Artif Intell Rev (2006) 26:191–209
DOI 10.1007/s10462-007-9049-y

Combining rough decisions for intelligent text mining using Dempster's rule

Yaxin Bi · Sally McClean · Terry Anderson

Published online: 10 November 2007

Abstract An important issue in text mining is how to make use of multiple pieces knowledge discovered to improve future decisions. In this paper, we propose a new approach to combining multiple sets of rules for text categorization using Dempster's rule of combination. We develop a boosting-like technique for generating multiple sets of rules based on rough set theory and model classification decisions from multiple sets of rules as pieces of evidence which can be combined by Dempster's rule of combination. We apply these methods to 10 of the 20-newsgroups—a benchmark data collection (Baker and McCallum 1998), individually and in combination. Our experimental results show that the performance of the best combination of the multiple sets of rules on the 10 groups of the benchmark data is statistically significant and better than that of the best single set of rules. The comparative analysis between the Dempster–Shafer and the majority voting (MV) methods along with an overfitting study confirm the advantage and the robustness of our approach.

Keywords Rule induction · Text mining · Rough set · Dempster's rule of combination

1 Introduction

For the problem of text mining, in general, and classification, in particular, different machine learning methods obtain different degrees of success, but none of them is totally perfect, and usually they are not good enough for real-world applications. Therefore it is desirable to develop an effective methodology for taking advantage of different learning methods by combining their decisions, so that more precise decisions can be achieved. In this research,

Y. Bi (✉) · T. Anderson
School of Computing and Mathematics, University of Ulster, Newtownabbey, Antrim
BT37 0QB, Northern Ireland, UK
e-mail: y.bi@ulster.ac.uk

S. McClean
School of Computing and Information Engineering, University of Ulster, Coleraine, Londonderry
BT52 1SA, Northern Ireland, UK

we investigate a novel approach for combining multiple decisions inferred from multiple sets of rules by using Dempster's rule of combination. The advantage of our approach is its ability to combine multiple sets of rules into a highly accurate classification rule by modelling the accumulation of evidence as boosting methods do (Freund and Schapire 1996).

Boosting learning techniques have been developed by Freund and Schapire (1996). These techniques work by repeatedly running a given weak learning algorithm on various distributions over training data, and then combining the classifiers produced by the weak leaner into a single composite classifier. The algorithms built on boosting techniques have attractive theoretical properties, and have also been shown to perform well experimentally on more standard machine learning tasks (Quinlan 1996; Freund and Schapire 1997). Although such methods produce complex solutions, they have the advantage that the extra computation they require is known in advance—if T classifiers are generated by a learning algorithm, then the boosted solutions require T times the computational effort of the corresponding algorithm (Quinlan 1996; Weiss and Indurkhya 2000).

Several derivatives based on boosting techniques have been developed in recent years. In Friedman et al. (1998), a method for decomposing a decision tree of larger size into a number of very small trees in terms of truncated tree induction was developed. Their work shows that a single decision tree is often dramatically outperformed by voting based on multiple smaller decision trees. In Cohen and Singer (1999), boosting techniques are used in a system called SLIPPER to generate a set of weighted rules. This approach generally outperforms standard rule induction techniques and these rules also maintain clarity of explanation. In Schapire and Singer (2000) and Nardiello et al. (2003), the authors report their work on text categorization using boosting techniques, respectively. Of particular interest to our work is the approach developed by Weiss and Indurkhya (2000), where a lightweight rule induction (LRI) method is used to generate compact Disjunctive Normal Form (DNF) decision rules. For each class, there are an equal number of the corresponding unweighted rules. An unseen instance is classified by all the rules and the instance is assigned to the class with the most satisfied rules.

Rough Set theory is a mathematical tool for dealing with vagueness and uncertainty (Pawlak 1991), and it is increasingly being applied to various data or text mining tasks (Guang and Bell 1998; Chouchoulas and Shen 2001; Bi et al. 2004b). In our work we develop a rough sets-based learning algorithm for generating multiple sets of weighted rules on the basis of work described in (Chouchoulas and Shen 2001; Grzymala-Busse 1992), instead of using conventional boosting techniques. The main idea of our algorithm is to evaluate dependency between the attributes of a data set. This dependency between different combinations of attributes is be analysed, and the subsets of attributes with the same dependency as that of the whole set of attributes are generated. Each of the subsets could have the same discriminant ability as the entire set of attributes does. The subsets of attributes generated in this way are called *reducts*, and they are in turn used to construct multiple sets of weighted rules, each of which plays the same role as a classifier in classification.

There are some recent research activities concerning how to combine classification decisions and what is best way to combine classification decisions (Opitz and Maclin 1999; Whiteaker and Kuncheva 2003; Tumer and Ghosh 2002; Kittler 1998). Although similar ideas to ours have been mentioned in (Xu Krzyzak and Suen 1992; Denoeux 2000), they have not been systematically investigated the method and techniques for incorporating the Dempster—Shafer (DS) theory of evidence into combining decisions of multiple sets of rules, which are, in particular, generated by rough sets-based methods. The distinguishing aspect of our approach is we have developed a method for modelling classification decisions from each set of rules as a piece of evidence where rule strengths are degrees of belief that indicate how likely it is that new instances belong to classes. This provides an effective way to improve decisions making in classification by accumulating more pieces of evidence derived from discovered rules (knowledge).

Various experiments have been conducted on the 10 out of 20 newsgroups of the benchmark data (Baker and McCallum 1998) to evaluate our rough sets-based algorithm and the DS method. In order for our experiments to faithfully reflect the performance of combining multiple sets of rules, we randomly partition the benchmark data set into a training set, a validation set and a testing set with different sizes to avoid likely overfitting that may be caused in experiments. The validation set is used for optimizing the threshold α and selecting different sets of rules for combinations and the test set is used to evaluate the performance of our algorithms.

Our experimental results show that the estimated performance of the best combination of the multiple sets of rules using our DS method is consistently better that of the best single set of rules. A t-test is utilized to show that this performance difference is statistically significant at the 0.05 level. A comparative analysis between the DS and the MV methods is also carried out, demonstrating the advantage of DS over MV in combining multiple sets of rules. To examine the robustness of our method, we also analyze the generalization performance of the best single set of rules and the best combined sets of rules on the validation and testing sets, and we have found that DS has more overfitting than MV.

2 Rough Sets for rule generation

Inductive learning can be loosely defined as *learning general rules* from specific instances (Mitchell 1999). In other words, induction learning can be seen as a process of synthesizing mappings from a sample space consisting of individual instances. The result often is to reduce the space, leading to a new smaller space containing a set of representative instances, which serves the same role as the original one. To develop the rough sets-based learning algorithm, we introduce several essential concepts below (Pawlak 1991; Guang and Bell 1998).

