International Journal of Approximate Reasoning 103 (2018) 394–413

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

A three-way selective ensemble model for multi-label classification ☆

Yuanjian Zhang [a,b], Duoqian Miao [a,b], Zhifei Zhang [b,c,*], Jianfeng Xu [a,b,d], Sheng Luo [a,b]

[a] Department of Computer Science and Technology, Tongji University, Shanghai, 201804, China
[b] Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, 201804, China
[c] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, 210023, China
[d] Software College, Nanchang University, Jiangxi, 330047, China

ARTICLE INFO

Article history:
Received 14 March 2018
Received in revised form 9 September 2018
Accepted 10 October 2018
Available online 17 October 2018

Keywords:
Multi-label classification
Three-way decisions
Selective ensemble
Uncertainty
Probabilistic rough set

ABSTRACT

Label ambiguity and data complexity are widely recognized as major challenges in multi-label classification. Existing studies strive to find approximate representations concerning label semantics, however, most of them are predefined, neglecting the personality of instance-label pair. To circumvent this drawback, this paper proposes a three-way selective ensemble (TSEN) model. In this model, three-way decisions is responsible for minimizing uncertainty, whereas ensemble learning is in charge of optimizing label associations. Both label ambiguity and data complexity are firstly reduced, which is realized by a modified probabilistic rough set. For reductions with shared attributes, we further promote the prediction performance by an ensemble strategy. The components in base classifiers are label-specific, and the voting results of instance-based level are utilized for tri-partition. Positive and negative decisions are determined directly, whereas the deferment region is determined by label-specific reduction. Empirical studies on a collection of benchmarks demonstrate that TSEN achieves competitive performance against state-of-the-art multi-label classification algorithms.

1. Introduction

Multi-label classification has been prevailing in past decades. Unlike traditional single-label learning paradigm whose label semantics is unique, instances may be associated with multiple labels. To name a few, in sentiment analysis, emotions of a person may be both *delighted* and *peaceful*; In music information retrieval, a piece of waltz contains the semantics of *Chopin*, *Austria*, and *piano*. The number of labels per instance differs across the instances, leading to the problem of label sparsity. The goal of multi-label classification is to predict labels for unseen instances with maximal accuracy (or minimal loss).

Performing multi-label classification requires much more efforts on complicated data characteristics. Even for the simplest case, i.e., perfect information of binary labels is available, the complexity from high dimensional small-sample-size requires delicate operations. To date, multi-label classification algorithms can be roughly summarized into two categories

☆ This paper is part of the Virtual special issue on Uncertainty in Granular Computing, Edited by Duoqian Miao and Yiyu Yao.

* Corresponding author at: Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, 201804, China.
E-mail address: zhifeizhang@tongji.edu.cn (Z. Zhang).

https://doi.org/10.1016/j.ijar.2018.10.009
0888-613X/

[1,2]: algorithm adaptation methods, and problem transformation methods. The algorithm adaptation methods extends specific algorithms developed in single-label scenario directly, such as ML-*k*NN [3], ML-Forest [4] and ML-TSVM [5]. Major limitation of these methods is that they are tailored from particular classifiers, and lack generality. In contrast, the problem transformation methods, which are the focus of this paper, transform learning task to fit single-label learning paradigm. The robustness can be enhanced if label correlations can be well considered. With regard to the label co-occurrence phenomenon, there are generally three strategies. The simplest kind is *binary relevance* (BR) [6,7], which is criticized for the absent considerations of label correlations. However, it is quite intuitive and interpretable, and inspires the researches on label-specific methods [8,9]. Recently, label-specific feature [10,11] is considered to reflect the specific characteristic on label level. To leverage label correlation, *second-order* and *high-order* strategies are subsequently proposed. The second-order strategy claims that label correlation only exists in *pair-wise* style. The comparison of pair-wise relevance flourishes the study of ranking-based solutions [12–14]. The high-order strategy is promised to approximate complicated label correlation at the expense of considerable computation [15]. A new direction is hybridization. For example, random k-labelset [16–18] ensembles an assemble of high-order classifiers to generate a more robust result without significantly improving complexity. The data complexity is gradually resolved as the tasks in each layer are simplified [19,20].

Selective ensemble [21], prunes the unnecessary base classifiers and achieves more accurate performance in some cases. A hybrid approach which integrates two types of selective ensemble techniques for multi-label classification is proposed [22], where the label-level selection criterion is composed of majority voting error and interrater agreement. Randomness of subset in random *k*-labelset is reduced by considering selective criterion, and subset selection is formulated as a minimum set covering problem [23]. In addition to performance improvements, problems of uncovered label and imbalanced label coverage are greatly alleviated. The selective ensemble mechanism on classifier chain (ECC) [14] is employed to minimize coverage loss and solves the problem with an efficient stochastic optimization.

The quality of feature is another vital factor that greatly determines the performance of classifier, especially for the high-dimensionality of multi-label data. Currently, there are two different solutions in algorithm implementation. The first solution behaves much similar as the idea algorithm adaptation. A pioneer work [24] is the extension from mutual information to multivariate mutual information, in which the potential useful features are generated incrementally. To cope with hybrid data, neighborhood relation is extended to multi-label scenario [25]. By introducing the concept complementary decision reduct, a rough-set based approach [26] is presented by emphasizing inclusion relation of equivalent class to positive label. Fuzzy rough set *FRS-LIFT* [10] is demonstrated to be applicable in multi-label classification after lifting the features from original space. An alternative solution is multi-objective optimization based. Criterion combining label dependency, feature redundancy and label-feature relevance is studied in a single-label view [27]. Mutual information is further extended to measure both the dependency with feature-label and conditional redundancy within feature subset [28]. Feature selection is explained as finding a feature subset which maximizes the correlation within label space [29]. Local refinement on chromosome-level (feature subset) is performed iteratively so that fitness value (defined by feature-label mutual information) can be gradually optimized. A novel score evaluation [30] on feature subset is presented, which is supposed to alleviate the imbalanced effects on assessing redundancy and relative. More recently, feature selection is combined with a granulation strategy on label side [31] and both relevance and redundancy are considered.

Although aforementioned solutions seem to be effective, an indisputable fact is that the uncertainty in multi-label classification is much more sophisticated [32]. For example, in Pascal Visual Object Classes (VOC2007) [33], a person is difficult to be recognized when he is getting off a bus in a foggy day. Obviously, either learning or predicting such labels is a non-trivial task. Label distribution learning [34,35] can reflect the relative importance of label, but the roughness of concept boundary is not involved. Such kind of uncertainty can be realized via Rough Set [36,37]. With the aid of approximation operators, a concept can be approximated by two crisp sets. The semantics of lower approximation set refers to instance assembles which are totally affiliated to the concept, whereas the upper approximation set indicates the instances which are partially associated to the concept. Rough set, as well as its variants, has been studied for multi-label classification [26,38–40], where both approximations and reductions are extended. However, rough set is still not adequate for this issue. On one hand, studying the concept approximation only cannot reflect the semantic differences of different labels comprehensively; on the other hand, merely addressing label-specific reduction is easily affected by the distribution of example set. Therefore, it seems more rationale to balance the label-specific and high-order. Unfortunately, it is beyond the scope of rough set theory.

Three-way Decisions (3WD) [41,42] is an emerging methodology which simulates the way human behave with uncertainty. Under the trisecting-and-acting framework, a number of interesting conclusions are subsequently reported. Hu [43] introduced two novel three-way decisions from the axiomatized three-way decision spaces. Qian [44] demonstrated that performing attribute reduction sequentially can accelerate the problem solving. Li [45] extended the three-way decisions for set-based analysis. Sun [46] claimed that decision-theoretic rough fuzzy set is capable of dealing with uncertain linguistic description. Currently, the semantics of 3WD is not confined to rough set, learning mechanisms which stratify uncertainty are also included. For example, predictions of labels are initially determined by a collection of LP classifiers with smaller subsets, and is corrected by majority voting [16]. Similar idea is also held in [47], except that the motivation is to replenish outcome deduced from classifier chain. It should be noticed that these results may be non-monotonic, i.e. instances of arbitrary label may be changed from relevant to irrelevant, and vice versa. In [48], uncertainty in multi-label classification is interpreted as label co-occurrence, conditional relevance and label-condition redundancy respectively. Three two-stage algorithms named as Two-stage Voting System (*TSVM*), Two-stage Classifier Chain Method (*TSCCM*) and Two-stage Pruned Classifier Chain Method (*TSPCCM*) are presented. Imbalanced distribution of labels is exclusively addressed in [49], in which

a two-stage multi-label hypernetwork is constructed. A recent work [50] identifies the Chinese text sentiment analysis by evaluating orientations from sentiment lexicon to sentence topic, in which the deferred sentences are further assessed by combinations of affective characteristics of words. Our primary motivation is to find a label-specific low-dimensional representation to reduce the label vagueness, and refine the label classification by comprehensively utilizing the voting results from a collection of simple classifiers. We assume that concealed label correlation can be optimized by ensemble learning, and label-specific ensemble can be synthesized for classification under the framework of three-way decisions. Based on this assumption, this paper aims at solving multi-label classification problem with a three-way selective ensemble (*TSEN*) model. Our contributions are as follows:

- We provide a new tri-partition solution for multi-label classification. By selectively integrate the label-specific reduct, we extract the positive and negative regions from low-level boundary region. Empirical experiments demonstrate that our model is superior than a collection of state-of-the-art methods. It is worth noting that decision-making process is non-parametric.
- A novel view for construction of base classifier is proposed. By concentrating on label-specific reduction with shared attributes, relevant features are selected. The process for the generation of base classifier is also non-parametric.
- Probabilistic rough set is initially considered as both the tie-breaking classifier and reduction generator. A novel three-way attribute reduction algorithm is designed to ensure that reducts are both compact and relevant. Proposed attribute reduction algorithm can be regarded as a supplementary of reduct theory.

The rest of paper is organized as follows. Section 2 outlines preliminary knowledge of proposed method. Section 3 elaborates the details of presented model. Experiments and analyses are described in Section 4. Finally, we conclude our proposal in Section 5.

2. Preliminary

The basic notions and concepts for three-way decisions [41,51] defined in single-label learning paradigm. Corresponding definitions can be extended to multi-label learning paradigm, which are reviewed in next section.

Definition 1. [37] An information system is defined by a quadruple tuple: $IS = (U, A, V, f)$ where U is a finite non-empty set of data objects called universe. $A = C \bigcup D$ is a finite non-empty set of attributes, where C is a set of condition attribute, D is a set of decision attribute. V is a non-empty set of values of $a \in A$, and f is an information function from U to V, denoted as $f : U \times A \to V$.

The hidden structure of information, or information granular, represents the similarity/dissimilarity relations among instances. Equivalence relation is regarded as the fundamental criterion to discern objects.

Definition 2. [37] Given a subset of attributes $B \subseteq A$ in IS, the $IND(B)$ denotes an equivalence relation, which can be defined as follows:

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

It can be easily shown that attributes B partition U into a number of non-overlapped sets. The affiliation of objects to class can be determined by adopting maximum inclusion degree of information granular among all classes, and the decisions should suffice the requirement of thresholds meanwhile. In real practice, applicable rules can be extracted given the thresholds located in the interval [0,1]. The conditional probability can be regarded as a kind of inclusion degree.

Definition 3. [37] Given a subset $D_j \subseteq U$, the conditional probability of an object belonging to D_j given that the object belongs to [x]. This probability may be simply estimated as follows:

$$\Pr\left([x]_{D_j} \left| [x]_C \right.\right) = \frac{\left|[x]_{D_j} \cap [x]_C\right|}{\left|[x]_C\right|}, \tag{2}$$

where $|\bullet|$ denotes the cardinality of a set.

The result of conditional probability divides the whole universe into three regions named as positive region (*POS*), boundary region (*BND*) and negative region (*NEG*) respectively. Details of three regions are described as follows.

Definition 4. [52] Given a pair of thresholds α and β with $0 \leq \beta < \alpha \leq 1$, the positive, boundary and negative regions are defined as follows:

$$
\begin{aligned}
&POS_{(\alpha,\bullet)}([x]_{D_j}) = \{x \in U \| \Pr([x]_{D_j}|[x]_C) \geq \alpha\};\\
&BND_{(\alpha,\beta)}([x]_{D_j}) = \{x \in U | \beta < \Pr([x]_{D_j}|[x]_C) < \alpha\};\\
&NEG_{(\bullet,\beta)}([x]_{D_j}) = \{x \in U | \Pr([x]_{D_j}|[x]_C) \leq \beta\},
\end{aligned} \tag{3}
$$

where $\underline{apr}_C(D_j) = POS_{(\alpha,\bullet)}([x]_{D_j})$; $\overline{apr}_C(D_j) = BND_{(\alpha,\beta)}([x]_{D_j}) \bigcup \underline{apr}_C(D_j)$.

