Text Classification using Multi-word Features

Wen Zhang, Taketoshi Yoshida, Xijin Tang

Abstract—**We carried out a series of experiments on text classification using multi-word features. An automated method was proposed to extract the multi-words from text data set and two different strategies were developed to normalize the multi-words into two different versions of multi-word features. After the texts were represented respectively using these two different multi-word features, text classification was conducted in contrast to examine the effectiveness of these two strategies. Also the linear and nonlinear polynomial kernel of support vector machine (SVM) was compared on the performance of text classification task.**

I. Introduction

AUTOMATED text classification utilizes a supervised learning method to assign predefined category labels to new documents based on the likelihood suggested by a trained set of labels and documents. During the process of transforming the unstructured text into structured data as numerical vectors for the data mining methods, bag of words (BOW) [1] is often used to represent the text with single words obtained from the given text data set. As a simple and intuitive method, BOW method makes the representation and learning easy and highly efficient. But an obvious disadvantage of BOW method is that it ignores the ordering and composing of words occurring in the text which are used to describe a concept, not merely a mixture of single words, and this kind of concept usually can provide more evidential information for text classification. With this disadvantage of BOW method, it is reasonable to conjecture that adopting ordering and composing information in text representation purposely might improve the text classification performance. Generally, multi-word features are not found too frequently in a text data set, but when they do occur they are often highly predictive. Based on this motivation, the method of multi-word features [3] was proposed in this paper, and the effectiveness of this method was examined with text classification using multi-word features.

Recently, a lot of research has been undertaken in the text mining field with the expectation to enhance BOW by both linguistic characteristics and logic character-istics of words in text. The ordering and position of a word in document was considered as the background relation between text categories for classification in [2]. The concept of ontology was introduced in [4] to represent text, and different strategies to make use of ontologies, the linguistic relationships between words and concepts, are discussed. However, these studies are all based on single word representation and concentrate on the hidden relationships between single words in the aspect of linguistics or logic. In this paper, multi-word features, i.e., a group of consecutive words were proposed for text representation and their effectiveness was examined with the text classification task. Two strategies were developed to post-process the extracted multi-words into multi-word features (multi-word features are the multi-words after post-processing) to represent the documents, and their text classification performances were compared. Also, the linear kernel and nonlinear kernel of support vector machine (SVM) were compared in this paper on text classification task.

The rest of this paper is organized as follows. Section 2 describes the data set used for text classification. Section 3 describes data preprocessing procedures which aim to extract the multi-word features from a raw data set. The text classification task with multi-word features was designed, carried out and the results were demonstrated in Section 4. Also, two strategies for feature set construction together with the linear kernel and nonlinear kernel of SVM were compared respectively in this section. Finally, concluding remarks and further research plans are given in Section 5.

II. Data Selection

The Reuters-21578 data set [5] was selected as our experiment data. It appeared as Reuters-22173 in 1991 and was indexed with 135 categories by personnel from Reuters Ltd in 1996. By our statistics, it contains in total 19403 valid texts with average 5.4 sentences for each text. For convenience, the texts from 4 categories, “grain”, “crude”, “trade” and “interest” were selected as our target data set, on the condition that the number of sentences for each text in these categories is between 4 and 7. With this method, 252 texts from “grain”, 208 texts from “crude”, 133 texts from “interest” and 171 texts from “trade” were assigned as our target data set. Thus, the text collection for the experiment was performed and all the processing described in this paper was conducted on this data set.

III. Data Preprocessing

The purpose of this section is to explain the processing of the

.Manuscript received April 15, 2007. This work is supported by Ministry of Education, Culture, Sports, Science and Technology of Japan under the “Kanazawa Region, Ishikawa High-Tech Sensing Cluster of Knowledge-Based Cluster Creation Project” and partially supported by the National Natural Science Foundation of China under Grant No.70571078 and 70221001.

Wen Zhang is with the School of Knowledge Science, Japan Advanced Institute of Science and Technology, 1-1 Ashahidai, Tatsunokuchi, Ishikawa 923-1292, Japan (e-mail: zhangwen@ jaist.ac.jp).