2.1 Definitions and notation

In rough sets, data objects or instances are organized into decision systems. A decision system can defined as $S = \langle U, A, V, f \rangle$, where $U = \{u_1, \ldots, u_{|U|}\}$ is a collection of objects, $A = \{a_1, \ldots, a_{|A|}\}$ is a set of attributes, $V = \{V_{a1,\ldots,} V_{a|A|}\}$ is a set of attribute values, in which $V_{a_i} = \{V_{a_{i1}}, \ldots, V_{a_{ik}}\}$ is the domain of attribute a_i, and $V_{a_{i\,j}}$ is a categorical value $(1 < k \leq |V_{a_i}|)$. We define f as a function over U, A, and V, and $f(u, a)$: $U \times A \rightarrow V_a$ assigns particular values from the domain of attributes to instances such that $f(u, a) \in V_a$, for $a \in A$ and $u \in U$. Attribute A is further divided into two parts—the condition attribute W and decision attributes H such that $A = W \cup H$ and $W \cap H = \phi$.

Table 1 presents a decision system, containing information about patients' symptoms and the disease from which patients suffer. The symptoms and the disease name correspond to the attribute names. Each row corresponds to an individual patient's symptoms and a possible disease in terms of object and relationships between symptoms and disease are derived from doctors' subjective judgement, historical records of patients, and so forth.

Definition 1 With an attribute $a \in A$, two instances $u, v \in U$ are defined as an *equivalence relation* over U if and only if $f(u, a) = f(v, a)$, denoted by τ.

Table 1 A decision system (Pawlak 1991), where *he* = Headache, *m* = Muscle-pain, *t* = Temperature, *h* = Flu, y = yes, n = no, hi = high, vhi = very high)

U/A	*he*	*m*	*t*	$\cup$	*h*
1	n	y	hi		y
2	y	n	hi		y
3	y	y	vhi		y
4	n	y	n		n
5	y	n	hi		n
6	n	y	vhi		y

Definition 2 With an *equivalence relation* τ associated with the set of attributes, a partition operation on a decision system under τ_A is defined as U/τ_A (U/A for short), where $U/A = \{X_1, \ldots, X_n\}$, and each X_i is called an *equivalence class*, such that for any two instances u and v, if $u, v \in X_i$ and $a \in A$, then $f(u, a) = f(v, a)$.

Definition 3 Given a subset of attributes $B \subseteq A$, if there is $Q \subseteq B$, s.t. $U/\tau_B = U/\tau_Q$ and Q is minimal among all subsets of B, then Q is defined as a *reduct* of B. The component attributes of a reduct are significant so that none of them can be omitted. Notice that more than one reduct of B may exist.

Definition 4 Suppose the decision attribute H is a singleton, i.e. $H = \{h\}$, and let $U/\tau_d = \{X_1, \ldots, X_k\}$ be a partition over U with respect to h, for each subset $X \subseteq U/\tau_h$ and a subset of condition attributes $B \subseteq W$, we associate two subsets with X as follows:

$$\underline{B}X = \cup\{Y \in U/B | Y \subseteq X\} \tag{1}$$

$$\overline{B}X = \cup\{Y \in U/B | Y \cap X \neq \phi\} \tag{2}$$

where $\underline{B}X$ and $\overline{B}X$ are called the *lower* and *upper approximations*, respectively, and the difference between $\underline{B}X$ and $\overline{B}X$ is called boundary, denoted by BN_B. To express the quantitative relationship between a subset of condition attributes $B \subseteq C$ and a decision attribute h, we define a measure, the *dependency degree*, denoted by $\gamma_B(X)$ below:

$$\gamma_B(H) = \sum_{X \in U/H} \frac{|\underline{B}X|}{|U|} \tag{3}$$

Definition 5 A decision rule is the logical expression of cause-effect relation which is composed of the antecedent and the consequent in the following form:

$$\boxed{\text{IF } (a_1 = v_1) \wedge (a_2 = v_2) \wedge_{\ldots} \wedge (a_{|A-1|} = v_{|V|}) \text{ THEN } h_i|stg}$$

where $a_i \in A$ is a attribute, $v_i \in V_{ai}$ is an attribute value of a_i and $h_i \in V_h$ (or $h_i \in 2^{Vh}$) is a decision attribute value, and *stg* is a degree of belief for possible conclusions which lies in a range [0, 1].

By applying these concepts to Table 1 as an example, we have obtained a reduct $B = \{m, t\}$, the corresponding approximations $\underline{B}X = \{1, 3, 4, 6\}$, $\overline{B}X = \{1, 2, 3, 4, 5, 6\}$ and $BN_B = \{2, 5\}$, and $\gamma_B(H) = 4/6$.

R_1:	R_2:
r_{11} IF (he = no) ∧ (t = hi) THEN h = y | 1/6	r_{21} IF (m = y) ∧ (t = hi) THEN h = y | 1/6
r_{12} IF (he = y) ∧ (t = vhi) THEN h = y | 1/6	r_{22} IF (m = y) ∧ (t = vhi) THEN h = y | 2/6
r_{13} IF (he = no) ∧ (t = vhi) THEN h = y | 1/6	r_{23} IF (m = y) ∧ (t = n) THEN h = n | 1/6
r_{14} IF (he = no) ∧ (t = n) THEN h= n | 1/6	r_{24} IF (m = n) ∧ (t = hi) THEN h = {y, n} | 2/6
r_{15} IF (he =y) ∧ (t = hi) THEN h = {y, n} | 2/6	

Fig. 1 Two sets of rules constructed from two reducts $\{m, t\}$ and $\{h, t\}$

2.2 Rule generation based on rough sets

The rough set-based approach to inductive learning consists of two steps. The first step is to find multiple single covering solutions for all training instances held in a decision table. Specifically, given a set of condition attributes A and a subset $B \subseteq A$, a covering attribute set is found directly by computing its dependency degree $\gamma_B(H)$. The direct solution involves adding an attribute at a time, removing the attribute covered by the attribute set, and then the process is repeated until $\gamma_B(H)$ is equal (or approximately equal) to $\gamma_A(H)$. At the end of the induction of conjunctive attributes, more than one covering set—reduct—will be found.

The second step is to transform multiple sets of reducts to the corresponding sets of rules as seen in Definition 5. In general, rules can be generated from three different regions, i.e. lower and upper approximations, and boundary. In our work we consider rules to be generated from the lower approximation and boundary. For the former case, formally, let h be decision attribute, $V_h = \{h_1, \ldots, h_k\}$ be decision attribute values and $U/h = \{X_1, \ldots, X_k\}$ be a set of partitions with respect to the decision attribute h, and $\underline{B}X$ be the lower approximation, then each $\underline{B}X_i$ consists of a number of equivalent classes, denoted by $\underline{B}X_i = Y_1 \cup \ldots \cup Y_q$, where $Y_j = \{u | f(v, B) = f(u, B), v, u \in \underline{B}X_i\}$. Thus a rule can be represented in the form of IF $(B = f\ (u, B))$ THEN $h_i | stg_B(Y_j)$, where $u \in Y_j$, $h_i \in V_h$, and $stg_B(Y_j)$ is given as follows:

$$stg_B(Y_j) = \frac{|Y_j|}{|U|}(1 \leq j \leq k) \tag{4}$$

For the later case, by using a similar way to the above, we include a set value of a decision attribute as the rule conclusion that represents undeterministic status about a proposition.

