second and the third columns of the upper half shows the number of attributes generated and the minimum support level.

Table 2 indicates that the values of *Precision* are higher than those of *Recall* for each category. Generally speaking, the values of *Recall* and *Precision* have a trade off relation by their definitions. Tabel 2 shows that *Precision* is much better than *Recall* in the trade off for the higher minimum support values. The values of *Error rate* almost lies between 8% and 16%. The results also show a tendency that the values of *Error rate* are lower for the lower support values. In case of lower minimum support, the decision tree can have better accuracy, i.e., lower *Error rate* and higher *Recall*, because it is induced from larger number of attributes, i.e., more information on the given data. On the other hand, the decision tree uses only a limited number of significant attributes under higher minimum support, and this effect increases the value of *Precision*, because the

Table 2. Means for decision trees classfying examples(%)

data	attribute	minsup (%)	"*Arts & Humanities*"			"*Business & Economy*"		
			Error rate	*Recall*	*Precision*	*Error rate*	*Recall*	*Precision*
Sup10	823	10	12.6	50.5	79.2	14.3	56.5	67.6
Sup20	78	20	13.3	44.0	80.8	15.0	45.5	69.6
Sup30	19	30	13.9	32.0	95.3	13.6	45.5	77.4

data	"*Education*"			"*Government*"			"*Health*"		
	Error rate	*Recall*	*Precision*	*Error rate*	*Recall*	*Precision*	*Error rate*	*Recall*	*Precision*
Sup10	8.30	69.0	86.7	13.8	45.5	76.1	8.90	65.0	87.2
Sup20	10.9	65.2	77.5	14.2	38.5	80.4	16.1	46.6	64.7
Sup30	10.4	57.5	86.6	14.5	32.5	86.7	15.3	29.3	83.0

Table 3. Concrete instances of the attributes(in case of Sup20)

Arts & Humanities	*Business & Economy*	*Education*	*Government*	*Health*
illustration	enterprise	success	society	research
renewal	business	classroom	politics	life
image	guide	school	policy	age
reproduction	month	learning	election	environment
without-notice	information	education	opinion	medical
{without-notice, reproduction}	{month, information}	{learning, education }	activity	health
...	...	...	...	...

Web-pages characterized by the significant attributes are selected for each class.

Table 3 shows some concrete instances of the attributes derived from the transactions of Web-pages. Notation {A,B..} in Table 3 represents an attribute which is a frequent itemset consisting of multiple nouns. It is clear that specific nouns tends to appear in the Web-pages that belong to a specific class. Moreover, some specific combinations of nouns characterizes the Web-pages of each class. Thus, the application of attribute generation based on basket analysis has a contribution to provide some information for the inductive classification.

We present a concrete decision tree under the conditions of data: Sup20 and class: *Arts & Humanities*. `‘‘Arts & Humanities’’` labeled on the leaf means that the conclusion is classified as "*Arts & Humanities*". On the contrary, `non-‘‘Arts & Humanities’’` means the conclusion is classified as not "*Arts & Humanities*". The decision tree contains some decision nodes conditioned by the combinations of multiple nouns. This clarifies that the attributes consisting of multiple nouns have some contribution to the classification.

```
illustration = 1: ‘‘Arts & Humanities’’
illustration = 0:
|   guide = 1: non-‘‘Arts & Humanities’’
|   guide = 0:
```

```
|   |   {without-notice,reproduction} = 0: non-‘‘Arts & Humanities’’
|   |   {without-notice,reproduction} = 1:
|   |   |   contents = 1: non‘‘Arts & Humanities’’
|   |   |   contents = 0:
|   |   |   |   {month,information} = 1: non-‘‘Arts & Humanities’’
|   |   |   |   {month,information} = 0:
|   |   |   |   |   election = 0: ‘‘Arts & Humanities’’
|   |   |   |   |   election = 1: non-‘‘Arts & Humanities’’
```

A simple rule derived from this decision tree by tracing a branch is

```
illustration = 1 ->  ‘‘Arts & Humanities’’.
```

When a certain Web-page contains the noun “illustration”, we can recognize that this Web-page belongs to the category of *Arts & Humanities* from this rule.

```
illustration = 0
guide = 0
{without-notice,reproduction} = 0
->  non-‘‘Arts & Humanities’’
```

This rule is for the other class. When a certain Web-page doesn't contain the nouns “illustration”,“guide” and the combination of “without-notice” and “reproduction” at all, this Web-page is concluded not to belong to the category of “*Arts & Humanities*”.

