Keyword Reduction for Text Categorization using Neighborhood Rough Sets

Si-Yuan Jing

Sichuan Province University Key Laboratory of Internet Natural Language Intelligent Processing, Leshan Normal University, Leshan, 614000, China

Abstract

Keyword reduction is a technique that removes some less important keywords from the original dataset. Its aim is to decrease the training time of a learning machine and improve the performance of text categorization. Some researchers applied rough sets, which is a popular computational intelligent tool, to reduce keywords. However, classical rough sets model, which is usually adopted, can just deal with nominal value. In this work, we try to apply neighborhood rough sets to solve the keyword reduction problem. A heuristic algorithm is proposed meanwhile compared with some classical methods, such as Information Gain, Mutual Information, CHI square statistics, etc. The experimental results show that the proposed methods can outperform other methods.

Keywords: *Text Categorization; Keyword Reduction; Neighborhood Rough Sets; Heuristic Algorithm.*

1. Introduction

Automatic text categorization is always a hot issue in pattern classification and web data mining. There are several key steps in automatic text categorization, including text pre-processing, keyword reduction (or called feature selection), learning machine training, etc. Keyword reduction technique remove less important keywords from original keyword set, thus improve the result of text categorization as well as the training time. Statistical methods are the most commonly used in keyword reduction, such as statistics, information gain, mutual information [1], etc. Some researchers use some other techniques which can also play well in keyword reduction, such as rough sets [2, 13].

Rough sets is a powerful computational intelligent tool which proposed by Z. Pawlak in 1982 [3]. It has been proven to be efficient to handle imprecise, uncertain and vague knowledge. Attribute reduction is one of the most important applications for rough sets. In last decade, some researchers applied rough sets to solve the keyword reduction problem and obtained promising results. For example, A. Chouchoulas and Q. Shen firstly introduced a common method to apply rough sets to keyword reduction in text categorization [4]. Miao et al. proposed a hybrid algorithm for text categorization which combined traditional classifiers with variable precision rough set [2]. Y. Li et al. presented a novel rough set-based case-based reasoner for use in text categorization [5]. However all the research adopt classical rough sets model (i.e. Pawlak rough sets model), which can just deal with nominal values. The number of keyword is continuous, thus they must discrete data before computation. This will loss some important information and impacts the final result.

To address this problem, we try to apply neighborhood rough sets to solve the keyword reduction problem in this paper. A new heuristic algorithm for keyword reduction is proposed. We use two famous classifiers, i.e. kNN and SVM, to test the performance of the new algorithm. The experimental datasets are Reuters-21578 and 20-newsgroup. Experimental results show that the proposed method can achieve good performance and outperform some well-known statistical methods.

The rest of the paper is organized as follow: section 2 will recall some basic concept and knowledge of neighborhood rough sets as well as keyword reduction techniques. Section 3 introduces a new heuristic algorithm based on neighborhood rough sets for keyword reduction. Section 4 gives the experimental result and the discussion. Finally, section 5 gives a conclusion.

2. Preliminaries of Neighborhood Rough Sets

Neighbor theory is proposed by Lin et al. in 1990 [9]. It has been an important tool for many artificial intelligence tasks, such as machine learning, pattern classification, data mining, natural language understanding and information retrieval etc. [8]. Neighborhood rough set is a new model which combines neighbor theory and rough set theory together, and it can be used to deal with numeric value or hybrid value. In this section, we briefly recall some basic concepts and theory of neighborhood rough sets in [6, 10].

Definition 2.1: U is a non-empty finite set of objects, Δ is a given function. We say $NAS=(U,\Delta)$ is a neighbor approximation space where:

1). $\Delta(x_1,x_2)\geq 0, \Delta(x_1,x_2)=0$, if and only if $x_1=x_2$, $\forall x_1,x_2\in U$

2). $\Delta(x_1,x_2)=\Delta(x_2,x_1), \forall x_1,x_2\in U$

3). $\Delta(x_1,x_3)\leq \Delta(x_1,x_2)+\Delta(x_2,x_3), \forall x_1,x_2,x_3\in U$

We say Δ is the distance function in this neighbor space. There are many useful distance function can be used, such as Euclidean distance function, Minkowski distance, etc.

To construct a neighborhood rough set model for universe granulation on the numerical attribute, Hu et al. proposed a δ neighbor [7].