Taketoshi Yoshida is with the School of Knowledge Science, Japan Advanced Institute of Science and Technology, 1-1 Ashahidai, Tatsunokuchi, Ishikawa 923-1292, Japan (e-mail: yoshida@ jaist.ac.jp).

Xijin Tang is with Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100080, P.R.China (e-mail: xjtang@amss.ac.cn).

texts selected in Section 2 into standard format and to extract multi-word features from the selected texts used as training data. The usually adopted preprocessing methods in text mining area were employed, such as stop word elimination, stemming, sentence boundary determin-ation. Furthermore, a hand-crafted method for multi-word extraction and post-processing of the multi-words into normalized features were included in this section as well. The following sections give the details of each procedure.

A. Stop word elimination

Stop words, or stopwords, is a name given to words which are filtered out prior to the processing of natural language data (text). They are generally regarded as 'functional words' which do not carry meaning. In this research, we obtained the stop words from USPTO (United States Patent and Trademark Office), which contains about 100 functional words in English [6]. By our observation concerning the occurring positions of the stop words in a sentence, three kinds of full matching of stop words are programmed to eliminate the stop words as "stop word + white space" for the beginning position, "white space + stop word + punctuation" for the end position and "white space + stop word + white space" for the middle position while "+" means "followed by".

B. Stemming

In English, as well as in many other languages, words usually occur in text in more than one form. It is always advantageous to eliminate this kind of variation before further processing. Generally, two kinds of main tasks are included in stemming in English: one is singular/plural regularization and another is present/past modification. In this research, only the singular/plural regularization was carried out, to transform all the singular words into plural words, because most of the multi-words identified were nouns rather than other parts of speech. For example, the "mln dlr" was regularized into "mln dlrs" in our text collection

C. Sentence boundary determination

In order to extract the multi-words from texts, it is necessary to break up the full text into separate sentences. Sentence boundary determination is essentially the problem of deciding which instances of a period followed by white space are sentence delimiters (full stop) and which are not. In this research, sentence boundary determination is also conducted with a hand-crafted program. The punctuation marks "?" and "!" are regarded as the end of a sentence, and many other rules were made to identify a period as a delimiter for a sentence, especially on distinguishing it from decimal point [3].

D. Multi-word extraction

Many methods could be applied to extract the multi-words from text, such as the frequency approach, correlation approach and mutual information approach, etc.

However, a newly programmed approach was proposed in this research to extract the multi-words from training texts. A basic hypothesis was conceived with a multi-word, that if a multi-word appears in a text and has the power enough to discriminate the category of this text from others, it should occur more than once in all the texts of this category. For example, "money market" is a significant multi-word feature for the category "interest", so we assumed that it is impossible for it to occur only once in all the texts of the "interest" category. Otherwise, it could not be regarded as a multi-word with the power of discrimination, which can be used to distinguish its category from other categories.

Based on the above hypothesis, the multi-words were extracted by the comparison between any two sentences in the same category so as to find out the same consecutive matching in both sentences. At the first step, the same parts were identified in both sentences. Secondly, the words of the same parts in the consecutive positions were extracted. Finally, the single words were eliminated from the extracted words. With this method, some meaningful multi-words were extracted from our selected data, such as "U.S. agriculture", "U.S. agriculture department", etc. Our multi-word extraction algorithm is shown below.

Input:

s_1, the first sentence

s_2, the second sentence

Output:

Multi-word extracted from s_1 and s_2.

Procedure:

$s_1 = \{w_1, w_2, \ldots, w_n\}$, $s_2 = \{w_1', w_2', \ldots, w_m'\}$, k=0

For each word w_i in s_1

  For each word w_j in s_2

    While(w_i = =w_j)

      k++

    End while

    If k>1

      combine the words from w_i to w_{i+k} into a multi-word

    End if

  End for

End for

Algorithm 1. Multi-word Extraction from Sentences

After the processing of multi-word extraction, 468 multi-words were obtained from training data "grain", 407 multi-words from training data "crude", 366 multi-words from training data "trade", and 273 multi-words from training data "interest".

E. Multiword post-processing with two strategies

The multi-words generated in Section 3.4 are usually common to two sentences in the same category. More often than not, the multi-words overlapped each other such as "U.S. agriculture department", "U.S. agriculture" and "agriculture department". For this reason, it is necessary to develop some practical strategies to make these multi-words into uniform features so as to represent textual documents in both training and test documents. In this research, we develop two types of strategies to post-process the multi-words extracted in

Section 3.4.