Figure 1 presents two sets of rules R_1 and R_2 which are generated from two reducts $\{m, t\}$ and $\{he, t\}$ obtained from Table 1. Here we denote multiple sets of rules by $\mathfrak{R} = \{R_1, R_2, \ldots, R_{|\mathfrak{R}|}\}$, where $R_i = \{r_{i1}, r_{i2,\ldots,}r_{i|Ri|}\}$, $1 \leq i \leq |\mathfrak{R}|$, and each r_{ij} is called a *intrinsic* rule. The relationship between two sets of rules R_i and R_j $(i \neq j)$ is in DNF (disjunctive normal form) as are the rules within R_i (Weiss and Indurkhya 2000).

From Fig. 1, it is noted that the DNF model does not require mutual exclusivity of rules within a set of intrinsic rules and/or between different sets of intrinsic rules. The DNF used in this context differs from the conventional way in which only one of the rules is used to determine the class for a given instance. Instead, all rules will be evaluated for a new instance, and all the trigged rules will be used together to classify it. Rules for either the same classes or different classes can potentially be satisfied simultaneously. In the former case, conflicting conclusions occur. One solution to this is to rank all the classes according to the strengths of confidence derived from the triggered rules. The class with the highest strength is taken as the final conclusion (Apte et al. 1994). Another solution is based on the majority voting principle, in which the conflicting conclusions are resolved by identifying the most frequently satisfied rules (Weiss and Indurkhya 2000). In contrast, our approach makes use of the rule strengths aggregated by using Dempster's rule of combination to reconcile conflicting conclusions.

3 Dempster–Shafer theory of evidence

The Dempster–Shafer theory of evidence allows us to combine pieces of evidence derived from subsets of the frame of discernment that consists of a number of exhaustive and mutually exclusive propositions (Shafer 1976). These propositions form a universal set Θ. For any subset $H = \{h_1, \ldots, h_{|H|}\} \subseteq \Theta$, h_i represents a proposition, called the *focal element*. When H is a one element subset, it is called a *singleton*. All the subsets of Θ constitute a powerset 2^{Θ}, i.e. $H \subseteq \Theta$, if and only if $H \in 2^{\Theta}$. The DS theory uses a numeric value in the range [0, 1] to represent the strength of evidence supporting a subset $H \subseteq \Theta$ based on a given piece of evidence, denoted by $m(H)$, called the *mass function*, and uses a sum of the strengths for all subsets of H to indicate the strength of belief about a proposition H, denoted by $bel(H)$, often called the *belief function*. Notice that $bel(H)$ is equal to $m(H)$ if the subset H is a singleton. The formal definition for Dempster's combination rule is given below:

Definition 6 Let m_1 and m_2 be two mass functions on the frame of discernment Θ, and for any subset $H \subseteq \Theta$, the *orthogonal sum* $\oplus$ of two mass functions on H is defined as:

$$(m_1 \oplus m_2)(H) = \frac{\sum_{X \cap Y = H} m_1(X) * m_2(Y)}{1 - \sum_{X \cap Y = \phi} m_1(X) * m_2(Y)} \tag{5}$$

The orthogonal sum allows two mass functions to be combined into a third mass function, which pools pieces of evidence to support propositions of interest.

4 Modelling rule decisions as evidence

In Bi (2004), we have provided the theoretical justification about the correspondence between a set of intrinsic rules and evidential functions. Sophisticated theoretical results of relationships between rough set theory and evidence theory are also documented in (Yao and Lingras 1998; Skowron and Grzymala-Busse 1994). Instead of studying these theoretical properties, here we focus our interests on developing a method for modelling decisions of multiple sets of rules as pieces of evidence and empirically assessing its validity and applicability. In this section, we start by briefly discussing a general form that a text categorization algorithm based on Rough Sets may have, and then describe our method for modelling the decisions of multiple sets of rules.

4.1 Learning approximations for text categorization

Given a collection of training documents, which is modelled as a decision table, the task of inductive learning for text categorization is to discover rules to assign documents into predefined categories. Formally, let $D = \{d_1, \ldots, d_{|D|}\}$ be a collection of documents, where $d_i = \{w_{i1}, \ldots, w_{in}\}$, and $C = \{c_1, \ldots, c_{|C|}\}$ be a set of predefined categories, then the classification task of assigning documents into predefined categories can be regarded as a mapping function which maps a boolean value to each pair $\langle d, c \rangle \in D \times C$. If a value of 1 is assigned to $\langle d, c \rangle$, then that means a decision has been made that document d belongs to category c, whereas a value of 0 indicates a decision not to classify document d into category c. Therefore, inductive learning in this context aims at constructing an approximation function R for the rules making $R : D \times C \rightarrow \{0, 1\}$, where R is called the classification model, also referred to as a set of intrinsic rules.

Due to inductive learning being an approximating process, moving from the specific to the general, function R may not guarantee that the propositional expression $R : D \times C \rightarrow \{0, 1\}$ is always true, but may instead be partially true. In the other words, there is uncertainty about determining if a document belongs to a particular category or categories. We can therefore define an alternative function $R : D \rightarrow C \times [0, 1]$, where numeric values between 0 and 1 is expressed by stg, indicating how likely it is that a given document d belongs to category c. Without loss of generality, we denote the above process by the notation given in Definition 5, i.e. IF $R(d)$ THEN $h|stg$.

4.2 Mass function definition

Having given a general form of the decision rules for text categorization, we now describe how to model the outputs (conclusions) of intrinsic rules to pieces of evidence in order to formulate an application-specific mass function. Let $C = \{c_1, c_2, \ldots, c_{|C|}\}$ be a frame of discernment, and let $R_i = \{r_{i1}, r_{i2,\ldots,}r_{i|Ri|}\}$ be a set of intrinsic rules. Given a test document d, threshold α and matching function μ that determines the matching degrees between the document and rule conditions (antecedents), if q rules are triggered, i.e. $r_{i,j+1}, r_{i,j+2}, \ldots, r_{i,j+q}$ where $\alpha \leq \mu_{r_{i,j+1}}(d), \ldots, \alpha \leq \mu_{r_{i,j+q}}(d)$, and $1 \leq j, q \leq |R_i|$, then q conclusions are inferred from R_i. Formally, we can represent this inference process by the expression of $r_{i,j+1}(d) \rightarrow h_1|stg_{j+1}, r_{i,j+2}(d) \rightarrow h_2|stg_{j+2}, \ldots, r_{i,j+q}(d) \rightarrow h_q|stg_{j+q}$, where $h_s \in 2^C_, 1 \leq s \leq q$, and stg_{j+s} are rule strengths expressing the extent of which documents belong to the respective categories in terms of degrees of confidence. At the end of the inference process, a set of conclusions will be obtained, and denoted by $H' = \{h_1, \ldots, h_q\}$, where $H' \subseteq 2^C$.

For the rules triggered with respect to q, there are two situations of either $q = |R_i|$ or $q < |R_i|$. When $q = |R_i|$, this means all the rules in R_i are completely satisfied with a given document, resulting in $stg_{j+1} + stg_{j+2} + \cdots + stg_{j+q} = 1$. These conclusions can be directly used to define a mass function. When $q < |R_i|$, $stg_{j+1} + stg_{j+2} + \cdots + stg_{j+q} < 1$, Demspter's rule cannot be directly applied. To make use of Demspter's rule of combination appropriately to pool all the conclusions to draw a final classification decision, it is essential to develop a method for normalizing the outcomes of rules. For convenience later, we define a function ϖ such that $\varpi(h_j) = stg_{i+j}$.

