An Efficient Centroid Based Chinese Web Page Classifier[1]

Liu Hui Peng Ran Ye Shaozhi Li Xing

ABSTRACT

In this paper, we present an efficient centroid based Chinese web page classifier that has achieved satisfactory performance on real data and runs very fast in practical use. Except for its clear design, this classifier has some creative features: Chinese word segmentation and noise filtering technology in preprocessing module; combined χ^2 Statistics feature selection method; adaptive factors to improve categorization performance. Another advantage of this system is its optimized implementation. Finally we show performance results of experiments on a corpus from Peking University of China, and some discussions.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Probabilistic algorithms – χ^2 *Statistics, Mutual Information*; I.5.1 [**Pattern Recognition**]: Models – *Statistical*; I.5.2 [**Pattern Recognition**] Design Methodology – *Classifier design and evaluation, Feature evaluation and selection*; I.5.4 [**Pattern Recognition**] Applications –*Text processing*.

General Terms

Algorithms, Design, Experimentation, Performance

Keywords

Text categorization, Centroid based, Feature Selection, χ^2 statistics, Chinese word segmentation

1. INTRODUCTION

We live in a world of information explosion. The phenomenal growth of the Internet has resulted in the availability of huge amount of online information. Therefore the ability to catalog and organize online information automatically by computers is highly desirable. Automatic text categorization is such a research field that begins with the emergence of Digital Library, flourished in Web environment, and increasingly became a common network application.

Numerous text categorization algorithm and classifier systems have emerged during the past twenty years. Nowadays the most popular algorithms are Rocchio algorithm [8], Naïve Bayes method [10], Support Vector Machine [1], Boosting method [3], k-Nearest Neighbor [16] and so on. Comparatively, practical classification systems, especially those can provide stable services are much less than the new classification technologies coming out every year. The reasons may be, firstly there still exist many problems to be addressed to apply an algorithm to a system design and implementation, and secondly, many algorithms attained perfect results on public corpora, but failed to achieve satisfactory results on real data.

Asia Pacific Advanced Network 2003, 25-29 August 2003, Busan, Republic of Korea.
Network Research Workshop 2003, 27 August 2003, Busan, Republic of Korea.

Considering web page categorization, a subfield of text categorization, the task is more difficult, since there is a great diversity among the web pages in terms of document length, style, and content. Another aspect of a web page is its fast update status, with the topic, trend and sometimes even writing style changing quickly with time. At the same time, corpora of web page are less than professional document or high-quality content corpora that are used mainly for Digital Library or algorithm testing. But there are also some advantages of web pages. For example, web pages have special structures and links, which provide useful information of their classes, and several systems have exploited this feature to classify web pages. [4]

Turning our attention further to the Chinese Web Page Categorization, we could easily find another two obstacles: the need of segmentation and the lack of Chinese corpora. Chinese web page classification mainly adopts algorithms like k-Nearest Neighbor, Naïve Beyes, Centroid based method and etc, with some exploitations of hyperlinks and structures information to improve classification accuracy. [13][12]

In this paper we present the detailed introduction of a Chinese web page categorization system that has excellent performances on real data at a very fast speed. Our approach has many advantages such as a clear system design, Combined χ^2 Statistics Feature Selection, optimized implementation and some other new features.

We carried out our experiments on a corpus provided by Peking University of China, and we attended the Chinese web page categorization competition held by them on March 15th, 2003. This corpus is available to public, and many Chinese classification systems have done experiments on it, hence the result could be compared and retested.

We have laid out the rest of the paper as follows. In Section 2, we outline the system architecture, and briefly introduce the function of each module, which helps to understand the whole process. In Section 3, we give detailed information on the new features of our system, among which combined χ^2 Statistics algorithm is introduced and analyzed. Sections 4 show some tricks of our system's implementation. In Section 5 we present experimental results and some analysis. At last Section 6 provides conclusions and future plan.

