Classification of Deep Web Data Sources Based on Feature Weight Estimate

Xiaoqing ZHOU [1], Jiaxiu SUN[2], Shubin Wang[1]

1Center of Computing, China West Normal University, Nanchong, China

2Colleges of Business, China West Normal University, Nanchong, China

Abstract:
The traditional search engine is unable to correct search for the magnanimous information in Deep Web hides. The Web database's classification is the key step which integrates with the Web database classification and retrieves. This article has proposed one kind of classification based on machine learning's web database. The experiment has indicated that after this taxonomic approach undergoes few sample training, it can achieve the very good classified effect, and along with training sample's increase, this classifier's performance maintains stable and the rate of accuracy and the recalling rate fluctuate in the very small scope.

Keywords: deep web; web database; feature extraction; feature valuation; Naive Bayes classifier

1 Introduction

In order to use the information including in Deep Web database more effectively, it is need to categorize by its fields. At present the classification of Web database is primary and needs further improvement [1]. The paper advances the classification of web database based on machine learning and uses a new method to calculate weight. The classification method can work very effectively after trained with some samples tested by experiment; it will work better with more samples. And the stable performance can be available, accuracy and recall will vary within small range accordingly.

2 Classification of Web database

2.1 Feature extraction of the query interface

Feature extraction is the preface to categorize automatically. HTML form contains complicated information and also magnanimous useful information. How to choose proper feature to describe HTML form is very urgent. The pattern of HTML form is <form action= “URL” method= “get|post” >, and the content of HTML form is </form>. The argument of Action means the processing program to submit HTML data to server. The argument of Method means the method to transfer the content of HTML form to sever [2].

HTML includes so much useful information that it can show its features by using all the control. But some property of outlook can be emitted, such as: size, length, and so on. Additionally the features can be made by the show times of control; HTML can be in one textbox or several textboxes.

According to the above-mentioned, the following features of HTML will be withdrawn:

The property of name in HTML <form> tag;

These words which withdrawn from the property of action in <form> tag;

The type of control which included in the HTML form;

The value of name property and value property in input control;

The property of name in select control and textarea control;

These words between control tags

Many features of HTML have been gotten after the above process. Then we will standardize the features to make it more accurate. Examples as the following figure1 and Figure2. Figure1 is for the code of HTML and figure2 is the features withdraw automatically.

```
<Form name=”searchadvformnew” method=”get”
Action=”http://search.book.dangdang.com/search.aspx”>
<input type=”text” name=”title”>
<input type=”text” name=”author”>
<input type=”text” name=”publisher”>
<input type=”image” name=”search_btn”>
</form>
```

Fig. 1 The HTML code for web form

InputType=text
InputType=image
Textname=title
Textname=author
Textname=publisher
Imagename=search_btn
Formname=searchadvformnew
Action=http
Action=search.book.dangdang.com
Action=search.aspx

Fig. 2 Features extracted from the web form automatically

2.2 Naïve Bayes Classifier

Classifier of Naïve Bayes is the most common of Probability-based classifier instructors [3]. The classifier parameters are formed by the priori class probabilities $P(c_j)$ and class-based entry of conditional probability $P(w_t \mid c_j)$, determined entirely by the training set of documents have been marked。 The formula of each class c_j of priori class probability is such as the equation (1):

$$P(c_j) = \frac{1 + \sum_{d_i \in D} P(c_j \mid d_i)}{|C| + |D|} \quad (1)$$

In equation (1), $|C|$ is the number for the class; $|D|$ is the text number of training in the collection. $P(w_t \mid c_j)$ is estimated by the equation (2):

$$P(w_t \mid c_j) = \frac{1 + \sum_{d_{i \in D}} N(w_t, d_i) P(c_j \mid d_i)}{|V| + \sum_{k=1}^{|V|} \sum_{d_{i \in D}} N(w_k, d_i) P(c_j \mid d_i)} \quad (2)$$

In equation (2), $N(w_j \mid d_i)$ are frequencies of features appear in the text d_i;$|V|$ are on behalf of numbers of all the different features of term in text collection. For the text d_i, when d_i is defined as belonging to category c_j, $P(c_j \mid d_i)$ =1, otherwise, $P(c_j \mid d_i)$ =0.

For the text without annotations in test text set, we can be obtained the posterior probability $P(c_j \mid d_i)$ by using the classifier has been trained. Characteristics of the first k-term in the text d_i are expressed with $w_{d,k}$.Formula as the equation (3):

$$P(c_j \mid d) \propto P(c_j) \prod_{k=1}^{|d|} P(w_{d,k} \mid c_j) \quad (3)$$

2.3 The weight estimates of inquiring interface feature

It is need to create a model after getting the features of query interface. And it is also need evaluate the weight of the features by the method of statistics and mathematics in order to improve the classification accuracy. The famous weight function is the TF-IDF formula by Salton in 1988[4].