Definition 2.2: Given a neighbor space (U,Δ), $\forall x\in U$, $\delta\geq 0$, we say $\delta(x)$ is a δ neighbor of x whose centre is x and radius is δ, where:

$$\delta(x)=\{y|\Delta(x,y)\leq\delta, y\in U\} \tag{2.1}$$

It is easy to find that a set of δ neighbors will induce a family of information granules and they can be used to approximate any concepts in universe.

Definition 2.3: A neighborhood information system is a triple $NIS=(U,A,\Delta)$, where U is a non-empty finite set of objects; A is a non-empty finite set of attributes; Δ is a distance function; A and Δ form a family of neighborhood relation on U.

We give an example to illustrate how to compute δ neighbors in a give information system.

Table 1 shows an information table, in which there are six objects and each of them has two attributes. We use an improved Euclidean distance proposed in [10] as the distance function. The results are shown in table 2. If we set threshold to 0.3, we can get: $\delta(x_1)=\{x_1,x_5\}$, $\delta(x_2)=\{x_2,x_4\}$, $\delta(x_3)=\{x_3,x_5\}$, $\delta(x_4)=\{x_2,x_4\}$, $\delta(x_5)=\{x_1,x_3,x_5\}$, $\delta(x_6)=\{x_6\}$.

Table 1. An illustrated example for δ neighbor computation

obj.	attribute1	attribute2	obj.	attribute1	attribute2
x_1	2	4	x_4	10	8
x_2	12	7	x_5	3	3
x_3	5	3	x_6	6	9

Table 2. The results of distance computation

	x_1	x_2	x_3	x_4	x_5	x_5
x_1	0	1.11	0.34	1.04	0.14	0.92
x_2	1.11	0	0.97	0.26	1.12	0.68
x_3	0.34	0.97	0	0.97	0.2	1
x_4	1.04	0.26	0.97	0	1.09	0.43
x_5	0.14	1.12	0.2	1.09	0	1.04
x_5	0.92	0.68	1	0.43	1.04	0

Based on above definition, we can construct lower approximation and upper approximation in neighborhood rough sets.

Definition 2.4: Given a neighborhood approximation space $NAS=(U,\Delta)$ and $X\subseteq U$, the lower approximation and upper approximation about X can be defined as:

$$\begin{cases}\underline{N}(X)=\{x|\delta(x)\subseteq X, \forall x\in U\}\\ \overline{N}(X)=\{x|\delta(x)\cap X\neq\varnothing, \forall x\in U\}\end{cases} \tag{2.2}$$

And $\forall X\subseteq U, \underline{N}(X)\subseteq X\subseteq\overline{N}(X)$. Moreover, we can respectively define the boundary domain, positive domain and negative domain as follow:

$$BN(X)=\overline{N}(X)-\underline{N}(X) \tag{2.3}$$

$$POS(X)=\underline{N}(X) \tag{2.4}$$

$$NEG(X)=U-\overline{N}(X) \tag{2.5}$$

Positive domain $POS(X)$ represents the granules contained by X; $NEG(X)$ represents the granules not contained by X; $BN(X)$ represents the granules partially contained by X. If $\overline{N}(X)=\underline{N}(X)$, we say X in this neighborhood approximation space is definable; otherwise, it's indefinable, namely it's rough.

In practical environment, some data are noisy. The model defined above is crisp and without tolerance ability of noise. Therefore, we adopt variable precision rough sets to improve formula 2.2. The function $card(\cdot)$ is to count the object number in a set. Parameter k is usually set between 0.5 to 1.

$$\begin{cases}\underline{N^k}(X)=\left\{x\left|\left(\frac{card(\delta(x)\cap X)}{card(\delta(x))}\right)\geq k, \forall x\in U\right.\right\}\\ \overline{N^k}(X)=\left\{x\left|\left(\frac{card(\delta(x)\cap X)}{card(\delta(x))}\right)<1-k, \forall x\in U\right.\right\}\end{cases} \tag{2.6}$$

Data in many pattern classification problems, such as the keyword reduction problem, can be represented by a decision table. Therefore, we introduce some concepts and a neighborhood rough sets model to handle decision table.

Definition 2.5: Given a neighborhood information system $NIS=(U,A,\Delta)$, we say it's a neighborhood decision table $NDT=(U,C\cup D,\Delta)$ if $A=C\cup D$, C represents the condition attribute set, D represents the decision attribute, and C forms a family of neighborhood relation on U.