1) Strategy 1: Also named "decomposition strategy". With this strategy, if a short multi-word is included in a long multiword, the long multi-word will be eliminated from the multi-word feature set. For example, if "U.S. agriculture department", "U.S. agriculture" and "agriculture department" were extracted from texts, the "U.S. agriculture department" will be eliminated from the multi-word feature set because it includes the multi-word "U.S agriculture" and "agriculture department".

2) Strategy 2: Also named "combination strategy". With this strategy, if a short multi-word is included in a long multiword, the short multi-word will be eliminated from the multi-word feature set. For the same example with three multi-words as "U.S. agriculture department", "U.S. agriculture" and "agriculture department", the "U.S. agriculture" and "agriculture department" will be eliminated from the multi-word feature set because they are included in the multi-word "U.S agriculture department".

With the decomposition strategy, total 1514 multi-words obtained in Section 3.4 were decomposed into 984 multi-word features, and with the combination strategy, the extracted multi-words was combined into 1037 multi-word features.

IV. TEXT CLASSIFICATION WITH MULTI-WORD FEATURES

The main task devised for this section is to conduct the text classification using multi-word features obtained in Section 3. Firstly, both training and test documents in the text data set were represented using the multi-word features. Then, the information gain of each feature was calculated out to evaluate their discrimination power. Next, SVM was employed to classify the predefined test data. We examined the test documents in the text data set with linear kernel and nonlinear kernel respectively. Finally, the experiment results are presented so as to analyze the performance of the two strategies specified previously and the kernels of SVM.

A. Text representation with multiword features

In Section 3.5, two strategies were developed to post-process the extracted multi-words, and two types of multi-word feature sets were established. For this reason, two kinds of text representation methods were developed to represent the text using the above two different multi-word feature sets.

With the multi-word features generated from Strategy 1, simple full matching was used to represent texts in vector space model.

With the multi-word features generated from Strategy 2, a fuzzy matching method was developed to determine the occurrence of a multi-word in a text because the long multi-word usually does not occur fully in a text.

Two indicators were introduced to determine whether a long multi-word feature occurred in a given text. One is the ratio of single words included in the multi-word (multi-word comprises single words) occurring in the text, and the other is the minimum distance that these single words occurred in a given text.

For example, if we got two sentences as *"u.s. agriculture secretary richardlyng declined to confirm statements made today by a farm state congressman that the united states will offer subsidized wheat to the soviet union within the next 10 days"* and *"senate agriculture committee chairman patrick leahy (d-vt.) charged Japan with lying and cheating in its trade practices with U.S",* and the matching multi-word is "U.S. department of agriculture", we can calculate the ratio of single words is 2/4 and the minimum distance of single words is 2 for the first sentence and respectively 2/4, 18 for the second length. Following is the algorithm designed to calculate these two indicators.

Input:

$D = \{s_1, s_2, \ldots, s_n\}$ // D is a text in selected text collection and s_i is the ith sentence in D, n>=4 and n<=7;

$W = \{w_1, w_2, \ldots, w_m\}$ // W is a multi-word feature and w_i is the *ith* single word in W, m is the number of single words for a multi-word feature;

Output:

//In order to describe the program clearly, we adopt W' that means a set of single words of the W occurring in D.

r --- the ratio of single words of the multi-word feature occurring in the text, that is, |W'|/|W| ;

l --- minimum distance of the elements in W' occurring in D;

Procedure:

$S = \bigcup_{i=1}^{k} s_i$;

W' = ∅;

For each w_i in W

  If w_i exists in S

    W' = W'∪{w_i}

  End if

End for

r = |W'|/|W|;

For each w_i in W'

  L_j = {$a_{j,k}$ | $a_{j,k}$ is the *kth* position for w_j occurring in S};

End for

// the total number of L_j is |W'|;

L = ∅;

For each $a_{i,k}$ in L_1

  $L_{k'}$ = {$a_{i,k}$};

  While $|L_{k'}| < |W|$

    next = $|L_{k'}| +1$;