Definition 5. [52] According to the three probabilistic regions, one can make three-way decisions of acceptance, deferment and rejection, respectively.

$$
\begin{aligned}
&DES_{Accept}([x]_C \to [x]_{D_j}), \quad \text{for} [x]_C \subseteq POS_{(\alpha,\bullet)}([x]_{D_j});\\
&DES_{Defer}([x]_C \to [x]_{D_j}), \quad \text{for} [x]_C \subseteq BND_{(\alpha,\beta)}([x]_{D_j});\\
&DES_{Reject}([x]_C \to [x]_{D_j}), \quad \text{for} [x]_C \subseteq NEG_{(\bullet,\beta)}([x]_{D_j}).
\end{aligned} \tag{4}
$$

Uncertainty in rough set can be measured from a number of numerical features. Approximation accuracy and roughness are frequently considered to characterize the concept uncertainty.

Definition 6. [37] Given an information system $IS = (U, A, V, f), A = C \bigcup D$, $B \subseteq C$, accuracy of D_i with respect to R is defined as follows:

$$
\alpha_B^{(\alpha,\beta)}(D_i) = \frac{\left|\underline{apr}_B(D_i)\right|}{\left|\overline{apr}_B(D_i)\right|}. \tag{5}
$$

The roughness of D_i with respect to R is defined as follows:

$$
\rho_B^{(\alpha,\beta)}(D_i) = 1 - \alpha_B(D_i). \tag{6}
$$

However, accuracy and roughness do not distinguish partitions with different granularity, making the measuring result flawed. Granularity follows the monotonic decreasing property as partitions are refined, and one famous definition of granularity is as follows:

Definition 7. [37] Given an information system $IS = (U, A, V, f), A = C \bigcup D, B \subseteq C$, and $U/B = \{X_1, X_2, \cdots, X_k\}$ knowledge granulation of R is defined as follows:

$$
GK(B) = \frac{1}{|U|^2} \sum_{i=1}^{k} |X_i|^2. \tag{7}
$$

Consequently, $\alpha_B^{(\alpha,\beta)}(D_i)$, $\rho_B^{(\alpha,\beta)}(D_i)$ is redefined as follows: $Accuracy_B(D_i) = 1 - \rho_B^{(\alpha,\beta)}(D_i)GK(B)$, $Roughness_B(D_i) = \rho_B^{(\alpha,\beta)}(D_i)GK(B)$. The modified version of accuracy approximation can be used for constructing reduction principle. Reduction is a crux issue in Rough Set theory which aims at finding a subset of attributes with certain predefined property preserved. The selected attributes are jointly sufficient and individually necessary. Taking $\alpha_B^{(\alpha,\beta)}(U/D)$ for example, reduction principle can be written as follows:

(1) $\alpha_B^{(\alpha,\beta)}(U/D) = \alpha_C^{(\alpha,\beta)}(U/D)$

(2) $\alpha_{B-\{b\}}^{(\alpha,\beta)}(U/D) \neq \alpha_B^{(\alpha,\beta)}(U/D)$, for $\forall b \in B$

Although the reduct of an information system may be multiple, the core set is indispensable. Given $\alpha^{(\alpha,\beta)}(U/D)$, the core of an information system are the intersections of all related reducts ($RED_{\alpha^{(\alpha,\beta)}(U/D)}(C)$), denoted as: $Core_{\alpha^{(\alpha,\beta)}(U/D)}(C) = \bigcap RED_{\alpha^{(\alpha,\beta)}(U/D)}(C)$.

3. The TSEN model

One crucial issue is to find a suitable set of feature representations for each label. The optimized feature representations should not only reflect differences of label semantics, but should also consider differences of instances. There are two questions to be clarified:

(1) How to find label-specific features which can tolerant certain degree of uncertainty?

(2) How to utilize label correlations automatically to approximate semantics of label?

Fortunately, the probabilistic rough set and selective ensemble can answer the questions satisfactorily. As illustrated in Fig. 1, the problem is resolved in two stages. In the first stage, we employ probabilistic rough set for label-specific

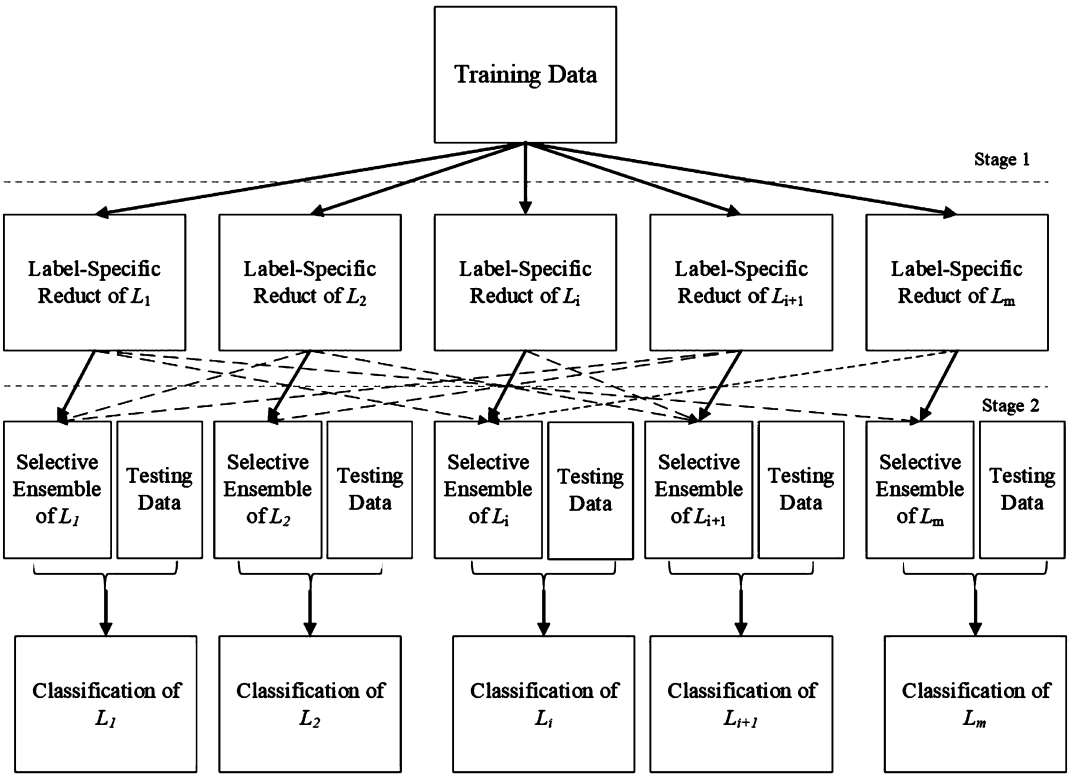

Fig. 1. Framework of *TSEN*.

Table 1
A summary of frequently used notations.

Notation	Meaning
L_i	The i-th label
x_i	The feature vector of the i-th instance
R_i	The reduct regarding label L_i
$x_j^{R_i}$	The projection of instance x_j on attributes of reduct regarding label L_i
P_i^j	The assemble of instances which shows positive on label L_i and is similar to instance x_j
N_i^j	The assemble of instances which shows negative on label L_i and is similar to instance x_j
$f_i(\cdot)$	The integrated binary function which predicts the information of an instance belonging to label L_i
$g_i(\cdot)$	The primary binary function which predicts the information of an instance belonging to label L_i determined by R_i
(α^+, β^+)	Pair-wise parameter for positive class of an arbitrary label
(α^-, β^-)	Pair-wise parameter for negative class of an arbitrary label
$L_i^{h(j)}$	The j-th relevant label of L_i
$C_{(i,j)}^{h(k)}$	The projections of instance x_i on the k-th intersected reduct of label j
$\left(g_p^j, R_i\right)$	The j-th granular structure with the positive labelling of L_i determined by attribute included in R_i
$\left(g_n^j, R_i\right)$	The j-th granular structure with the negative labelling of L_i determined by attribute included in R_i

learning. The fault tolerance can be reflected from parameter (α, β), but some necessary revisions are required, which will be discussed in section 3.1. Reducts regarding each label are learned independently. In the second stage, we consider a selective ensemble strategy based on a collection of label-specific reducts. Different from rough ensemble proposed in [53], we resort to reducts that are correlated within labels, and the size of ensemble is also label-specific. To distinguish the reduct of label itself and the non-trivial intersected reducts, solid arrow and dotted arrow are marked respectively. As illustrated in Fig. 1, reduct information w.r.t. different labels is not fully connected. For example, reducts of L_i is not considered for ensemble learning regarding label L_1, but is associated with ensemble learning regarding label L_{i+1}. The result w.r.t. an arbitrary label L_i is determined by the choice of label-specific selective ensemble and testing data, and the details will be elaborated later. Table 1 summarizes the frequently used notations.

3.1. Stage 1: label-specific reduct with probabilistic rough set

Compared to single-label learning, the complexity of information system embodies on the multiple labels and multiple mapping, and is thus extended as follows:

Definition 8. An multi-label information system (MLIS) is defined by a quadruple tuple: $IS = (U, A, V, f)$ where U is a finite non-empty set of data objects called universe. $A = C \bigcup L$ is a finite non-empty set of attributes, where $C = \{c_1, c_2, \cdots, c_n\}$ is a set of condition attribute, $L = \{L_1, L_2, \cdots, L_m\}$ is a set of labels. V is a non-empty set of values of $a \in A$, and f is an information function from U to V, denoted as $f : U \times A \to V$.

Attribute reduction conducted in multi-label is an adaptation of single-label. Generally speaking, there are three categories of attribute reduction algorithms [51], i.e., deletion-based, addition–deletion based and addition-based. The core idea of addition–deletion based approach is to search a subset which preserves a property of information system measured by attribute importance γ initially, and then shrink the sets until the cardinal reaches minimum. The critical components in reduct computing are two folds: a) definition of preserved goal; b) selection of γ. In our model, the preserved property is defined as the probability approximation accuracy so that basic prediction concerning label is available. The addition–deletion reduct algorithm cost a complexity of $max(O(|U||C|), O(|U/(C-R)||C-R|^2), O(|U/R||R|^2))$, where U denotes the universe, C refers to the condition attributes and R signifies the reduct.

As suggested by [54], the attribute importance can be computed as follows:

$$\gamma_B^{(\alpha,\beta)}(U/D, m(\cdot)) = EG_m(\pi_1) - (1 - \alpha_B^{(\alpha,\beta)}(U/D))EG_m(U/B), \tag{8}$$

where $\alpha_B^{(\alpha,\beta)}(U/D) = \frac{\sum_{Y_i \in U/D} |\underline{apr}_B(\underline{apr}_C^{(\alpha,\beta)}(Y_i))|}{\sum_{Y_i \in U/D} |\overline{apr}_B(\overline{apr}_C^{(\alpha,\beta)}(Y_i))|}$ and $\pi_1 = \{U\}$ denotes the coarsest partition. Symbol $m(\cdot)$ represents a measure of granularity of subsets regarding U.

However, the imbalanced distribution of labels are not well studied in Eq. (8), which is quite ubiquitous in multi-label learning. The number of label combinations grows exponentially as the increment of labels, resulting in the limited number of positive relevant instances in an individual label. As positive label attracts more attentions, we make the following adjustments of (α, β):

$$(\alpha, \beta) \to \{(\alpha^+, \beta^+), (\alpha^-, \beta^-)\}, \tag{9}$$

where (α^+, β^+) and (α^-, β^-) are considered as tri-partition for positive concept and negative concept respectively. Intuitively, the performance of positive class can be enhanced by setting a more strict acceptance for negative class.

By stipulating particular (α^+, β^+) as well as (α^-, β^-), negative labelling of label L_i is penalized. The proportion of positive class in generated reduct is lifted, compared to that of negative class.

In retrospect of addition–deletion method, the syntax of deletion should be more profound. Traditionally, deletion is equivalent to remove attribute that is proved to be redundant one-by-one. However, the complementary view should also be considered. Inspired by the work in [55], an alternative solution to implement the idea of deletion is investigated. To explain the idea clearly, we define strategy-independent label-dependent core and strategy-independent label-independent core.