Recently, some scholars represent with the rationality of the TF-IDF to weight the features [5]. They think that the useful word sheet is only small ratio and most word sheet is not relative to the type which needs to confirm, so they are noise word sheet. The noise may cover useful information and induce the classification accuracy. So we use a new method to adjust weight, which is to evaluate every feature with evaluation function. The new weight function is called TF-TEF function, TEF is for the evaluation function of the features. The TF-TEF function is as follows:

$$w_{ik} = TF - TEF(t_{ik}) = TF(t_{ik}) \times TEF(t_{ik}) \quad (4)$$

Among them, $TF(t_{ik})$ are the frequencies of first k-term feature in the text d_i, $TEF(t_{ik})$ is the evaluation function to rate various characteristics of term and reflect the degree of correlation between various types of entry. This article uses information gain as evaluation function (equation (5)). $P(w)$ is the probability of Feature words w; $P(\overline{w})$ is probability of feature words w not appear. $P(c_t | w)$ means probability of feature words w to belong to the class c_t.

$$TEF_{\inf oGain}(w) = P(w)\sum_t P(c_t | w)\log\frac{P(c_t | w)}{P(c_t)} + P(\overline{w})\sum_t P(c_t | \overline{w})\log\frac{P(c_t | \overline{w})}{P(c_t)} \quad (5)$$

Weight adjustment techniques are on the basis of feature adjustment to evaluation function, and not Simple feature selection. During weight-adjusted, the classifier role is modified the characteristics. The Calculation of $P(c_j | d)$ changes as the equation (6):

$$P(c_j | d) \propto P(c_j)\prod_{k=1}^{|d|} P(t_{d,k} | c_j)^{TE-TEF(t_{j,k})} \quad (6)$$

In equation (6), $TF - TEF(t_{d,k})$ is the new weighting function of feature entry $t_{d,k}$. The higher weight means greater role in Naïve Bayes model, and the lower weight means smaller role .As $TF - TEF(t_{d,k})$ =0, $P(w_{d,k} | c_j)$ ceases actually.

3 Experimental Design

This article has selected 195 database interfaces from three areas under UIUC Web in order to verify the feasibility of classification algorithm of Web database. We conduct manual classification of these data sources by selecting randomly 10,20,30,40,50,60,70 deep Web query interfaces as training samples, the rest are test samples.

Index of Web Database Categories can adopt accuracy、recall and F value. We suppose Web database of dw_1, dw_2 … dw_n, and predetermined categories of l_1, l_2 … l_n. TP assigned the dw_i to the correct category l_i; FN means not to assign the dw_i to the correct category l_i (Leakage points). FP Incorrectly assigned the dw_i to the l_i (False alarm), Then:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F_1 = \frac{2PR}{P + R} \tag{7}$$

The results shown in table 1.Firstly,the application of classification achieve accuracy of 89.4% and recall of 88.2% after training random sample of 40(Random 10 for each area).Secondly, the classifier performance is stable fundamentally, and the accuracy、recall and F_1 value wave in the small range with the increase of training sample.

Table 1 Accuracy rate、Recalling rate and F1

Number of Training samples	Accuracy rate	Recall rate	F1
40	89.4	88.2	88.1
60	89.5	88.3	88.5
80	88.5	86.8	87.3
100	88.9	87.2	87.8
120	89.6	87.4	87.9
140	88.8	87.6	88
160	88.6	87.8	87.9

4 Conclusions

Web data includes query interface and result. The classification of web data is the key to integrate classification and search. The paper discusses feature extraction from HTML and classification with naïve Bayes. The result shows that the method used in the paper can work well with small samples. But the paper does not consider more content of the words, so it is our further study.

References

[1] GAO Ling, ZHAO Ping-Pong, CUI Zhi-ming. Automatic Judgment of Deep Web Query Interfaces. Computer Technology and Development Vol.17-5(2007), p.150-151

[2] YU Wei, LI Shi-jun, WEN Li-juan, TIAN Jian-wei.Ranking of Deep Web Sources Dased on Data Quality. Journal of Chinese Computer Systems Vol.31-4(2010), p.641-646

[3] JIN Ling-zhi, WANG Xiao-ling, ZHU Shou-zhong.Automatic classification of Deep Web sources. Microcomputer Information Vol.25-4(2009), p.227-230

[4] Tang Huanling, Sun Jiantao, Lu Yuch. A Weight Adjustment Technique with Feature Weight Function Named TEF-WA in Text Categorization. Journal of Computer Research and Development Vol.42-1(2005), p.47-51

[5] Wei Huang，Zhongliang Jing.Multi-focus image fusion using pulse coupled neural network.Pattern Recognition Letters Vol.28-9(2007), p.1123-1132.