International Journal of Approximate Reasoning 52 (2011) 1103–1117

Contents lists available at SciVerse ScienceDirect

International Journal of Approximate Reasoning

journal homepage: www.elsevier.com/locate/ijar

Diverse reduct subspaces based co-training for partially labeled data

Duoqian Miao, Can Gao *, Nan Zhang, Zhifei Zhang

Department of Computer Science and Technology, Tongji University, Shanghai 201804, PR China
The Key Laboratory of "Embedded System and Service Computing", Ministry of Education, Shanghai 201804, PR China

ARTICLE INFO

Article history:
Received 16 June 2010
Revised 19 May 2011
Accepted 24 May 2011
Available online 7 June 2011

Keywords:
Rough set theory
Markov blanket
Attribute reduction
Rough co-training
Partially labeled data

ABSTRACT

Rough set theory is an effective supervised learning model for labeled data. However, it is often the case that practical problems involve both labeled and unlabeled data, which is outside the realm of traditional rough set theory. In this paper, the problem of attribute reduction for partially labeled data is first studied. With a new definition of discernibility matrix, a Markov blanket based heuristic algorithm is put forward to compute the optimal reduct of partially labeled data. A novel rough co-training model is then proposed, which could capitalize on the unlabeled data to improve the performance of rough classifier learned only from few labeled data. The model employs two diverse reducts of partially labeled data to train its base classifiers on the labeled data, and then makes the base classifiers learn from each other on the unlabeled data iteratively. The classifiers constructed in different reduct subspaces could benefit from their diversity on the unlabeled data and significantly improve the performance of the rough co-training model. Finally, the rough co-training model is theoretically analyzed, and the upper bound on its performance improvement is given. The experimental results show that the proposed model outperforms other representative models in terms of accuracy and even compares favorably with rough classifier trained on all training data labeled.

1. Introduction

Since the initial work of Pawlak [1,2], rough set theory, as an effective approach to dealing with imprecise, uncertain and incomplete information, has been used in many research fields successfully, such as pattern recognition, artificial intelligence, machine learning, knowledge acquisition and data mining [3–7]. In Pawlak's rough set model, the lower and upper approximations are defined on equivalence relation. However, this binary relation is too restrictive for many practical applications. To address this issue, some extended models have been put forward by replacing equivalence relation with other binary relations, such as neighborhood rough sets [8–14], tolerance rough sets [15–18], fuzzy rough sets [19–22], dominance-based rough sets [23–25], covering rough sets [26–28], etc. By incorporating probabilistic approaches into rough set theory, several probabilistic generalization models like decision-theoretic rough sets [29–31], variable precision rough sets [32,33], Bayesian rough sets [34,35], and others [36,37] have also been proposed. These models enrich the theory of rough sets as well as its practical application.

In general, Pawlak's rough set model and its extensions rely on a large number of labeled data to train a classifier (rough classifier). However, in many practical learning domains (e.g. web-page classification, anti-spam and image retrieval), we often face the problem where the labeled data are fairly expensive to obtain since labeling example requires much human effort, whereas the unlabeled data are often cheap and readily available. In such situation, traditional rough set approaches may be not applicable because of the scarcity of the labeled data. Therefore, it would be desirable to capitalize on the abundant unlabeled data to improve learning performance.

[*] Corresponding author at: Department of Computer Science and Technology, Tongji University, Shanghai 201804, PR China. Tel.: +86 15000600177.
E-mail addresses: miaoduoqian@163.com (D. Miao), 2005gaocan@163.com (C. Gao), zhangnan0851@163.com (N. Zhang), tjzhifei@163.com (Z. Zhang).

0888-613X/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.ijar.2011.05.006

Duan et al. [38] studied the problem of building web-page classifier using positive and unlabeled examples. The notion of tolerance class in tolerance rough sets [18] was introduced to approximate the concepts that existed in web-pages and enrich the representation of web-pages. The experimental results indicated that their method markedly dominated some existing ones. Based on an extension of decision theoretic model proposed by Yao [39–41] , Lingras et al. [42] presented a semi-supervised decision theoretic rough set model and successfully applied it to model the promotional campaign in a real-world retail store. The learning of partially labeled data with rough set theory was also investigated in [43,44] and some promising results were shown in their experiments. The aforementioned works apply the concept of supervised rough sets to the learning of partially labeled data successfully. However, little attention has been paid to semi-supervised rough set model to deal with both labeled and unlabeled data directly.

In this paper, we are principally concerned with the theoretical and experimental study of Pawlak's rough set model for partially labeled data. Our first contribution is to propose a novel attribute reduction algorithm for partially labeled data. Traditional discernibility matrix in rough set theory could deal with either labeled data or unlabeled data. Motivated in part by the works of Skowron and Rauszer [45] and Slezak [46], we propose a new discernibility matrix for partially labeled data. And the theory of Markov blanket is introduced to conduct the discernibility matrix based algorithm for attribute reduction. The relationship between traditional discernibility matrix and our proposed one is discussed, and the validity of the proposed algorithm is also analyzed.

The second contribution of this paper is to introduce a method for constructing a semi-supervised rough set model for partially labeled data. In contrast to traditional rough set approaches, which usually train only one rough classifier on the labeled data in the learning process, our proposed rough co-training model uses two classifiers learned from different attribute subspaces and could benefit from the unlabeled data. In fact, the main principle behind rough co-training is the theory of ensemble learning, which generally obtains better performance than single classifier. The base classifiers of rough co-training are trained in two diverse reduct subspaces of partially labeled data, therefore rough co-training could make the best possible use of the diversity of the base classifiers on the unlabeled data to enhance its performance.

The third contribution of this paper is to give the theoretical analysis and comparison experiment. We theoretically explain why the rough co-training model could work well for partially labeled data and analyze the upper bound on the performance improvement. The comparison experiments with other representative models, such as self-training and standard co-training, are performed in different situations, and the reasons for better performance of rough co-training are clearly interpreted. Moreover, the latent property that the performance of rough co-training with only few labeled data and adequate numbers of unlabeled data even outperforms that of rough classifier with all training data labeled is discovered in the experiments. This salient feature could be used to conduct the design of effective learning algorithm and reduce the cost of labeling example in practical learning domains.

The rest of this paper is organized as follows: Section 2 reviews the fundamental principles of rough set theory. In Section 3, a Markov blanket based attribute reduction algorithm is proposed for partially labeled data. Section 4 describes the rough co-training model and analyzes its effectiveness. The experimental results are reported in Section 5. Finally, Section 6 concludes the paper and indicates the intended directions of future research.

2. Preliminary knowledge on rough set theory

This section will review some basic concepts of rough set theory. Detailed description of the theory can be found in [3–7,47–52].

In rough set theory, an information system is described by a bivariate table, whose columns are labeled by attributes, rows are labeled by examples of interest and entries of the table are attribute values. Formally, an information system is defined as $S = (U, A, V, f)$, where U is a nonempty and finite set of examples, called the universe; A is a nonempty and finite set of attributes; V is the union of attribute domains, i.e., $V = \bigcup V_a$, where V_a denotes the domain for each attribute $a \in A$;and f is an information function which associates a unique value of each attribute with every example belonging to U. If the attribute set A can be divided into condition attribute set C and decision attribute set D, this information system is also called as decision information system or decision table.

For arbitrary attribute subset B of A, it determines a binary relation $IND(B)$, which is called as indiscernibility relation and defined as follows:

$$IND(B) = \{\langle x, y\rangle \in U \times U | \forall a \in B, f(x, a) = f(y, a)\} \tag{1}$$

Obviously, an indiscernibility relation is an equivalence relation which satisfies reflexivity, symmetry and transitivity. The family of all equivalence classes of $IND(B)$, i.e., a partition of the universe determined by B, will be denoted by $U/IND(B)$ or simply by U/B ;an equivalence class of $IND(B)$, i.e., the block of the partition U/B is denoted by

$$[x]_B = \{y \in U | \langle x, y\rangle \in IND(B)\} \tag{2}$$

For arbitrary ordered pair $\langle x, y\rangle$ in $IND(B)$, it means that examples x and y are indiscernible with respect to B. Equivalence classes induced by $IND(B)$ are referred to as B-elementary sets or B-elementary granules.

Rough set theory hinges on two basic concepts, namely the lower and upper approximations of a set. Let X be a subset of the universe U, its lower and upper approximations with respect to $B(B \subseteq A)$ are denoted as $\underline{B}(X)$ and $\overline{B}(X)$ respectively.

$$\underline{B}(X) = \{x \in U | [x]_B \subseteq X\} \tag{3}$$

$$\overline{B}(X) = \{x \in U | [x]_B \cap X \neq \emptyset\} \tag{4}$$

The B-lower approximation of concept X is the union of all B-elementary sets that are included in X, whereas the B-upper approximation of concept X is the union of all B-elementary sets that have a nonempty intersection with X. If $\underline{B}(X) = \overline{B}(X)$, X is a crisp(definable) set with respect to B. Otherwise, X is a rough(indefinable) set. $BND_B(X) = \overline{B}(X) - \underline{B}(X)$ is called as the boundary of X over U.