Definition 9. Given a $MLIS = (U, A = C \bigcup L, V, f)$ and a reduct principle $\theta_C^{(\alpha,\beta)}(U/L_i)$, an arbitrary attribute b $(b \in B \bigcap B \subseteq C)$ is called strategy-independent label-dependent core of an information system if the following two conditions are satisfied:

(1) $\theta_B^{(\alpha,\beta)}(U/L_i) = \theta_C^{(\alpha,\beta)}(U/L_i)$;

(2) $\theta_{B-\{b\}}^{(\alpha,\beta)}(U/L_i) \neq \theta_C^{(\alpha,\beta)}(U/L_i)$.

Definition 10. Given a $MLIS = (U, A = C \bigcup L, V, f)$ and a reduct principle $\theta_C^{(\alpha,\beta)}(U/L_i)$, an arbitrary attribute b $(b \in B \cap B \subseteq C)$ is called strategy-dependent label-dependent core of an information system if the following two conditions are satisfied:

(1) $\theta_B^{(\alpha,\beta)}(U/L_i) = \theta_C^{(\alpha,\beta)}(U/L_i)$;

(2) b is the last element confirmed by addition method of reduction.

The strategy-independent label-dependent in multi-label learning is an extension of core attribute, whereas the strategy-dependent label-dependent is not intensively considered. It is evident that strategy-independent label-dependent is indispensable in representing positive/negative class regarding label L_i since it is independent of attribute searching strategy, However, the strategy-dependent label-dependent is associated with the process of attribute addition. We will demonstrate the necessity of strategy-dependent label-dependent by the following lemma.

Algorithm 1 The revised addition–deletion method based probabilistic reduct of an information system.

Input: A multi-label information system $MLIS = (U, C \bigcup L, V, f)$, threshold pair $\{(\alpha^+, \beta^+), (\alpha^-, \beta^-)\}$, preserving property $\theta = POS$

Output: A collection of (α, β) Reducts of IS, denoted as R, a label-specific classifier g, $g(L) = \{g_1(\cdot), g_2(\cdot), \cdots, g_m(\cdot)\}$

Step 1: Compute core attributes, denoted as $Core_\theta(U/L_i)$

Step 1.1: Initialize $Core_\theta(U/L_i) = \emptyset$

Step 1.2: Calculate $POS_c(U/L_i), \forall c \in C$

Step 1.3: $Core_\theta(U/L_i) = Core_\theta(U/L_i) \bigcup \{c\}$ iff $\theta^{(\alpha,\beta)}_{C-\{c\}}(U/L_i) \neq \theta^{(\alpha,\beta)}_{C}(U/L_i)$

Step 1.4: Loop step 1.2 and 1.3 until all attributes are visited.

Step 2: Add attributes one-by-one to generate a super-reduct R_i

Step 2.1: Initialize $R_i = Core_\theta(U/L_i), \quad CA_i = C - R_i$

Step 2.2: Select attribute with maximal discernibility regarding U/L_i, set $R_i = R_i \bigcup \underset{c \in C - R_i}{\arg\max}\ \gamma^{\{(\alpha^+,\beta^+),(\alpha^-,\beta^-)\}}_{R_i \bigcup \{c\}}(U/L_i)$ and update the candidate set CA_i as $CA_i = CA_i - \{c\}$

Step 2.3: Loop 2.2 until $\theta^{(\alpha,\beta)}_{R_i}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$, store the last appended attribute b

Step 3: Select relative-core attribute to generate final reduct R_i

Step 3.1: Initialize $CB_i = CT_i = Core_\theta(U/L_i) \bigcup \{a\}, CS_i = R_i - \{a\}$

Step 3.2: Select attribute b from CS_i, let $CT_i = CT_i \bigcup \underset{b \in R_i - CT_i}{\arg\max}\ \gamma^{\{(\alpha^+,\beta^+),(\alpha^-,\beta^-)\}}_{R_i \bigcup \{b\}}(U/L_i)$,

Step 3.3: Loop step 3.2 until $\theta^{(\alpha,\beta)}_{CT_i}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$, store the last appended attribute b,

Step 3.4 $CB_i = CB_i \cup \{b\}, CS_i = CT_i, CT_i = CB_i$

Step 3.5: Loop from step 3.2 to 3.4 until $\theta^{(\alpha,\beta)}_{CB_i}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$ and construct a classifier $g_i(\cdot)$ which maps from C to L_i

Step 3.6: $R_i = CB_i$

Step 4: Loop from Step 1 to Step 4 until all L_i in L are examined

Lemma 1. *Given a $MLIS = (U, A, V, f)$ and a reduct R which preserve the property defined by $\theta^{(\alpha,\beta)}_{C}$, $b \in R$ holds if b is a strategy-dependent label-dependent core of MLIS.*

Proof. According to Definition 10, attribute b ($b \in B$) and $\theta^{(\alpha,\beta)}_{B}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$ holds. The process of addition process in reduction can be referred to Algorithm 1, which signifies that the last selected attribute is indispensable in reduct. The reasons are two folds: firstly, the iteration will not be terminated until such b is selected, i.e. $\theta^{(\alpha,\beta)}_{B}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$, and secondly, property preserving does not hold in $B - \{b\}$, which implies $\theta^{(\alpha,\beta)}_{B-\{b\}}(U/L_i) \neq \theta^{(\alpha,\beta)}_{C}(U/L_i)$. □

The multiple reduct can be regarded as different combinations of cores and marginal [55]. Obviously, the relative-core is an element of marginal set. The uncertainty for the components of reduct is reduced yet possibly not removed after selecting a strategy-dependent label-dependent core. A more refined reduct composing of both strategy-dependent label-independent cores and strategy-dependent label-dependent cores is generated. It suggests a promising direction for computing a reduct, as shown in following theorem.

Theorem 1. *Given a $MLIS = (U, A, V, f)$ and a reduct R which is derived from addition process of Algorithm 1, the final reduct RED can be calculated by iteratively select strategy-dependent label-dependent from R until marginal set reaches empty.*

Proof. It is apparent that core is indispensable in constituting a reduct. Thus, the theorem can be demonstrated if we can prove that attribute sets derived from Theorem 1 are the assemble of cores, regardless of strategy-independent label-dependent or strategy-dependent label-dependent.

Firstly, we will prove that marginal set is strictly monotonously decreasing in calculating reduct. The cardinal of marginal is at least one less than the original marginal set, as an attribute b is selected as strategy-dependent label-dependent and transfer to the core set, that is $CORES = CORES \bigcup \{b\}$. It may be more condensed since the combination of original cores and newly obtained core may gain a more powerful representation, i.e. $MARGINAL = MARGINAL - NS$, where NS denotes the redundant attributes. The redundant attributes, NS are moved to the non-useful set, i.e. $NON\_USEFUL = NON\_USEFUL \bigcup NS$. The renewed $CORES$ may still be flawed in preserving property, as there may still be multiple choices in combinations of cores and marginal. The non-useful set, however, can be safely discarded in determining reducts. For similar operations in the coming iteration, it is not exceptional that marginal set becomes smaller on condition that both cores and non-useful set keeps increasing.

Secondly, we will prove that the operations are finite. As suggested above, the selection of strategy-dependent label-dependent continues until marginal set is empty. It is naturally that within finite rounds (at most $|MARGINAL|$), we can identify the eligibility of attribute as strategy-dependent label-dependent core.

Finally, we will prove that derived set is minimal as reduct. Obviously, the cardinal cannot be smaller since all attributes in derived set serve as core, and cores cannot be removed from reduct. As for the representation power, it remains unchanged since in every iteration step it is preserved (i.e. $\theta^{(\alpha,\beta)}_{CORES}(U/L_i) = \theta^{(\alpha,\beta)}_{C}(U/L_i)$). □

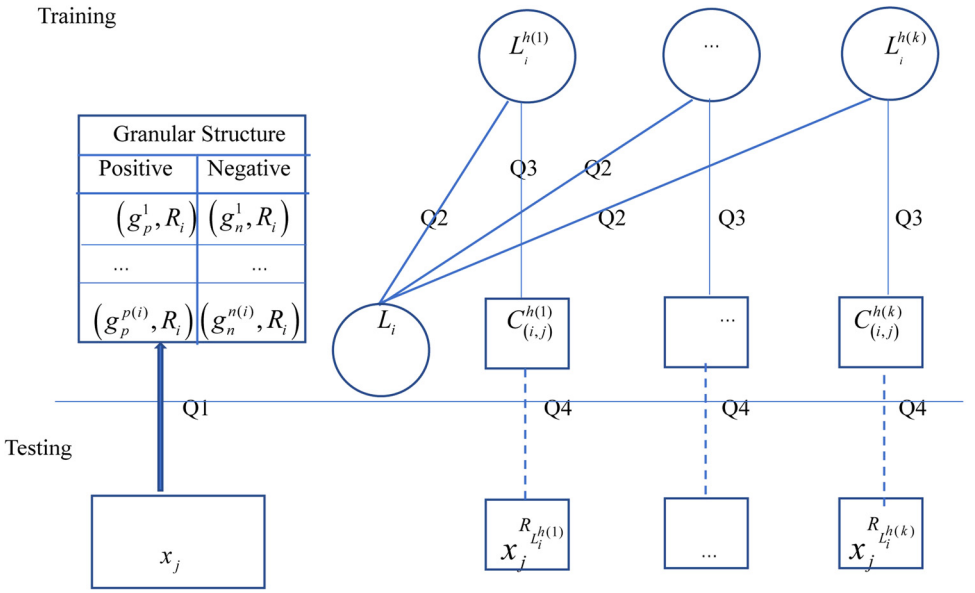

Fig. 2. Details of selective ensemble for label L_i.

Given positive region preserving ($\theta = POS$), the label-specific learning of multi-label on the basis of addition–deletion reduct is explained in Algorithm 1.

3.2. Stage 2: label-based selective ensemble with reducts

Although label-based knowledge is generated, label of instance may be replenished by correlations which may support or oppose the label-specific decision. In other words, there should be some potential relations for the probability of information granular co-occurrence. However, it is time-consuming if we assume each label is associated with the rest. Thus, one rationale hypothesis is that label correlation is limited to a particular set of conditions in a style of ensemble pair-wise, which is formally defined as follows:

Definition 11. Given a $MLIS = (U, A = C \bigcup L, V, f)$ with $L = \{L_1, L_2, \cdots, L_m\}$, set $PWRL = \{PWRL_1, PWRL_2, \cdots, PWRL_m\}$ is defined as pair-wise related label assembles. It represents the projections of C_i on related instances regarding $L_i, i = 1, 2, \cdots, m$, with $PWRL_i \subseteq C$ be the correlated label set of L_i and include the information concerning l, where $l \subseteq \{L - L_i\}$.

The generation of $PWRL$ is label-specific, and techniques which are employed in first-stage should be well integrated. Different from single-label learning which an attribute is either included or excluded for a reduct, the same attribute may occur in more than one reduct regarding label. Accordingly, we simulate the local correlation from those intersected granulars. Let c_k belong to the reduct of L_i and L_j meanwhile, we believe there are some special connections of reduct representing L_i and L_j. In other words, reduct of L_i is potentially beneficial to alleviate uncertainty of L_j and vice versa. How to calculate $PWRL_i$ will be discussed later.

The uncertainty of multi-label classification is reduced yet still remained after generating basic predictions. In other words, although preliminary prediction is better than random guess, it is still possible for a seemingly positive to be negative, and vice versa. For simplicity, we will discuss the prediction of an arbitrary label L_i and the remaining are analogous. For the prediction of L_i, there are four questions to be answered, as shown in Fig. 2.

Q1: How to measure the similarity of an unseen instance x_j with positive granular structure $(g_p^{p(i)}, R_i)$ or negative granular structure $(g_n^{n(i)}, R_i)$ which is determined by stage 1?

Q2: Which group of labels $L_i^{h(k)}$ are relevant to the considered label L_i?

Q3: What are the associated attributes (denoted as $C_{(i,j)}^{h(k)}$) regarding label L_i^j, which are the j-th associated attributes of L_i?

Q4: What are the constraints between projection of unseen instance x_j on reducts regarding L_i (denoted as $x_j^{R_{L_i^{h(k)}}}$) and associated instances of instances on pertinent label L_i^j (denoted as $C_{(i,j)}^{h(k)}$)?