Definition 2.6: Given a neighborhood decision table $NDT=(U,C\cup D,\Delta)$, $\forall P\subseteq C$, decision attribute D divide the universe U into some equivalence classes, $X_1,X_2,\cdots,X_n$, the lower approximation and upper approximation about D relative to P can be defined as:

$$\begin{cases}\underline{N}_P^k(D)=\underline{N}_P^k(X_1)\cup\underline{N}_P^k(X_2)\cup\cdots\cup\underline{N}_P^k(X_n)\\ \overline{N}_P^k(D)=\overline{N}_P^k(X_1)\cup\overline{N}_P^k(X_2)\cup\cdots\cup\overline{N}_P^k(X_n)\end{cases} \quad (2.7)$$

Obviously $POS_P(D)=\underline{N}_P^k(D)$, $NEG_p(D)=U-\overline{N}_P^k(D)$, $BN_p(D)=\overline{N}_P^k(D)-\underline{N}_P^k(D)$.

3. Algorithm

Generally speaking, there are two steps in text categorization task, i.e. text preprocessing and classifier building. Text preprocessing has five sub-steps which are word extracting, stopping word omitting, word stemming, keyword reducing and keyword weighting. The former three sub-steps is responsible to extract words from raw text meanwhile delete some useless words which are in a given stopping word list, and combine the words with same stem. After that, we commonly use VSM (Vector Space Model) to represent the texts. The last two sub-steps, i.e. keyword reducing and weighting, remove keywords which have little help to classification and assign a new value to each keyword which can emphasize the importance of different keywords. Classifier building in text categorization is same with other machine learning tasks and the explanation is omitted here.

Keyword reduction is a key step in text categorization. Its aim is to remove some keywords which are less important. A good algorithm of keyword reduction can keep the most important keywords and it will greatly improve classifier's performance as well as decrease the training time.

In this section, we firstly give some important concepts and theories to measure the importance degree of keywords based on neighborhood rough sets, and then we propose a heuristic algorithm for key reduction in text categorization.

To apply neighborhood rough set s to text categorization, we must represent the problem by neighborhood decision table $NDT=(U,C\cup D,\Delta)$, where U represents a set of samples (i.e. texts), C represents a set of keywords, D represents the label of documents, Δ is a pre-defined distance function. Then, for $\forall P\subseteq C$, we can define the dependency relation between P and D as:

$$\gamma_P(D)=\frac{card(POS_P(D))}{card(U)} \quad (3.1)$$

Obviously, $0\le\gamma_P(D)\le 1$. The dependency $\gamma_P(D)$ reflects the dependence degree of D relative to P. This definition is in accordance with the classical rough set theory.

Theorem 3.1: Given a neighborhood information system $NIS=(U,A,\Delta)$ and threshold δ, $P_1,P_2\subseteq C$, if $P_1\subseteq P_2$, then $\forall x\in U,\delta_{P_1}(x)\subseteq\delta_{P_2}(x)$.

Proof. It is obvious thus the proof is omitted here.

Theorem 3.2: Given a neighborhood decision table $NDT=(U,C\cup D,\Delta)$, $P_1,P_2\subseteq C$, if $P_1\subseteq P_2$, then $x\in POS_{P_1}(D)\Rightarrow x\in POS_{P_2}(D)$.

Proof. Without loss of generality, let $\forall x\in POS_{P_1}(D)$, D_j represents the jth equivalence class divided by decision D. According to theorem 3.1, if $P_1\subseteq P_2$, then $\delta_{P_2}(x)\subseteq\delta_{P_1}(x)$. Thus, $x\in POS_{P_2}(D)$. This completes the proof.

Theorem 3.3: $\gamma_P(D)$ is monotone, that's to say, if $P_1\subseteq P_2\subseteq\cdots\subseteq C$, then $\gamma_{P_1}(D)\le\gamma_{P_2}(D)\le\cdots\le\gamma_C(D)$.

Proof. According to theorem 3.2, if $P_1\subseteq P_2\subseteq\cdots\subseteq C$, then $POS_{P_1}(D)\le POS_{P_2}(D)\le\cdots\le POS_C(D)$. According to formula 3.1, we can get $\gamma_{P_1}(D)\le\gamma_{P_2}(D)\le\cdots\le\gamma_C(D)$. This completes the proof.