Assume C and D are the sets of condition and decision attributes in a given decision table respectively, partitions U/C and U/D will be induced by attribute sets C and D over U. The positive and boundary regions of D with respect to C are defined as

$$POS_C(D) = \bigcup_{X \in U/D} \underline{C}(X) \tag{5}$$

$$BND_C(D) = \bigcup_{X \in U/D} \overline{C}(X) - \bigcup_{X \in U/D} \underline{C}(X) \tag{6}$$

The boundary region is the set of C-elementary sets which can not be perfectly described by C, and the positive region is the set of C-elementary sets which completely belong to one block of the partition U/D.

The discernible information among the examples can be described by a matrix, which is called as discernibility matrix. Let $S = (U, A, V, f)$ be a decision table, the element m_{ij} of discernibility matrix M is denoted as

$$m_{ij} = \begin{cases} \{a \in C | a(x_i) \neq a(x_j)\}, & d(x_i) \neq d(x_j) \\ \emptyset, & otherwise \end{cases} \tag{7}$$

For inconsistent decision table, there are some different definitions for discernibility matrix. Without specific statement, the decision tables in this paper are consistent.

Attribute reduction is a key problem in rough set theory. Given a decision table $S = (U, A, V, f)$ and discernibility matrix M, for any subset $P \subseteq A$, if P satisfies the conditions:

(I) for any element r of M, P has a nonempty intersection with r;
(II) no attribute can be eliminated from P without affecting the requirement (I).

then P is a reduct of the given decision table.

The reduct is a subset of all condition attributes, which retains the discriminating power of the original data, and has no redundant attribute. Usually, there exist a number of reducts for a given decision table, while the intersection of all reducts is called the core.

3. Markov blanket based attribute reduction algorithm for partially labeled data

In theory, a classifier with more attributes should have more discriminating power, but in practice, with a limited number of training data, excessive attributes will not only significantly slow down the learning process, but also cause the classifier to overfit the training data as irrelevant or redundant attributes may confuse the learning algorithm. Attribute reduction(feature selection) is just a research field which has been proven effective in enhancing learning efficiency, increasing predictive accuracy, and reducing the complexity of the learning process. In rough set theory, attribute reduction is a key research problem and many useful algorithms have been proposed at present [53–56]. A reduct is a minimum subset of attributes that provides the same descriptive or classification ability as the entire set of attributes. In other words, attributes in a reduct are jointly sufficient and individually necessary for classification.

Generally, a partially labeled data consists of few valuable labeled examples and large numbers of exploitable unlabeled examples. Intuitionally, a learned classifier should take full advantage of the precious labeled examples and also use the unlabeled ones to improve its performance. To this end, in the process of attribute reduction, we desire to take all training examples into consideration to conduct the design of an effective algorithm. More specifically, all training examples are first partitioned into disjoint equivalence classes by all condition attributes. For any equivalence class, if there exists a labeled example, we will assign its class symbol to other examples in the equivalence class. Otherwise, a special pseudo-class symbol which differs from that of all labeled examples is attached to every unlabeled example in the equivalence class. Then each unlabeled example will have a class symbol, while the partially labeled data is transformed into a decision table.

For instance, in the following table, there is a partially labeled data, where two examples have class symbols and the others are unlabeled.

Table 1
A partially labeled data.

	$a1$	$a2$	$a3$	$a4$	$a5$	$a6$	$a7$	d
o_1	0	0	0	0	0	0	0	0
o_2	0	0	1	0	1	0	0	?
o_3	0	0	1	0	1	1	0	?
o_4	1	1	0	0	0	0	0	1
o_5	2	0	2	0	0	0	0	?
o_6	2	2	2	1	0	0	0	?
o_7	2	2	2	2	2	2	1	?
o_8	0	0	0	0	0	0	0	?
o_9	1	1	0	0	0	0	0	?
o_{10}	2	0	2	0	0	0	0	?

With all condition attributes, the universe can be partitioned into seven disjoint equivalence classes, namely $\{o_1, o_8\}, \{o_4, o_9\}, \{o_5, o_{10}\}, \{o_2\}, \{o_3\}, \{o_6\}$ and $\{o_7\}$. In the table, examples o_1 and o_8 are indiscernible with respect to all condition attributes. But example o_1 has class symbol "0", thus example o_8 will be assigned class symbol "0". Example o_9 can be attached class symbol "1" in the same way as example o_8. For equivalence class $\{o_5, o_{10}\}$, it consists of two unlabeled examples, in which no classification information is available, hence the examples in this equivalence class can only be labeled a nominal class symbol "*" which differs from other class symbols in the table. The other examples in the table could be deduced by analogy. Then Table 1 becomes a decision table. This process not only avoids the inconsistency of examples in the way of simply labeling every unlabeled example with a nominal class symbol "*", but also reduces the complexity of the learning process.

Formally, we denote a partially labeled data as $MS = (U = L \cup N, A = C \cup D, V, f')$, where L is a nonempty and finite set of labeled examples and N is a nonempty and finite set of unlabeled examples. The decision table derived from MS is denoted as $TS = (U', A = C \cup D, V, f'')$. And the decision table in which the unlabeled examples in MS are attached true class symbols is denoted as $S = (U, A = C \cup D, V, f)$(underlying decision table). Then we can present a new definition of discernibility matrix and analyze the relationships among MS, TS and S.

Definition 1. Let $MS = (U = L \cup N, A = C \cup D, V, f')$ be a partially labeled data, and $TS = (U', A = C \cup D, V, f'')$ be the decision table derived from MS. Then, the element m_{ij} of discernibility matrix M of TS is denoted as

$$m_{ij} = \begin{cases} \{a \in C | a(x_i) \neq a(x_j)\}, & d(x_i) \neq d(x_j) \vee d(x_i) = * \vee d(x_j) = * \\ \emptyset, & otherwise \end{cases} \tag{8}$$

Proposition 1. *Let $MS = (U = L \cup N, A = C \cup D, V, f')$ be a partially labeled data. If M_1 is the collection of elements in the discernibility matrix of the derived decision table TS, and M_2 is the collection of elements in the discernibility matrix of the underlying decision table S. Then M_2 is a subset of M_1.*

Proof. From Definition 1, we can see that, for the labeled examples in MS, there is no difference between the discernibility matrixes of TS and S in the element. For any unlabeled example in N, it will be labeled a pseudo-class symbol during the transformation. Therefore, some discernible information related to unlabeled examples may be produced in the discernibility matrix of TS because there is a difference between the unlabeled examples and labeled ones in the class symbol. However, in the underlying decision table S, those unlabeled examples may have the same class symbol as the labeled one. As a result, some discernible information related to unlabeled examples will not appear in the discernibility matrix of S. This case could also happen on two unlabeled examples. In short, some elements in the discernibility matrix of TS may not appear in the discernibility matrix of S, while each element in the discernibility matrix of S definitely exists in the discernibility matrix of TS. Hence, the collection of elements in the discernibility matrix of S is a subset of that of elements in the discernibility matrix of TS. The proposition is proved. □

Proposition 2. *Let $MS = (U = L \cup N, A = C \cup D, V, f')$ be a partially labeled data. If $Core_1$ is the core attribute set of the derived decision table TS, and $Core_2$ is the core attribute set of the underlying decision table S. Then the formula $Core_2 \subseteq Core_1$ holds.*

Proof. Reductio ad absurdum. Assume that there is a core attribute $a \in Core_2$, but $a \notin Core_1$. Then it follows that there exist two examples X_i and X_j which have different class symbols and only attribute a can discern them. In other words, examples X_i and X_j are indiscernible without attribute a. If X_i and X_j are both labeled examples in MS, the discernibility matrix of TS definitely contains a singleton set "$\{a\}$" and the formula $a \in Core_1$ holds. If only one of examples X_i and X_j is unlabeled in MS, the unlabeled one will be assigned a distinct class symbol "*" during the transformation. In order to discern examples X_i and X_j, the discernible information "$\{a\}$" will appear in the discernibility matrix of TS. In the case of two unlabeled examples, same result could be deduced by analogy. Thus, a is a core attribute of TS, which contradicts the assumption. The proposition is proved. □

Proposition 3. *Let $MS = (U = L \cup N, A = C \cup D, V, f')$ be a partially labeled data. If RED_1 is a reduct of the derived decision table TS, there must exist a reduct RED_2 in the underlying decision table S and the formula $RED_2 \subseteq RED_1$ holds.*

Proof. From Proposition 1 above, we see that the collection M_2 in the underlying decision table *S* is a subset of the collection M_1 in the derived decision table *TS*. Assume that e_1 is an element of the difference set of the collections M_1 and M_2. Then there exist three different relationships between e_1 and the element of M_2.