For the first question (Q1), we refer to the Jaccard similarity which are frequently considered in measuring similarity between sets. Concretely, let $x_j^{R_i}, R_i^k$ be the attributes included in reduct of label L_i of j-th instance and k-th of granular structure, similarity can be computed as:

$$Sim(R_i^j, R_i^k) = \frac{|x_j^{R_i} \cap R_i^k|}{|x_j^{R_i} \cup R_i^k|}. \tag{10}$$

In multi-label classification, similarity defined in equation can be further customized by introducing polarity of granular structure. The basic aim is to generate two mutually disjoint sets which represent most relevant positive (P_i^j) and most relevant negative (N_i^j) respectively in consideration of imbalanced class distribution.

Definition 12. Given a $MLIS = (U, A, V, f), A = C \cup L, L = \{L_1, L_2, \cdots, L_i, \cdots, L_m\}$, $R_i^k \to \{P, N\}$ represents the possibly positive result (abbreviated as P) of the k-th reduct regarding label L_i. For an unseen instance x_j, the positive related sets of instances P_i^j are denoted as:

$$P_i^j = \bigcup_{R_i^k} \arg\max_{R_i^k} \left[\!\left[R_i^k \to P \right]\!\right] \left[\!\left[Sim(x_j^{R_i}, R_i^k) > 0.5 \right]\!\right]. \tag{11}$$

Definition 13. Given a $MLIS = (U, A, V, f), A = C \cup L, L = \{L_1, L_2, \cdots, L_i, \cdots, L_m\}$, $R_i^k \to \{P, N\}$ represents the possibly negative result (abbreviated as N) of the k-th reduct regarding label L_i. For an unseen instance x_j, the negative related sets of instances N_i^j are denoted as:

$$N_i^j = \bigcup_{R_i^k} \arg\max_{R_i^k} \left[\!\left[R_i^k \to N \right]\!\right] \left[\!\left[Sim(x_j^{R_i}, R_i^k) > 0.5 \right]\!\right]. \tag{12}$$

For the second question (Q2), we explain the detailed semantic of pair-wise related labels as follows:

$$PWRL_i = \bigcup \{L_j | R_i \bigcap R_j \neq \emptyset\}, \forall j \neq i. \tag{13}$$

We have the following properties regarding $PWRL$.

Property 1.

1. $|PWRL| = |L|$;
2. $0 \leq |PWRL_i| \leq |L| - 1, \forall i$
3. $L_k \in PWRL_j$ *if* $L_k \in PWRL_i \wedge PWRL_i = \bigcup \{L_j | R_i \bigcap R_j \neq \emptyset\}$.

Proof. The cardinal of $PWRL$ should be $|L|$ because of the arbitrariness of label L_i.

The cardinal of $PWRL_i$, however, achieve its minimal 0 if label-dependent reduct has no intersections with the remaining. Since itself is excluded, the maximal of $PWRL_i$ should be $|L| - 1$.

The third property is straightforward since the set intersection operation is exchangeable. □

It can be observed that no additional parameters are required, and thus is exempted from parameter optimization.

For the third question (Q3), we solve it by referring to the projections of instances (acquired in Q1) on reducts regarding selected attributes (granular structure obtained in Q2) in the style of *label-by-label*. It is worth noting that $C_{(i,j)}^{\cdot} = \{PS_{(i,j)}^{\cdot}, NS_{(i,j)}^{\cdot}\}$, where $PS_{(i,j)}^{\cdot}$ corresponds to the assemble of projections whose reduct (P_i^j) is pertinent to positive class for j-th relevant label regarding L_i, and $NS_{(i,j)}^{\cdot}$ corresponds to the assemble of projections whose reduct (N_i^j) is pertinent to negative class for j-th relevant label regarding L_i.

For the last question (Q4), the primary goal is to utilize the polarity information of reduct to improve the classification result regarding L_i. To elaborate our idea more clearly, we firstly present a definition.

Definition 14. Given a $MLIS = (U, A, V, f), A = C \bigcup L, \forall L_i \in L$, positive label correlation degree lcp_i^j is defined as the proportion of the count of projection of x_j on R_i are totally affiliated to positive related sets of instances P_i^j in cardinal of related labels of L_i.

$$lcp_i^j = \sum_k \frac{\left[\!\left[x_j^{R_{L_i h(k)}} \subseteq C_{(i,j)}^{h(k)} \right]\!\right]}{|PWRL_i|}, \quad \forall L_k \in PWRL_i \to g(x_j^{R_i}) = 1. \tag{14}$$

Dually, we have the definition of negative label correlation degree lcn_i^j as follows:

Definition 15. Given a $MLIS = (U, A, V, f)$, $A = C \bigcup L$, $\forall l_i \in L$, negative label correlation degree lcn_i^j is defined as the proportion of the count of projection of x_j on R_i are totally affiliated to negative related sets of instances N_i^j in cardinal of related labels of L_i.

$$lcn_i^j = \sum_k \frac{\left[\!\left[x_j^{R_{L_i h(k)}} \subseteq C_{(i,j)}^{h(k)} \right]\!\right]}{|PWRL_i|}, \quad \forall L_k \in PWRL_i \rightarrow g(x_j^{R_i}) = 0. \tag{15}$$

Some properties on lcp_i^j and lcn_i^j are discussed as follows:

Property 2.

1. *non-negativity:* $lcp_i^j \geq 0, lcn_i^j \geq 0$ *if* $|PWRL_i| \neq 0$;
2. *boundness:* $lcp_i^j \leq 1, lcn_i^j \leq 1$ *if* $|PWRL_i| \neq 0$;

Proof. We will prove the non-negativity and boundness of lcp_i^j first. Let L_k be an arbitrary label whose label-dependent reduct R_k are intersected with R_i, the expression $\left[\!\left[x_j^{R_{L_i h(k)}} \subseteq C_{(i,j)}^{h(k)} \right]\!\right] \times \left[\!\left[g(x_j^{R_i}) = 1 \right]\!\right] \in \{0, 1\}$ is obviously held. According to Eq. (13), $L_k \in PWRL_i$ holds, thus the local result on L_k is either $\frac{0}{1}$ or $\frac{1}{1}$. The non-negativity of lcp_i is straightforward, since it is the sum of all labels like L_k, whose local result is always non-negative. As the maximal result reaches 1 if all $\left[\!\left[x_j^{R_{L_i h(k)}} \subseteq C_{(i,j)}^{h(k)} \right]\!\right] \times \left[\!\left[g(x_j^{R_i}) = 1 \right]\!\right]$, we get the property of boundness concerning lcp_i. The proof of lcn_i^j is similar to that of lcp_i^j. □

Based on Definition 14 and Definition 15, the association of an instance x_j w.r.t L_i is defined as:

$$f_i(x_j) = \begin{cases} 1 & lcp_i^j > lcn_i^j; \\ 0 & lcp_i^j < lcn_i^j; \\ g_i(x_j) & otherwise, \end{cases} \tag{16}$$

where $g_i(x_j)$ represents the corresponding result of R_i derived in stage 1. We concatenate results on each label as the final prediction for test set, i.e. $f = \{f_1(\cdot), f_2(\cdot), \cdots, f_m(\cdot)\}$.

Remark 1. Although Eq. (16) is trilateral in form, the results obtained cannot be guaranteed to be consistent when the positive label correlation degree lcp_i and negative label correlation degree lcn_i cannot be distinguished or the two degrees are unavailable. This is because the function $g_i(x_j)$ has three possible outcomes. The fusion of stage 1 and stage 2 can be regarded as optimistic for stage 2 and pessimistic for stage 1.

To summarize the work in stage 2, we present Algorithm 2. It is worth noting that execution of Q2 has no relations to the rest yet the label relevance can be shared for construction of classifiers in selective ensemble. Consequently, steps illustrated in Fig. 2 are performed in sequence of Q2, Q1, Q3, Q4.

3.3. Algorithm complexity

The computational complexity of proposed model is reflected in Algorithm 1 and Algorithm 2. The complexity of first two stages of Algorithm 1 are increased to $O(|U||C||L|)$ and $O(|U/(C-R)||C-R|^2|L|)$ respectively. These steps can be accelerated by introducing parallel computing paradigm, since the sequence in label selection makes no difference. For the third step of Algorithm 1, the complexity is $O(|U/R||R|^2|L|)$ since only marginal attributes are to be selected. Although comparable computing expenditure is required for an individual label L_i, the speed in deleting attributes can be faster if there are a wealth of redundancy in marginal attributes. This lies on the fact that $|U/Core| \leq |U/R|$.

Analysis for Algorithm 2 is as follows. Let $|U|$ and $|U'|$ be the instances count for training and testing respectively, we will discuss the time required for classification of a single instance x_j on an arbitrary label L_i. In step 1, the effort to search relevant labels requires $O(|L|)$. Similarity measurement in step 2 requires $O(|U'||U/R_i||R_i|)$. Step 3 is the construction for base classifier for selective ensemble, and it occupies $O(1)$. For step 4, it requires $O(k|CR_k|)$, where k represents the classifier count and $|CR_k|$ represents the average number of attributes which are considered. In most cases, $k \ll |L|, |CR_k| \ll |C|$. Consequently, the worst case of computational complexity for Algorithm 2 is $O(|L|^2) + O(|U'||U/R||R||L|) + O(|L||C||U'||L|)$, which signifies that selective ensemble is non-trivial for all labels.

Algorithm 2 The selective ensemble of reduct for multi-label classification.

Input: A multi-label information system which are partitioned into training set $MLIS_{TR}$ and testing set $MLIS_{TE}$
Reduct R_i of L_i, $i = 1, 2, \cdots, m$, label-specific function $g_i(\cdot)$
Output: $\{x_j^{L_i}\}, \forall x_j \in MLIS_{TE}, i = 1, 2, \cdots, m$

Step 1: Find relevant labels $\forall L_i$ to construct base classifier for selective ensemble
  Step 1.1: Calculate $PWRL_i$ according to Eq. (13).
  Step 1.2: Loop Step 1.1 until information of $PWRL_i$ are available $\forall L_i$.
Step 2: Compute similarity of x_j with R_i
  Step 2.1: Find positive relevant granular P_i^j for L_i according to Eq. (11).
  Step 2.2: Find negative relevant granular N_i^j for L_i according to Eq. (12).
  Step 2.3: Loop Step 2.1 and Step 2.2 until all L_i are visited.
Step 3: Find projections of related granular on relevant labels for selective ensemble
  Step 3.1: Find $C_{(i,j)}^{h(k)}$, $\forall L_i$;
Step 4: Selective ensemble for classification of x_j
  Step 4.1: Compute lcp_i^j based on Eq. (14);
  Step 4.2: Compute lcn_i^j based on Eq. (15);
  Step 4.3: Determining L_i classification of x_j according to Eq. (16).
  Step 4.4: Loop Step 4.1 to Step 4.3 until all L_i of x_j is determined.
Step 5: Loop from Step 2 to Step 4 until all $x_j \in MLIS_{TE}$ are visited.

Table 2
Description of data sets.

Data set	# Instances	# Features	# Labels	# Cardinality	Domain
Genbase	662	1185	27	1.252	Biology
Medical	978	1449	45	1.245	Text
Enron	1702	1001	53	3.39	Text
Slashdot	3782	1079	22	1.18	Text
LangLog	1460	1004	75	1.18	Text
Bibtex	7395	1836	159	2.402	Text

Although we cannot accurately point out which step dominates the complexity, we can anticipate that Algorithm 1 requires more calculations than Algorithm 2. This is due to the fact that normally the label count is smaller than attribute count.

4. Results and discussions

In this section, we conduct extensive experiments to verify the performance of *TSEN*. Altogether three groups of comparisons are considered. The first experiment seeks to explore the sensitivity of α^+ on *TSEN* given $\alpha^- = 1$, $\beta^+ = \beta^- = 0$. The second experiment manages to testify the superiority performance over a collection of state-of-art methods from ten measures defined in section 3.2. The third experiment, however, makes a further investigation by introducing Friedman test [56] to examine whether statistical discrepancy exists. All experiments are implemented in Matlab 2017b and performed on a computer with Intel (R) Core(TM) i7-8550U CPU with memory equipped by 8 GB RAM. For each considered algorithms, we repeatedly run each comparing methods five times on six sets of randomly partitioned training (80 percent) and testing (20 percent) data.

4.1. Datasets

We conduct experiments on six multi-label benchmark data sets, and the details of which are summarized in Table 2. To circumvent information loss, attributes type of benchmark are all nominal. The term "cardinality" is abbreviated for label cardinality representing the average labels count regarding instances. They can be downloaded from the websites of Mulan[1] [57] and Meka[2] [58].

Genbase data set is composed of 662 proteins, and each protein chain is represented by a 1185 motif sequence. There are 27 protein function families, including oxydoreductases, isomerases and transferases.