Theorem 3.3 shows an important property of dependency function. According to it, we can define the significance of each keyword a relative to decision D (i.e. labels), as follow:

$$sig(a,P,D)=\gamma_{P\cup a}(D)-\gamma_P(D) \quad (3.2)$$

Based on the significance metric, we can design a heuristic keyword reduction algorithm. The algorithm begins with a

relative core set searched by an algorithm proposed in [11]. For each iteration, it computes significance of all rest keywords and selects a keyword that has the highest significance and adds it to a final keyword set. The algorithm does not end until the pre-defined keyword number is achieved. The pseudo-code is shown below.

NRS-KR Algorithm:

Input: $NDT=(U,C\cup D,\Delta)$, a predefined threshold λ and keyword number N

Output: A keyword reduction, denoted as red

1. $red=core$;
2. **for** k=1:$(N-|core|)$ **do**
3. m = 0, candidate=null;
4. **for** $\forall a_i \in C-red$ **do**
5. **if** $sig(a_i,red,D)$ >m **do**
6. candidate = a_i ; m= $sig(a_i,red,D)$;
7. **end if**
8. **end for**
9. $red=red\cup$candidate;
10. **end for**

4. Experiments

In this section, we perform experiments to evaluate the proposed algorithm. Firstly, we will introduce experimental datasets, the selected methods which are used for comparison, and performance metrics in the experiments. Then, we will explain how we perform the experiments and show the results.

4.1 Selected Datasets

We select two datasets to evaluate the proposed algorithm. The first one is Reuters-21578, which is a standard text categorization benchmark and collected by the Carnegie group from the Reuters newswire in 1987. We only choose the most populous ten categories from this corpus, which are *earn*, *acq*, *coffee*, *sugar*, *trade*, *ship*, *crude*, *interest*, *money-fx*, gold. The second one is 20-newsgroup, which contains approximately 20,000 newsgroup texts being divided nearly among 20 different newsgroups.

4.2 Methods for Comparison

We choose four methods to compare with the proposed method. The first one is classical rough sets based keyword reduction algorithm and the other three are χ^2 statistics (CHI), Information Gain (IG) and Mutual Information (MI), respectively.

The χ^2 statistics measures the lack of independence between keyword and label and can be compared to the χ^2 distribution with one degree of freedom to judge extremeness. The χ^2 statistics is defined as formula 4.1, where t means term (i.e. keyword) and c means category (i.e. label), A is the number of times t and c occur at the same time, B is the number of times t occurs without c, C is the number of times c occurs without t, and D is the number of times neither t or c occur. N is the total number of texts in dataset.

$$\chi^2(t,c)=\frac{N\times(AD-CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \quad (4.1)$$

The information gain measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a text. The information gain of a term t is defined as follow.

$$\begin{aligned} IG(t)=&-\sum_{i=1}^{m}\Pr(c_i)\times\log\Pr(c_i)+\Pr(t)\times \\ &\sum_{i=1}^{m}\Pr(c_i|t)\times\log\Pr(c_i|t)+\Pr(\bar{t})\times \\ &\sum_{i=1}^{m}\Pr(c_i|\bar{t})\times\log\Pr(c_i|\bar{t}) \end{aligned} \quad (4.2)$$

Mutual information is a criterion commonly used in statistical modeling. The basic idea behind this metric is that the larger mutual information is, the higher the co-occurrence probability between term t and category c is. MI can be defined as follow. The symbol N, A, B, C is same with formula 4.1.

$$MI(t,c)=\log\frac{\Pr(t,c)}{\Pr(t)\Pr(c)}\approx\log\frac{A\times N}{(A+C)\times(A+B)} \quad (4.3)$$

4.3 Performance Metrics

For different purpose, there are several performance metrics can be chose in text categorization evaluation, including recall, precision, F-measure, Micro-average of F-measure and Macro-average of F-measure. In the experiments, we choose Micro-average of F1-measure (abbreviated as F1-Micro) and Macro-average of F1-measure (abbreviated as F1-Macro) as the performance metrics. Since the two metrics are based on other basic metrics, we briefly explain here.