(1) $\exists e_2 \in M_2$, it has $e_2 \subseteq e_1$. According to the definition of reduct, RED_2 will have a nonempty intersection with each element in M_2. Therefore, the intersection of RED_2 and e_1 is definitely nonempty on the condition $e_2 \subseteq e_1$. In other words, RED_2 is sufficient to discern the examples which produce discernible information e_1. If this case holds for all elements in the difference set of M_1 and M_2, the reduct of *TS* is the same as that of *S*.

(2) $\exists e_2 \in M_2$, it has $e_2 \supset e_1$. Under the circumstances, some reducts of *S* may be not enough to discern the examples which produce discernible information e_1. But the reduct that contains an element of the intersection set of e_2 and e_1 is sufficient to discern those examples. Therefore, for any reduct in M_2 and e_1, there must exist a reduct in M_2 and these two reducts have same attributes.

(3) $\forall e_2 \in M_2$, it has $e_2 \not\subset e_1$ and $e_2 \not\supset e_1$. In this case, the reduct RED_2 of *S* may be not sufficient to discern all examples in *TS*, and some attributes will be added to the reduct of *TS*. Therefore, the reduct of *S* will be included by that of *TS*.

In all cases, the reduct of *TS* includes at least one reduct of *S*. The proposition is proved. □

The propositions mentioned above describe the implicit relationships of the derived decision table and underlying decision table. They guarantee that the reduct of the derived decision table can hold the discriminating power of the underlying decision table. Therefore, traditional discernibility matrix based attribute reduction algorithms could be used to deal with partially labeled data because of no loss of classification information.

It is well known that finding a minimal reduct of a given decision table is a NP-hard problem [45]. Although some heuristic approaches have been proposed, their algorithms are incomplete. In other words, the reduct of those approaches not only differs from the minimal reduct, but also contain redundant attribute. Table 1 is a good instance. The reduct of the discernibility matrix based forward-heuristic algorithm is $\{a1, a2, a3, a6\}$, but attribute set $\{a2, a3, a6\}$ is minimal. Inspired by the work of Slezak [46], we propose a Markov blanket based attribute reduction algorithm for partially labeled data, which could eliminate redundant attribute effectively and also generate high quality reduct. In order to explain our algorithm, we first present some related concepts.

Definition 2 [46,57]**.** Let *B* be a subset of all attributes *C* and $C_i \notin B$. *B* is a Markov blanket for C_i if C_i is conditionally independent of $C - B - C_i$ given *B*, namely $P(C - B - \{C_i\}|C_i, B) = P(C - B - \{C_i\}|B)$.

In fact, it is easy to see that if *B* is a Markov blanket of C_i, then it is also the case that the decision attribute *D* is conditionally independent of the attribute C_i given *B*, namely $P(D|C_i, B) = P(D|B)$. The Markov blanket condition requires that *B* subsumes not only the information for C_i with respect to *D*, but also about all of the other attributes.

Theorem 1 [57]**.** *Let G be current set of attributes, and assume that attribute $C_i \notin G$ (previously removed) has a Markov blanket within G. Let $C_j \in G$ be the attribute that is about to be removed based on some Markov blankets within G. Then C_i also has a Markov blanket within $G - \{C_j\}$.*

Theorem 1 guarantees that an attribute removed in an earlier phase will still find a Markov blanket in any later phase. That is to say, removing an attribute in a later phase will not affect the previously removed attributes. According to the previous definition of reduct, we can prove that there is no Markov blanket for any core attribute.

Definition 3. Let *M* be the discernibility matrix of a given decision table $S = (U, A = C \cup D, V, f)$. For any subset *B* of *C*, the directly relevant set of *B* within *M* is defined as

$$RS_M(B) = \{K | K \in M \wedge K \cap B \neq \emptyset\} \tag{9}$$

Definition 4. Let *M* be the discernibility matrix of a given decision table $S = (U, A = C \cup D, V, f)$. For any subset *B* of *C*, the directly irrelevant set of *B* within *M* is defined as

$$IS_M(B) = \{K | K \in M \wedge K \cap B = \emptyset\} \tag{10}$$

Definition 5. Let *M* be the discernibility matrix of a given decision table $S = (U, A = C \cup D, V, f)$. For any subset *B* of *C*, the relative complement set of *B* to its directly relevant set within *M* is defined as

$$RC_M(B) = \{K - B | K \in RS_M(B)\} \tag{11}$$

Definition 6. Let M be the discernibility matrix of a given decision table $S = (U, A = C \cup D, V, f)$. For any subset B of C, the indirectly relevant set of B within M is defined as

$$IR_M(B) = \{K | K \in IS_M(B) \wedge Q \in RC_M(B) \wedge K \cap Q \neq \emptyset\} \tag{12}$$

Definition 7. Let M be the discernibility matrix of a given decision table $S = (U, A = C \cup D, V, f)$. For any attribute $a \in C$, its Markov blanket is defined as

$$MB_M(a) = \{x \in K | K \in RC_M(\{a\}) \vee K \in IR_M(\{a\})\} \tag{13}$$

Actually, the Markov blanket for an attribute includes two parts of attributes, namely its relative complement set and indirectly relevant set. Although the sequence of attributes does not affect the Markov blanket for an attribute, it is closely related to the quality of the reduct. Intuitionally, if one attribute has high frequency in discernibility matrix, this attribute may be more important for classification. Hence, the attributes can be sorted by frequency, and then the attribute which has not only a Markov blanket but also lowest frequency should be first taken into consideration to be removed from the whole attribute set. This process can be depicted by Algorithm 1.

Algorithm 1 Markov blanket based attribute reduction algorithm (MBARA)

```
Input:
   A partially labeled data $MS = (U = L \cup N, A = C \cup D, V, f')$;
Output:
   An optimal reduct of $MS$;
1:  Let $Core = \emptyset$, $RED = \emptyset$, $C_{list} = \emptyset$;
2:  Transform the partially labeled data $MS$ into a decision table $TS = (U', A = C \cup D, V, f'')$;
3:  Compute the discernibility matrix $M$ of $TS$;
4:  Exclude the unnecessary elements of $M$ with the law of absorption;
5:  Add the attribute in the singleton set of $M$ to $Core$ and remove the attributes that do not appear in $M$ from $C$, $RED = Core$, $C = C - Core$;
6:  Sort the attributes in $C$ by frequency ascendingly and add them to $C_{list}$; {The attributes in $C_{list}$ are candidates to be removed}
7:  while $M \neq \emptyset$ do
8:    Get the first attribute $a$ in $C_{list}$;
9:    if $MB_M(a) = \emptyset$ then
10:     $RED = RED \cup \{a\}$, $C_{list} = C_{list} - \{a\}$, $M = M - RS_M(\{a\})$;
11:   else
12:     $M = IS_M(\{a\}) \cup RC_M(\{a\})$, $C_{list} = C_{list} - \{a\}$;
13:   end if
14:   Condense $M$ with the law of absorption and update $C_{list}$;
15: end while
16: return $RED$.
```

Algorithm 1 involves two closely integrated stages: (1) Categorizing all condition attributes into core, candidate and irrelevant attribute sets with discernibility matrix, and (2) Removing redundant attributes from candidate list with Markov blanket theory. In the first stage (from line 1 to line 6), it computes discernibility matrix by Definition 1, and some redundant elements are removed from the discernibility matrix with the law of absorption. Then all condition attributes are partitioned into three different attribute sets with respect to decision attribute. In the second stage (from line 7 to line 15), it further excludes redundant attributes from candidate list C_{list}. The criterion for redundant attribute is whether there is a Markov blanket for this attribute within current C_{list}. Because there is no Markov blanket for any attribute in the core set, these attributes will be first added to the reduct and not be taken into consideration in the following process. In C_{list}, if there is a Markov blanket for a lowest-frequency attribute a, this attribute will be removed from C_{list} and discernibility matrix. After one round of filtering, in C_{list}, some attributes that relate to attribute a will be necessary for classification (formally, they present in discernibility matrix in the form of singleton set, and there is no Markov blanket for these attributes). These attributes will be added to the reduct in the following round selection, while their directly relevant information in discernibility matrix will be disposed of properly. The algorithm terminates when discernibility matrix is empty, which means the reduced attribute set has a nonempty intersection with any nonempty element of discernibility matrix. Therefore, the reduct in the algorithm holds the same discriminating power as all condition attributes.

As shown in Algorithm 1, its major computation lies in the establishment of discernibility matrix. Assume $|C| = m$, $|U/C| = n$, $C_m^{\lceil m/2 \rceil} = k$, the time complexity of building a discernibility matrix is $O(mn^2)$. Because of the symmetry, only $n(n-1)/2$ elements will be generated in discernibility matrix. But the number of necessary elements in discernibility matrix will decrease to worst-case k from $n(n-1)/2$ with the law of absorption [58]. In line 7, if an attribute is selected, this attribute and its supersets will be removed from discernibility matrix. In the worst-case, discernibility matrix will be empty after $|C| = m$ times. Therefore, based on discernibility matrix, the time complexity of computing a reduct is $O(mk)$. While the total time complexity of Algorithm 1 is $O(mn^2 + mk)$, which is approximate to $O(mn^2)$, and its total space complexity is $O(k)$.