Medical data set contains 978 clinical free text from Cincinnati Children's Hospital Medical Center Department of Radiology. For each radiology text report, 45 candidates labels are inspected independently by coding staffs and coding companies and observe the corpus recommended by ICD-9-CM.

Enron data set is prepared by the CALO Project (cognitive assistant that learns and organizers) containing 1702 emails of senior management of Enron. Altogether 53 labels, provided by UC Berkeley Enron Email Analysis Project, are categorized into four groups, including coarse genre, included/forwarded information, primary topics and emotional tone.

[1] http://mulan.sourceforge.net/datasets.html.
[2] http://meka.sourceforge.net/.

Table 3
Characteristics of comparing algorithms.

Algorithm	Order of correlations			Categories of algorithms	
	First-order	Second-order	High-order	Problem transformation	Algorithm adaptation
ML-KNN	√				√
ML-LOC			√		√
LIFT	√			√	
FRS-SS-LIFT	√			√	
fRAkEL			√		√
CDR			√		√
LLSF		√		√	
LPLC		√			√
MIMLK			√		√
TSEN		√		√	

Slashdot data set consists of 3782 titles and partial blurbs published on website of Slashdot.org, and each instance is associated with at most 22 labels. Possible tags include entertainment, hardware and politics.

LangLog data set is a collection of blogs launched by Mark Liberman and Geoffrey Pullum. The accumulated 1460 articles are associated with at most 75 labels. Possible tags include ethics, pragmatics and humor.

Bibtex data set contains metadata for bibtex items such as author, organization, journal volume. For 7395 records, the dimension of candidate tag space is 159.

4.2. Comparing methods

We compare our proposed method *TSEN* with six state-of-the-art multi-label classification algorithms. Parameter setting for each comparing methods are presented as follows:

ML-kNN[3] [3] Multi-Label k-Nearest Neighbor. It learns a classifier which takes an action of maximum a posteriori principle on the neighbor of training instances. As recommended in [3], the number of neighbors is set to 10 and euclidean metric is employed to generate neighborhood.

ML-LOC[4] [14] Multi-Label LOcal Correlation. It exploits local label correlations by assuming that instances with similar features are with similar codes represented in LOcall Correlation (LOC for short) code space. Default settings of *ML-LOC* are utilized, such as m=15, $\lambda_1 = 1, \lambda_2 = 100$. The maximal iteration is set as 1000.

LIFT[5] [8] Label specIfic FeaTures. It constructs label-specific classifier by rewriting instances as distance to the clustered centroid. Setting for the number of clusters retained for positive and negative classes r is set to be 0.1.

FRS-SS-LIFT [10] Fuzzy Rough Set Sample Selection LIFT. It derives a label-specific reduct assemble on the basis of LIFT and accelerated by adopting sample selection strategy. Setting for the number of clusters retained for positive and negative classes r is set to be 0.2.

fRAkEL[6] [59] fast RAkEL. It speeds up algorithm *RAkEL* by shrinking the samples with irrelevant labels on subset. The size of feature subset is fixed as 3, whereas the number of base classifier is twice the number of label cardinality.

CDR [26] Complementary Decision Reduct. It emphasizes the preservation toward vague concept positive class distribution of label with more compacted representation. No additional parameters are required.

LLSF[7] [11] Learning Label-Specific Features. It constructs a collection of label-specific classifier which consider both the sparsity and sharing of attributes in representing label information. Parameter α, β are tuned in $\{2^{-10}, 2^{-9}, \cdots, 2^{10}\}$.

LPLC[8] [13] Local Pairwise Label Correlation. It analyzes both positive and negative pairwise label correlation locally by weighing result from most positive relevant and more negative relevant. Parameter k is searched in $\{3, 5, \cdots, 21\}$, and α is tuned in $\{0.6, 0.7, 0.8, 0.9, 1.0\}$.

MIMLK [31] Mutual Information Multi-Label with K-nearest neighborhood. It aims at finding compact representations of label granular which are with maximal correlation and minimal redundancy simultaneously. The feature number is empirically determined as 10, whereas k is fixed as 10.

The characteristics for comparing algorithms are summarized in Table 3.

4.3. Evaluation metrics

The evaluation measures for multi-label classification is roughly classified into two taxonomies, i.e. *example-based* metrics and *label-based* metrics. The first category of metrics evaluate prediction performance on each example separately, and

[3] Source code: http://lamda.nju.edu.cn/code_MLkNN.ashx.
[4] Source code: http://lamda.nju.edu.cn/code_MLLOC.ashx.
[5] Source code: http://cse.seu.edu.cn/PersonalPage/zhangml/files/LIFT.rar/.
[6] Source code: http://github.com/KKimura360/fast_RAkEL_matlab.
[7] Source code: http://www.escience.cn/people/huangjun/index.html.
[8] Source code: http://www.escience.cn/people/huangjun/index.html.

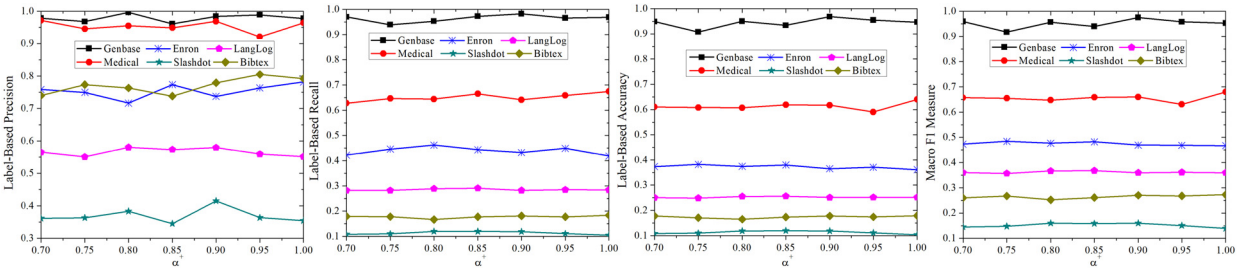

Fig. 3. Sensitivity analysis of α^+ regarding label-based measures.

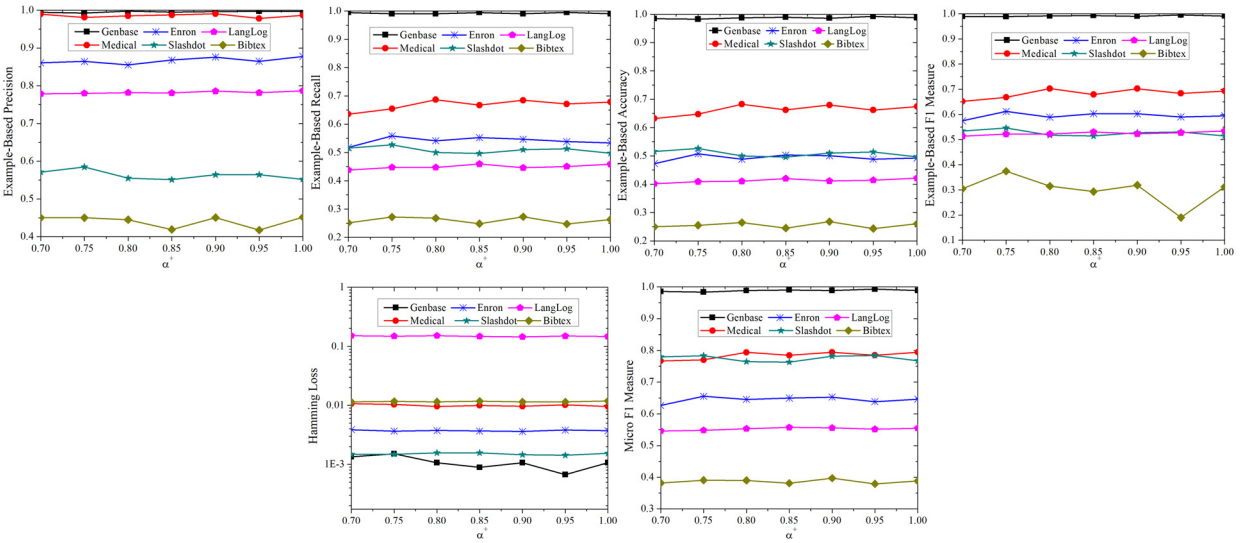

Fig. 4. Sensitivity analysis of α^+ regarding example-based measures.

returns the mean value across the test set. The second category of metrics evaluate prediction performance on each label separately, and returns the mean value across all labels. As *TSEN* focuses on discriminating the relevance of label w.r.t. instance, we take the view of information retrieval and consider the following metrics: *Label-based Precision* [60], *Label-based Recall* [60], *Label-based Accuracy* [60], *Macro F1* [60], *Example-based Precision* [61], *Example-based Recall* [61], *Example-based F1* [61], *Hamming-Loss* [62], *Micro F1* [62].

We notice that some other evaluation metrics, such as one-error, coverage, ranking loss and average precision [63], are also frequently referred. It suggests that multi-label classification can also be realized by determining pair-wise relative relevance degree. However, the premise is that a real-valued function $f(\cdot,\cdot)$ must be defined. *TSEN*, instead, determines the relevant labels in a qualitative solution (see Eq. (16)). Therefore, they are not applicable in our approach.

4.4. Experimental evaluation

Since positive class usually occupies a small proportion of instances, we permit some mistakes for the approximation of positive concept, whereas the negative concept should be approximated more accurately meanwhile. To examine the sensitivity of α^+, we tune α^+ from 0.7 to 1.0 with a step of 0.05 and conduct five-fold cross validation on six datasets referred in section 4.1. Fig. 3 and Fig. 4 illustrate the experimental results of *TSEN* in terms of label-based and example-based criterion respectively. The horizontal axis of each sub-figure indicates different selections regarding α^+, whereas the vertical axis represents the performance of evaluation criteria over six benchmarks.

It can be shown from Fig. 3 that different selections of α^+ have limited influence on label-based evaluations. The fluctuations are not so dramatic, especially for *Label-based Accuracy*. The absolute performance for considered dataset, however, exhibits significant differences and the relative ranking with regard to data sets are not incongruous. Concretely, the performance of *Enron* and *Bibtex* on *Label-based Precision* are comparative, whereas the performance of *Enron* regarding *Label-based Accuracy* dominates the other. Additionally, performance can be enhanced when the α^+ varies, which may imply that introducing uncertainty on positive class can be beneficial for robustness of classifier.

It can be observed from Fig. 4 that fluctuations of example-based evaluation on selected datasets are generally negligible, especially for the cases in *Genbase* and *LangLog*. However, results such as *Example-based F1* are not so stable for dataset *Bibtex*. Similar to performance displayed in Fig. 3, rankings of datasets on example-based evaluations are inconsis-

tent. For example, *Slashdot* outperforms *Enron* in terms of *Hamming Loss*, whereas results are degenerated with respect to *Example-based Recall*.

Combining Fig. 3 and Fig. 4, it can be concluded that proposed *TSEN* is rather steady for a wide range of α^+. To validate the effectiveness of *TSEN* more comprehensively, results of average results as well as standard derivations against comparing algorithms are reported subsequently from Tables 4 to 13. For each evaluation, best results are highlighted in bold, and evaluations except for Hamming Loss are more desirable if the value is bigger, which is indicated by symbol (↑) and otherwise (↓). Relative performance across algorithms are also considered and recorded in the style of ranking. For ten methods on each data set, their ranks are marked by 1–10 in the brackets. The average rank, as well as the overall performance ranking are listed in the last two rows.

Based on aforementioned results, we claim that the superiority of *TSEN* over other algorithms are both data-relevant and metric-relevant. For dataset *Enron*, *TSEN* is more likely to acquire a dominant position. The possibility of ranking first is as high as 8/10. As for dataset *Slashdot*, the overall performance is much more degraded (2/10 possibility of ranking first). For metrics such as *Label-based Precision* and *Hamming Loss*, *TSEN* achieves an overwhelming position since the probability of ranking first are 6/6 and 4/6 respectively. However, the result is a bit frustrated when it comes to *Macro F1*, *Example-based Accuracy*, and *Example-based F1*. The underlying reason is that a mechanism of weighing relative importance of relevant label is lacked. It is also worthy mentioning that performance on precision is much encouraging than recall, which reflects that *TSEN* is more sensitive for misclassification on negative classes. It makes sense since the possibility for inclusion is more likely to be effective on negative classes. For all 60 predictive performance results (6 data sets × 10 evaluation metrics), *TSEN* ranks in first place among the ten comparing algorithms at 36.7% cases, in second place at 28.3% cases, in third place at 10% cases, and only 8.33% cases locate in the second half. Thus, *TSEN* is generally more powerful than all selected methods.