Firstly, we give the definition of recall and precision, as follow.

$$recall = \frac{a}{a+c} \times 100\% \quad (4.4)$$

$$precision = \frac{a}{a+b} \times 100\% \quad (4.5)$$

Where a is the number of texts correctly classified to that class, b is the number of texts incorrectly classified to that class, c is the number of text incorrectly rejected from that class. Based on the two definitions, we further give the definition of F1-measure as formula 4.6.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (4.6)$$

Finally, the F1-Micro and F1-Macro are defined as follow.

$$\begin{aligned} F1-macro &= \sum_{i=1}^{n} \frac{N_i}{N} \times F1_i \\ &= \sum_{i=1}^{n} \frac{N_i}{N} \times \frac{2 \times precision_i \times recall_i}{precision_i + recall_i} \end{aligned} \quad (4.7)$$

$$F1-Micro = \frac{2 \times precision_{total} \times recall_{total}}{precision_{total} \times recall_{total}} \quad (4.8)$$

Where $precision_{total}$ and $recall_{total}$ is computed by the total number of a, b, c.

4.4 Experimental Results

The experiments are performed on Matlab R2012b. We choose two popular classifiers for evaluation, which are SVM and kNN. SVM is implemented by libsvm which is a widely used open-source library for SVM [12]. Linear kernel is adopted in SVM. For category ranking in kNN, we individually select 10 nearest neighbors for each text and adopt the category which is the most. Cosine similarity is used for neighbor's selection. Before keyword reduction, we need to preprocess the raw datasets. Firstly, we extract words from the raw texts. These 'keywords' will not be used to construct VSM immediately. We remove some useless words based on a stop word list which is obtained from Internet and has 887 stop words. After that, we use the Porter Stemming Algorithm to combine the words that has same stem. For the proposed algorithm, the threshold is set to 0.2. Euclidean distance is adopted as the distance function.

We compare the proposed algorithm (NRS) with other four methods, i.e. classical rough sets (RS), statistics (CHI), information gain (IG) and mutual information (MI). The results are shown from figure 2 to 5. In experiments, we choose a dataset and apply different methods to select a number of keyword and evaluate by a specific classifier. The keyword number changes from 1000 to 10000. From figure 2 to 5, we can easily find that the proposed algorithm plays very well and it can select much better keywords than other four methods. Furthermore, we list the average results of each algorithm in table 3 and 4. From table 3 and 4, we can see that the proposed algorithm also outperforms other methods in this aspect.

Table 3: Performance of different algorithms on Reuters-21578 (%)

Methods	SVM		kNN	
	F1-Micro	F1-Macro	F1-Micro	F1-Macro
NRS	0.9590	0.9135	0.9206	0.8628
RS	0.9512	0.8912	0.8892	0.8258
CHI	0.9550	0.9068	0.8687	0.8501
IG	0.9489	0.8756	0.8633	0.8316
MI	0.8969	0.7311	0.7932	0.6624

Table 4: Performance of different algorithms on 20-newsgroups (%)

Methods	SVM		kNN	
	F1-Micro	F1-Macro	F1-Micro	F1-Macro
NRS	0.7074	0.7070	0.6495	0.6447
RS	0.6253	0.6183	0.5904	0.5835
CHI	0.6374	0.6301	0.5982	0.5914
IG	0.5713	0.5669	0.5309	0.5225
MI	0.4308	0.4240	0.3834	0.4104

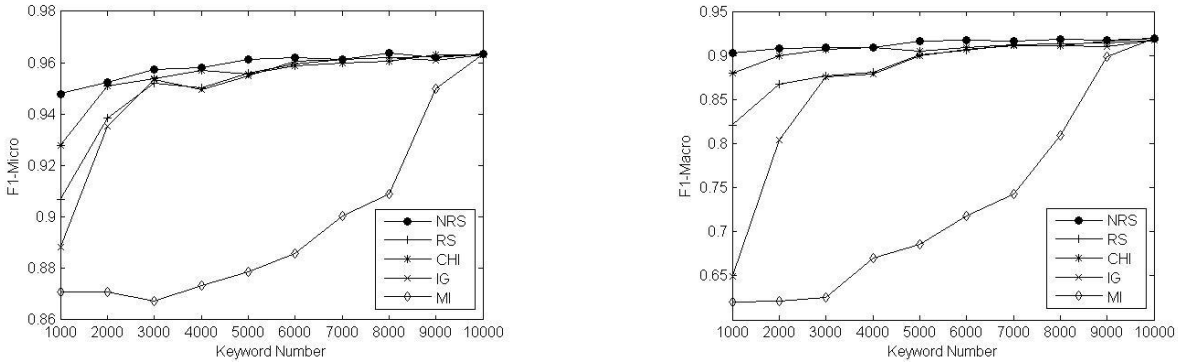

Fig. 1 Comparison of different algorithms on Reuters-21578 using SVM Classifier(a) F1-Micro; (b) F1-Macro