The normalization process starts by finding the duplicate conclusions within H', and then the corresponding rule strengths are added up, resulting in a new set of the conclusions. Formally, for any two $h_j, h_{i+s} \in H'$, if $h_j = h_s$, $j \neq s$, then $\varpi(h_j) \leftarrow \varpi(h_j) + \varpi(h_s)$ and h_s is eliminated. After this processing, a set of conclusions is reconstructed, denoted by $H = \{h_1, h_2, \ldots, h_{|H|}\}$, where $H \subseteq 2^C$. The definition of a mass function for H is as follows:

Definition 7 A mass function is defined as m: $H \rightarrow [0, 1]$. There are four different situations based on the inclusive relations between C and H.

(1) if $C \in H$, then we define a mass function as follows:

$$m(h_i) = \frac{\varpi(h_i)}{\sum_{j=1}^{|H|} \varpi(h_j)} (1 \leq i \leq |H|) \tag{6}$$

(2) if $C \notin H$, then $H \leftarrow H \cup C$ and we define a mass function as follows:

$$m(h_i) = \varpi(h_i)(1 \leq i \leq |H| - 1) \tag{7}$$

$$m(C) = 1 - \sum_{i=1}^{|H|-1} \varpi(h_i) \tag{8}$$

(3) if $H = C$ and $\varpi(h_i) \neq 0$ for any element $h_i \in H(1 \leq i \leq |H|)$, then we define:

$$m(h_i) = \varpi(h_i)(1 \leq i \leq |H|) \tag{9}$$

(4) if $H = C$, and $\varpi(h_i) = 0$ for any element $h_i \in H(1 \leq i \leq |H|)$ then we define: $m(H) = 1$.

We have elsewhere provided a proof that the rule strength satisfies the condition of a mass function (Bi 2004). However, as in the first case above, some conclusions cannot be inferred from a specific piece of evidence, so these conclusions remain unspecified. Thus it is necessary to redistribute mass among known conclusions. Such redistribution for the unknown state of hypotheses could be valuable in the coherent modelling and basic assignment of probabilities to potential hypotheses and in making decisions over an incomplete frame of discernment.

The second case means that the added conclusion C represents our ignorance about the unknown state of hypotheses in inference processes. It absorbs the unassigned portion of the belief after the commitment to H. The addition of ignorance about the likelihood of hypotheses provides us with the information we need for the inference process. This also means that the system does not require complete knowledge about all potential hypotheses since we represent an implicit set of unmodelled hypotheses by including an additional unknown state, C.

For the third case, the conclusions obtained are exactly the same as these integral hypotheses within C, through we directly replace strengths with a mass function.

The fourth case means that the conclusion obtained does not have knowledge about any individual hypotheses within the frame of discernment C, and its complement is an empty element. In this situation, we reassign its degree of total belief as 1.0.

5 The combination methods

In this work, we implement two combination methods: the majority voting method and the Demspter-Shafer method. For majority voting, each set of rules is equally treated as a vote. However when an even number of rule sets participate voting, a conflict decision may occur, resulting in a difficult situation in making the final decision. To cope with such a situation, we take the performance of rulesets into account, giving different weights to each vote based on their performance. More details about the majority voting method used here can be found in (Xu et al. 1992). The rest of the section is focused on how Dempster's rule can be used to combine multiple sets of rules.

testing docs	ruleset$_1$	$\oplus$	ruleset$_2$	$\longrightarrow$	combined ruleset
d_1	m_{11}	$\oplus$	m_{21}	=	combined result$_1$
d_2	m_{12}	$\oplus$	m_{22}	=	combined result$_2$
d_3	m_{13}	$\oplus$	m_{23}	=	combined result$_3$
…	…		…		
d_n	m_{1n}	$\oplus$	m_{2n}	=	combined result$_n$
accuracy	p/n		s/n		?%

Fig. 2 The process of combining classification outputs represented by mass functions which are obtained from two sets of rules

Suppose we are given multiple sets of rules $\mathfrak{R} = \{R_1, R_2, \ldots, R_{|\mathfrak{R}|}\}$ and a set of categories $C = \{c_1, c_2, \ldots, c_{|C|}\}$, for a new document d, the decisions on which document d belongs to categories will be made by $R_1, R_2, \ldots, R_{|\mathfrak{R}|}$, resulting in $R_i(d) = m_i(H_i)$. If only one of the sets of rules is triggered, such as $R_1(d) = H_1$, then H_1 will be ranked in decreasing order on the basis of rule strengths. If the top choice of H_1 is a singleton, it will be assigned to the new document, otherwise lower ranked decisions will be considered for the further selection. When K sets of rules are triggered, after the normalization by using the method in Definition 7, multiple decisions $H_1, H_{2,\ldots,} H_K$ are obtained, where $H_i = \{h_{i1}, h_{i2}, \ldots, h_{|H_i|}\}$, $H_i \subseteq 2^C$ and the corresponding mass function is $m_i(H_i) = \{m_i(h_{i1}), m_i(h_{i2}), \ldots, m_i(h_{i|H_i|})\}$ which is further organized into a triplet structure as given in Definition 8. As a consequence of this process, K pieces of evidence will be obtained, we can combine them to decide the final decisions by using Dempster's rule of combination as follows:

$$m_1 \oplus m_2 \oplus \ldots \oplus m_K \tag{10}$$

By using Eq. 10 we can obtain a set of classification decisions to which documents should belong along with the corresponding belief values, we define a decision rule for determining a final category in general cases below:

$$\phi(x) = h = argmax(\{bel(h)|h \subseteq C\}) \tag{11}$$

Figure 2 illustrates the process of combining the outputs of two sets of intrinsic rules represented by triplets given $K = 2$ and n testing documents. Outputs of different sets of rules on the same testing document can be combined using the orthogonal sum. Each of the combined results will be ranked and the final decision is made by Eq. 11. Notice that we are only interested in the case where h_{ij} is a singleton, i.e. a single category, thus we have $m(h_{ij}) = bel(h_{ij})$ as stated Sect. 3.

6 Experiments and evaluation

In this section we describe the experiments which have been performed to evaluate our methods described in the previous sections. In the experiments, we used a single holdout evaluation since the computation complexity of calculating reducts is exponential in terms of both time and storage and the current version of the Rough Sets-based algorithm does not have the ability to perform sequential calculations of reducts in the multiple folds of a cross-validation method.

For our experiments, we have chosen a public benchmark dataset, often referred to as 20-newsgroup. It consists of 20 categories, and each category has 1,000 documents (Usenet articles), so the dataset contains 20,000 documents in total. Except for a small fraction of the articles (4%), each article belongs to exactly one category (Baker and McCallum 1998).

In this work, we have randomly selected 10 categories of documents to form an experimental data set, containing 10,000 documents in total, to reduce the computational requirements. We also randomly divide the data set into a training set consisting of 63% of the dataset, a validation set consisting of 27% of the data set, and a testing set consisting of 10% of the data set as suggested by Aphinyanaphongs and Aliferis (2003). These three data sets are mutually exclusive and independent of one other. The role of each set is given as follows:

- The training set is used to compute reducts (attribute subsets) and construct sets of rules
- The validation set is used to evaluate the classification performance of the sets of rules and determine how many and which sets of rules should be selected for the performance evaluation of combining the outputs of multiple sets of rules, and choose an optimal threshold α that is used to determine which rule would be triggered.
- The testing set is used to evaluate the performance of the D-S algorithm and the majority voting method using the selected rule sets from the previous step.