2. ARCHITECTURE

The system is divided into two parts: the training part and the testing part. Training part functions as reading the prepared training samples, implementing a series preprocessing, and then

[1] This classifier ranked first at Chinese web page categorization competition during the meeting of National Symposium on Search Engine and Web Mining, which was hosted by Peking University of China on March 14th –15th, 2003. Further information about the meeting please refer to http://net.cs.pku.edu.cn/~sedb2002/.

extracting characteristics of predefined classes and model for class decision. Testing part is used to label input testing samples by the model generated in training part, and after this process, some statistics are generated to evaluate the performance of the classifier. The testing result affect training part by the feedback module, which could utilize testing statistics to adjust some empirical parameters and the weight of selected features. This makes the classifier more adaptable to the changing data source, especially web.

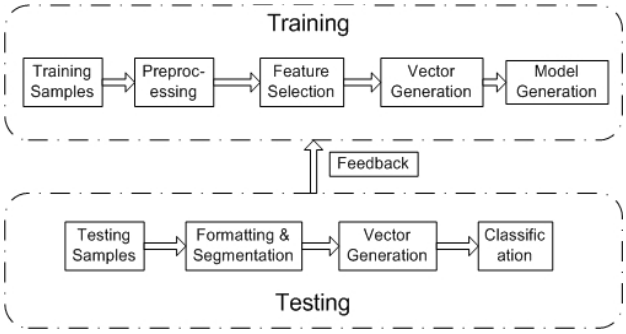


Figure 1. Architecture of classifier

Preprocessing To extract useful information from web page structure, generate attribute-value form texts, and fulfill the Chinese word segmentation. During this process, the statistical data about the training set is recorded into a database, including keyword and its Term Frequency and Inverse Document Frequency (TF-IDF) [9], and each text is saved in a new keyword-frequency form. Here frequency refers to times of this keyword appearing in this single text.

Feature Selection Originally each keyword is looked on as a feature in the text space, and thus each text can be represented as a point in such space. Due to the computation difficulty that huge dimension causes and considerable amount of redundant information that bad features bring with, feature selection is necessary. In this system, a novel combined χ^2 statistics method is adopted, and we choose features in subspace of each class.

Vector Generation We represent a text using Vector Space Model (VSM)[12], so that training samples will be transformed as a vector in subspace of their own class, and testing samples in subspace of every class. Each dimension is a keyword chosen as a feature. The weight of each feature is computed by the popular TF-IDF equation [9].

Model Generation To calculate the centroid of each class. Centroid means the vector that has the same dimension with sample texts and can represent a special class. The Model file contains parameters as class number, feature dimension, training set size, and the centroids.

Formatting & Segmentation To extract information from HTML structured file and segment the Chinese paragraph into word sequence. This work is done using our own Chinese Segmentation algorithm and dictionary, which will be introduced in the later part.

Classification To decide which class a new vector should belong to. This is done by computing the similarities between the test vector and the centroid vector of each class, and then choosing one or two classes as the decision. We just use vector dot product as the similarity measure.

Feedback This module aims to make use of testing statistics to improve classifier applicability for practical data. Adaptive factors will be optimized through user feedback. Correctly decided document could be added as new training sample, and thus may revise feature set.

3. ADVANCED FEATURES

This classifier has some new features compared with other systems that we have studied. Especially for Chinese web page classification, this classifier shows good performance as high precision and very fast speed. For each module of the system, whose main function has been introduced scarcely in, we have done many-detailed work to test its intermediate result and tried our best to improve its performance.

3.1 Preprocessing Techniques

As we have discussed above, word stems is regarded as features, the representation units of text. However, unlike English and other Indo-European languages, Chinese texts do not have a natural delimiter between words. As a consequence, word segmentation becomes one of the major issues of preprocessing. Another problem is caused by the particular characteristics of Web pages, the large diversification of their format, length and content. We also see a lot of hyperlinks in web pages. Some are related to the content of web pages, and others, such as advertisement, are not. We call the irrelevant information "noise", and it is certain that in web pages there exists an amount of noise that must be filtered before document representation.