For Table 1, its discernibility matrix after the law of absorption is $\{\{a1, a2\}, \{a1, a3\}, \{a2, a4\}, \{a3, a5\}, \{a6\}\}$. Core attribute $a6$ is first put into reduct by Algorithm 1, and all condition attributes except $a6$ and $a7$ are added to candidate list. Then calculates the Markov blanket for $a5$, which has lowest frequency. The directly relevant set of $a5$ is $\{a3, a5\}$; its directly irrelevant set is $\{\{a1, a2\}, \{a1, a3\}, \{a2, a4\}\}$; its relative complement set is $\{a3\}$; the indirectly relative set of that attribute is $\{a1, a3\}$. Actually, the Markov blanket for $a5$ includes two parts of condition attributes: its near neighbors (the relative complement set) and its near neighbor's neighbor (the indirectly relevant set), namely $\{a3\}$ and $\{a1, a3\}$. Therefore, the Markov blanket for $a5$ is $\{a1, a3\}$. Attribute $a5$ will be excluded from the candidate list and discernibility matrix because of the redundancy with respect to attributes $a1$ and $a3$, and the element $\{a1, a3\}$ will also be removed with the law of absorption. Then the discernibility matrix becomes $\{\{a1, a2\}, \{a2, a4\}, \{a3\}\}$. In the second round of selection, attribute $a4$ will be eliminated, and the remains of the discernibility matrix is $\{\{a2\}, \{a3\}\}$. The attributes $a2$ and $a3$ in the singleton sets will be added to the reduct in sequence because there is no Markov blanket for these attributes within the candidate list. Finally, Algorithm 1 gets an optimal reduct $\{a2, a3, a6\}$, but many representative algorithms obtain a superset $\{a1, a2, a3, a6\}$.

4. Rough co-training for partially labeled data

4.1. Basic idea of rough co-training

Co-training [59] proposed by Blum and Mitchell is an important model for partially labeled data. It has been used in many applications successfully, such as web-page categorization, image retrieval and intrusion detection [60]. The standard co-training assumes that there exist two sufficient and redundant sets of attributes or views that describe the data. Two base classifiers are first trained on the initial labeled examples using two attribute sets respectively. Then, alternately, one classifier labels some confident unlabeled examples and adds those examples with predicted labels to the training set of the other one. The classifiers are iteratively retrained until the predefined stopping criterion is met. But in many practical applications, there is only a single natural set of attributes, and the assumption for two sufficient and redundant attribute sets is difficultly satisfied. Although some relaxed co-training models [61–64] have been proposed at present, it is still an open question that how to split a single natural attribute set into two effective attribute sets.

Generally, with correlation measure between the condition and decision attributes, the entire set of condition attributes can be classified into three disjoint categories, namely strongly relevant, weakly relevant and irrelevant attributes [65]. Strong relevance of an attribute indicates that this attribute is always necessary for classification; it can not be removed without affecting the original conditional class distribution. Weak relevance suggests that the attribute is not always necessary but may be important for classification under certain conditions. Irrelevance indicates that the attribute is not necessary at all. An optimal reduct should include all strongly relevant attributes, none of irrelevant attributes, and a subset of weakly relevant attributes.

In fact, with the concept of discernibility matrix in rough set theory, it is easy to classify all condition attributes into three different sets. Each attribute in core set is closely relevant to decision attribute, namely strongly relevant attribute; the attribute which does not appear in discernibility matrix is totally irrelevant to decision attribute, namely irrelevant attribute; and the others are weakly relevant attributes. In previous Table 1, $a6$ is strongly relevant to decision attribute; $a7$ is totally irrelevant to decision attribute; and the remaining attributes $a1$, $a2$, $a3$, $a4$ and $a5$ are weakly relevant.

Usually, there are a number of reducts for a given partially labeled data. Each reduct subspace preserves the discriminating power of the original data. Therefore, it is sufficient to train a good classifier. Moreover, different reduct subspaces describe the data in different views, which suggests that we could employ two diverse reduct subspaces to train the base classifiers of co-training, and then capitalize on the unlabeled examples to improve the learning performance. The structure of rough co-training is shown in Fig. 1.

4.2. Diverse reduct subspaces based co-training for classification

As mentioned earlier, the condition attributes can be classified into strongly relevant, weakly relevant and irrelevant attribute sets, while an optimal reduct should include all strongly relevant attributes, none of irrelevant attributes, and a subset of weakly relevant attributes. Based on Markov blanket theory, algorithm MBARA could acquire an optimal reduct of partially labeled data effectively. As for the other reduct, the theoretically optimal way is to get all reducts of partially labeled data and select the most distinct one from the optimal reduct. Unfortunately, the process of finding all reducts is very time-consuming. Actually, two diverse reducts can be obtained by modifying algorithm MBARA.

As shown in algorithm MBARA, it explores the Markov blanket for each attribute within candidate list C_{list}. If there exists a Markov blanket for an attribute a within C_{list}, this attribute will be excluded from the reduct. This means that the attributes in the Markov blanket have all discernible information about attribute a. In other words, attribute a is redundant with respect to the attributes in its Markov blanket. Intuitively, the attribute and its Markov blanket are interactional. More specifically, there

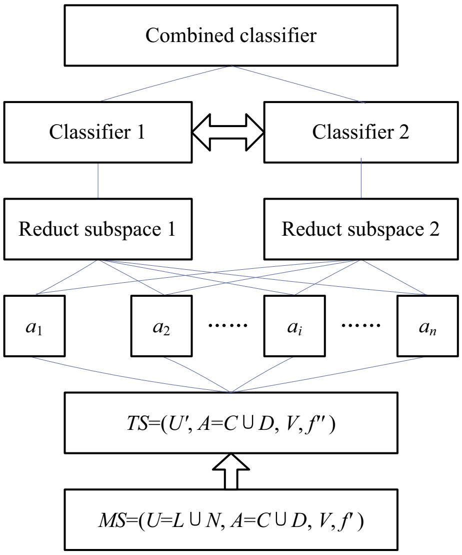


Fig. 1. Structure of rough co-training.

exists an attribute a' contained in the Markov blanket for attribute a and this attribute has another Markov blanket which includes attribute a. Then attribute a' will be excluded from the reduct, and attribute a may be necessary for classification. Therefore, it is rational to use the relationship between the attribute and its Markov blanket to design a practicable algorithm for two diverse reducts. The detailed procedure is shown in Algorithm 2.

Algorithm 2 Search diverse reduct subspaces of partially labeled data

```
Input:
    A partially labeled data $MS = (U = L \cup N, A = C \cup D, V, f')$;
Output:
    Two diverse reducts of $MS$;
1:  Let $Core = \emptyset, RED = \emptyset, C_{list} = \emptyset$;
2:  Transform the partially labeled data $MS$ into a decision table $TS = (U', A = C \cup D, V, f'')$;
3:  Compute the discernibility matrix $M$ of $TS$;
4:  Exclude the unnecessary elements of $M$ with the law of absorption;
5:  Add the attribute in the singleton set of $M$ to $Core$ and remove the attributes that do not appear in $M$ from $C$;
6:  $RED_1 = Core, C_1 = C - Core$; sort the attributes in $C_1$ by frequency ascendingly and add them to $C_{list}$;
7:  Call lines 7–15 of Algorithm 1 to output $RED_1$;
8:  $RED_2 = Core, C_2 = C - Core$; put the attributes in $RED_1 - Core$ on the head of $C_{list}$ and add other attributes in $C_2$ to $C_{list}$
    in proper order;
9:  Call lines 7-15 of Algorithm 1 to output $RED_2$;
10: return $RED_1$ and $RED_2$.
```

There is no difference between Algorithms 1 and 2 from line 1 to line 5. In lines 6 and 7, Algorithm 2 first calculates an optimal reduct with the principle of Algorithm 1. But in line 8, we adjust the strategy for ordering candidate attributes. For the attributes in the first reduct, we expect that these attributes would not appear in the second reduct. The algorithm will put those attributes into candidate list with priority, which means the attributes in the optimal reduct will be first taken into consideration to be excluded from the second reduct. Consequently, two reducts will have fewer common attributes. For Table 1, Algorithm 2 will get the first reduct $\{a2, a3, a6\}$. Attributes $a2$ and $a3$ will be first put into candidate list for the second reduct, and then other attributes $a1$, $a4$, and $a5$ are added in proper order. As a result, the second reduct $\{a1, a4, a5, a6\}$ will be generated by Algorithm 2. There is only one common attribute between two diverse reducts, namely core attribute $a6$.