The idea of dividing the data set into a validation set and a testing set is to fairly select rule sets for combinations and optimise the model parameters without using the testing set, as indicated in (Baker and McCallum 1998; Aphinyanaphongs and Aliferis 2003), in order to avoid a high likelihood of the overfitting of the classification models over the testing set.

6.1 Evaluation measure

In our analysis, we use evaluation measures of precision (p) and recall (r), and combined measure F_1 defined as $F_1 = 2pr/(p+r)$, which are widely used in information retrieval and text categorization (van Rijsbergen 1979). In particular, we base our performance analyses on the macroaverage F_1 (calculated for each experiment as the average F_1 measure on all categories) (Yang 1999).

6.2 The experimental results

For our experiments, we use information gain to select about 270 keywords (features) after removing stopwords and applying stemming. The selected number of features is only based on the ability of the Rough Sets algorithm in handling the dimensionality of features, rather than an optimum number of features. By using the rough sets-based algorithm, ten reducts have been selected and ten corresponding sets of intrinsic rules have been constructed. We denote these sets of rules by $R_0, R_1, \ldots, R_9$. In our method, there are two parameters that may affect the estimation of classification accuracy: (a) threshold values α determining if a rule will be triggered, and (b) the number of sets of intrinsic rules to be combined.

Let us first see how a threshold value is chosen. Based on the empirical study, we define eight threshold values: 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, 0.95 and 1.00, and then randomly pick three sets of rules from the ten sets of rules to evaluate the performance of these rulesets against these threshold values on the validation set to determine an optimal threshold α. The estimated classification accuracy of these rule sets on each threshold point is averaged, and

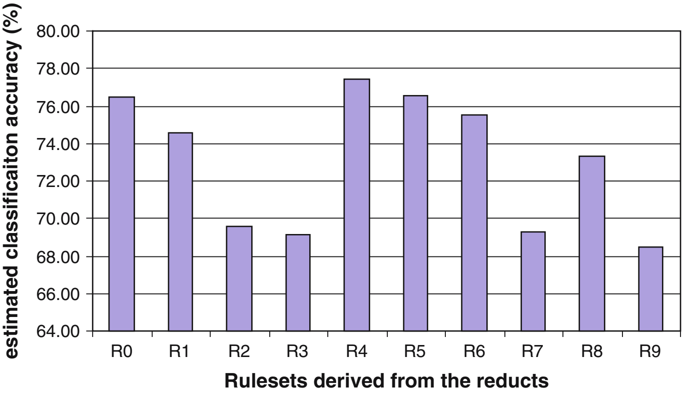

Fig. 3 The performance of different sets of intrinsic rules on the validation set (note *R*i is the same as *R*i throughout the rest of sections)

then the threshold which corresponds to the highest classification accuracy is chosen, i.e. 0.90 above. This threshold is thus used throughout all the experiments below.

It is noted that the threshold of 0.90 is not absolute one; it may vary when different sets of rules are used for choosing a threshold. However, it is not easy and may be not possible to choose a universal threshold which ensures all sets of rules would achieve the best classification accuracy simultaneously. Whatever a threshold value is chosen, the trade-off between the performance of one set of rules and multiple sets of rules must be made.

Prior to the experiments to evaluate the effectiveness of combinations of different number of sets of rules, we first remove the rules which have low strengths and do not contribute the performance increase, and then we evaluate the performance of individual rule sets based on a threshold of 0.90. Figure 3 presents the estimated classification accuracy of 10 sets of intrinsic rules on the validation data set. It can be seen that ruleset R_4 achieves the highest classification accuracy.

To examine the performance of the resulting combinations of different rule sets in classification in terms of the combinations of the sets of rules, we first rank these rulesets in descending order based on their estimates of classification accuracy, and then divide the 10 sets of rules into two groups to see the effectiveness of the combinations of the rulesets since the combinations of lower performance of rules may not bring any benefits to the combined performance of rulesets (Opitz and Maclin 1999). If we use 70% classification accuracy as a cut-off point, then the first group consists of R_0, R_1, R_4, R_5, R_6, R_8, and the second group includes R_2, R_3, R_7, R_9.

For the first group of rulesets, we first take R_4 with the best classification accuracy, and then combine it with R_0, R_1, R_5, R_6, R_8 using Demspter's rule of combination. The combined results are denoted by DS_{40}, DS_{41}, DS_{45}, DS_{46}, DS_{48}. Following the similar process, we take R_5 that is the second best rule set to combine R_0, R_1, R_6, R_8 using DS. These combined results are ranked and the best combined ruleset DS_{45} is chosen for the next round of combination. It is combined with R_0, and then combined with R_6, and so forth, resulting in a ranked list of the result combinations DS_{450}, DS_{4506}, DS_{45061}, and DS_{450618}. As illustrated in Fig. 4, their classification accuracies have dropped with the addition of more rulesets and there is no indication that the combinations of more than two rulesets can outperform the combined ruleset DS_{45}. Therefore, from these combinations of the rulesets, it is observed that

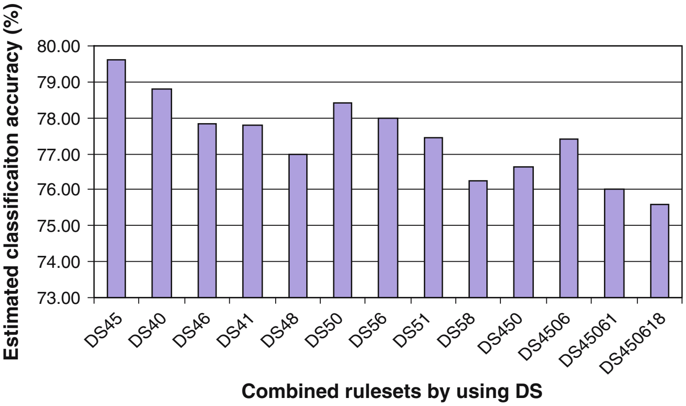

Fig. 4 The performance of the combined rulesets from the first group on the validation set (note *DS*i is the same as *DS*i throughout the rest of sections)

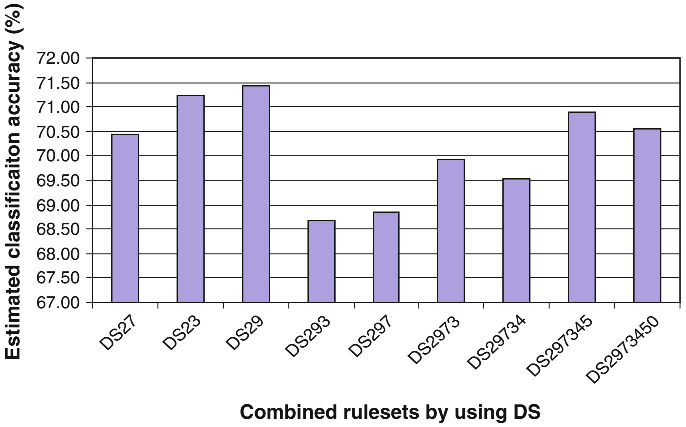

Fig. 5 The performance of the combined rulesets from the second group on the validation set

the combination of the best ruleset with the second best ruleset provides the best combination in achieving the biggest predictive performance.