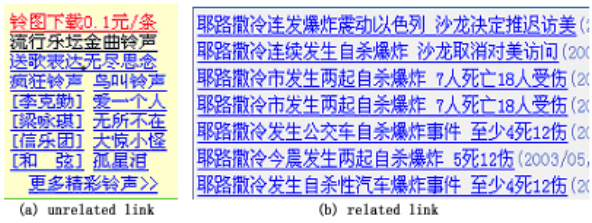
3.1.1 Chinese word segmentation

Not using the traditional dictionary that set up by professionals, we establish our own dictionary based on log data from search engine. We extract words and phrases from log data and exploit the query times as their frequency. This dictionary contains 17000 word or phrases, and [15] said that considering both word and phrase together will effectively enhance categorization precision, and also guarantee independency between different features. As an illustration, "搜索引擎 (search engine)" will not be segmented as "搜索 (search)" and "引擎 (engine)", thus "搜索引擎" will be treated as a single feature in text space. It is a very interesting way that we use dictionary obtained from web service to analyze web pages, so that it has better quality than others in adaptability to the changing web, simplicity to manage and update, and categorization accuracy. For segmenting word, we use Maximum Matching Method [7], which is to scan a sequence of Chinese letters and returns the longest matched word. Although it is simple and not as accurate as more advanced method such as Association-Backtracking Method, its fast speed attracts us and we find its result satisfactory on our dictionary.

3.1.2 Noise filtering

Firstly, stop word, those common words that have not tendency to any class, such as empty word and pronoun, will be removed. Our Chinese stop word list is mainly from Chinese statistical dictionary, combined with high-frequency words in web pages as 'copyright', 'homepage' and etc; secondly advertising links should be deleted, which is usually appeared in commercial sites and homepages. After studying main commercial sites in China, such as Sina (http://www.sina.com.cn), 263 (http://www.263.net) and Sohu (http://www.sohu.com.cn), we find a rule that the length of most advertising or unrelated links are relatively shorter than related links (see Figure 2). And the keywords of advertising links are usually in a limited set. Based on above research, we set 10 as the threshold length of link. If a link is shorter than 10 letters or it contains keywords listed in

the above limited set, it will be considered as noising link and discarded.



(a) unrelated link (b) related link

Figure 2. Comparison of related and unrelated link

3.2 Combined χ^2 Statistics Feature Selection

We adopt a new feature selection algorithm that combines χ^2 Statistics method with Mutual Information (MI) method, and this combination successfully retained the merit of each algorithm and compensate for their limitations.

3.2.1 χ^2 Statistics

A table of events occurrence helps to explain the idea behind this method. And these two events refer to a word ***t*** and a class ***c***. A is the number of documents with ***t*** and ***c*** co-occur, B is the number of documents with ***t*** occurs but not ***c***, C is the reverse of B, and D is the number of documents with neither ***t*** nor ***c*** occurs. Here the number of documents can also be represented as probability, and the corresponding values are proportional. **N** refers to the total number of files in the dataset you are choosing features. This algorithm could be formulated by equation (1).

$$\chi^2(t,c) = \frac{N \times (AD - CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \qquad (1)$$

χ^2 Statistics is based on the hypothesis that high frequency words, whatever class they are in, are more helpful to distinguish different classes. Yang compared main feature selection technologies and demonstrated that χ^2 Statistics method outperformed other methods on public corpora [17]. And a number of Chinese text categorization systems have adopted this method [12][6].

However this method has its limitation. Firstly when A->0 and B->N, which means a word common in other class, but not appearing in the studied class, $\chi^2(t,c)$ will be relatively large, so the weight of common words of other classes often precede many important words in this class; secondly when A->0 and B->0, $\chi^2(t,c)$->0, showing that low frequency words in this class are tend to be removed, which will cause much information loss.

3.2.2 Mutual Information (MI)

The idea of this method is to measure how dependent a word and a class on each other by Mutual Information. According to the definition of Mutual Information, this method can be expressed as equation (2).

$$I(t,c) = \log \frac{P_r(t \mid c)}{P_r(t)} = \log \frac{P_r(t,c)}{P_r(t)\cdot P_r(c)} \qquad (2)$$

It is interesting that MI has two properties, which happen to compensate for limitations of χ^2 statistics method. One is for high frequency word in other class but not the studied class, $P_r(t \mid c)$ is low and $P_r(t)$ is high, so that $I(t,c)$ is comparatively small; the other is for words with the same $P_r(t \mid c)$, those with higher total frequency will be given lower value.