As mentioned above, based on Markov blanket theory, Algorithm 2 could be used to compute two diverse reducts of partially labeled data. Since each reduct subspace retains the discriminating power of the original data, it is sufficient to train the classifier with good generalization. What is more, two reduct subspaces describe the partially labeled data from different points of view, and the diversity of the two classifiers comes into existence as a result. Based on two diverse reduct subspaces, the rough co-training model depicted in Fig. 1 can be formulated by Algorithm 3.

Algorithm 3 first decomposes all condition attributes into two diverse reducts. Then, on the initial labeled examples, two base classifiers are trained in each reduct subspace respectively. For any unlabeled example, there are three different cases for the results of the two classifiers, namely only one classifier is predictable, and both of classifiers are predictable or unpredictable. In the first case, the classifiers could learn from each other, while rough co-training just capitalizes on these

Algorithm 3 Rough co-training for partially labeled data

```
Input:
    A partially labeled data $MS = (U = L \cup N, A = C \cup D, V, f')$;
Output:
    A combined classifier $f$;
1:  Decompose condition attribute set $C$ into two diverse reducts $RED_1$ and $RED_2$ by Algorithm 2;
2:  Let $L_1 = L_2 = L$ and train two base classifiers $f_1$ and $f_2$ on $L$ using reducts $RED_1$ and $RED_2$ respectively;
3:  Add the unpredictable examples of classifiers $f_1$ and $f_2$ to sets $N_1$ and $N_2$ respectively;
4:  while $N_1 \cup N_2 \neq N_1 \cap N_2$ do
5:      Add the predictable examples $N_2 - (N_1 \cap N_2)$ of $f_1$ with the class symbols to the training set $L_2$ of $f_2$;
6:      Add the predictable examples $N_1 - (N_1 \cap N_2)$ of $f_2$ with the class symbols to the training set $L_1$ of $f_1$;
7:      Retrain classifiers $f_1$ and $f_2$ and update the unpredictable example sets $N_1$ and $N_2$ respectively;
8:  end while
9:  Combine classifiers $f_1$ and $f_2$;
10: return Combined classifier $f$.
```

Table 2
Diversity matrix of the classifiers on the unlabeled data.

	f_2 predictable (p)	f_2 unpredictable (u)
f_1 predictable (p)	n_{pp}	n_{pu}
f_1 unpredictable (u)	n_{up}	n_{uu}

unlabeled examples to enhance its performance. After retraining the classifiers in line 7, the first case will happen again on other unlabeled examples. The classifiers will learn from each other once again. In an optimal situation, every unlabeled examples will be labeled a class symbol. Therefore, the performance of rough co-training may be improved to a large extent.

We now analyze the time complexity of Algorithm 3 before an empirical study of the effectiveness. In terms of condition attribute $|C| = m$ and example $|U| = n$, the time complexity of training a base classifier is $O(mn)$. In each round, the classifiers learn from each other on the unlabeled examples. After retraining the classifiers, some new divergent examples (only one classifier is predictable) may appear, and the classifiers could learn from each other again. Algorithm 3 has a best-case complexity $O(mn)$ when all unlabeled examples could be predicted by two base classifiers in the first round, and a worst-case complexity $O(mn^2)$ when there is only one useful unlabeled example in each round of co-training. On the whole, with two diverse reducts of a given partially labeled data, the time complexity of Algorithm 3 is less than $O(mn^2)$.

4.3. The theoretical analysis on the effectiveness of rough co-training

In order to work, co-training requires two distinct properties of the underlying data distribution. One is that there should at least exist two sufficient attribute sets(views) for classification. The other is that two attribute sets should on the other hand not be too highly correlated. In the view of rough set theory, each reduct does not lose any discriminating power of the original data. Therefore, the classifiers trained in reduct subspaces will have a good generalization power. As for the second property, Balcan [64] proven that the weaker "expansion" assumption on the underlying data distribution was sufficient for iterative co-training to succeed given appropriately strong classifier on each attribute set. Wang and Zhou [66] also showed that the co-training process can succeed even when the two classifiers had large difference. Theoretically, those conclusions guarantee the effectiveness of rough co-training.

Assume that a partially labeled data is consisted of labeled examples $L(|L| = l)$, unlabeled examples $N(\ |N| = n)$ and testing examples $T(|T| = t)$. Each example can be described as $X = (X_1, X_2)$, where X_1 and X_2 correspond to two different reduct subspaces. Two classifiers trained in different reduct subspaces are denoted as f_1 and f_2. For any unlabeled example, each classifier will have two different results, namely predictable or unpredictable. On all unlabeled examples N, the diversity of the two classifiers can be denoted as what is shown in Table 2.

Where n_{pp} is the number of unlabeled examples predicted by both classifiers and n_{uu} is the number of unpredictable examples of the two classifiers. The numbers of unlabeled examples predicted by only one of the classifiers are denoted as n_{pu} and n_{up} respectively. In the first round, classifier f_2 will label n_{up} unlabeled examples to the training set of classifier f_1. The number of unpredictable examples of classifier f_1 will decrease from n_{up}+n_{uu} to n_{uu}. Analogically, classifier f_2 will have only n_{uu} unpredictable examples. On the whole, the number of labeled examples will increase from n_{pp} to $n_{pp} + n_{pu} + n_{up}$. After updating the classifiers, some unpredictable examples for both classifiers in the first round may be predictable. Then the second round of co-training could happen. In an optimal situation, each classifier could predict arbitrary instance of input space after several rounds of co-training.

On all testing examples T, the performance of the classifiers before co-training can also be presented in Table 3.

In Table 3, t_{cc}, t_{ii} and t_{uu} are the numbers of testing examples predicted by both classifiers correctly, incorrectly and uncertainly respectively. The symbols t_{ci} and t_{ic} denote the numbers of testing examples predicted by only one of the classifiers correctly. While t_{cu} and t_{uc} are the numbers of testing examples that one classifier predicts correctly but the other one predicts uncertainly. The testing examples that one classifier predicts incorrectly but the other one predicts

Table 3
Diversity matrix of the classifiers on the testing data.

	f_2 correct(c)	f_2 incorrect(i)	f_2 unpredictable(u)
f_1 correct(c)	t_{cc}	t_{ci}	t_{cu}
f_1 incorrect(i)	t_{ic}	t_{ii}	t_{iu}
f_1 unpredictable(u)	t_{uc}	t_{ui}	t_{uu}

Table 4
UCI data sets.

Data set	Attributes	Instances	Missing	Reducts
Wisconsin Breast Cancer Database (WBCD)	9	699	Yes	20
Tic-Tac-Toe (TTT)	9	958	No	9
Mushroom (MR)	22	8124	Yes	292
Wisconsin Diagnostic Breast Cancer (WDBC)	30	569	No	212
Ionosphere (Iono)	34	351	No	203
King Rook versus King Pawn (KRVSKP)	36	3196	No	4

Table 5
The reduct of data set under the label rate α =10%.

Data set	Attributes	Min	Max	Average	Reduct	Approx. rate (%)
WBCD	9	6	9	8	4	66.7
TTT	9	8	8	8	8	100
MR	22	14	15	15	4	28.6
WDBC	30	13	15	14	8	61.5
Iono	20	16	19	17	8	50
KRVSKP	36	30	32	31	29	96.7

uncertainly are denoted by t_{iu} and t_{ui}. Before co-training, the performance of classifiers f_1 and f_2 are $(t_{cc} + t_{ci} + t_{cu})/t$ and $(t_{cc}+t_{ic}+t_{uc})/t$ respectively. In an optimal situation, the performance of classifiers f_1 and f_2 will be $(t_{cc}+t_{ci}+t_{cu}+t_{uc}+t_{uu})/t$ and $(t_{cc} + t_{ic} + t_{uc} + t_{cu} + t_{uu})/t$ respectively after co-training.

5. Empirical analysis

5.1. Benchmark data sets

Six UCI data sets [67] are used in the experiments. The detailed information of these data sets is shown in Table 4. The fourth column indicates that whether the data set has missing values and the last column is the number of reducts in each data set. Data set "Ionosphere" contains 351 samples described by 34 continuous attributes. We use the principle of equal frequency [68] to discretize continuous data. Data sets "Wisconsin Breast Cancer Database" and "Mushroom" which have missing values in different attributes will be completed by conditioned mean(or mode) [68].

For each data set, 10-fold cross validation is employed for evaluation. In each fold, the training set is randomly partitioned into labeled set L and unlabeled set N for a given label rate (α), which can be computed by the size of L over the size of $L \cup N$. For instance, if a training set has 1000 examples, under the label rate α =10%, it will produce a set with 100 labeled examples and a set with 900 unlabeled examples. In order to simulate the effectiveness on different numbers of unlabeled examples, the data sets are investigated with different label rates.

5.2. Attribute reduction for partially labeled data

In Section 3, a Markov blanket based attribute reduction algorithm is proposed to deal with partially labeled data. In order to show its effectiveness, we collect the reduct information of each data set under the label rate α =10%. The detailed information is shown in Table 5.