For the second group of rulesets, we use a similar process to that for the first group to examine the performance of combined rulesets. We first take R_2 to combine with R_7, R_3, R_9 using DS. The performance of the combined rulesets is shown in Fig. 5. Following the same principle as above, we combine DS_{29} with R_3, R_7, and then combine DS_{297} with R_3. As can be seen, the performance of these combinations drops on average relative to DS_{27}, DS_{23} and DS_{29}. However, when DS_{297} is combined with $R_{3,}$ the performance increases again. A similar pattern to the first group of rulesets is observed for this group. To analyze the effectiveness of adding more rulesets, we take the best performing R_4, the second and third best performing R_5 and R_0 from the first group to combine with DS_{2973}. The performances of DS_{29734}, DS_{297345} and $DS_{2973450}$ are not better than that of DS_{29}, which is a similar pattern to the first group of rulesets.

According to the figures as seen in Figs. 4 and 5, in both the cases, it is seen that the best combination of rulesets is composed of just two sets of rules, one with the highest

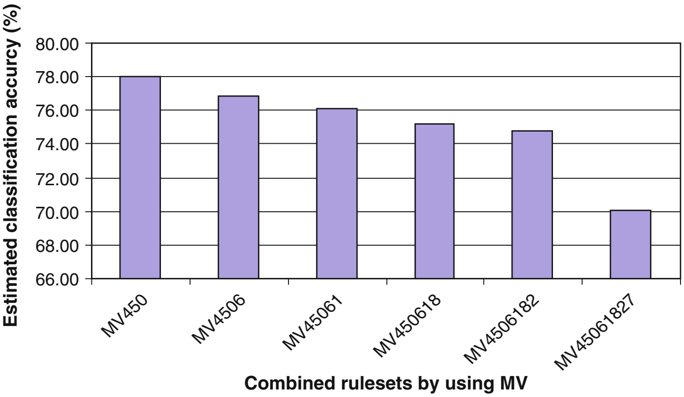

Fig. 6 Performance of the combined rulesets using MV on the validation set

classification accuracy and the other with the second highest accuracy. However, as seen in our previous study (Bi et al. 2004a), the combination of the best and the second best sets of rules may not always be the best combination. The performance of combined rulesets is actually dependent on the closeness of two sets of rules as described in (Bi 2004).

To perform a comparative analysis with the majority voting method, we use the similar procedure to combine outputs obtained from different rule sets. We first rank the ten rulesets based on their performance on the validation set in descending order, i.e. R_4, R_5, $R_0, \ldots, R_9$, and then combine them using the majority voting algorithm, and the combined results are denoted by MV_{45}, MV_{450}, and so on. Specifically, we first take R_4 with the best performance, and then combine it with R_5, and R_0. The combined result MV_{450} is combined with R_6, and this combination process is repeated until all the rulesets are combined. At the end of the process, a list of the combined rulesets MV_{450}, MV_{4506}, MV_{45061}, MV_{450618}, $MV_{4506182}$ and $MV_{45061827}$ are obtained, as illustrated in Fig. 6.

From Fig. 6, it is observed that the performance of the combined rulesets decreases with more rulesets being combined, and the best combination is the combination of the top three rulesets R_4, R_5, and R_0. Figure 7 presents a performance comparison among the best individual ruleset, the best combined ruleset obtained by the DS algorithm and the best combined ruleset obtained by MV on the validation set. It can be seen that in comparison to the best individual ruleset R_4(77.44%), the estimated classification accuracy of DS_{45} can achieve 79.62%, which is 2.18% better than R_4, and MV_{450} can achieve 78.04%, which is 0.6% better than R_4.

6.3 Generalization analysis

In order for our experiments to faithfully reflect the combined performance of multiple sets of rules, we take the best individual ruleset R_4, and the best combined rulesets DS_{45} and MV_{450} based on the validation set, and evaluate their generalization performance on the testing set further. As illustrated in Fig. 8, the estimated accuracy of R_4 is 76.33%, and the estimated accuracy of DS_{45} is 78.10%, so it is 1.77% better than R_4, and the estimate of the MV_{450} accuracy is 76.89%, which is 0.56% better than R_4. Based on these difference of classification accuracies, we perform a paired t-test on the ten pairs of estimated classification accuracies for ten document categories produced by R_4 and DS_{45}, and a paired t-test on the ten pairs of classification accuracies for ten document categories produced by R_4 and MV_{450}, to assess if

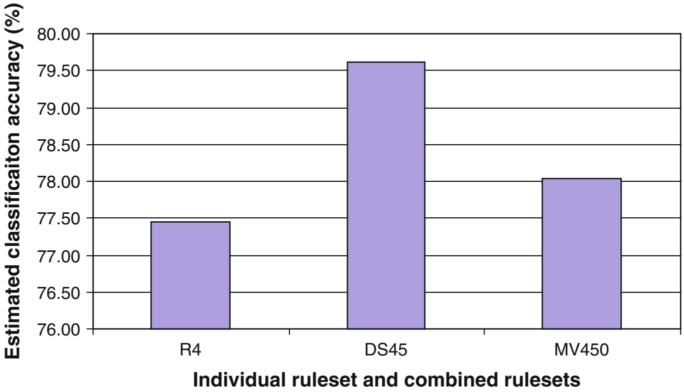

Fig. 7 The performance of the best individual ruleset and the best combined rulesets by using DS and MV on the validation set

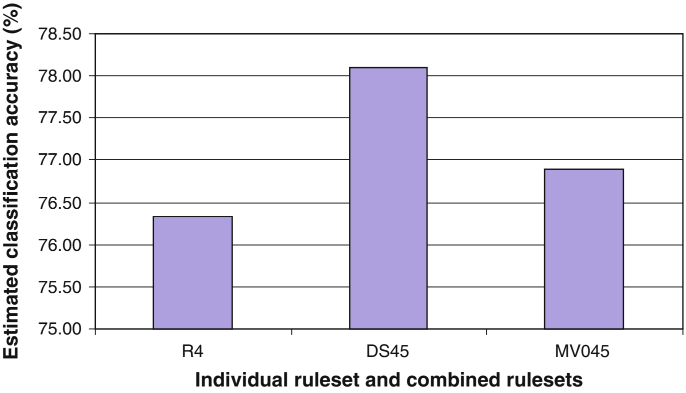

Fig. 8 The performance of the best individual ruleset and the best combined rulesets by using DS and MV on the testing set

the best combined rulesets outperforms the best individual set of rules. The one-tailed t-test result concludes that the estimated accuracy difference between R_4 and DS_{45} is statistically significant at the 0.05 significance level. However the difference between R_4 and MV_{450} is not.

To analyze overfitting of the best single ruleset and the best combined ruleset using DS and MV, we make the performance comparison between the best individual and the best combined ruleset on the validation and testing sets. Putting the two experimental results in Figs. 7 and 8 together in Fig. 9, as we can see from the figure, the difference between the validation set and the testing set accuracies of R_4 is 1.1%, the performance difference between the validation set and the testing set of DS_{45} is 1.52%, and the difference between the validation set and the testing set accuracies of MV_{450} is 1.15%. The over-fitting rate in this experiment is 1.26% on average. It is noted that the Dempster-Shafer method shows more over-fitting than the majority voting method.