3.2.3 Combined χ^2 statistics

The above analysis shows that χ^2 statistics and Mutual Information algorithm have complementary properties, so we put forward a combined feature selection method that has demonstrated its advantages in our system.

The combined χ^2 statistics can be formulated as equation (3).

$$W(t,c) = \lambda \cdot \chi^2(t,c) + (1-\lambda)\cdot I(t,c) \quad 0 < \lambda < 1 \qquad (3)$$

$W(t,c)$ is the combined weight of word ***t*** to class ***c***, and we use this weight to select features for each class. In our system the best result is achieved when λ happen to be 0.93.

3.3 Subspace

Traditional VSM generates a space that all training samples could be represented in one single space, and this space is called total space. However in Chinese web page categorization, we find that there are obstacles in using total space. An obvious problem emerged when the number of predefined classes or total training samples is large. To include adequate represented features for each class, total feature dimension will be extremely high, and such high dimension bring about much difficulty in computation. If we reduce the dimension to a practical level, precision will decline due to the sparse document vector and inadequate knowledge to tell apart different classes.

To address this problem, we adopt subspace method. Feature subset is chosen for each class, and training samples in this class are represented as points in a subspace generated by this feature subset. As illustrated by Figure 3, each subspace has much fewer features than total space, and these features could reveal the character of the special class more comprehensively and accurately. On the other hand, training samples could be expressed as vectors that have a predestined direction, and this representation tends to preserve class-related information. Therefore, it is not only easier to accomplish the computation, but also to discriminate different classes.

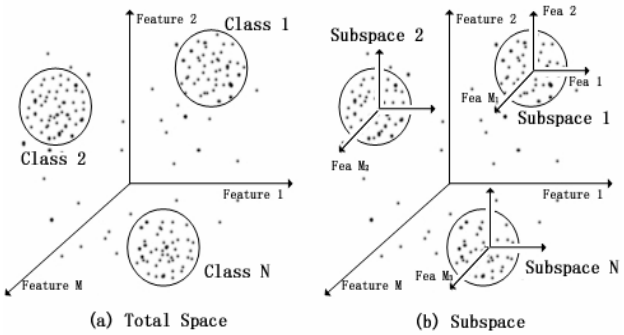


Figure 3. Sketch Map of Totals pace and Subspace

3.4 Adaptive Factors

The task we are dealing with is to automatic classify huge amount and highly dynamic web pages, hence a classifier trained on public corpus is often not compatible with real data, and to update the training corpus frequently so as to catch the changes of web data is also not practical. Although the corpus we use is totally composed of Chinese web pages, it is still not satisfactory because the samples distribute greatly unbalanced among different classes, and the content of these samples cannot cover the majority subfield of these classes. Therefore we adopt adaptive factors to adjust the classifier model and make the classifier more adaptable to the web data.

We incorporate two kinds of adaptive factors in our system.

3.4.1 Class Weight

A default hypothesis in text categorization is that the probability of different classes is equal. However, we observed that not only in real world there exists imparity between different classes but also our classifier is discriminatory to some classes. For example, on one hand, web pages belong to "computer" class are more than those of "social science" class, and on the other hand, our classifier tends to recognize a "computer" related document as belongs to "social science" class because the content of this file contains many not-explicit features and the later class covers a much wider range of features than computer class.

Class weight is a vector whose dimension equals with the number of classes. At the beginning, we set Class weight according to the training samples contained in each class, and normalized them to a real number between 0 and 1. Then in open tests, we find many users to test our system with random selected web pages, and bring feedback results to us. We optimized the class weight factor and achieved much higher precision in later open testing process.

3.4.2 VIP factor

We notice that there are some critical words that are very important for web page categorization, such as 'movie' for entertainment class and 'stock' for financial class, so we set up a very important feature list, and accordingly an array of VIP factors for each class. VIP factors are different among classes because VIP words' effects on different class are not the same. Our definition of VIP factor is as simple as Class Weight, and if a word is in VIP word list, a VIP factor will be considered. Initially the factors were all the same, and were adjusted by user feedback later.

