Enriched Format Text Categorization Using A Component Similarity Approach

Fei Zhu
School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu, China, 215006
Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou, China
Email: zhufei@suda.edu.cn

Jiong Yang and Yong Zhou
School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu, China, 215006
Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou, China
Email: {youngj, yongz}@suda.edu.cn

***Abstract*—Text categorization has been widely studied for years. However, conventional plain text categorization approaches which work good in plain text behave poor when they are simply applied to enriched format texts. An categorization approach that is applicable to enriched format text is proposed. During feature selection, we get feature structure distribution weight by using extended structure model so that structure affections to categorization are fully considered. Text formats are also taken into account in feature weighting. The combined feature weighting approach strengthens important parts and weakens less important ones. The text categorization is fulfilled by document component similarity, which first decomposes document, gathers features by components and other user-defined rules, completes document component tree, and then achieves categorization by it. We implement a CSBC based Naïve Bayes classifier in which the final result is the combination of all classifiers of component tree. Finally we parse OpenOffice.org document, draw components that are most related to classification from OpenOffice.org documents, and then use the classifier to categorize OpenOffice.org documents. The experiment results show that the classifier can automatically classify OpenOffice.org documents and work quite well.**

***Index Terms*—text classification, enriched format text classification, OpenDocument, OpenOffice.org, Naïve Bayes**

I. INTRODUCTION

Text categorization is the process of automatically assigning documents to predefined categories[1]. It plays an important role in information extraction and summarization, text retrieval, and question-answering[2]. With the increasing number of digital documents, automated text categorization has become more and more promising.

Text categorization has been applied to many industries. Therefore it is expected to meet application demands, which means that text categorization systems should deal with practical documents but not experimental data. Meanwhile, lots of documents being processed by computers in real world now are enriched format texts. As a result, in industrial applications, the text categorization objects are being transferred from plain text to enriched format text.

Enriched format texts, however, usually have more formats, descriptive information, metadata and other elements than plain texts. Enriched format text is generally composed of heterogeneous data, such as texts and pictures. Web pages, Microsoft Office documents, OpenOffice.org documents and StarOffice documents are typical enriched format documents that we often use[3].

Researchers have paid much attention to text categorization algorithms and related approaches in text categorization domain. Little work is focused on how to apply the text categorization approaches to enriched format text. Conventional plain text categorization approaches, however, are often dedicated to text content itself. Information, such as structure information, format information, metadata and other kinds of descriptive information in enriched format text, which is usually helpful to categorization, is usually ignored or incorrectly treated. Categorization preprocessing to clear description information in enriched format text may result in loss of main topic and finally lead to bias when we simply use conventional plain text categorization algorithms on enriched format text. Researches and experiment results show plain text categorization algorithm cannot deal with enriched format text effectively and efficiently. Thus it is necessary to propose a new categorization method for enriched format text.

II. RELATED WORK

Automated text categorization is an important field for many web applications, such as document indexing, document filtering, and cataloging web resources. It has witnessed a booming interest in the past decades, due to the increasing availability of documents in digital form and the ensuing need to organize them. Researchers have put forward varieties of approaches with good performance.

Jialun Lin, Xiaoling Li and Yuan Jiao applied cluster idea to text categorization[4]. They attempted to use the text categorization pattern of self-initiated learning to design a clustering-based text categorization algorithm, in

doi:10.4304/jsw.6.9.1713-1720

the purpose of reducing the dimension of training set and raising the efficiency of categorization implementation. The experiments showed that their algorithm raised the efficiency while slightly reducing the precision.

M. Arun Kumar and M. Gopal performed a study on Linear Support Vector Machines efficiency and efficacy for text categorization tasks[5]. The results showed that SVMlin and Proximal SVM performed better in terms of consistent performance and reduced training time. They further investigated fuzzy proximal SVM on both the text corpuses; it showed improved generalization over proximal SVM.

Peerapon Vateekul and Miroslav Kubat used multiple decision trees algorithm to fulfill automated text categorization[6]. Their solution, called fast decision-tree induction, took advantage of a two-pronged strategy with feature-set pre-selection and induction of several trees.

Hao Pan, Ying Duan and Longyuan Tan put forward a text categorization method based on Naive Bayes learning support vector machine[7]. After text pre-processing, the method used vector space model and linked list of technical extract text features so as to reduce dimensions. In their approach, a Naive Bayes algorithm was used to train the support vector machines and the resulting support vector machines were used for text categorization.

Donghong Fan and Lixia Song proposed a text multi-categorization method based on fuzzy correlation analysis[8]. A fuzzy simple correlation analysis showed the strength and the direction of linear relationship between two fuzzy attributes.

Sheng Gao, Wen Wu, Chinhui Lee and Tatseng Chua introduced a maximal figure-of-merit-learning approach for robust classifier design, which directly optimized performance metrics of interest for different target classifiers[9]. The approach, embedding the decision functions of classifiers and performance metrics into an overall training objective, learned the parameters of classifiers in a decision-feedback manner to effectively take into account both positive and negative training samples, thereby reducing the required size of positive training data.