The third and fourth columns denote the minimum and maximum cardinality of reducts in 10-fold cross validation. The column "Reduct" means the cardinality of the optimal reduct of data set under the label rate 100%. The "Approx. rate" shows the effectiveness of Algorithm 1, which can be computed by the number of "Reduct" over the number of "Min". On data set "TTT", we can see that Algorithm 1 could produce an optimal reduct that bears comparison with true reduct even with 10% training examples labeled. And some promising results are also shown on other data sets. By observing the experimental results, we find that the irrelevant attributes which have no valuable information for classification will be removed from the reduct in each fold cross validation, whereas different parts of weakly relevant attributes will be excluded from the reduct with different labeled examples. The higher the label rate, the fewer weakly relevant attributes.

5.3. The effectiveness of rough co-training

In order to show the advantage of our model over traditional rough set approaches, rough co-training employs two homogeneous rough classifiers. As a matter of fact, if we use two heterogeneous classifiers (e.g. decision tree and SVM) to expand the diversity, rough co-training could obtain better performance. In each round of co-training, there may exist

Table 6
Average accuracy of the compared algorithms under the label rate 10%.

Data set	Self-training			Random co-training			Rough co-training		
	Initial	Final	Improvement (%)	Initial	Final	Improvement (%)	Initial	Final	Improvement (%)
WBCD	0.8511	0.8586	1.4	0.8272	0.9051	**25.2**	0.8653	0.9239	17.3
TTT	0.5372	0.5924	12.6	0.4709	0.5963	21.9	0.5044	0.6177	**24.1**
MR	0.9907	0.9907	0	0.9624	0.9838	**3.7**	0.9871	0.9911	2.3
WDBC	0.8512	0.8512	0	0.7522	0.8354	20	0.8442	0.8863	**28.3**
Iono	0.6586	0.6729	11.4	0.5854	0.6968	**34.3**	0.6454	0.7079	27.1
KRVSKP	0.7252	0.7709	11.7	0.6108	0.7106	18.5	0.7027	0.8102	**18.8**
Avg.	0.7690	0.7895	6.2	0.7015	0.7880	20.6	0.7582	0.8229	19.7

many useful unlabeled examples for each classifier. In order not to increase the classification noise of the classifiers, we only pick some unlabeled examples with higher quality. More specifically, because the rule with more attributes tends to predict the example conservatively, we rank the unlabeled examples by the length of corresponding rule in the classifier and only 10% unlabeled examples with higher score will be selected for co-training. For the final performance of rough co-training, there are many proposed approaches to combining classifiers [69,70]. We use the average accuracy of the two classifiers to represent the final performance.

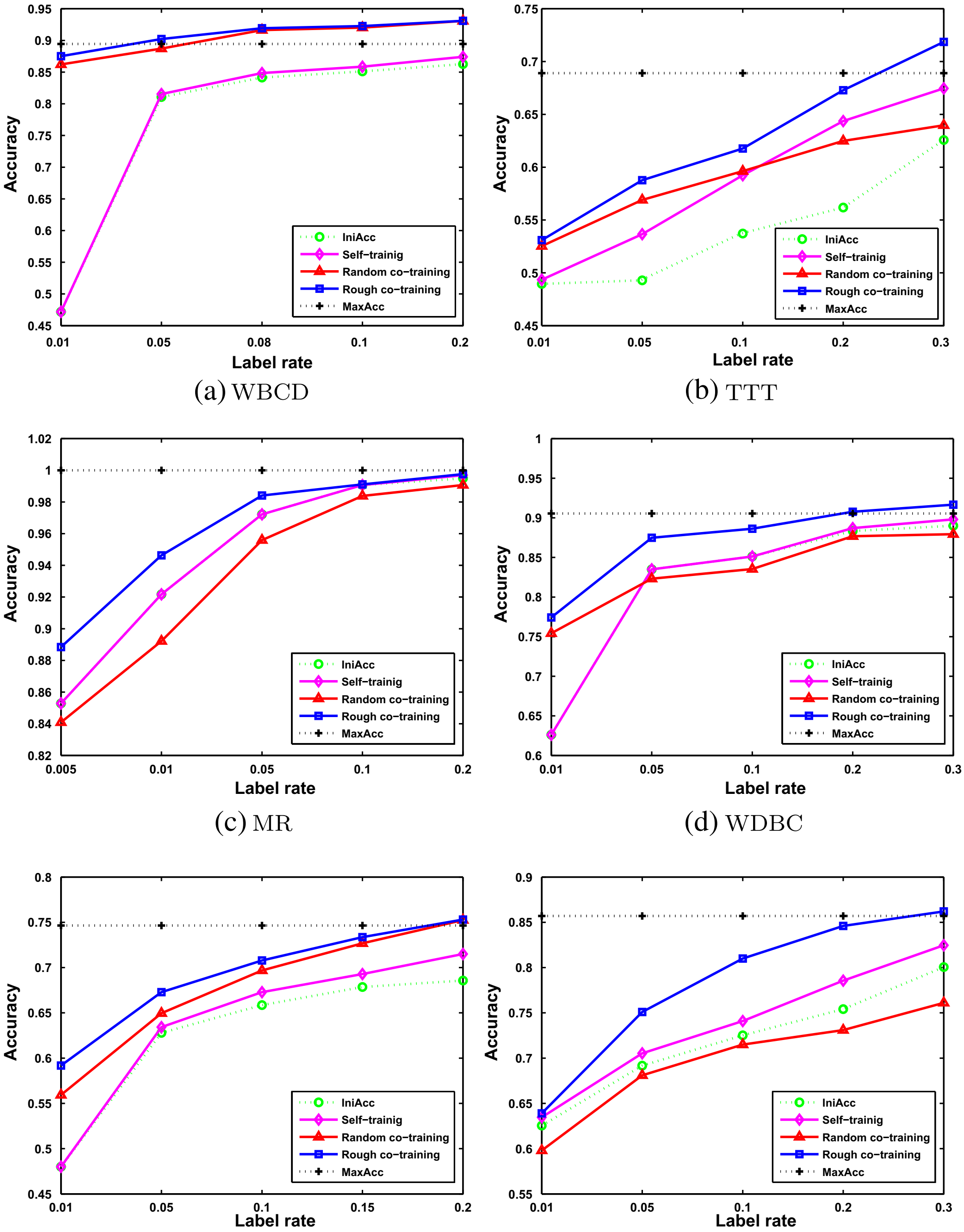
For comparison, standard co-training, self-training and traditional rough set approaches are also evaluated in our experiments. As mentioned above, the standard co-training assumes that there exist two sufficient and redundant sets of attributes or views to describe the data. Therefore, it could not be directly applied to the data sets because of there is no natural separation of the attributes. However, some previous research [62,71] indicated that co-training could still benefit from the unlabeled examples by randomly splitting all attributes into two sets. Thus, in our experiments, we split the attributes in each data set into two disjoint sets with almost equal size and then make standard co-training work on them(random co-training). Self-training [62] is also an effective model for partially labeled data. The model first trains a classifier on the labeled examples and then keeps on refining the classifier with the self-labeled examples. As for traditional rough set approaches, they only use the labeled examples to train one rough classifier. The performance of these approaches could be acquired in the first round of self-training. Therefore, we do not investigate them again in the experiments.

For each data set under a specific label rate, 10-fold cross validation is applied, and the results are averaged. Table 6 shows the average accuracy of the learned classifiers under the label rate 10%. For each algorithm in the table, "initial" and "final" denote the average accuracy of the classifiers learned only with the labeled examples and those further refined with the unlabeled examples respectively. The maximum performance improvement of the algorithms is denoted by "improvement.", which can be computed by subtracting the average accuracy of the initial classifiers from that of the final classifiers in 10-fold cross validation. The highest performance improvement among three different algorithms is boldfaced. The row "Avg." in the table shows the average results over all data sets.

In Table 6, the performance of each algorithm is boosted, but in different ways. Self-training attains slight improvement on most of data sets. But we also see that self-training fails in data sets "MR" and "WDBC". Although there is a great improvement in the performance, the initial accuracy of random co-training is so bad that its final performance may be lower than that of traditional rough set approaches (i.e. the initial accuracy of self-training). Because of using two classifiers, the initial accuracy of rough co-training may be lower than that of self-training. But rough co-training achieves significant improvement on most of data sets. As for the maximum performance improvement in 10-fold cross validation, random co-training is equally matched to rough co-training, nevertheless its final performance is far from that of rough co-training. By averaging the performance of each algorithm over all data sets, self-training and random co-training obtain an improvement over traditional rough set approaches by 2.1% and 1.9% respectively, whereas rough co-training achieves an overall 5.4% improvement.

In order to further investigate the effectiveness of rough co-training, the experiments under other different label rates are also performed. The performance of each algorithm is shown in Fig. 2. In each subfigure, "IniAcc" is the performance of traditional rough set approaches (without any benefit of the unlabeled examples). The algorithms learned from both labeled and unlabeled examples are denoted by "Self-training", "Random co-training" and "Rough co-training" respectively. "MaxAcc" shows the accuracy of rough classifier trained on all examples with true class symbols (i.e. the case under the label rate 100%). Note that the label rate varies with different data sets.