Friedman test is employed to conduct performance analysis among the comparing algorithms. It is recommended in examining whether there is statistically superiority among multiple algorithms over a collection of data sets. Given k comparing algorithms and N data sets, Let $Rank_j^2 = \frac{1}{N}\sum_i Rank_i^j$ be the average rank for the j-th algorithm, and $Rank_i^j$ be the rank of the j-th algorithm on the i-th data set. The Friedman statistic F_F is distributed according to the F-distribution with (k-1) numerator degrees of freedom and (k-1)(N-1) degrees of freedom as denominator, denoted as:

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2} \tag{17}$$

where $\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j Rank_j^2 - \frac{k(k+1)^2}{4}\right]$.

Table 14 provides the Friedman statistics F_F and the corresponding critical value in terms of each evaluation metric. As shown in Table 14, at significance level $\alpha = 0.05$, the null hypothesis that all the comparing algorithms perform equivalently is clearly rejected in terms of most evaluation measures. Consequently, we can proceed with a post-hoc test to analyze the relative performance among the comparing algorithms except for Example-Based Recall. The Nemnyi test is employed to test whether our proposed method *TSEN* achieves a competitive performance against the comparing algorithm. The performance between two classifiers will be significantly different if the corresponding average ranks differ at least the critical difference *CD*.

$$CD = q_\alpha\sqrt{\frac{k(k+1)}{6N}} \tag{18}$$

For Nemenyi test $q_\alpha = 3.164$ at significance level $\alpha = 0.05$, and thus $CD = 5.531 (k = 10, N = 6)$.

It can be inferred from Fig. 5 that *TSEN* receives statistically superior performance against *MIMLK* in terms of *Label-based Precision*, *Label-based Accuracy*, *Macro F1*, *Example-based Precision* and *Micro F1*. For metric *Example-based Accuracy*, *Hamming Loss* and *Micro F1*, performance of *TSEN* gains a statistically superiority than *fRAkEL* in terms of *Hamming Loss*, and better than *CDR* in terms of *Label-based Precision*. Thus, *TSEN* is a very competitive algorithm for multi-label classification.

4.5. Discussions

The presented *TSEN* achieves an encouraging performance on selected multi-label benchmarks. This section intends to discuss more characteristics of *TSEN*. The first issue is that the robustness performance for different selections of α^+ does not necessary imply that the extension $\{(\alpha^+, \beta^+), (\alpha^-, \beta^-)\}$ is meaningless. One possible reason is that the ensemble based decision is qualitative rather than quantitative. In other words, *TSEN* cannot distinguish the relative relevancy of two labels if both of them receive more than half effective votes. Here effectiveness refers to the votes that satisfy $\left[\!\left[x_j^{R_{L_i^{h(k)}}} \subseteq C_{(i,j)}^{h(k)} \right]\!\right] = 1$. The second issue concerns about the computation complexity. Due to the exponential increasing combinations and a greedy-based searching policy, it is widely recognized as a drawback of rough set based approach. It may be suffered for large dataset and is expected to be accelerated. Despite this bottleneck, one shall not ignore that feature selection can remove irrelevant features and generates a low-dimensional representation.

Table 4
Experimental results of each comparing algorithm (mean ± std) in terms of *Label-based Precision*.

Data set	Label-based Precision (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.5671 ± 0.0151(8)	0.7277 ± 0.0318(5)	0.6593 ± 0.0423(6)	0.7280 ± 0.0446(4)	0.7639 ± 0.0188(2)	0.5617 ± 0.0464(9)	0.7407 ± 0.7037(3)	0.6426 ± 0.0232(7)	0.5270 ± 0.0607(10)	**0.9780 ± 0.0250**(1)
Medical	0.2551 ± 0.0245(6)	0.2025 ± 0.0190(9)	0.2516 ± 0.0336(7)	0.3003 ± 0.0363(4)	0.3234 ± 0.0489(3)	0.2129 ± 0.0423(8)	0.3944 ± 0.3559(2)	0.2660 ± 0.0196(5)	0.1910 ± 0.0429(10)	**0.9209 ± 0.0320**(1)
Enron	0.1501 ± 0.0309(7)	0.2194 ± 0.0182(4)	0.1231 ± 0.0177(9)	0.2638 ± 0.0262(2)	0.1865 ± 0.0348(5)	0.1384 ± 0.0266(8)	0.2260 ± 0.0224(3)	0.1569 ± 0.0143(6)	0.0703 ± 0.0391(10)	**0.7372 ± 0.0315**(1)
Slashdot	0.2682 ± 0.0113(2)	0.0628 ± 0.0226(7)	0.0659 ± 0.0117(8)	0.1305 ± 0.0388(5)	0.1621 ± 0.0262(3)	0.0428 ± 0.0184(10)	0.1233 ± 0.1614(6)	0.1344 ± 0.0332(4)	0.0499 ± 0.0119(9)	**0.3635 ± 0.0861**(1)
LangLog	0.1400 ± 0.0095(5)	0.1223 ± 0.0028(6)	0.0294 ± 0.0050(10)	0.1842 ± 0.0227(4)	0.0523 ± 0.0156(9)	0.2390 ± 0.0221(3)	0.0610 ± 0.0782(8)	0.4266 ± 0.0079(2)	0.0835 ± 0.0046(7)	**0.5794 ± 0.0305**(1)
Bibtex	0.1683 ± 0.0129(6)	0.1268 ± 0.0995(10)	0.1648 ± 0.0123(7)	0.4592 ± 0.0127(2)	0.3712 ± 0.0847(4)	0.1462 ± 0.0203(9)	0.4091 ± 0.0107(3)	0.3319 ± 0.0114(5)	0.1463 ± 0.0100(8)	**0.7793 ± 0.0075**(1)
Avg. rank	5.667(6)	6.833(7)	7.833(8.5)	3.5(2)	4.333(4)	7.833(8.5)	4.167(3)	4.833(5)	9(10)	1(1)
Total order: TSEN ≻ FRS-SS-LIFT ≻ LLSF ≻ fRAkEL ≻ LPLC ≻ ML-KNN ≻ ML-LOC ≻ LIFT/CDR ≻ MIMLK										

Table 5
Experimental results of each comparing algorithm (mean ± std) in terms of *Label-based Recall*.

Data set	Label-based Recall (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.5286 ± 0.0334(10)	0.6935 ± 0.0328(6)	0.6956 ± 0.0475(5)	0.7401 ± 0.0046(4)	0.7745 ± 0.0197(2)	0.5605 ± 0.0710(9)	0.7407 ± 0.7037(3)	0.6825 ± 0.0143(7)	0.5852 ± 0.0511(8)	**0.9687 ± 0.0317**(1)
Medical	0.1733 ± 0.0120(9)	0.1147 ± 0.0190(10)	0.3183 ± 0.0294(5)	0.3472 ± 0.0579(4)	0.4448 ± 0.0470(2)	0.1889 ± 0.0163(8)	0.4033 ± 0.3370(3)	0.2213 ± 0.0135(7)	0.2360 ± 0.0658(6)	**0.6585 ± 0.0627**(1)
Enron	0.0749 ± 0.0071(10)	0.0885 ± 0.0109(9)	0.2366 ± 0.0531(4)	**0.5206 ± 0.0247**(1)	0.2573 ± 0.0933(3)	0.0903 ± 0.0097(8)	0.2145 ± 0.1266(5)	0.1004 ± 0.0057(7)	0.1137 ± 0.0277(6)	0.4323 ± 0.0410(2)
Slashdot	0.2016 ± 0.0092(2)	0.0436 ± 0.0010(9)	0.1450 ± 0.0291(4)	**0.2293 ± 0.0686**(1)	0.1695 ± 0.0300(3)	0.0427 ± 0.0014(10)	0.1156 ± 0.0730(5)	0.0561 ± 0.0068(8)	0.1015 ± 0.0755(7)	0.1110 ± 0.0252(6)
LangLog	0.0760 ± 0.0063(8)	0.1455 ± 0.0059(6)	0.0937 ± 0.0104(7)	0.2931 ± 0.0341(2)	0.1672 ± 0.0880(5)	0.2069 ± 0.0126(4)	0.0598 ± 0.0232(9)	**0.4714 ± 0.0061**(1)	0.0216 ± 0.0151(10)	0.2821 ± 0.0075(3)
Bibtex	0.0495 ± 0.0018(9)	0.0087 ± 0.0326(10)	**0.4585 ± 0.0276**(1)	0.3178 ± 0.0080(4)	0.3853 ± 0.0964(3)	0.0137 ± 0.0019(8)	**0.3992 ± 0.0092**(1)	0.1405 ± 0.0023(7)	0.1625 ± 0.0192(6)	0.1813 ± 0.0184(5)
Avg. rank	8(9)	8.333(10)	4.333(3.5)	5.333(5)	3(1.5)	7.833(8)	4.333(3.5)	6.167(6)	7.167(7)	3(1.5)
Total order: fRAkEL/TSEN ≻ LIFT/LLSF ≻ FRS-SS-LIFT ≻ LPLC ≻ MIMLK ≻ CDR ≻ ML-KNN ≻ ML-LOC										

Table 6
Experimental results of each comparing algorithm (mean ± std) in terms of *Label-based Accuracy*.

Data set	Label-based Accuracy (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.5255 ± 0.0297(9)	0.6913 ± 0.0322(5)	0.6586 ± 0.0410(6)	0.7274 ± 0.0453(4)	0.7614 ± 0.0200(2)	0.5314 ± 0.0699(8)	0.7407 ± 0.9624(3)	0.6362 ± 0.0257(7)	0.5242 ± 0.0573(10)	**0.9467 ± 0.0211**(1)
Medical	0.1518 ± 0.0126(9)	0.1106 ± 0.0189(10)	0.2307 ± 0.0266(5)	0.2819 ± 0.0382(4)	0.2964 ± 0.0340(3)	0.1637 ± 0.0155(8)	0.3422 ± 0.3079(2)	0.1883 ± 0.0150(6)	0.1700 ± 0.0524(7)	**0.5903 ± 0.0479**(1)
Enron	0.0610 ± 0.0061(9)	0.0746 ± 0.0111(6)	0.1041 ± 0.0143(5)	0.1256 ± 0.0240(3)	0.1233 ± 0.0096(4)	0.0696 ± 0.0095(8)	0.1416 ± 0.0985(2)	0.0734 ± 0.0046(7)	0.0537 ± 0.0143(10)	**0.3649 ± 0.0506**(1)
Slashdot	**0.1785 ± 0.0134**(1)	0.0389 ± 0.0011(9)	0.0589 ± 0.0107(6)	0.1245 ± 0.0394(2)	0.1134 ± 0.0097(3)	0.0363 ± 0.0016(10)	0.0856 ± 0.0675(5)	0.0501 ± 0.0066(7)	0.0433 ± 0.0113(8)	0.1109 ± 0.0252(4)
LangLog	0.0612 ± 0.0046(6)	0.1055 ± 0.0031(5)	0.0264 ± 0.0048(9)	0.1832 ± 0.0228(3)	0.0368 ± 0.0080(8)	0.1639 ± 0.0109(4)	0.0387 ± 0.0196(7)	**0.2981 ± 0.0024**(1)	0.0045 ± 0.0038(10)	0.2509 ± 0.0072(2)
Bibtex	0.0458 ± 0.0020(9)	0.0086 ± 0.0296(7)	0.1506 ± 0.0108(5)	0.1545 ± 0.0060(4)	0.2256 ± 0.0186(2)	0.0130 ± 0.0018(10)	**0.2495 ± 0.0047**(1)	0.1155 ± 0.0018(6)	0.0666 ± 0.0077(8)	0.1778 ± 0.0056(3)
Avg. rank	7.167(8)	7(7)	6(6)	3.333(2)	3.667(3.5)	8(9)	3.667(3.5)	5.667(5)	8.833(10)	2(1)
Total order: TSEN ≻ FRS-SS-LIFT ≻ fRAkEL/LLSF ≻ LPLC ≻ LIFT ≻ ML-LOC ≻ ML-KNN ≻ CDR ≻ MIMLK										

Table 7
Experimental results of each comparing algorithm (mean ± std) in terms of *Macro F1*.