Dwi Sianto Mansjur, Ted S. Wada and Biing Hwang Juang tried to solve automatic text categorization problem using kernel density classifier with topic model and cost sensitive learning[10]. A Latent Semantic Analysis technique was used to construct a topic space. Each dimension represented a single topic, from which text features were extracted by the system.

T. Mouratis and S. Kotsiantis improved the accuracy of Discriminative Multinomial Bayesian Classifier by using feature selection technique[11] that evaluated the worth of an attribute by computing the value of the chi-squared statistic with respect to the class.

Zijun Yu, Weigang Wu, Jing Xiao, Jun Zhang, Rui-Zhang Huang and Ou Liu proposed a keyword combination extraction algorithm[12], which is based on ant colony optimization, to search the optimal keyword combination of a target category.

Zhenyu Lu, Yongmin Liu, Shuang Zhao and Xuebin Chen presented an approach of Chinese text feature selection and weighting based on semantic statistics[13]. They used synonym merging and a weight function based on term frequency and entropy.

Wei Huang, Yi Liu, Bing Gao and Kewei Yang proposed a method of word segmentation according to lexical chunk as segmentation unit[14]. The approach used traditional segmentation method to segment Chinese text. It calculated mutual information between two lexical entries and adjacent frequency of two or more lexical entries. The results proved the effect of feature selection in Chinese text categorization and enhanced the capability of text classification.

Yanjun Li, D. Frank Hsu and Soon M. Chung combined multiple feature selection methods by using the Combinatorial Fusion Analysis[15]. A rank-score function and its graph, called rank-score graph, were adopted to measure the diversity of different feature selection methods.

Xiaofei Zhang, Heyan Huang and Keliang Zhang introduced a KNN text categorization algorithm based on semantic centre, which replaced the large number of original sample documents with a small amount of sample semantic centers[16]. Experiments proved it worked over 10 times as fast as that of the traditional KNN and its F_1 value was approximately equal to SVM and traditional KNN algorithm.

Hu Guan, Jingyu Zhou and Minyi Guo designed a fast Class-Feature-Centroid classifier for multi-class, single-label text categorization, in which a centroid was built from inter-class term index and inner-class term index[17]. The classifier proposed a combination of these indices and employed a denormalized cosine measure to calculate the similarity score between a text vector and a centroid.

Yunliang Zhang, Lijun Zhu, Xiaodong Qiao and Quan Zhang took advantage of a flexible KNN algorithm for text categorization by authorship based on features of lingual conceptual expression[18]. The approach was combined with K-variable algorithm and weighting algorithm. It improved the effect of text categorization.

Xiaoyu Jiang, Xiaozhong Fan, Zhifei Wang and Keliang Jia applied automatic summarization to text categorization so as to reduce the dimensionality of feature vector space and the computing complexity of categorization[19]. Text summarization was directly used for feature selection and categorization instead of the original text. Each summary was used to select and weight features for each document. Finally texts were classified using KNN algorithm.

Yi Guo, Zhiqing Shao and Hua Nan introduced a content-oriented automatic text categorization algorithm to simulate the human cognitive procedure in the text categorization task[20]. The approach included a process of lexical or semantics analysis. It reduced the time and effort spent on training and corpus maintenance.

Jauji Shen and Jiachiuan Wu regarded document as a container of term, and generated rules by using the term distribution in documents[21]. They believed there

existed some kind of semantic relevance between term and paragraph in a document, called Meaningful Inner Link Objects-MILO, which varied with different semantics of a document itself.

III. TEXT CATEGORIZATION AND NAIVE BAYES TEXT CLASSIFIER

A. Text Categorization

Text categorization is to determine the class or the classes of a given text by its property within predetermined classes. That is, for a given corpus $D=\{d_1,d_2,\cdots,d_{|D|}\}$ with $|D|$ texts and $|C|$ irrelevant classes $C=\{c_1,c_2,\cdots,c_{|C|}\}$, text categorization is to assign a Boolean value *true* or *false* to each item $<d_j,c_i>$ where $<d_j,c_i>\in D\times C \quad (1\le i\le|C|,1\le j\le|D|)$. If d_j belongs to the class c_i, $<d_j,c_i>$ is assigned a value *true*; otherwise it is assigned a value *false*.

Text categorization can be expressed more formally as: by using categorization function $\Phi: D\times C\to\{true, false\}$, find an approximate expression of unknown objective function $\hat{\Phi}: D\times C\to\{true, false\}$, such that $\hat{\Phi}$ is as close to Φ as possible.

Generally speaking, text categorization has three parts, including training, testing and classifying. The purpose of training is to set up a classifier. Testing is to evaluate the classifier and classifying is to get resulting class of unknown texts by using the classifier. The framework of a text categorization system is shown as figure 1.