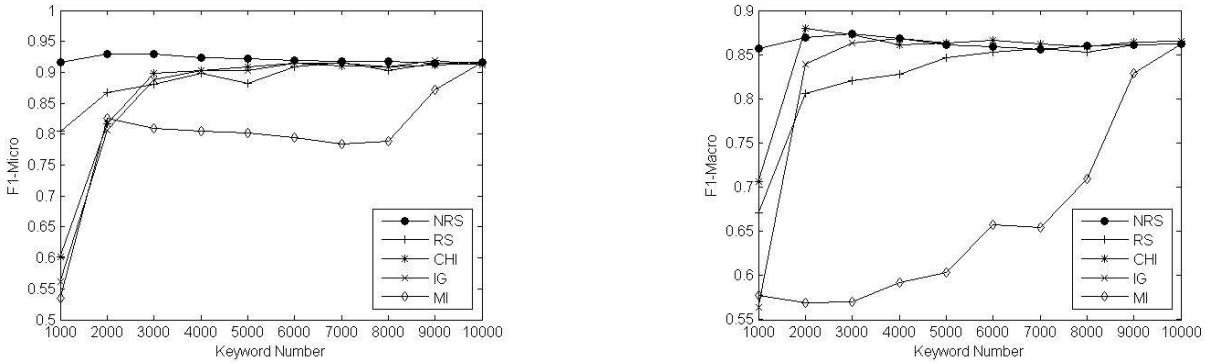

Fig. 2 Comparison of different algorithms on Reuters-21578 using kNN Classifier(a) F1-Micro; (b) F1-Macro

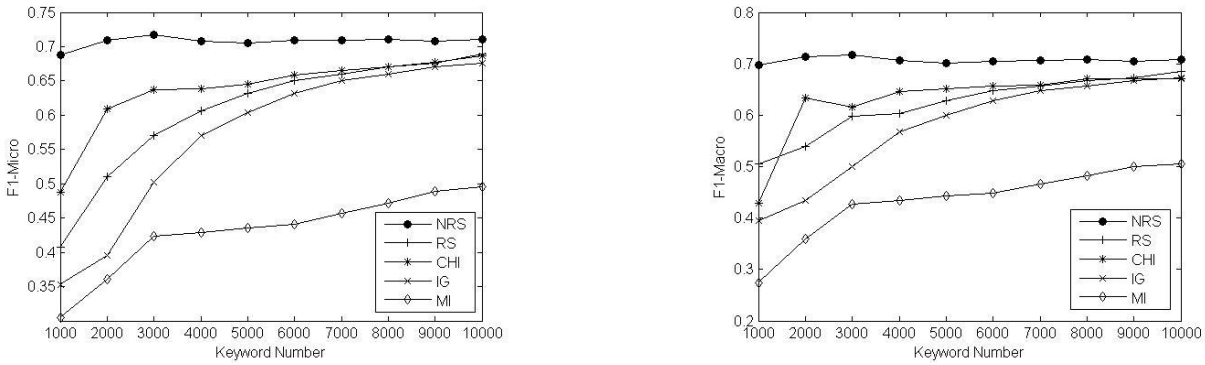

Fig. 3 Comparison of different algorithms on 20-newsgroups using SVM Classifier(a) F1-Micro; (b) F1-Macro