Data set	Macro F1 Measure (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.5424 ± 0.0254(10)	0.7052 ± 0.0313(5)	0.6722 ± 0.0399(6)	**0.9857 ± 0.0454**(1)	0.7676 ± 0.0192(3)	0.5530 ± 0.0634(8)	0.7407 ± 0.7037(4)	0.6554 ± 0.0198(7)	0.5477 ± 0.0538(9)	0.9529 ± 0.0215(2)
Medical	0.1937 ± 0.0146(8)	0.1394 ± 0.0203(10)	0.2705 ± 0.0268(5)	**0.7378 ± 0.0396**(1)	0.3530 ± 0.0272(4)	0.1925 ± 0.0197(9)	0.3914 ± 0.3425(3)	0.2320 ± 0.0148(6)	0.2011 ± 0.0503(7)	0.6308 ± 0.0493(2)
Enron	0.0876 ± 0.0089(9)	0.1060 ± 0.0101(6)	0.1465 ± 0.0206(5)	0.4138 ± 0.0058(2)	0.1854 ± 0.0130(4)	0.0973 ± 0.0118(8)	0.2038 ± 0.1362(3)	0.1052 ± 0.0055(7)	0.0764 ± 0.0182(10)	**0.4694 ± 0.0564**(1)
Slashdot	0.2196 ± 0.0089(2)	0.0431 ± 0.0019(9)	0.0770 ± 0.0162(6)	**0.5850 ± 0.0429**(1)	0.1499 ± 0.0098(4)	0.0411 ± 0.0024(10)	0.1138 ± 0.0840(5)	0.0639 ± 0.0088(7)	0.0536 ± 0.0175(8)	0.1507 ± 0.0356(3)
LangLog	0.0885 ± 0.0068(6)	0.1270 ± 0.0040(5)	0.0426 ± 0.0072(9)	0.1406 ± 0.0264(4)	0.0618 ± 0.0113(7)	0.2188 ± 0.0132(3)	0.0568 ± 0.0331(8)	**0.4146 ± 0.0048**(1)	0.0078 ± 0.0066(10)	0.3593 ± 0.0103(2)
Bibtex	0.0664 ± 0.0033(8)	0.0157 ± 0.0407(10)	0.2183 ± 0.0156(5)	0.2698 ± 0.0071(4)	0.3289 ± 0.0245(2)	0.0229 ± 0.0029(9)	**0.3589 ± 0.0076**(1)	0.1634 ± 0.0028(6)	0.0910 ± 0.0098(7)	0.2700 ± 0.0083(3)
Avg. rank	7.167(7)	7.5(8)	6(6)	2.167(1.5)	4(3.5)	7.833(9)	4(3.5)	5.667(5)	8.5(10)	2.167 (1.5)
Total order: FRS-SS-LIFT/TSEN ≻ fRAkEL/LLSF ≻ LPLC ≻ LIFT ≻ ML-KNN ≻ ML-LOC ≻ CDR ≻ MIMLK										

Table 8
Experimental results of each comparing algorithm (mean ± std) in terms of *Example-based Precision*.

Data set	Example-based Precision (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.9709 ± 0.0075(9)	0.9911 ± 0.0078(4)	0.9712 ± 0.0120(8)	0.9836 ± 0.0153(6)	0.9938 ± 0.0021(3)	0.9773 ± 0.0351(7)	**0.9989 ± 1.0000**(1)	0.9846 ± 0.0094(5)	0.9458 ± 0.0239(10)	0.9972 ± 0.0025(2)
Medical	0.5857 ± 0.0137(9)	0.4336 ± 0.0717(10)	0.6621 ± 0.0304(7)	0.7378 ± 0.0467(3)	0.6734 ± 0.0667(6)	0.6862 ± 0.0149(4)	0.7500 ± 0.8001(2)	0.6824 ± 0.0180(5)	0.6250 ± 0.0965(8)	**0.9784 ± 0.0066**(1)
Enron	0.5396 ± 0.0255(6)	0.6408 ± 0.0236(4)	0.4730 ± 0.0145(8)	0.6941 ± 0.0058(2)	0.4451 ± 0.0878(9)	0.6261 ± 0.0188(5)	0.5302 ± 0.7031(7)	0.6437 ± 0.0185(3)	0.3500 ± 0.1544(10)	**0.8759 ± 0.0049**(1)
Slashdot	0.6345 ± 0.0259(2)	**0.6390 ± 0.0181**(1)	0.5626 ± 0.0186(9)	0.5649 ± 0.0149(8)	0.5875 ± 0.0358(6)	0.6344 ± 0.0149(3)	0.6089 ± 0.6287(5)	0.6144 ± 0.0120(4)	0.5511 ± 0.1208(10)	0.5646 ± 0.0155(7)
LangLog	0.5580 ± 0.0148(4)	0.6616 ± 0.0136(2)	0.1021 ± 0.0150(8)	0.1289 ± 0.0255(7)	0.0980 ± 0.0269(9)	0.6187 ± 0.0084(3)	0.1309 ± 0.0896(6)	0.5197 ± 0.0208(5)	0.0233 ± 0.0037(10)	**0.7855 ± 0.0065**(1)
Bibtex	0.2505 ± 0.0062(7)	0.0201 ± 0.2266(9)	0.2850 ± 0.0118(6)	0.4441 ± 0.0089(3)	0.4328 ± 0.0757(4)	0.0152 ± 0.0018(10)	**0.4587 ± 0.0048**(1)	0.4027 ± 0.0116(5)	0.1964 ± 0.0241(8)	0.4506 ± 0.0107(2)
Avg. rank	6.167(7.5)	5(5)	7.667(9)	4.833(4)	6.167(7.5)	5.333 (6)	3.667 (2)	4.5(3)	9.333(10)	2.333(1)

Total order TSEN ≻ LLSF ≻ LPLC ≻ FRS-SS-LIFT ≻ ML-LOC ≻ CDR ≻ ML-KNN/fRAkEL ≻ LIFT ≻ CDR

Table 9
Experimental results of each comparing algorithm (mean ± std) in terms of *Example-based Recall*.

Data set	Example-based Recall (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.9461 ± 0.0208(10)	0.9831 ± 0.0046(7)	0.9889 ± 0.0049(4)	0.9905 ± 0.0138(3)	0.9950 ± 0.0028(2)	0.9673 ± 0.0298(9)	0.9874 ± 0.9824(5)	0.9871 ± 0.0039(6)	0.9723 ± 0.0398(8)	**0.9972 ± 0.0064**(1)
Medical	0.5401 ± 0.0182(9)	0.3943 ± 0.0594(10)	0.6917 ± 0.0359(4)	0.7588 ± 0.0388(3)	**0.8714 ± 0.0365**(1)	0.6335 ± 0.0130(8)	0.8384 ± 0.8120(2)	0.6676 ± 0.0321(6)	0.6417 ± 0.0908(7)	0.6714 ± 0.0329(5)
Enron	0.3543 ± 0.0234(10)	0.3955 ± 0.0102(9)	**0.6757 ± 0.0243**(1)	0.5206 ± 0.0093(5)	0.5994 ± 0.1031(2)	0.4560 ± 0.0199(7)	0.5133 ± 0.4989(6)	0.4541 ± 0.0238(8)	0.5559 ± 0.1698(3)	0.5465 ± 0.0207(4)
Slashdot	0.6073 ± 0.0201(4)	0.5762 ± 0.0195(7)	0.6226 ± 0.0186(2)	**0.6280 ± 0.0187**(1)	0.6047 ± 0.0105(5)	0.5726 ± 0.0101(8)	0.5962 ± 0.5684(6)	0.5520 ± 0.0123(9)	0.6124 ± 0.1237(3)	0.5138 ± 0.0131(10)
LangLog	0.3626 ± 0.0184(5)	0.4084 ± 0.0186(4)	0.1186 ± 0.0172(9)	0.1705 ± 0.0344(7)	0.3274 ± 0.1170(6)	0.4155 ± 0.0069(3)	0.1456 ± 0.0701(8)	**0.6131 ± 0.0140**(1)	0.0289 ± 0.0155(10)	0.4458 ± 0.0104(2)
Bibtex	0.1296 ± 0.0043(8)	0.0129 ± 0.1036(9)	0.4234 ± 0.0183(3)	0.3178 ± 0.0079(5)	**0.5066 ± 0.0814**(1)	0.0150 ± 0.0015(10)	0.4970 ± 0.0157(2)	0.2746 ± 0.0069(6)	0.3416 ± 0.0313(4)	0.2729 ± 0.0033(7)
Avg. rank	7.167(8.5)	7.167(8.5)	3.833(2)	4(3)	2.833(1)	7.5(10)	4.833(4.5)	5.667 (6)	5.833(7)	4.833(4.5)

Total order: fRAkEL ≻ LIFT ≻ FRS-SS-LIFT ≻ LLSF/TSEN ≻ LPLC ≻ MIMLK ≻ ML-KNN/ML-LOC ≻ CDR

Table 10
Experimental results of each comparing algorithm (mean ± std) in terms of *Example-based Accuracy*.

Data set	Example-based Accuracy (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.9450 ± 0.0203(9)	0.9822 ± 0.0041(5)	0.9707 ± 0.0120(7)	0.9831 ± 0.0157(4)	**0.9901 ± 0.0024**(1)	0.9622 ± 0.0306(8)	0.9874 ± 0.9824(3)	0.9734 ± 0.0087(6)	0.9442 ± 0.0238(10)	0.9876 ± 0.0051(2)
Medical	0.5253 ± 0.0113(9)	0.3899 ± 0.0602(10)	0.6429 ± 0.0325(5)	0.7162 ± 0.0496(2)	0.6543 ± 0.0594(4)	0.6310 ± 0.0133(6)	**0.7234 ± 0.7619**(1)	0.6199 ± 0.0274(7)	0.5921 ± 0.0905(8)	0.6621 ± 0.0320(3)
Enron	0.3066 ± 0.0201(9)	0.3434 ± 0.0146(8)	0.4196 ± 0.0135(3)	0.4553 ± 0.0058(2)	0.3477 ± 0.0381(7)	0.3881 ± 0.0171(5)	0.3758 ± 0.4406(6)	0.3964 ± 0.0134(4)	0.3013 ± 0.1237(10)	**0.5007 ± 0.0154**(1)
Slashdot	**0.5829 ± 0.0232**(1)	0.5760 ± 0.0193(2)	0.5601 ± 0.0181(5)	0.5639 ± 0.0148(4)	0.5372 ± 0.0296(9)	0.5726 ± 0.0101(3)	0.5573 ± 0.5658(6)	0.5502 ± 0.0117(7)	0.5498 ± 0.1185(8)	0.5138 ± 0.0131(10)
LangLog	0.3145 ± 0.0104(5)	0.3551 ± 0.0180(4)	0.0973 ± 0.0156(8)	0.1277 ± 0.0251(6)	0.0871 ± 0.0196(9)	0.3652 ± 0.0061(3)	0.1072 ± 0.0682(7)	**0.4325 ± 0.0188**(1)	0.0233 ± 0.0037(10)	0.4113 ± 0.0107(2)
Bibtex	0.1257 ± 0.0041(8)	0.0121 ± 0.1022(10)	0.2707 ± 0.0113(4)	0.2771 ± 0.0075(3)	0.3331 ± 0.0391(2)	0.0126 ± 0.0017(9)	**0.3626 ± 0.0054**(1)	0.2531 ± 0.0061(6)	0.1911 ± 0.0232(7)	0.2683 ± 0.0028(5)
Avg. rank	6.833(9)	6.5(8)	5.333(5.5)	3.5(1)	5.333(5.5)	5.667(7)	4(3)	5.167(4)	8.833(10)	3.833(2)

Total order: FRS-SS-LIFT ≻ TSEN ≻ LLSF ≻ LPLC ≻ LIFT/fRAkEL ≻ CDR ≻ ML-LOC ≻ ML-KNN ≻ MIMLK

Table 11
Experimental results of each comparing algorithm (mean ± std) in terms of *Example-based F1*.