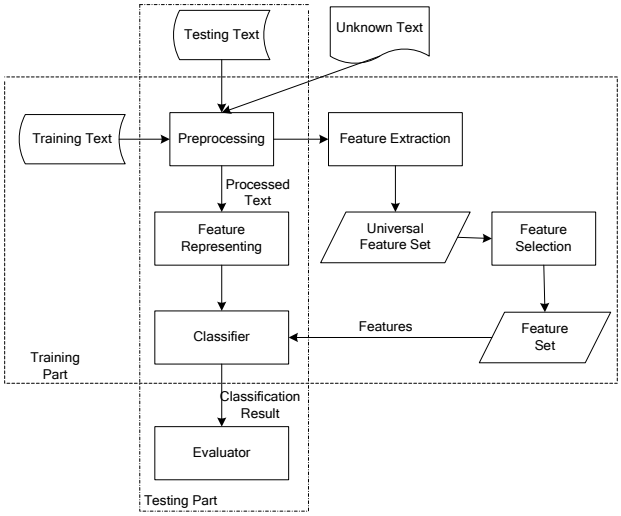

Figure 1. System Framework of Text Categorization

B. Naive BayesText Classifier

Naïve Bayes text classifier is widely used in many research fields and applications. It assumes a document as a collection of independent words. Naïve Bayes text classifier behaves quite well[22,23], although the above hypothesis is usually not true. One of the advantage of the hypothesis is that it decreases calculation and increases capability of Naïve Bayes classifier as well.

Naïve Bayes text classifier uses probability of a word vector belonging to a class c_i to get the probability of text belonging to c_i, as shown in equation 1[24].

$$P(c_i\mid d_j)=\frac{P(c_i)P(d_j\mid c_i)}{P(d_j)} \tag{1}$$

Here $P(d_j)$ denotes the stochastic probability of getting vector d_j and $P(c_i)$ is the stochastic probability of belonging to class c_i.

Due to the huge dimensions of vector d_j, it is hard to get the value of $P(d_j\mid c_i)$. For the simplification of computation, we treat the two vectors of document as statistically independently. Then $P(d_j\mid c_i)$ can be calculated by equation 2.

$$P(d_j\mid c_i)=\prod_{t\in d_j}P(t\mid c_i) \tag{2}$$

IV. ENRICHED FORMAT TEXT CATEGORIZATION

A. Plain Text and Enriched Format Text

Plain texts are those that only have few limited basic descriptive information for content, such as formats for text.

Enriched format text uses a standardized method to code various text properties, format descriptions and structure information as well as other metadata. Web pages, Microsoft Office documents, OpenOffice.org documents and StarOffice documents are enriched format documents that we often use.

Enriched format text, as opposed to plain text, has styling information beyond colors, styles, sizes and special features. It has more formats and elements than plain text. It has logic structures and metadata, and is usually composed of heterogeneous data, such as texts and pictures.

B. Enriched Format Text Categorization

Text categorization has been studied for many years in information retrieval and machine learning. Researcher has put forward various approaches which work quite well. With there are more and more enriched format texts, lots of documents being processed by computers are exactly enriched format documents. As a result, the practical processing objects are transforming from plain texts to enriched format ones. Meanwhile, conventional text categorization still stays on solving plain text categorization.

In plain text categorization, researchers have to devote most to the content of the text because lack of information describing the texts. On the other hand, plain text categorization is relatively simple as the main processing objects are texts. The descriptive information in plain text is much less important so that ignoring the descriptive information will not affect categorization result much. Conventional categorization algorithms which focus more on text content work quite well on plain text.

Plain text categorization considers more about texts but less cares about descriptive information. Some researchers tried to applied conventional text categorization approaches to enriched format texts but the results were much worse than expected as plenty of descriptive information that useful to categorization were ignored or processed incorrectly.

Enriched format text contains abundant formats, such as font, styles, size, underline and color. It can also have structure information such as paragraph settings which unique in enriched format documents. As we know, variations like text colors, styles, fonts and sizes are remarkable contrasts and help to understand the topic of the text. Hence, when we design a classifier for enriched format text, these factors should be taken into consideration.

V. Structure Description Information of Text

A. Basic Structure of Document Text

In a document, there will be various structures such as paragraph, sentence and phrase. Structures in text are usually set by the writer intentionally. Normally one is willing to put all closely related text to a single structure, e.g. a paragraph. We found structures are really helpful to understand topic and theme of the text.

Document text generally contains paragraph, sentence and word. It may have several paragraphs; a paragraph is composed of many sentences; while a sentence consists of words. Document text structure usually looks like a hierarchical upside down tree, as shown in Figure 2[25], which we call basic structure.