To explain how these factors affect the final decision of class label, we first present equation (4) expressing how to compute the weight of a feature.

$$W(t,d) = freq(t,d) \times \log(\frac{N}{n_t} + 0.01) \qquad (4)$$

It is the TF-IDF method, $freq(t,d)$ refers to times of word ***t*** appearing in document ***d***, N and n_t here are confined within one class, respectively meaning total number of files and number of files with word ***t*** appearing in this class. If word **t** is a VIP word, then equation (4) is changed to equation (5).

$$W'(t,d) = class\_weight[class\_id] \cdot VIP\_factor[class\_id] \cdot freq(t,d) \times \log(\frac{N}{n_t} + 0.01) \qquad (5)$$

4. IMPLEMENTATION

The system is written in ANSI C. The required libraries are all open-source and free. It is tested under Linux, with 256M Memory. It compiles successfully under Solaris and MS Windows with a few changes. And it is also easily portable and customizable.

To set up an efficient database is significant in training process, especially when we are facing with massive collections. We use Berkeley DB [14], the most widely used embedded data management software in the world. Berkeley DB provides a programming library for managing (key, value) pairs, both of which can be arbitrary binary data of any length. It offers four access methods, including B-trees and linear hashing, and supports transactions, locking, and recovery. [2] Another merit of this embedded database is that it is linked (at compile-time or run-time) into an application and act as its persistent storage manager. [11]

In our system, a total training statistic DB is established, which contains the frequency and file numbers that a word appeared in each class and all the training set. During the same process, a small file DB is generated for each file recording the word and its frequency in this file.

In testing process, we do not use any custom implementation in order to avoid extra disk I/O, with the needed statistics or word list is loaded into memory initially. We optimized each step of testing process to improve system speed: simplifying Chinese word segmentation algorithm; adopting two-level-structured dictionaries, making full use of Chinese letter coding rule, and loading dictionary and other word lists into B-trees. Therefore we achieved fast speed, and it could test **3000** medium-sized Chinese web page within **50 seconds**.

5. EXPERIMENT

5.1 Corpus

Currently, there is a lack of publicly available Chinese corpus for evaluating various Chinese text categorization systems [5]. Although Chinese corpus is available from some famous English corpus resources such as TREC, whose corpus is mainly attained from Chinese news sites, we studied those corpora and found their contents to some extent outdated and topic-limited, so they are not suitable for building up a practical Chinese web page system.

Fortunately, Peking University held a Chinese web page categorization competition and provided a public available corpus as the standard (called PKU corpus for short), which became our testbed. This corpus is created by downloading various Chinese web pages that cover the predefined topics. And there is a great diversity among the web pages in terms of document length, style, and content.

Our experiment is based on the corpus consisting of 11 top-level categories and around 14000 documents. The corpus is further partitioned into training and testing data by one attribute of each document, which is set by the provider. The training sample distribution is far from balanced, and the documents in Class 2 cover only a small area of topics in this category, so we enlarged this corpus by adding several hundred web pages to strengthen such too weak classes a little. Detailed information of this corpus is shown in Table 1.

Table 1. PKU corpus statistics (revised version)

#	Category	Train	Test	Total
1	Literature and Art	396	101	497
2	News and Media	284	18	302
3	Business and Economy	852	211	1063
4	Entertainment	1479	369	1848
5	Politics and Government	341	82	423
6	Society and Culture	1063	290	1353
7	Education	401	82	483
8	Natural Science	1788	470	2258
9	Social Science	1700	460	2160
10	Computer Science and Network	829	217	1046
11	Medicine and Health	2240	601	2841
	Total	11373	2901	14274

5.2 Evaluation

Common performance measures for system evaluation are:

Precision (P): The proportion of the predicted documents for a given category that are classified correctly.

Recall (R): The proportion of documents for a given category that are classified correctly.