In Fig. 2, rough co-training achieves significant improvement on most of the data sets. Self-training and random co-training also attain improvement in their performance, but they could not be comparable to rough co-training. In self-training, there is only one classifier involved in the learning process, thus the classifier has to label the unlabeled examples totally by itself. If the initial classifier is biased, the final performance of self-training may be very poor. This claim is confirmed by Fig. 2(a)–(e) in the first two label rates. Random co-training uses two classifiers in the learning process, however its attribute sets are generated by randomly split. Consequently, the initial performance of random co-training is much worse than that of single classifier trained in original attribute set. This claim is consistent with Table 6, where the initial accuracy of random co-training is much lower than that of self-training. As a result of poor performance, the classifiers in random co-training may mislabel the unlabeled examples for each other in the process of co-training. When the diversity of the two classifiers on the unlabeled examples could not compensate for their errors, the final performance of random co-training is worse than that of self-training. Data sets except "WBCD" and "Iono" are good instances. Although our rough co-training model has



(a) WBCD

(b) TTT

(c) MR

(d) WDBC

(e) Iono

(f) KRVSKP

Fig. 2. Average accuracy of data sets under different label rates.

similar working style with random co-training, it uses two diverse reducts to train the base classifiers. Each reduct subspace preserves the discriminating power of the original data. Therefore, it is sufficient to train a good base classifier. What is more, each classifier could be refined by the examples labeled by its concomitant classifier instead of itself. Some unpredictable examples for one classifier may be classifiable after co-training. Therefore, the performance of rough co-training will be improved. This interprets that rough co-training is better than self-training and random co-training on all data sets.

Interestingly, on some data sets, the performance of rough co-training under certain label rates even outperforms that of the classifier learned from all training examples labeled(i.e. α =100%). For example, when 20% training examples are labeled in data sets “WDBC” and “Iono”, rough co-training, by exploiting the unlabeled examples, is able to reach the performance comparable to that of rough classifier trained on all training examples with true class symbols. Data sets “TTT” and “KRVSKP”

Table 7
The effectiveness of rough co-training on all data sets.

Data sets	Attributes	Reduct	Avg. improvement (%)	Turning-point
WBCD	9	4	12.0	0.05
TTT	9	8	10.1	0.3
MR	22	4	2.1	0.4
WDBC	30	8	5.3	0.2
Iono	20	8	7.7	0.2
KRVSKP	36	29	6.7	0.3
Avg.	–	–	7.3	0.24

under the label rate α =30% are also good cases. This phenomena should be ascribed to the utilization of the unlabeled data and multi-classifiers. Generally, the label rate under which rough co-training outperforms the maximum accuracy of rough classifier varies with different data sets(turning-point). In the extreme situation(under the label rate α =100%), our rough co-training model degenerates to an ensemble system [72], which often achieves better performance than single classifier. Table 7 shows the detailed information about performance improvement and turning-point on each data set.

In summary, the unlabeled examples are beneficial for training the classifier. Traditional rough set approaches, self-training and random co-training are ineffective to deal with partially labeled data, while rough co-training is able to improve the performance by capitalizing on the unlabeled examples in effective and efficient way. Furthermore, the remarkable property "turning-point" is shown in the rough co-training model, which could be used to conduct the design of effective learning algorithm and alleviate human effort in labeling the example of practical problem.

6. Conclusions

Rough set theory is a supervised learning model which could deal with imprecise, uncertain and incomplete information effectively. However, quite a few practical problems involve both labeled and unlabeled examples. In this paper, the rough co-training model is proposed for partially labeled data, which could use the unlabeled examples to enhance the performance of the classifier trained on few labeled examples. More specifically, rough co-training uses two diverse reduct subspaces to train its base classifiers, and then each classifier is iteratively refined on the unpredictable examples labeled by its concomitant classifier. Theoretically speaking, the reducts are optimal subsets of the original attributes because they avoid the loss of discriminating information and have the least redundancy. Therefore, the classifiers trained in reduct subspaces will have good generalization. At the same time, the classifiers constructed in different reduct subspaces will have a great opportunity to get the diversity. Experiments on UCI data sets also verify the effectiveness of rough co-training. In terms of classification accuracy, rough co-training is better than self-training and random co-training, and greatly dominates traditional rough classifier trained only on the labeled examples. What is more, under certain label rates, the performance of rough co-training is even better than that of single rough classifier with all training examples labeled. Although some promising results are attained in the rough co-training model, there are several research works we expect to do in the future. Since rough co-training is sensitive to the initial labeled data, incorporating active learning into rough co-training may be a good solution. Another interesting future work is to employ two heterogeneous classifiers to expand the diversity, which is anticipated to make rough co-training perform better.

Acknowledgements

The authors would like to thank Editor-in-Chief, Area Editor, and the anonymous referees for their valuable comments and helpful suggestions. The work is partially supported by the National Natural Science Foundation of China (Serial No. 60970061, 61075056) and Shanghai Leading Academic Discipline Project (Serial No. B004).

References

[1] Z. Pawlak, Rough sets, International Journal of Computer and Information Science 11 (1982) 341–356.
[2] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht, 1991.
[3] Q. Liu, Rough Sets and Rough Reasoning, Science Press, Beijing, 2001.
[4] G.Y. Wang, Rough Set Theory and Knowledge Acquisition, Xi'an Jiaotong University Press, Xi'an, 2001.
[5] L. Polkowski, Rough sets: Mathematical foundations, in: Advances in Soft Computing, Physica-Verlag, Heidelberg, 2002.
[6] W.X. Zhang, W.Z. Wu, J.Y. Liang, et al, Rough Set Theory and Methods, Science Press, Beijing, 2003.
[7] D.Q. Miao, D.G. Li, Rough Set Theory, Algorithms and Applications, Tsinghua University Press, Beijing, 2008.
[8] T.Y. Lin, Neighborhood systems and approximation in database and knowledge base systems, in: Proceedings of the Fourth International Symposium on Methodologies of Intelligent Systems, Poster Session, October 12–15, 1989, pp. 75–86.
[9] T.Y. Lin, Granulation and nearest neighborhoods: rough set approach, in: W. Pedrycz (Ed.), Granular Computing: An Emerging Paradigm, Physica-Verlag, Heidelberg, Germany, 2001, pp. 125–142.
[10] T.Y. Lin, Neighborhood systems: mathematical models of information granulations, in: Proceeding of 2003 IEEE International Conference on Systems, Man and Cybernetics, Washington, DC, USA, October 5–8, 2003, pp. 3188–3193.
[11] Y.Y. Yao, Relational interpretations of neighborhood operators and rough set approximation operators, Information Sciences 111 (1998) 239–259.
[12] Y.Y. Yao, Granular computing using neighborhood systems, in: R. Roy, T. Furuhashi, P.K. Chawdhry (Eds.), Advances in Soft Computing: Engineering Design and Manufacturing, Springer-Verlag, London, 1999, pp. 539–553.