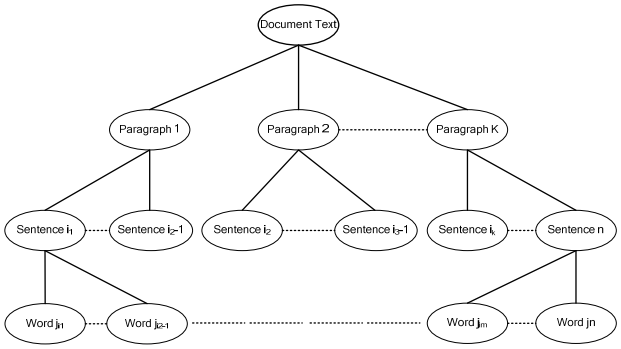

Figure 2. Basic structure of text which is hierarchically organized

B. Extended Structure of Enriched Format Text

As enriched format documents have metadata including title and keywords, and text body, basic structures are not fit any more. To solve the problem, we put forward an extended structure which is based on basic structure. In extended structure, document text is composed of metadata and body. Title, description, keyword and subject are four kinds of typical and important metadata. Body has sections. A section consists of extended paragraphs which are combined by texts with related topic. An extended paragraph is composed of sentences. A sentence is composed of words. The extended structure of enriched format text is shown as Figure 3[25].

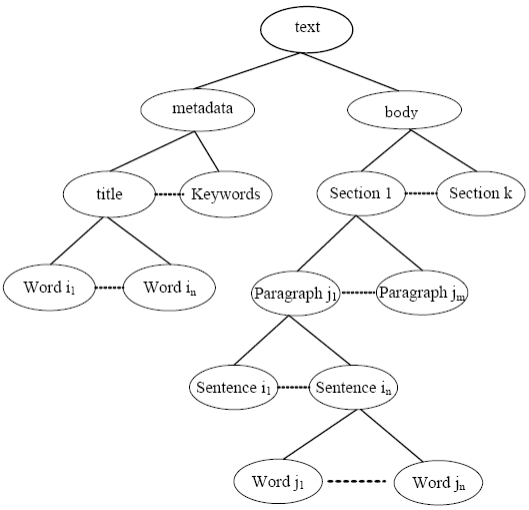

Figure 3. Extended structure of text which is hierarchically organized and contains more semantic elements

C. Feature Distribution Among Structures

As we have already known, enriched format text has structures which are helpful to categorization. Since writers tend to put related texts to a single structure, feature distribution among structures should be taken into account.

According to C.E. Shannon[26], the relevance of a feature to a topic is directly proportional to its occurrence times in the text. That is, if a feature occurs more times in a text, it is more likely relevant to the topic. On the other hand, the relevance of a feature to a topic is inversely proportional to its occurrence times in the classes. That is, the more classes contain the feature, the less distinctive the feature is.

Presently, most studies consider frequency of a feature in a document and distribution among documents, such as term frequency-inverse document frequency (tf-idf), while few consider its distribution among structures within a document which is in fact very important to categorization. In the following example, tf-idf and other conventional feature weighting algorithms cannot handle well. Given a text with n paragraphs, $p_1,\ldots,p_k,\ldots,p_n$, feature t concentratedly occurs in p_k, and rarely occurs in other paragraphs. Meanwhile the topic of p_k just partially reflects one aspect of the text or even has an entirely different topic with the overall topic.

This example is common for those with long text and large numbers of paragraphs that topic of one or few paragraphs is sometimes different with that of whole document. In such case, only considering a feature's local paragraph affect and ignoring its global text affect is unreasonable. Therefore feature distribution among structures within a document as well as its distribution among documents should both be taken into consideration.

We believe wide distribution of a feature among a single document shows the whole document is permeated by the feature, which implies the feature is strongly relevant to the text and thus able to reflect the main topic of the text. Therefore such kind of features should be

assigned to greater weights. As paragraph is a frequent kind of structure, we consider the feature's paragraph distribution. We hereby use equation 3 to get structure distribution weight of a feature.

$$wd_{kj} = \frac{\log\left(1+\frac{n_{kj}}{NP_j+1}\right)}{\sqrt{\sum_{i=1}^{M}\left(\log\left(1+\frac{n_{ij}}{NP_j+1}\right)\right)^2}} \quad (3)$$

Here n_{kj} denotes the total number of paragraphs containing feature t_k in document d_j, NP_j is the total paragraph number of d_j, and M is total feature number in the corpus.

In equation 3, we call $\frac{n_{kj}}{NP_j+1}$ structure distribution factor which evaluates feature's structure distribution in the document. Its value grows with the infusibility of the feature. We here take logarithm so as to reduce the effects of large differences in frequencies.

VI. Component Similarity Based Categorization

We here propose a novel categorization for enriched format document which is fulfilled by document component tree. We call it Component Similarity Based Categorization (CSBC). CSBC first decomposes document, then gathers text or feature by components and/or other user-defined rules, constructs document component tree, and then achieves categorization by it. When using CSBC, final result is the combination of all classifiers of component tree.