    Find $a_{next,*}$ in L_{next} which has the minimum difference from any one element in $L_{k'}$

    $L_{k'} = L_{k'} \cup \{ a_{next,*} \}$

  End while

  $L = L \cup \{ L_{k'} \}$;

End for

For each L_i in L

  b_i = maximum in L_i – minimum in L_i

End for

l = minimum b_i (i=1,…,|L|)

Algorithm 2. Occurrence Determination for Multi-word Feature in Strategy 2

Although the threshold of these two indicators should be set carefully and used jointly according to the practical application, the threshold for the first indicator was simply set as 0.5 and the latter one as the 1.75 times the length of the given multi-word feature in this paper. That is, if more than half number of single words in multi-word feature occurred in the given text and the minimum distance of these words occurred in the given text is no more than 1.75 times the length of this multi-word, this multi-word would be regarded as occurring in the given text. Otherwise, it would be regarded as absence in the given text. Take the previous two sentences for example; "the U.S. department of agriculture" will be regarded as occurrence in the first sentence (2/4 >= 0.5 and 2 <= 1.75*4) while it will be regarded as absence in the second sentence (2/4 >=0.5 but 18>=1.75*4).

B. Feature selection with Information gain

Information gain (IG) is usually employed as a term goodness criterion in the field of machine learning [7]. It is reported in [8] that 98% removal of unique terms yields text classification accuracy up to 89.2% with the Reuters-22173 data set. After the text representation in the previous section, IG value of each feature was calculated based on the entropy of classes and feature values, also the features were sorted in an ascending order according to their IG values. The text classification experiment was designed varying the removal percentages of the sorted features as shown in Table 1. The motivation for us to do this is that we wanted to examine the robustness of designed classifiers with multi-word features.

C. Learning with SVM

SVM is a classifier derived from a statistical learning theory by Vapnik and Chervonenkis, and it was first introduced in 1995 [9]. Some published results [10, 11] reported that it can obtain better performance than other learning methods in text classification tasks. In this research, we carried out the text classification task with SVM using linear and nonlinear kernel, respectively, in order to compare the performance between them. To simplify, the $(u*v)^1$ was used as the linear kernel, and the polynomial kernel $(u*v+1)^2$ was used as the nonlinear kernel for the task. Furthermore, the different representation strategies developed in Section 4.1 are combined. In details, four types of experiments were designed, as shown in Table 2. The motivation for us to devise these experiments is that we also wanted to compare the effect of representation strategy and kernel types on the classification, besides the single comparison of strategies or kernels.

D. Experimental results

According to the designs in Section 4.3, we carried out the experiments with the help of libSVM [12]. Also the popular 5-fold validation was employed to average the accuracy of each designed examination. Although the traditional precision and recall were usually utilized to evaluate the performance of text classifier, we only use the accuracy as the measure in this paper because our examination is a four-class classification task. The results of these experiments were as follows. It should be noticed here that each type of experiment was carried out at 11 different removal percentages of features, as specified in Section 4.2. In detail, Figures 1-4 show the results of the experiments we carried out.

TABLE I
PERCENTAGE OF HIGH IG VALUE FEATURES CORRESPONDING TO EACH DESIGNED TEST

Test No.	1	2	3	4	5	6	7	8	9
Removal Percentage (%)	0	50	70	75	80	85	90	92	95

TABLE II
EXPERIMENTS DESIGNED WITH TWO KINDS OF TEXT REPRESENTATION STRATEGIES AND KERNELS.

Experiment No.	1	2	3	4
Representation strategies	Strategy 1 Vs Strategy 2	Strategy 1 Vs Strategy 2	Strategy 1	Strategy 2
Kernel type	Linear	Nonlinear	Linear Vs Nonlinear	Linear Vs Nonlinear

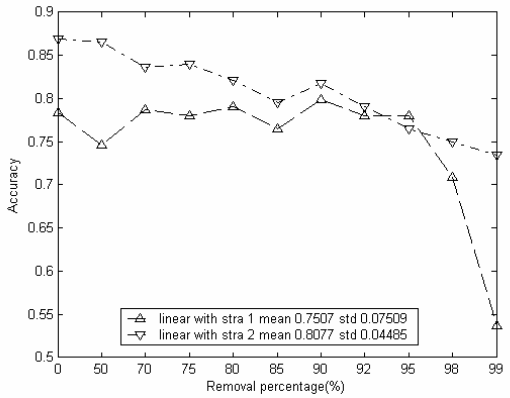

Fig. 1. Result from linear kernel with Strategy 1 and Strategy 2.