4 Discussion and Conclusion

In the research of text categorization, methods of attribute generation has been focused, and regarded as feature selection. In this field, respective words which occur in documents are assigned to features respectively. Many researchers in text categorization has attached importance to the aggressive dimensionality reduction of the feature space, and developed techniques to remove or lump comparatively uninformative features. Many approaches have been applied to text categorization, e.g. k-nearest neighbor, thesaurus. We proposed a method to use Japanese thesaurus precedently, for purpose of comparison with the technique based on basket analysis, which we proposed in 2.2. We applied this method of attribute generation to same data as described in 3.1, and evaluated the results in terms of error rate, recall, and precision, under same conditions as described in 3.3. In consequence of this comparison with the results mentioned in 3.3, the method generating attributes by thesaurus couldn't afford higher performance. We need to compare the proposed method based on basket analysis with other ones of feature selection. However, this method enables to generate attributes unique to given data and is easily understandable due to application of basket analysis popular in the field of data mining.

According to the result of Table 2, the minimum support level in the basket analysis is a parameter to change *Error rate* and the trade off between *Recall* and *Precision* resulted in the classification. Thus, the basket analysis used in the attribute generation provides a measure to tune these performances in addition to the generation of attributes. The minimum support level can be set depending on the objective of the Web-page classification. If the objective is to classify the Web-pages in a given data set, the minimum support should be set at a low value, since the high accuracy of the classification is needed in this objective. However, this application may not be very feasible, because the value of *Recall* does not become sufficiently high even if the minimum support level is set at a very low value. On the other hand, if the objective is to collect some Web-pages of a class from massive Web-page data, the minimum support should be set at a high value, because the high *Precision* of the classification is obtained. Only the high purity of the classified Web-pages is requested under this objective. Since the value of *Precision* can be very high for a high minimum support level, the proposed approach can provide an efficient measure to collect Web-pages of an objective class.

The proposed approach also provides a set of informative rules to classify the Web-pages. These rules can provide some useful insights to the analysts to manually classify the Web-pages for the directory-style search engines such as Yahoo! JAPAN. This is an advantage of our approach which uses rule-based inductive classification. Some other approaches to use numerical attributes, e.g., k-nearest neighbors and neural network, can not provide the comprehensive rules for the analysts.

Though a few studies worked on the Web-page classification for the directory-style search engines, none of them have reported the detailed performance of their approaches[6][13]. In contrast, we developed techniques including attribute generation and classification to classify Web-pages without using man power, and evaluated the detailed performance of the techniques. In summary, the proposed technique can be used for the automated collection of Web-pages of an objective class from massive data.

Some issues remain for our future work. One is the evaluation of the classification of Web-pages in various categories, because the evaluation has been made only for the top level category classification of Yahoo! JAPAN. We also intend to develop an approach for hierarchical classifications along the hierarchical categorization from the top to the bottom in the directory-style search engines.

References

1. Yahoo! Japan ¡http://www.yahoo.co.jp/¿.
2. ISIZE ¡http://www.isize.com/¿.
3. goo ¡http://www.goo.ne.jp/¿.
4. excite ¡http://www.excite.co.jp/¿.
5. altavista ¡http://www.altavista.com/¿.
6. Yahoo! U.S.A ¡http://www.yahoo.com/¿.

7. Yiming Yang and Xin Liu, A re-examination of text categorization methods.
8. Yiming Yang, An Evaluation of Statistical Approaches to Text Categorization. April 10, 1997.
9. Yiming Yang and Jan O. Pederson, A Comparative Study on Feature Selection in Text Categorization.
10. Iwayama, M., Tokunaga, T. Hierarchical Bayesian Clustering for Automatic Text Classification. 14th International Joint Conference on Artificial Intelligence(IJCAI'95), pp.1322-1327, Montreal, 1995.
11. Iwayama, M., Tokunaga, T. Cluster-Based Text Categorization: A Comparison of Category Search Strategies. 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'95), pp.273-280,Seattle, 1995.
12. Thorsten Joachims, Text Categorization with Support Vector Machines:Learning with Many Relevant Features.
13. Dunja Mladenić, Turning Yahoo into Automatic Web-Page Classifier. 13th European Conference on Artificial Intelligence Young Resarcher Paper(1998).
14. J. R. Quinlan. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993.
15. R.Agrwal and R.Srikant. First algorithms for mining association rules. In *Proceedings of the 20th VLDB Cnference,* pp.487-499, 1994.
16. chasen ¡http://chasen.aist-nara.ac.jp/index.html.en¿