For example, train_doc is a training text and test_doc is a testing text. By using CSBC, every component in test_doc retrieves equivalent components in train_doc, computes and compares similarity. Final categorization prediction is combination of all relevant parts. The illustration is shown as Figure. 4

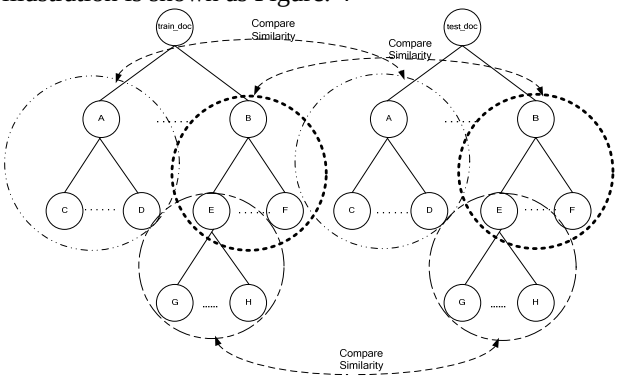

Figure 4. CSBC retrieves equivalent components and then computes similarity by document tree

By using Naive Bayes, the final categorization prediction of the document is the product of all component probabilities, as shown in equation 4.

$$P(d_j \mid c_i) = \sum_{s \in d_j} \sum_{t_k \in s} f_s(t_k, d_j) \log(P_s(t_k \mid c_i)) \quad (4)$$

VII. OpenOffice.org Word Document Categorization

A. Introduction to OpenOffice.org

OpenOffice.org is a leading open-source office software suite for word processing, spreadsheets, presentations, graphics, databases and more. It is available in many languages and works on all common computers. It stores all data in an international open standard format and can also read and write files from other common office software packages[27].

OpenOffice.org components include word processing, spreadsheets, presentations, drawings, data charting, formula editing, a database, and file conversion facilities. OpenOffice.org documents can be saved in an XML format or an XML-like format[28]. However, the resultant file is a binary since it is compressed. Table I shows some common applications in OpenOffice.org and their default file types.

TABLE I. OppenOffice.org applications and file type

Application	*File type*
OpenOffice.org Writer	*.odt
OpenOffice.org Impress	*.odp
OpenOffice.org Calc	*.ods
OpenOffice.org Draw	*.odg
OpenOffice.org Math	*.odf
OpenOffice.org Database	*.odb

B. Parsing of OpenOffice.org Word Document

OpenOffice.org word document is essentially a package including several files for setting, metadata and format information. OpenOffice.org word document has an upside down tree-like structure where each node is a component. The nodes in the document contain content and property which is descriptive information of component. The root node is office:document which has node office:meta for metadata, node office:setting for global settings, node office:script for scripts, node office:font-decls for font settings, node office:styles for common styles, node office:automatic-styles for automatic styles, node office:master-styles for primary styles, and node office:body for content body, as shown in Figure. 5.

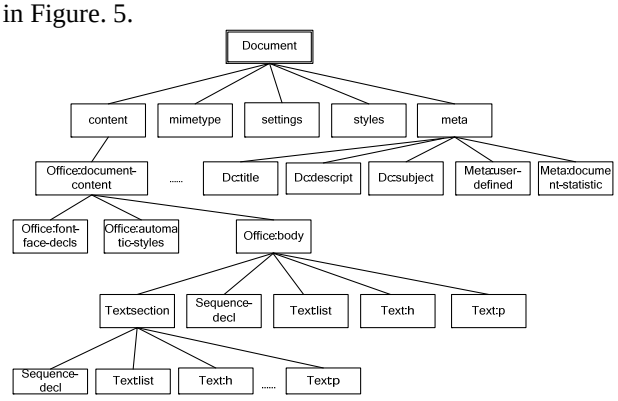

Figure 5. OpenOffice.org word document component tree

OpenOffice.org word document has extended structures and descriptive information, such as metadata,

title, subject, keyword and description, which, as we already discussed, are usually summarization of the document and therefore very important to categorization. Thus, when modeling categorization classifier, we take these factors into consideration.

C. Adding Formatting Weighting

Enriched format text could have various font formats such as bold, underline, italic, font-color and font-size. The texts with formats are different with those without formats. Normally, when someone uses a different format, he tends to emphasize the importance of the formatted text.

Yiming Yang and other researchers showed in their experiments that assigning different weights to formatted texts could improve precision rate and recall rate of categorization[29-31]. We hereby add format weighting to formatted text nodes in OpenOffice.org word component tree. Table II shows some formats, tags and related weights.