F-measure: The harmonic mean of precision and recall.

$$F = \frac{2 * R * P}{(R + P)} \qquad (6)$$

5.3 Results and Discussion

We show in Table 2 that the result of our system on previous described corpus. Micro-averaged scores are produced across the experiments, which means that the performance measures are produced across the documents by adding up all the documents counts across the different tests and calculated using there summed values [5].

Table 2. Experimental Results

#	Precision	Recall	F-measure
1	0.829787	0.772277	0.800000
2	0.259259	0.583333	0.358974
3	0.812183	0.884146	0.784314
4	0.961661	0.815718	0.882698
5	0.859873	0.823171	0.841121
6	0.802768	0.800000	0.801382
7	0.658768	0.847561	0.741333
8	0.903448	0.836170	0.868508
9	0.883978	0.695652	0.778589
10	0.735450	0.960829	0.833167
11	0.955932	0.938436	0.947103
Micro-ave	0.862267	0.828680	0.845140

From Table 2, we could find that all of the precision, recall or F-measure are distributed much unbalanced among these 11 classes, but the value of these three measures is comparable for the same class.

It is our first observation that the classifier's precision for a special class has a close relation with the number of training samples in this class. Figure 4 demonstrated that for unbalanced distributed corpus, the classes which own more training samples are tend to achieve better result in its scale. And this phenomenon can be explained by machine learning principle that only when the machine learn enough knowledge in a field, could it recognize new object of it.

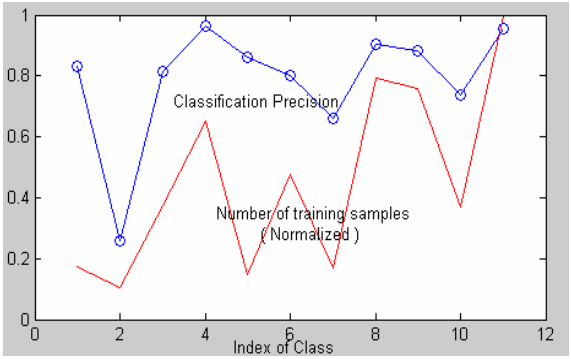


Figure 4. Relationship between Classifier Performance and Number of Training Samples in Each Class

Another observation is through checking the error classified samples and low precision classes. We find that class 2 is obviously difficult for the classifier, because of its lack of training samples and content inconsistency in training and testing part. Although the result seems not very attracting, we find the performance in practical use outperform the experiment, with open testing result above 85% stably.

6. CONCLUSIONS AND FUTURE WORK

Employing classification algorithm effectively into practical system is one of the main tasks of text categorization today. In this paper, we present an efficient Chinese web page categorization classifier and its advantages could be concluded as:

Clear Design We have not included many extra modules in the system, and just follow the traditional structure of text categorization system. This helps to clearly define function of each step and check the performance of each module. It is also very easy to employ other methods into this system, which means just take place the corresponding module.

Novel Technologies Involvement We believe that a perfect algorithm could not achieve good result if the prepared work is not done well. Each step of the system is significant, and should provide the best service for next step. The previous chapters have shown that this system has some tricks and new features in each module that contributes greatly to the final performance.

Optimized implementation Another important factor of a system is its implementation. We adopt high-efficiency database and optimized data structure and coding style, thus the speed of this system is very fast.

Above all, this is a classifier with good performance and fast speed. It is of great practical value, and has provided services for some search engines in China.

In the near future, we need to make it more customizable, including the class structure definition and class hierarchy

scalability. Another work to do is to further strengthen the feedback effect of training process, and the first step is establish user feedback interface at search engines and set up a mechanism to better utilize the information provided by users. In this way, we could more easily update training set and adjust the distribution and content of each class. We also envision being able to use unlabeled data to counter the limiting effect of classes with not enough examples.

7. ACKNOWLEDGMENTS

We thank Dr. Li Yue for her useful suggestions. This material is based on hard work in part by Xu Jingfang and Wu Juan of Tsinghua University.

8. REFERENCES

[1] Cortes, C. and Vapnik, V.N. Support Vector Networks. *Machine Learning*, 20:273-297, 1995.