It can be seen from Figure 1 that on average Strategy 2 outperforms Strategy 1 with the linear kernel, except for the result at 95% removal of total features. As shown in Figure 2, we could convincingly deduce that a better performance was obtained with Strategy 2 than Strategy 1 on nonlinear kernel. The IG feature selection method exhibited its effectiveness in Strategy 1, because the best result of Strategy 1 was obtained at 95% removal of total features. From Figure 3, we can draw the conclusion that the performance of linear kernel was better than that of the nonlinear kernel on Strategy 1, and also the IG method exhibited its effectiveness in the feature selection, as it kept the performance stable when more and more features were removed from the feature set. Although it also seemed in Figure 4 that the performance of linear kernel

is better than that of nonlinear kernel with Strategy 2, the robustness of Strategy 2 is not very good because its performance declined dramatically when more and more features were removed from the feature set.

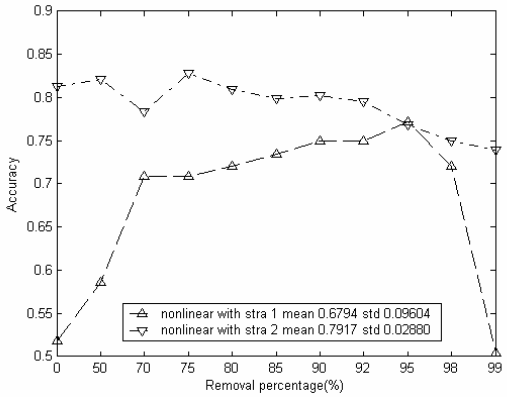

Fig. 2. Result from nonlinear kernel with Strategy 1 and Strategy 2.

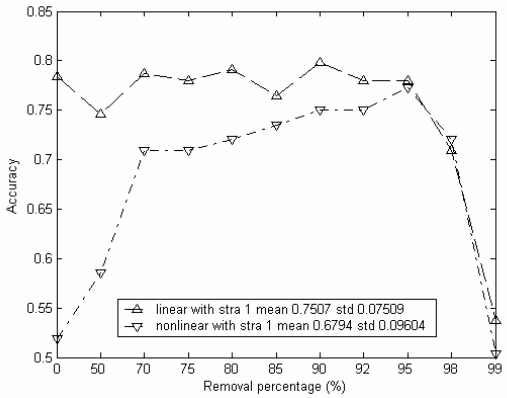

Fig. 3. Result from Strategy 1 with linear and nonlinear kernel

From the above analysis of Figure1-2, it can be summarized that Strategy 2 has better performance than Strategy 1, and from Figure 3-4, linear kernel also has a little better performance than the nonlinear kernel in text classification. The robustness of classifiers used in Figure 1-3 was fully demonstrated because the overall accuracy of classification is kept stable when more and more low IG value features were removed from the applied feature set. Further, it can also be deduced that the employment of different strategies on the text classification task has a greater influence than that of using different kernels.

V. Concluding Remarks

Multi-word feature is a newly practical method for text representation. In this paper, an automated method was proposed to extract the multi-words in the text based on our hypothesis that a multi-word cannot occur only once in the texts of its category. With this method, the multi-words are extracted from the texts of the same category. In order to normalize the initial multi-words into standard multi-word features, two strategies were developed: first is the decomposition strategy and second is combination strategy. Next, texts in data set were represented with these two types of different multi-word features, respectively. Then, IG method was employed to evaluate the importance of the features for text classification. The motivation of the feature evaluation was that we wanted to examine the robustness of each multi-word text classifier. Finally, the text classification was carried out with SVM in both linear and nonlinear kernels, and the results are compared on not only the different kernels but also the different strategies.