TABLE II. SOME USUAL FORMATS, TAGS AND WEIGHTS OF OPENOFFICE.ORG WORD DOCUMENT

Format	*Tag*	*Weight*
Italic	fo: font-style="italic"	W_t
Bold	fo: font-style	
Underline	style: text-underline-style	

D. Feature Weighting

We get whole text content to form a complete feature collection *T* for processing, and then raise weight of key nodes in the structure tree. Hence when we carry on categorization, four aspects including content of document text, feature distribution among structures, descriptive information of text, and text format, should be considered. We evaluate feature weight with equation 5 where wd_{kj} is calculated by equation 3.

$$w'_{kj} = \frac{\log(wf_{kj}+1.0)\times\log(\frac{N}{n_k})}{\sqrt{\sum_{i=1}^{M}(\log(wf_{ij}+1.0)\times\log(\frac{N}{n_i}))^2}}$$
$$wf_{kj} = \left(ws_{kj} + wd_{kj} + 1.0\right)\times f_{kj} \quad (5)$$
$$ws_{kj} = \begin{cases} w_u, t_k \notin T \\ w_t, t_k \in T \end{cases}$$

E. CSBC for OpenOffice.org Word Document

We believe content part of document text determines categorization result and non-content part affects categorization greatly. In CSBC model, each content or component belongs to one and only one component node.

As we have already discussed, among all the OpenOffice.org word document components, component title, description, subject and keyword of metadata document and component body of document text are the most important and the most useful to categorization. Therefore we construct a document component tree with these five kinds of components. And we just compute the similarity between two same kinds of component, as shown in Figure.6[25].

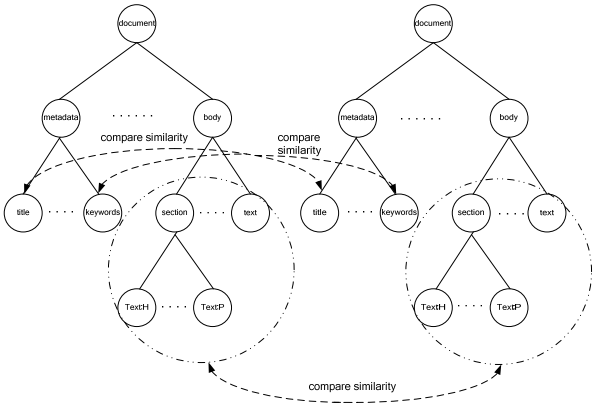

Figure 6. CSBC retrieves equivalent components in two documents and then computes similarity

We first decompose document, gather texts and then complete a document component tree. After that, we get similarities of the same kind of component. The final result is the combination of all classifiers of component tree by using equation 6.

$$P'(d_j \mid c_i) = \sum_{s\in d_j \wedge s\in SC} \sum_{t_k \in s} f_s(t_k, d_j)\log\left(P_s(t_k \mid c_i)\right) \quad (6)$$

Here *s* denotes each component in document and *SC* = {*title, description, subject, keyword, body*}.

VIII. EXPERIMENT AND RESULTS

We set up a categorization training and testing system with 9 objective classes from Fudan corpus[32], in which 3244 documents are used for training and 3164 documents are used for testing. Before categorization, we took some processing to transform the documents to OpenOffice.org word documents. Table III lists the testing results using conventional Naïve Bayes text classifier and Table IV shows results of CSBC based Naïve Bayes classifier.

TABLE III. TESTING RESULTS USING CONVENTIONAL NB

Class	*Documents*	*Precision(%)*	*Recall(%)*	*F_1(%)*
Transport	262	75.00	73.94	74.47
Sports	491	84.01	80.86	82.44
Military	359	79.27	85.53	82.40
Medical	305	75.00	79.34	77.17
Education	321	77.97	68.36	73.17
Environment	424	81.35	74.47	77.91
Economy	376	74.43	89.63	82.03
Art	375	74.91	82.08	78.50
Computer	251	80.87	61.22	71.05

TABLE IV. TESTING RESULTS USING CSBC BASED NB

Class	*Documents*	*Precision(%)*	*Recall(%)*	*F_1(%)*
Transport	262	83.33	82.06	82.70
Sports	491	80.27	83.71	81.99
Military	359	80.37	85.52	82.94
Medical	305	83.73	80.98	82.36
Education	321	75.08	71.34	73.21
Environment	424	80.61	81.37	80.99
Economy	376	77.89	82.45	80.17
Art	375	78.81	81.33	80.07
Computer	251	81.91	64.94	73.42