[13] Q.H. Hu, D.R. Yu, J.F. Liu, et al, Neighborhood rough set based heterogeneous feature subset selection, Information Sciences 178 (2008) 3577–3594.
[14] Q.H. Hu, D.R. Yu, Z.X. Xie, Neighborhood classifiers, Expert Systems with Applications 34 (2008) 866–876.
[15] L. Polkowski, A. Skowron, J. Zytkow, Rough foundations for rough sets, in: Proceedings of The Third International Workshop on Rough Sets and Soft Computing (RSSC'94), CA, USA, November 10–12, 1994, pp. 142–149.
[16] A. Skowron, J. Stepaniuk, Tolerance approximation spaces, Fundamenta Informaticae 27 (1996) 245–253.
[17] T.B. Ho, N.B. Nguyen, Nonhierarchical document clustering based on a tolerance rough set model, International Journal of Intelligent Systems 17 (2002) 199–212.
[18] C.L. Ngo, H.S. Nguyen, A tolerance rough set approach to clustering web search results, in: J.F. Boulicaut, et al (Ed.), PKDD 2004, Springer-Verlag, Berlin, Heidelberg, 2004, pp. 515–517.
[19] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, International Journal of General Systems 17 (1990) 191–209.
[20] A.M. Radzikowska, E.E. Kerre, A comparative study of fuzzy rough sets, Fuzzy Sets and Systems 126 (2002) 137–155.
[21] Q.H. Hu, Z.X. Xie, D.R. Yu, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognition 40 (2007) 3509–3521.
[22] Q.H. Hu, L. Zhang, D.G. Chen, et al, uncertainty measures and applications, International Journal of Approximate Reasoning 51 (2010) 453–471.
[23] S. Greco, B. Matarazzo, R. Slowinski, The use of rough sets and fuzzy sets in MCDM, in: T. Gal, T. Hanne, T. Stewart (Eds.), Advances in Multiple Criteria Decision Making, Kluwer Academic Publishers, Boston, 1999, pp. 14.1–14.59.
[24] S. Greco, B. Matarazzo, R. Slowinski, Rough sets theory for multi-criteria decision analysis, European Journal of Operational Research 129 (2001) 1–47.
[25] J. Blaszczynski, S. Greco, R. Slowinski, et al, Monotonic variable consistency rough set approaches, International Journal of Approximate Reasoning 50 (2009) 979–999.
[26] Z. Bonikowski, E. Bryniarski, U. Wybraniec-Skardowska, Extensions and intentions in the rough set theory, Information Sciences 107 (1998) 149–167.
[27] W. Zhu, F.Y. Wang, On three types of covering rough sets, IEEE Transactions on Knowledge and Data Engineering 19 (2007) 1131–1144.
[28] G.L. Liu, Y. Sai, A comparison of two types of rough sets induced by coverings, International Journal of Approximate Reasoning 50 (2009) 521–528.
[29] Y.Y. Yao, S.K.M. Wong, P. Lingras, A decision-theoretic rough set model, in: Z.W. Ras, M. Zemankova, M.L. Emrichm (Eds.), Methodologies for Intelligent Systems 5 – Proceedings of the 5th International Symposium on Methodologies for Intelligent Systems, North-Holland, New York, Knoxville, TN, USA, 1990, pp. 17–25.
[30] Y.Y. Yao, S.K.M. Wong, A decision theoretic framework for approximating concepts, International Journal of Man–machine Studies 37 (1992) 793–809.
[31] Y.Y. Yao, Three-way decisions with probabilistic rough sets, Information Sciences 180 (2010) 341–353.
[32] W. Ziarko, Variable precision rough sets model, Journal of Computer and Systems Sciences 46 (1993) 39–59.
[33] M. Beynon, Reducts within the variable precision rough sets model: a further investigation, European Journal of Operational Research 134 (2001) 592–605.
[34] D. Slezak, Rough sets and Bayes factor, in: Transactions on Rough Sets III, LNCS vol. 3400, Springer-Verlag (2005) 202–229.
[35] D. Slezak, W. Ziarko, The investigation of the Bayesian rough set model, International Journal of Approximate Reasoning 40 (2005) 81–91.
[36] Z. Pawlak, S.K. M Wong, W. Ziarko, Rough sets: probabilistic versus deterministic approach, International Journal of Man–Machine Studies 29 (1988) 81–95.
[37] T. Beauboef, F.E. Petry, G. Arora, Information-theoretic measures of uncertainty for rough sets and rough relational databases, Information Sciences 109 (1998) 185–195.
[38] Q.G. Duan, D.G. Miao, K.M. Jin, A rough set approach to classifying web page without negative examples, in: Proceedings of the 11th Pacific–Asia Conference on Knowledge Discovery and Data Mining, LNAI 4426, 2007, pp. 481–488.
[39] Y.Y. Yao, Probabilistic approaches to rough sets, Expert Systems 20 (2003) 287–297.
[40] Y.Y. Yao, Probabilistic rough set approximations, International Journal of Approximate Reasoning 49 (2008) 255–271.
[41] Y.Y. Yao, Y. Zhao, Attribute reduction in decision-theoretic rough set models, Information Sciences 178 (2008) 3356–3373.
[42] P. Lingras, M. Chen, D.Q. Miao, Semi-supervised rough cost/benefit decisions, Fundamenta Informaticae 94 (2009) 1–12.
[43] X.P. Gu, S.K. Tso, Applying rough set concept to neural network based transient stability classification of power systems, in: Proceedings of the 5th International Conference on Advances in Power System Control, Operation and Management, Hong Kong, 2000, pp. 400–404.
[44] S. Wang, X. Wang, D.W. Bi, et al., Collaborative statistical learning with rough feature reduction for visual target classification, in: Proceedings of International Joint Conference on Neural Networks, Hong Kong, 2008, pp. 1151–1156.
[45] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, in: R. Slowinski (Ed.), Intelligent Decision Support-Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, 1992, pp. 331–362.
[46] D. Slezak, Approximate Markov boundaries and Bayesian networks: rough set approach, in: M. Inuiguchi, S. Hirano, S. Tsumoto (Eds.), Rough Set Theory and Granular Computing, Springer, 2003, pp. 109–121.
[47] Y.Y. Yao, Two views of the theory of rough sets in finite universes, International Journal of Approximate Reasoning 15 (1996) 291–317.
[48] Y.Y. Yao, Constructive and algebraic methods of the theory of rough sets, Information Sciences 109 (1998) 21–47.
[49] Y.Y. Yao, A comparative study of fuzzy sets and rough sets, Information Sciences 109 (1998) 227–242.
[50] Z. Pawlak, A. Skowron, Rudiments of rough sets, Information Sciences 177 (2007) 3–27.
[51] Z. Pawlak, A. Skowron, Rough sets: some extensions, Information Sciences 177 (2007) 28–40.
[52] Z. Pawlak, A. Skowron, Rough sets and boolean reasoning, Information Sciences 177 (2007) 41–73.
[53] K. Thangavel, A. Pethalakshmi, Dimensionality reduction based on rough set theory: a review, Applied Soft Computing 9 (2009) 1–12.
[54] D.Q. Miao, J. Wang, Information-based algorithm for reduction of knowledge, in: Proceedings of the IEEE International Conference on Intelligent Processing Systems, Beijing, China, 1997, pp. 1155–1158.
[55] M. Inuiguchi, Y. Yoshioka, Y. Kusunoki, Variable-precision dominance-based rough set approach and attribute reduction, International Journal of Approximate Reasoning 50 (2009) 1199–1214.
[56] J. Qian, D.Q. Miao, Z.H. Zhang, et al, Hybrid approaches to attribute reduction based on indiscernibility and discernibility relation, International Journal of Approximate Reasoning 52 (2011) 212–230.
[57] D. Koller, M. Sahami, Toward optimal feature selection, in: Proceedings of the 20th International Conference on Machine Learning, Bari, Italy, 1996, pp. 284–292.
[58] J.Y. Wang, C. Gao, An improved algorithm for attribute reduction based on discernibility matrix, Computer Engineering 35 (2009) 66–68, (in Chinese).
[59] A. Blum, T.M. Mitchell, Combining labeled and unlabeled data with co-training, in: Proceedings of the 11th Annual Conference on Computational Learning Theory, Madison Wisconsin, USA, 1998, pp. 92–100.
[60] X. Zhu, A.B. Goldberg, Introduction to Semi-Supervised Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers, 2009.
[61] S. Goldman, Y. Zhou, Enhancing supervised learning with unlabeled data, in: Proceedings of the 17th International Conference on Machine Learning, San Francisco, USA, 2000, pp. 327–334.
[62] K. Nigam, R. Ghani, Analyzing the effectiveness and applicability of co-training, in: Proceedings of the 9th ACM International Conference on Information and Knowledge Management, McLean, VA, 2000, pp. 86–93.
[63] M. Li, Z.H. Zhou, Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples, IEEE Transactions on Systems, Man and Cybernetics – Part A 37 (2007) 1088–1098.
[64] M.F. Balcan, A. Blum, K. Yang, Co-training and expansion: Towards bridging theory and practice, in: L.K. Saul, Y. Weiss, L. Bottou (Eds.), Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, 2005, pp. 89–96.
[65] Q.H. Hu, D.R. Yu, Z.X. Xie, et al, EROS: ensemble rough subspaces, Pattern Recognition 40 (2007) 3728–3739.
[66] W. Wang, Z.H. Zhou, Analyzing co-training style algorithms, in: Proceedings of the 18th European Conference on Machine Learning, Warsaw, Poland, 2007, pp. 454–465.

[67] C. Blake, E. Keogh, C.J. Merz, UCI repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA, 1998. <http://www.ics.uci.edu/∼mlearn/MLRepository.html>.

[68] A. Øhrn, J. Komorowski, ROSETTA: a rough set toolkit for analysis of data, in: Proceedings of the 3rd International Joint Conference on Information Sciences, Fifth International Workshop on Rough Sets and Soft Computing (RSSC'97), Durham, NC, USA, March, 1997, pp. 403–407.

[69] J. Kittler, M. Hatef, R.P.W. Duin, et al, Combining classifiers: a theoretical framework, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (1998) 226–239.

[70] R.P.W. Duin, D.M.J. Tax, Experiments with classifier combining rules, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems (Proceedings of the 1th International Workshop, MCS 2000, Cagliari, Italy, June 2000), LNCS 1857, Springer, Berlin, 2000, pp. 16–29.

[71] J. Chan, I. Koprinska, J. Poon, Co-training with a single natural feature set applied to email classification, in: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Beijing, China, 2004, pp. 586–589.

[72] L. Hansen, P. Salamon, Neural network ensemble, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1990) 993–1001.