[2] Faloutsos C. and Christodoulakis S. Signature files: An access method for documents and its analytical performance evaluation. *ACM Transactions on Office Information Systems*, 2(4): 267-288, October 1984.

[3] Freund, Y. and Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1): 119-139, 1997.

[4] Giuseppe Attardi, Antonio Gull, and Fabrizio Sebastiani. Automatic Web Page Categorization by Link and Context Analysis. *In Chris Hutchison and Gaetano Lanzarone, editors, Proceedings of THAI'99, European Symposium on Telematics, Hypermedia and Artificial Intelligence*, 105--119, Varese, IT, 1999.1.

[5] Ji He, Ah-Hwee Tan and Chew-Lim Tan. Machine Learning Methods for Chinese Web Page Categorization. *ACL'2000 2nd Workshop on Chinese Language Processing*, 93-100, October 2000.

[6] Ji He, Ah-hwee Tan, Chew-lim Tan. On Machine Learning Method for Chinese Text Categorization. *Applied Science*, 18, 311-322, 2003.

[7] Jie Chunyu, Liu Yuan, Liang Nanyuan, Analysis of Chinese Automatic Segmentation Methods, *Journal of Chinese Information Processing*, 3(1):1-9, 1989.

[8] Joachims, T. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of International Conference on Machine Learning (ICML'97)*, 1997.

[9] Ken Lang. NewsWeeder: Learning to filter netnews. *In Machine Learning: Proceedings of the Twelfth International Conference*, Lake Taho, California, 1995.

[10] Lewis, D.D. and Ringuette, M. A Comparison of Two Learning algorithms for Text Categorization. In *Third Annual Symposium on Document Analysis and Information Retrieval*, 81-93, 1994.

[11] Melnik S. et al. Building a Distributed Full-Text Index for the Web. *Technical report, Stanford Digital Library Project*, July 2000.

[12] Pang Jianfeng, Bu Dongbo and Bai Shuo. Research and Implementation of Text Categorization System Based on VSM. *Application Research of Computers*, 2001.

[13] Peking University Working Report on Information Retrieval. http://net.cs.pku.edu.cn/opr/fsc_0628.ppt

[14] Resource of Berkeley-DB. http://www.sleepycat.com/

[15] Spitters, M. Comparing feature sets for learning text categorization. *Proceedings of RIAO 2000*, April 2000.

[16] Yang, Y. An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval Journal*, 1999, v1(1/2): 42-49.

[17] Yang, Y., Jan O.Pedersen, A comparative Study on Feature Selection in Text Categorization. *Proc. of, the 14th International Conference on Machine Learning, ICML-97*, pp.412-420, 1997.

9. About Authors

Liu Hui

She is pursuing master degree at DEE of Tsinghua University and is directed by Prof. Li Xing. She is interested in Information Retrieval, Machine Learning and Pattern Recognition.

Address: Room 304, Main Building, Tsinghua Univ. Beijing 100084, P.R.China

Telephone: 8610-62785005-525

Email: liuhui@compass.net.edu.cn

Peng Ran

She is an undergraduate student of Beihang University, and is doing her graduation project at Tsinghua University. Her research field is mainly text categorization and machine learning.

Address: Room 304, Main Building, Tsinghua Univ. Beijing 100084, P.R.China

Telephone: 8610-62785005-525

Email: peng@compass.net.edu.cn

Ye Shaozhi

He is pursing master degree at DEE of Tsinghua University. Directed by Prof. Li Xing, his research area is web crawler, IPv6 web development, Ftp search engine, Pattern Recognition and distributed system.

Address: Room 321, Eastern Main Building, Tsinghua Univ. Beijing 100084, P.R.China

Telephone: 8610-62792161

Email: ys@compass.net.edu.cn

Li Xing

He is the Professor at DEE of Tsinghua University as well as the Deputy Director of China Education and Research Network (CERNET) Center. Being one of the major architects of CERNET, his research interests include statistical signal processing, multimedia communication and computer networks.

Address: Room 225, Main Building, Tsinghua Univ. Beijing 100084, P.R.China

Telephone: 8610-62785983

Email: xing@cernet.edu