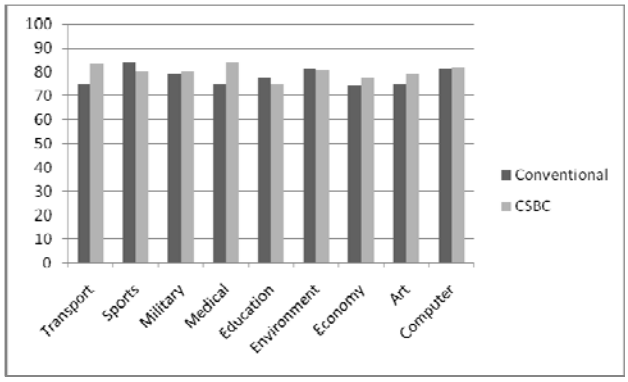

Figure 7. Precision of conventional and CSBC based NB classifiers

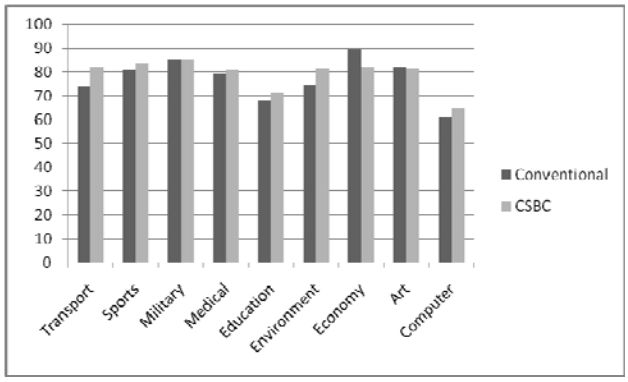

Figure 8. Recall rate of conventional and CSBC based NB classifiers

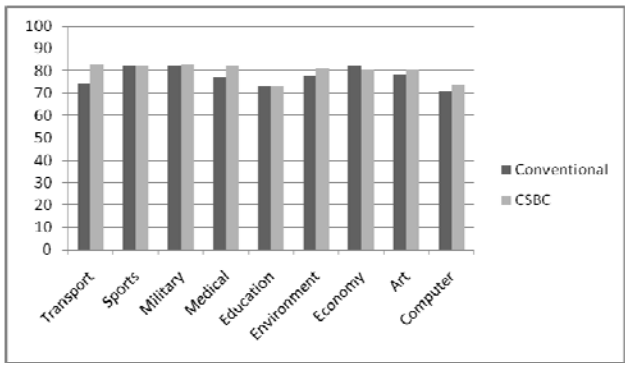

Figure 9. F_1 rate of conventional and CSBC based NB classifiers

We can see from Table III, Table IV, Figure 7, Figure 8 and Figure 9, both conventional Naïve Bayes and CSBC based Naïve Bayes work pretty good. Among 9 objective classes, 6 of 9 in precision rate, 6 of 9 in recall rate and 7 of 9 in F1 rate of CSBC based Naïve Bayes surpassed, showing overall performance of CSBC is better than conventional method.

IX. Conclusion

When we use conventional text categorization method to solve enriched format text categorization, much of the descriptive information is skipped or mistreated as text content. As we can imagine, neither of the cases is correct. Therefore it is necessary to separate text content from descriptive information. However the research of enriched format text categorization is not developed as expected.

We propose a text categorization approach for novel enriched format document. It considers descriptive information in texts and uses a new feature weighting approach. The experiment results show that approach can automatically conduct OpenOffice.org word documents categorization well.

Meanwhile, the component similarity approach based text categorization can also be implemented by other common algorithms, such as KNN. Moreover, the idea of CSBC can also be extended to in text mining and bioinformatics fields.

Acknowledgment

Supports from School of Computer Science and Technology, Soochow University and Provincial Key Laboratory for Computer Information Processing Technology, Soochow University are highly appreciated. Lurong Li and other contributors' work to Fudan corpus are also gratefully acknowledged.

References

[1] Kjersti, L.Elikvil, "Text Categorisation: A Survey," Rapport Nr. 941, Oslo, Norway: Norwegian Computing Center, 1999.

[2] F.Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, 2002.3, vol.34, pp. 1-47.

[3] OASIS, "Open Document Format for Office Application," www.oasis-open.org.

[4] Jialun Lin, Xiaoling Li, Yuan Jiao, "Text Categorization Research Based on Cluster Idea," ETCS, 2010 Second International Workshop on Education Technology and Computer Science, 2010, vol. 1, pp.483-486.

[5] M. Arun Kumar, M. Gopal, "An Investigation on Linear SVM and its Variants for Text Categorization," ICMLC, 2010 Second International Conference on Machine Learning and Computing, 2010, pp.27-31.

[6] Peerapon Vateekul, Miroslav Kubat, "Fast Induction of Multiple Decision Trees in Text Categorization from Large Scale, Imbalanced, and Multi-label Data," ICDMW, 2009 IEEE International Conference on Data Mining Workshops, 2009, pp.320-325.

[7] Pan Hao, Duan Ying, Tan Longyuan, "Application for Web Text Categorization Based on Support Vector Machine," IFCSTA, 2009 International Forum on Computer Science-Technology and Applications, 2009, vol. 2, pp.42-45.

[8] Donghong Fan, Lixia Song, "Text Categorization Method Based on Fuzzy Correlation Analysis," ICECS, 2009 Second International Conference on Environmental and Computer Science, 2009, pp.196-198.

[9] Sheng Gao, Wen Wu,Chin-Hui Lee,Tat-Seng Chua, "A maximal figure-of-merit (MFoM)-learning approach to robust classifier design for text categorization", ACM Transactions on Information Systems (TOIS), 2006.4, vol. 24 , Issue 2, pp. 190-218.

[10] Dwi Sianto Mansjur, Ted S. Wada, Biing Hwang Juang, "Using Kernel Density Classifier with Topic Model and Cost Sensitive Learning for Automatic Text Categorization," ICDAR, 2009 10th International Conference on Document Analysis and Recognition, 2009, pp.1086-1090.