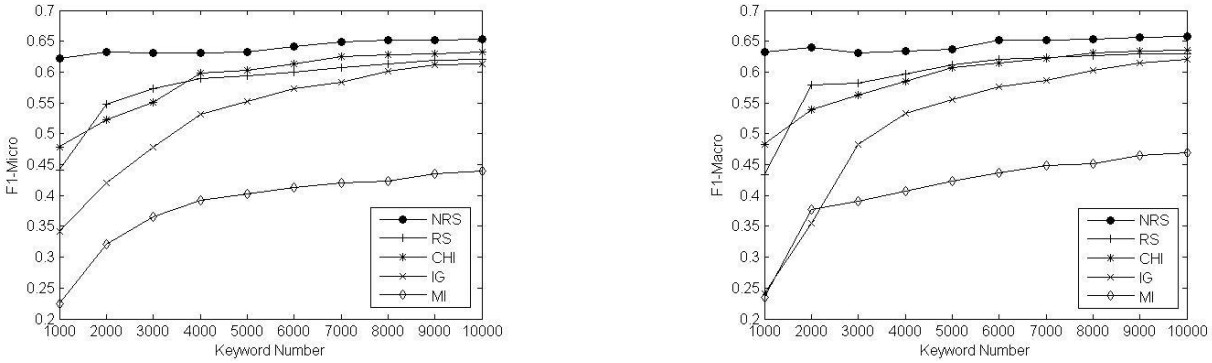

Fig. 4 Comparison of different algorithms on 20-newsgroups using kNN Classifier (a) F1-Micro; (b) F1-Macro

5. Conclusions

In this paper, we propose a novel way to keyword reduction in text categorization using neighborhood rough sets. As we knowledge, neighborhood rough sets is a model which combine classical rough sets and neighbor theory, and it is efficient to handle numeric value or even hybrid value. We apply some important concepts and theories of neighborhood rough sets to keyword reduction problem, moreover design a heuristic algorithm. The Experimental results prove that the proposed algorithm play well on keyword reduction problem and it outperform some classical methods on Reuters-21578 and 20-newsgroups.

Acknowledgments

This work is supported by the Scientific Research Fund of Sichuan Provincial Education Department (Grand No. 14ZB0247), Scientific Research Fund of Leshan Normal University (Grand No. Z1325), and Sichuan Provincial Department of Science and Technology Project (No.2014JY0036).Instead, write “F. A. Author thanks ... .” Sponsor and financial support acknowledgments are also placed here.

References

[1] Y. Yang, "A comparative study on feature selection in text categorization", in Proceeding of 14th International Conference on Machine Learning (ICML’97), 1997, pp. 412-420.

[2] D. Q. Miao, Q. G. Duan, H. Y. Zhang et al., "Rough set based hybrid algorithm for text classification", Expert Systems with Applications, Vol. 36, 2009, pp. 9168-9174.

[3] Z. Pawlak, "Rough sets", International Journal of Computer and Information Science, Vol. 11, 1982, pp. 341-356.

[4] A. Chouchoulas, Q. Shen, "Rough Set-Aided Keyword Reduction for Text Categorization", Applied Artificial Intelligence, Vol. 15, 2001, pp. 843-873.

[5] Y. Li, S. C. K. Shiu, S. K. Pal et al., "A rough set-based case-based reasoner for text categorization", International Journal of Approximate Reasoning, Vol. 41, 2006, pp. 229-255.

[6] Q. H. Hu, D. R. Yu, Z. X. Xie, "Numerical attribute reduction based on neighborhood granulation and rough approximation", Chinese Journal of software, Vol. 19, 2008, pp. 640−649.

[7] Q. H. Hu, D. R. Yu, J. F. Liu et al., "Neighborhood rough set based heterogeneous feature subset selection", Information Sciences, Vol. 178, 2008, pp. 3577-3594.

[8] H. Wang, "Nearest neighbors by neighborhood counting", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 28, 2006, pp. 942-953.

[9] T. Y. Lin, Q. Liu, K. J. Huang, "Rough sets, neighborhood systems and approximation", in the 5th International Symposium on Methodologies of Intelligent Systems, 1990, pp. 130-141.

[10] S. Y. Jing, K. She, S. Ali, "A Universal neighbourhood rough sets model for knowledge discovering from incomplete heterogeneous data", Expert Systems, Vol. 30, 2013, pp.89-96.

[11] T. F. Zhang, J. M. Xiao, X. H. Wang, "Algorithms of attribute relative reduction in rough set theory", Chinese Journal of Electronic, Vol. 33, 2005, pp. 2080-2083.

[12] C. C. Chang, C. J. Lin, "LIBSVM: a library for support vector machines", ACM Transactions on Intelligent Systems and Technology, Vol. 27, 2011, pp. 1-27.

[13] S. Ali, S. Y. Jing, K. She, "Profit-Aware DVFS Enabled Resource Management of IaaS Cloud", International Journal of Computer Science Issues, Vol. 10, 2003, pp. 237-247.

Si-Yuan Jing received the Ph. D. degree from University of Electronic Science and Technology of China, Chengdu China, in 2013. He is currently an associate professor with Leshan Normal University. He is the member of ACM and CCF. His research interest is in computational intelligence, big data mining etc.