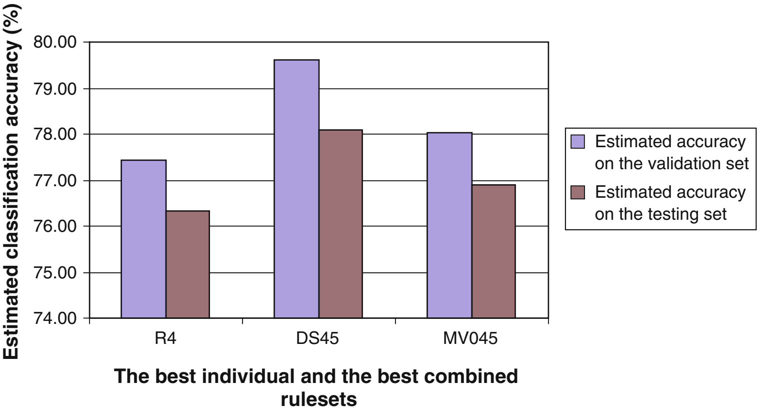

Fig. 9 Overfitting for the DS and MV methods on the validation set accuracy and the testing set accuracy

6.4 Discussion

In Figs. 8 and 9, our experimental results have shown that the performance of the DS algorithm in combining outputs of rulesets is better that of the majority voting method. In this section, we provide a comparison and discussion in an attempt to interpret why DS outperforms MV.

Majority voting is a simple decision making method that has been widely applied to aggregate classification decisions in multiple classification systems (Whiteaker and Kuncheva 2003; Xu et al. 1992). It treats each component classifier as a vote equally. However in this research we have taken their performance into account when a decision conflict occurs when combining an even number of classifiers. Both theoretical and empirical research has shown that more accurate performance can be achieved by combining classifiers since individual classifiers may make errors in different parts of the data space. This concept has been termed diversity (Whiteaker and Kuncheva 2003; Tumer and Ghosh 2002; Kittler et al. 1998; Xu et al. 1992). Whitaker and Kuncheva (2003) carried out an empirical study on the relationship between majority vote accuracy and diversity in Bagging and Boosting. This study is based on an enumerative example, which consists of three classifiers with the same classification accuracy 60%, and 28 combinations of the classifiers in classifying 10 instances. The results show that there is a situation where the combination of the three classifiers by using the majority voting method can produce 90% classification accuracy under some conditions, resulting in a 30% improvement on the individual classifiers. However in most situations, the combined three classifiers using majority voting outperform the individual classifiers with 10–20% increase or even less. In fact majority voting might not guarantee that the combinations of different classifiers can always do better than the best individual classifier. Those experimental results indicate that there is no clear relationship between the ten diversity measures used and the majority voting accuracy in that particular experimental setting.

Figure 10 presents our experimental results obtained by MV on the testing set, depicting the estimated classification accuracy of rulesets R_0, R_4, R_5 and their combination on the ten document categories. The performance of the combined rulesets has improved with respective to the best individual R_4. It has been observed that when there is a big difference between any pair of rulesets and an individual ruleset, the performance of their combinations can be improved, for example, on categories C5 and C9, but they cannot be better than the best individual. However, the performances on categories C3 and C6 may suggest that when

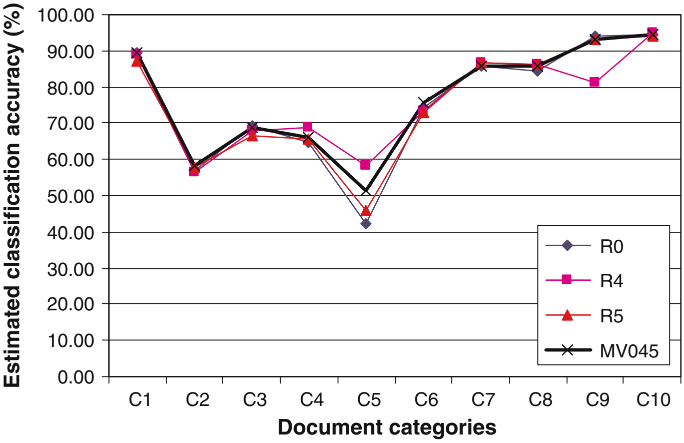

Fig. 10 The performance of individual ruleset R_0, R_4 and R_5vz the combined reduct MV_{045} on the testing set

there is certain amount of difference between any pair of rulesets and an individual ruleset, the predictive accuracy of the combined ruleset is better than that of individual rulesets. This certain amount of difference could be explained by the concept of diversity.

Dempster's rule of combination is built on the more sophisticated evidence theory. The key aspect of this method is how to transform different types of measures or scores of different classifiers into evidence. There are underlying differences between Dempster's rule of combination and the majority voting method in modelling practical applications. Looking at the calculation process of the core operation—orthogonal sum—in the DS theory of evidence, it is not only able to accumulate different pieces of evidence obtained from different sources (crossing categories), but it also takes account maximal agreements among the pieces of evidence. Importantly, the DS theory of evidence also provides an explicit representation for unknown or nondeterministic states of hypotheses in terms of ignorance which has been used to represent uncertainty in assigning documents into pre-defined categories. Thus these theoretical properties represent a significant improvement on the majority voting method. However, sophisticated theoretical complexity does not necessarily translate improved implementation of practical applications.

Figure 11 presents a graph describing the estimated classification accuracy of rulesets R_4 and R_5, and their combination on the ten document categories for the testing set. It can be observed that with the exception of categories C5 and C9, their classification accuracy is better that of the individuals on all the other categories. The performance variation on categories C5 and C9 may suggest that the predictive accuracy of the combined ruleset is not better than that of the two individual rulesets when there is a big gap between their predictive accuracies. In comparison with Fig. 10, there is similar performance behaviour on category C5, however there is not much similarity on category C9. From these results, it cannot be seen that there is a correlation between DS and MV in improving the performance of sets of rules in the combinations of multiple sets of rules.

In general, it is a very difficult task to understand why the combined classifier outperforms an individual classifier under some conditions (Lam 2002). In an effort to provide an insight into this issue, Kuncheva (2001) proposed an empirical method for measuring the diversity between pairs of classifiers based on a contingency table of correct/incorrect classification.