[11] T. Mouratis, S. Kotsiantis, "Increasing the Accuracy of Discriminative of Multinomial Bayesian Classifier in Text Classification," ICCIT, 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology, 2009, pp.1246-1251.
[12] Zijun Yu, Weigang Wu, Jing Xiao, Jun Zhang, Ruizhang Huang, Ou Liu, "Keyword Combination Extraction in Text Categorization Based on Ant Colony Optimization," SOCPAR, 2009 International Conference of Soft Computing and Pattern Recognition, 2009, pp.430-435.
[13] Zhenyu Lu, Yongmin Liu, Shuang Zhao, Xuebin Chen, "Study on Feature Selection and Weighting Based on Synonym Merge in Text Categorization," ICFN, 2010 Second International Conference on Future Networks, 2010, pp.105-109.
[14] Wei Huang, Yi Liu, Bing Gao, Kewei Yang, "Study on Method of Word Segmentation in Feature Selection in Chinese Text Categorization," WKDD, 2010 Third International Conference on Knowledge Discovery and Data Mining, 2010, pp.411-415.
[15] Yanjun Li, D. Frank Hsu, Soon M. Chung, "Combining Multiple Feature Selection Methods for Text Categorization by Using Rank-Score Characteristics," ICTAI, 2009 21st IEEE International Conference on Tools with Artificial Intelligence, 2009, pp.508-517.
[16] Xiaofei Zhang, Heyan Huang, Keliang Zhang, "KNN Text Categorization Algorithm Based on Semantic Centre," ITCS, 2009 International Conference on Information Technology and Computer Science, 2009, vol.1, pp.249-252.
[17] Hu Guan, Jingyu Zhou, Minyi Guo, "A class-feature-centroid classifier for text categorization," International World Wide Web Conference Proceedings of the 18th international conference on World Wide Web, Madrid, 2009, pp.201-210.
[18] Yunliang Zhang, Lijun Zhu, Xiaodong Qiao, Quan Zhang, "Flexible KNN Algorithm for Text Categorization by Authorship Based on Features of Lingual Conceptual Expression," CSIE, 2009 WRI World Congress on Computer Science and Information Engineering, 2009, vol. 2, pp.601-605.
[19] Xiaoyu Jiang, Xiaozhong Fan, Zhifei Wang, Keliang Jia, "Improving the Performance of Text Categorization Using Automatic Summarization," ICCMS, 2009 International Conference on Computer Modeling and Simulation, 2009, pp.347-351.
[20] Yi Guo, Zhiqing Shao, Hua Nan, "Content-Oriented Automatic Text Categorization with the Cognitive Situation Models," ISCSCT, 2008 International Symposium on Computer Science and Computational Technology, 2008, vol. 1, pp.512-516.
[21] Jauji Shen, Jiachiuan Wu, "Meaningful Inner Link Objects for Automatic Text Categorization," IIH-MSP, 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009, pp.266-269.
[22] Yiming Yang, "An Evaluation of Statistical Approaches to Text Categorization," Information Retrieval, 1999, pp.69-90.
[23] Yiming Yang, Xin Liu, "A Re-examination of text categorization methods," SIGIR'99 Berkley, 1999, USA, ACM.
[24] Mitchell, Tom M., "Machine Learning," McGraw-Hill College, 2005.6, pp.414.
[25] Fei Zhu, "Component Tree Based Categorization: An Novel Categorization Approach for Rich Format Text," 2008 Proceedings of Information Technology and Environmental System Sciences, vol.2, 2008.4, pp.1196-1202.
[26] C. E. Shannon, "A mathematical theory of communication," SIGMOBILE Mob. Comput. Commun. Rev. 5, 1, 2001.1, pp. 3-55.
[27] http://www.openoffice.org/
[28] Sun Microsystems Inc, "OpenOffice.org XML File Format Technical Reference Manual Version," http://xml.openoffice.org/ xml_specification.pdf.
[29] Monica Rogati, Yiming Yang, "High-Performing Feature Selection for Text Classification," CIKM, 2002, McLean, USA, ACM.
[30] Zhaohui Zheng, Xiaoyun Wu, Rohini Srihari, "Feature Selection for Text Categorization on Inbalanced Data," Newletter of the ACM Special Interest Group on knowledge Discovery and Mining,2004,6(1).
[31] Gang Wang, Frederick H Lochovsky, Qiang Yang, "Feature Selection with Conditional Mutual Information MaxiMin Text Categorization," CIKM, 2004, Washington, USA, ACM.
[32] www.nlp.org.cn.

Fei Zhu was born in Suzhou, Jiangsu, China, in 1978. He got his master's degree in 2006, majoring in computer science and technology. He is doing research on bioinformatics and systems biology. His interests include machine learning, biological text mining, biological and biomedical network analysis and construction.