The experimental results demonstrated that in multi-word

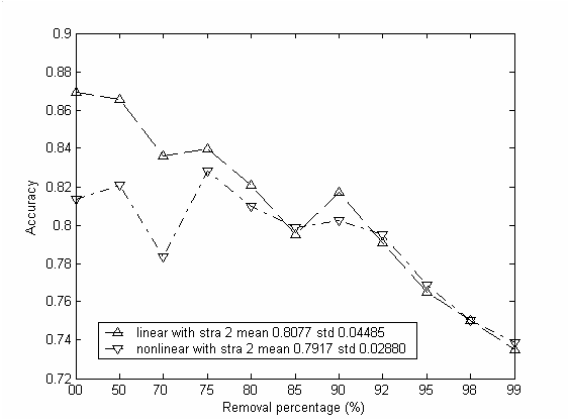

Fig. 4. Result from Strategy 2 with linear and nonlinear kernel

text classification, Strategy 2 outperforms Strategy 1, and linear kernel outperforms nonlinear kernel with SVM. However, it also appeared that Strategy 2 has poorer robustness than Strategy 1 when the low IG value features are removed from the applied feature set, and its performance also declined dramatically. Nevertheless, IG method was proved an effective approach for feature selection in most cases, because it kept the classification performance stable when the low IG value features were removed from the applied feature set gradually.

Although the experiment results have provided us with some clues on text classification with multi-word features, a generalized conclusion was not obtained from this examination because of the lack of theoretical proof. To be frank, our work is an initial step, and more examination and investigation should be undertaken for more convincing work

One of the promising directions in the text mining field concerns predictive pattern discovery from large amounts of documents. In order to achieve this goal, many kinds of work are involved in this field such as algorithm optimization, linguistics and machine learning. As for our further research, we would like to develop more precise algorithms [13] for multi-word extraction, and use linguistics in multiword extraction, instead of only literal extraction from texts. Also, the name entity will be considered seriously in extraction, as well as the adoption of support from a third-party dictionary. Another aspect that we also should advance is the improvement of the learning method such as SVM for multi-class classification. Despite the fact that the basic disciplines of the learning methods are well established, the

performance of classification will be improved if more processes are refined according to our practical research.

REFERENCES

[1] A. Chidanand, D. Fred, M.W. Sholom, *Automated learning of decision rules for text categorization.* ACM Transactions on Information Systems, 12(3), 1994, pp. 233-251.

[2] W. W. Cohen, *Learning to classify English text with ILP methods.* In Proceeding of 5th International Workshop on Inductive Logic Programming, 1995, pp. 3-24.

[3] I. Nitin, J. D. Fred, T. Zhang, *Text Mining: Predictive Methods for Analyzing Unstructured Information.* Springer Science and Business Media, Inc. 2005, pp15-37.

[4] A. Hotho, S. Staab, G.Stumme, *Ontologies improve text document clustering.* In Proceeding of IEEE International Conference on Data Mining (ICDM03), 2003, pp 541-544

[5] D. D. Lewis, *Reuters-21578 Text Categorization Test Collection.* Available: http://www.daviddlewis.com/resources/testcollections/reuters21578/

[6] USPTO Patent Full-Text And Image Database, *Stopwords.* Available: http://ftp.uspto.gov/patft/help/stopword.htm

[7] J. R. Quinlan, *Induction of decision trees*. Machine Learning, vol.1, 1986, pp 81-106

[8] Y. M. Yang, J. O. Pedersen, *A Comparative Study on Feature Selection in Text Categorization.* In Proceedings of the Fourteenth International Conference on Machine Learning, 1997, July 08-12, pp 412-420

[9] F. Mulier, "Vapnik-Chervonenkis (VC) Learning Theory and Its Application", *IEEE Trans on Neural Networks*, vol. 10,1999, pp 5-7

[10] C. Apte., F. Damerau., S.M. Weiss, *Text Mining with Decision Trees and Decision Rules.* In Conference on Automated Learning and Discovery, Carnegie-Mellon University, June 1998.

[11] Y.M. Yang, X. Lin, *A re-examination of text categorization methods.* In: Proceedings on the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, C A, 1999, pp 42-49.

[12] C. C. Chang, C, J, Lin, (2006, August) *libsvm 2.83.* Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[13] T. F. Smith, M. S. Waterman, *Identification of common molecular subsequences*. Journal of Molecular Biology, vol.147, (1981) pp195-197.