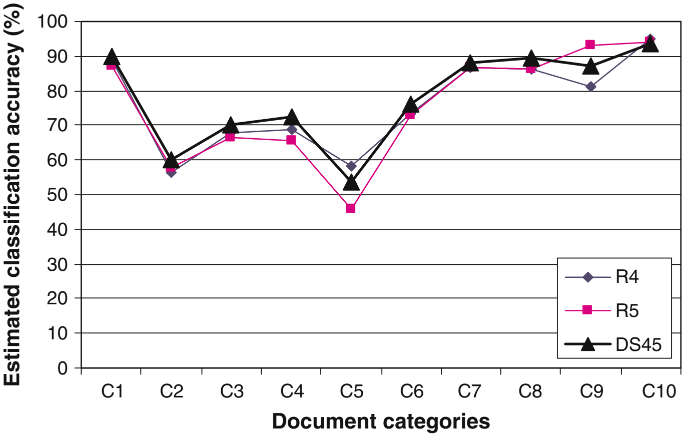

Fig. 11 The performance of individual rulesets R_4 and R_5 vz the combined ruleset DS_{45} on the testing set

As indicated by Whitaker and Kuncheva (2003), diversity is not a simple concept and in some situation it might not correlate the performance improvement of combined classifiers, and to date no "best" measure for diversity has been found. Due to diversity being measured on the training process, the contingency tables for any pairs of classifiers must be computed at the validation stage, these methods cannot be directly applied to examine the diversity of multiple sets of rules. Alternatively we propose an empirical measure called the *closeness*. Based on our empirical study, we found that when a ruleset has the highest closeness to the best ruleset, its combination with the best ruleset can achieve the best performance (Bi 2004).

7 Conclusion

In this paper, we have presented a method built on Rough Set theory for generating multiple sets of rules, a novel combination approach for combining classification decisions obtained from multiple sets of rules based on Dempster's rule of combination, and the comparative analysis between DS and MV. Various experiments have been carried out on 10 of 20-newsgroups benchmark data. It is observed that in the case of DS, the combination which can achieve the highest predictive performance is the combination of two sets of rules of which one is the best and the other is the second best, whereas in the majority voting case, the best combination is the combination of three rulesets, i.e. the combination of the best, the second best and the third best rulesets. To analyze the generalization performance of our method, we have compared the estimated performance of the best individual ruleset and the best combined rulesets using both DS and MV on the validation set with that on the testing set, and found that DS has more overfitting than MV.

We should emphasize that the focus of this research is on exploring methods and techniques which can make use of multiple discovered knowledge and evidence sources for improving text classification, instead of developing or improving single rough sets-based algorithms. Our empirical results show that the best combination of rulesets outperforms single sets of rules, which supports a common idea in artificial intelligence that decision making on the basis of multiple knowledge is more effective than single knowledge. On the other hand, we found that the performance of the combined ruleset is directly affected by the performance of individual sets of rules, the combination of the best and the second best rulesets outperforms any one of the single rulesets. This also means that given a data mining algorithm if our rough sets algorithm is comparable with it, the best combined ruleset using our DS method would outperform it.

References

Aphinyanaphongs Y, Aliferis CF (2003) Text categorization models for retrieval of high quality articles in internal medicine. In: Proceedings of the American Medical Informatics Association (AMIA) annual symposium, Washington, DC, USA, pp 31–35

Apte C, Damerau F, Weiss S (1994) Automated Learning of Decision Text Categorization. ACM Trans Inf Syst 12(3):233–251

Baker D, McCallum A (1998) Distributional clustering of words for text classification. In: Proceedings of 21st ACM international conference on research and development in information retrieval, pp 96–103

Bi Y (2004) Combining multiple classifiers for text categorization using Dempster's rule of combination. PhD dissertation, University of Ulster

Bi Y, Anderson T, McClean S (2004a) Combining rules for text categorization using Dempster's rule of combination. In: Proceedings of 5th international conference on intelligent data engineering and automated learning. LNCS 3177, Spring-Verlag, pp 457–463

Bi Y, Bell D, Guan JW (2004b) Combining evidence from classifiers in text categoriza-tion. In: Proceedings of the 8th international conference on knowledge-based intelligent information & engineering systems. LNCS 3215, Spring, pp 521–528

Chouchoulas A, Shen Q (2001) Rough set-aided keyword reduction for text categorization. Appl Artif Intell 15(9):843–873

Cohen WW, Singer Y (1999) Simple, fast, and effective rule learner. In: Proceedings of annual conference of American association for artificial intelligence, pp 335–342

Denoeux T (2000) A neural network classifier based on Dempster–Shafer theory. IEEE Trans Syst Man Cybern A 30(2):131–150

Freund Y, Schapire R (1996) Experiments with a new boosting algorithm. In: Machine learning: proceedings of the thirteenth international conference, pp 148–156

Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139

Friedman J, Hastie T, Tibshirani R (1998) Additive logistic regression: A statistical view of boosting (Technical Report). Stanford University Statistics Department. http://www.stat-stanford.edu/~tibs

Grzymala-Busse J (1992) LERS — A System for learning from examples based on Rough Sets. In: Slowinski R (ed), Intelligent decision support. Kluwer Academic, pp 3–17

Guang JW, Bell D (1998) Rough computational methods for information systems. Artif Intell 105:77–103

Kittler J, Hatef M, Duin RPW, Matas J (1998) On combining classifiers. IEEE Trans Pattern Anal Mach Intell 20(3):226–239

Kuncheva L (2001) Combining classifiers: soft computing solutions. In: Pal SK, Pal A (eds) Pattern recognition: from classical to modern approaches. World Scientific, pp 427–451

Lam L (2000) Classifier combinations: implementation and theoretical issues. In: Kittler J, Roli F (eds) Multiple classifier systems. LNCS 1857, Spring, pp 78–86

Mitchell T (1999) Machine learning and data mining. Commun ACM 42(11):31–36

Nardiello P, Sebastiani F, Sperduti A (2003) Discretizing continuous attributes in AdaBoost for text categorization. In: Proceedings of 25th European conference on information retrieval. LNCS 2633, Springer-Verlag, Berlin, pp 320–334

Opitz D, Maclin R (1999) Popular ensemble methods: an empirical study. J Artif Intell Res 11:169–198

Pawlak Z (1991) Rough Set: theoretical aspects of reasoning about data. Kluwer Academic

Quinlan JR (1996) Bagging, boosting, and C4.5. In: Proceedings of the thirteenth national conference on artificial intelligence, pp 725–730

Schapire RE, Singer Y (2000) BoosTexter: aboosting-based system for text categorization. Mach Learn 39(2/3):135–168

Shafer G (1976) A mathematical theory of evidence. Princeton University Press, Princeton

Skowron A, Grzymala-Busse J (1994) From rough set theory to evidence theory. In: Yager R, Fedrizzi M, Kacprzyk J (eds) Advances of the Dempster–Shafer Theory of Evidence. Wiley, New York pp 193–236

Tumer K, Ghosh JR (2002) Combining of disparate classifiers through order statistics. Pattern Anal Appl 6(1):41–46

Xu L, Krzyzak A, Suen CY (1992) Several methods for combining multiple classifiers and their applications in handwritten character recognition. IEEE Trans Syst Man Cybern 22(3):418–435

Yao YY, Lingras PJ (1998) Interpretations of belief functions in the theory of rough sets. Inf Sci 104(1–2):81–106

Yang Y (1999) An evaluation of statistical approaches to text categorization. J Inf Retr 1(1/2):67–88

van Rijsbergen CJ (1979) Information retrieval, 2nd edn. Butterworths

Weiss S, Kulikowski C (1991) Computer system that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems. Morgan Kaufmann

Weiss SM, Indurkhya N (2000) Lightweight rule induction. In: Proceedings of the seventeenth international conference on machine learning, pp 1135–1142

Whiteaker CJ, Kuncheva L (2003) Examining the relationship between majority vote accuracy and diversity in bagging and boosting. Technical report. University of Wales, Bangor