Data set	Example-based F1 Measure (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.9536 ± 0.0165(10)	0.9854 ± 0.0042(5)	0.9769 ± 0.0094(7)	0.9857 ± 0.0150(4)	**0.9932 ± 0.0017**(1)	0.9703 ± 0.0318(8)	0.9912 ± 0.9882(2.5)	0.9821 ± 0.0069(6)	0.9539 ± 0.0285(9)	0.9912 ± 0.0043(2.5)
Medical	0.5505 ± 0.0180(9)	0.4057 ± 0.0634(10)	0.6657 ± 0.0328(5)	0.7378 ± 0.0451(2)	0.7239 ± 0.0436(3)	0.6512 ± 0.0136(7)	**0.7702 ± 0.7908**(1)	0.6563 ± 0.0247(6)	0.6199 ± 0.0931(8)	0.6838 ± 0.0324(4)
Enron	0.4026 ± 0.0208(10)	0.4593 ± 0.0151(8)	0.5265 ± 0.0137(3)	0.5507 ± 0.0056(2)	0.4632 ± 0.0367(7)	0.5023 ± 0.0187(5)	0.4917 ± 0.5583(6)	0.5045 ± 0.0166(4)	0.4054 ± 0.1543(9)	**0.6025 ± 0.0154**(1)
Slashdot	**0.6085 ± 0.0223**(1)	0.5966 ± 0.0190(2)	0.5812 ± 0.0183(6)	0.5850 ± 0.0155(5)	0.5758 ± 0.0196(7)	0.5926 ± 0.0117(3)	0.5878 ± 0.5869(4)	0.5716 ± 0.0120(8)	0.5705 ± 0.1210(9)	0.5303 ± 0.0138(10)
LangLog	0.4122 ± 0.0115(5)	0.4732 ± 0.0162(4)	0.1054 ± 0.0156(9)	0.1406 ± 0.0271(6)	0.1280 ± 0.0185(8)	0.4746 ± 0.0070(3)	0.1283 ± 0.0755(7)	**0.5481 ± 0.0170**(1)	0.0250 ± 0.0056(10)	0.5227 ± 0.0100(2)
Bibtex	0.1581 ± 0.0047(8)	0.0144 ± 0.1322(9)	0.3181 ± 0.0133(5)	0.3682 ± 0.0079(3)	0.4158 ± 0.0308(2)	0.0140 ± 0.0016(10)	**0.4390 ± 0.0066**(1)	0.3005 ± 0.0068(6)	0.2313 ± 0.0266(7)	0.3186 ± 0.0043(4)
Avg. rank	7.167(9)	6.333(8)	5.833(6)	3.667(2)	4.667(4)	6(7)	3.583(1)	5.167(5)	8.667(10)	3.917(3)

Total order: LLSF ≻ FRS-SS-LIFT ≻ TSEN ≻ fRAkEL ≻ LPLC ≻ LIFT ≻ CDR ≻ ML-LOC ≻ ML-KNN ≻ MIMLK

Table 12
Experimental results of each comparing algorithm (mean ± std) in terms of *Hamming Loss*.

Data set	Hamming Loss (↓)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.0046 ± 0.0019(10)	0.0014 ± 0.0004(5)	0.0022 ± 0.0010(6)	0.0012 ± 0.0009(4)	**0.0009 ± 0.0003**(1)	0.0033 ± 0.0009(7)	0.0011 ± 0.0167(3)	0.0035 ± 0.0013(8)	0.0045 ± 0.0014(9)	0.0010 ± 0.0004(2)
Medical	0.0166 ± 0.0010(7)	0.0177 ± 0.0015(9)	0.0126 ± 0.0010(3)	**0.0102 ± 0.0019**(1.5)	0.0231 ± 0.0104(10)	0.0133 ± 0.0007(4)	0.0131 ± 0.0098(4)	0.0171 ± 0.0016(8)	0.0144 ± 0.0039(6)	**0.0102 ± 0.0008**(1.5)
Enron	0.0523 ± 0.0010(6)	0.0529 ± 0.0012(7)	0.0462 ± 0.0017(3)	0.0454 ± 0.0055(2)	0.0940 ± 0.0404(10)	0.0511 ± 0.0019(4)	0.0653 ± 0.0468(9)	0.0512 ± 0.0010(5)	0.0539 ± 0.0135(8)	**0.0362 ± 0.0011**(1)
Slashdot	0.0153 ± 0.0007(2)	0.0159 ± 0.0010(3)	0.0166 ± 0.0008(5)	0.0160 ± 0.0007(4)	0.0261 ± 0.0074(10)	0.0178 ± 0.0009(6)	0.0217 ± 0.0159(9)	0.0180 ± 0.0003(7)	0.0188 ± 0.0036(8)	**0.0143 ± 0.0003**(1)
LangLog	0.0537 ± 0.0018(5)	0.1738 ± 0.0058(9)	0.0154 ± 0.0002(3)	**0.0135 ± 0.0009**(1)	0.0770 ± 0.0471(6)	0.1604 ± 0.0007(8)	0.0231 ± 0.0149(4)	0.1867 ± 0.0056(10)	0.0157 ± 0.0010(2)	0.1461 ± 0.0032(7)
Bibtex	0.0136 ± 0.0001(6)	0.0152 ± 0.0002(8)	0.0124 ± 0.0002(3)	0.0121 ± 0.0062(2)	0.0201 ± 0.0090(10)	0.0139 ± 0.0001(7)	0.0153 ± 0.0003(9)	0.0127 ± 0.0002(4)	0.0132 ± 0.0002(5)	**0.0114 ± 0.0001**(1)
Avg. rank	6(4.5)	6.833 (8)	3.833(3)	2.583(2)	7.833(10)	6(4.5)	6.333(6.5)	7(9)	6.333(6.5)	2.25(1)
Total order: TSEN ≻ FRS-SS-LIFT ≻ LIFT ≻ ML-KNN/CDR ≻ LLSF/MIMLC ≻ ML-LOC ≻ LPLC ≻ fRAkEL										

Table 13
Experimental results of each comparing algorithm (mean ± std) in terms of *Micro F1*.

Data set	Micro F1 Measure (↑)									
	ML-KNN	ML-LOC	LIFT	FRS-SS-LIFT	fRAkEL	CDR	LLSF	LPLC	MIMLK	TSEN
Genbase	0.9481 ± 0.0209(10)	0.9847 ± 0.0046(5)	0.9755 ± 0.0098(6)	0.9876 ± 0.0589(3)	**0.9899 ± 0.0032**(1)	0.9617 ± 0.0294(8)	0.9875 ± 0.7037(4)	0.9628 ± 0.0137(7)	0.9486 ± 0.0190(9)	0.9885 ± 0.0041(2)
Medical	0.6369 ± 0.0138(9)	0.5429 ± 0.0531(10)	0.7403 ± 0.0154(4)	**0.7973 ± 0.0856**(1)	0.6865 ± 0.0789(7)	0.7208 ± 0.0187(5)	0.7716 ± 0.8121(3)	0.6753 ± 0.0257(8)	0.7023 ± 0.0870(6)	0.7850 ± 0.0217(2)
Enron	0.4712 ± 0.0152(7)	0.4682 ± 0.0162(8)	0.5533 ± 0.0149(2)	0.5392 ± 0.0100(3)	0.4514 ± 0.0515(10)	0.5221 ± 0.0258(4)	0.4972 ± 0.5714(6)	0.5100 ± 0.0164(5)	0.4611 ± 0.0991(9)	**0.6529 ± 0.0156**(1)
Slashdot	0.6768 ± 0.0163(10)	**0.7846 ± 0.0138**(1)	0.7756 ± 0.0132(4)	0.7820 ± 0.0090(3)	0.7102 ± 0.0512(9)	0.7646 ± 0.0084(5)	0.7380 ± 0.7836(8)	0.7566 ± 0.0055(7)	0.7481 ± 0.0509(7)	0.7842 ± 0.0080(2)
LangLog	0.4631 ± 0.0150(4)	0.4510 ± 0.0092(5)	0.1802 ± 0.0209(8)	0.2625 ± 0.0319(6)	0.1480 ± 0.0298(9)	0.5155 ± 0.0078(3)	0.1812 ± 0.1462(7)	**0.6079 ± 0.0045**(1)	0.0465 ± 0.0153(10)	0.5557 ± 0.0093(2)
Bibtex	0.2094 ± 0.0045(7)	0.0237 ± 0.0029(9)	0.3737 ± 0.0163(4)	0.0127 ± 0.0036(10)	0.4329 ± 0.0517(2)	0.0245 ± 0.0031(8)	**0.4877 ± 0.0061**(1)	0.3616 ± 0.0068(5)	0.2755 ± 0.0217(6)	0.3971 ± 0.0108(3)
Avg. rank	7.833(9.5)	6.333(7.5)	4.667(3)	4.333(2)	6.333(7.5)	5.5(5.5)	4.833(4)	5.5(5.5)	7.833(9.5)	2(1)
Total order: TSEN ≻ FRS-SS-LIFT ≻ LIFT ≻ LLSF ≻ CDR/LPLC ≻ ML-LOC/fRAkEL ≻ ML-KNN/MIMLK										

Table 14
Summary of the Friedman Statistics $F_F(k = 10, N = 6)$ and the critical value in terms of each evaluation measure (k:# comparing algorithms; N: # data sets).

Measure	F_F	Critical value ($\alpha = 0.05$)
Label-based Precision	34.5359	5.531
Label-based Recall	42.9927	
Label-based Accuracy	32.7308	
Macro F1	31.5677	
Example-based Precision	23.1644	
Example-based Recall	5.2718	
Example-based Accuracy	15.1189	
Example-based F1	15.8531	
Hamming Loss	21.7216	
Micro F1	18.8766	

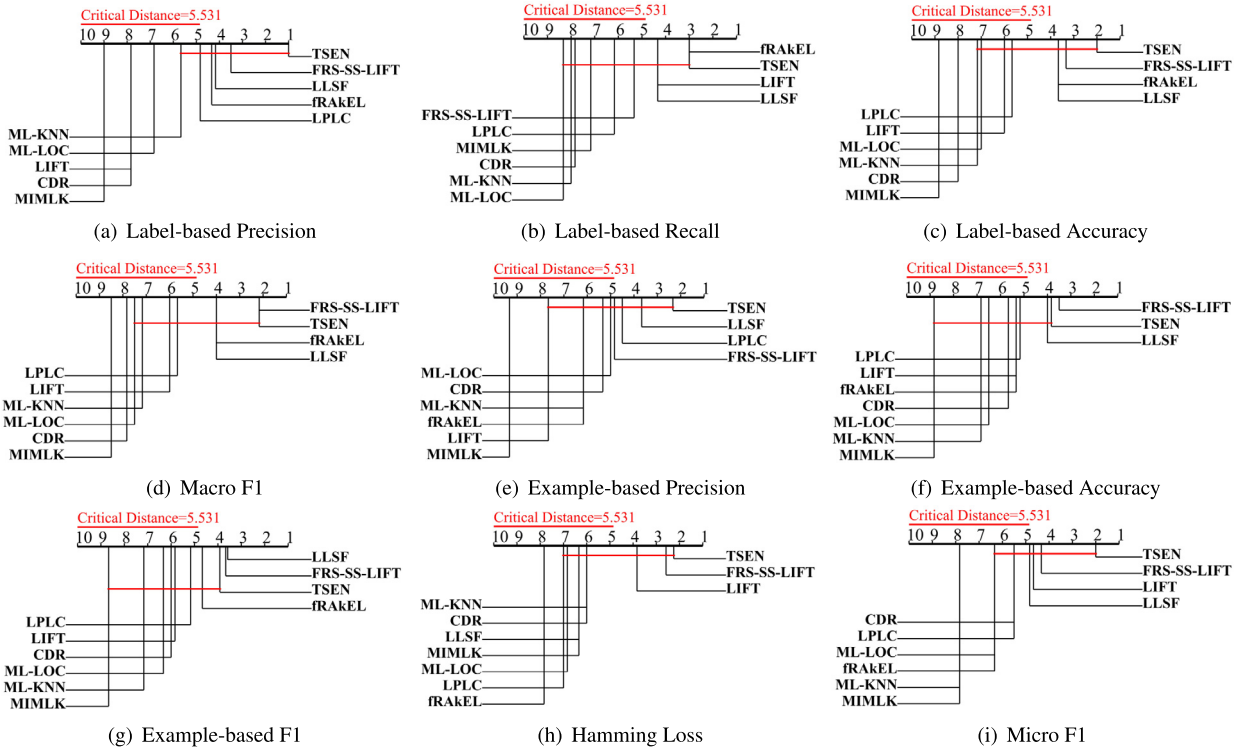

Fig. 5. Comparison of *TSEN* (control algorithm) against other comparing algorithms with the Nemenyi test regarding label-based measures. Groups of classifiers that are not significantly different from TSEN (at $p = 0.05$) are connected.

Attribute reduction is an indispensable procedure in *TSEN*. It is worthy mentioning that unlike *LLSF*, *CDR*, *FRS-SS-LIFT*, features generated in *TSEN* should be treated as a label-specific feature repository. There are three reasons accounting for this argument. Firstly, information from referred label served as constraint rather than ordinary feature set. Secondly, some of the attributes can be considered more than once. Last but not the least, the voting result on referred label can be trivial if neither Eq. (14) nor Eq. (15) holds. Thus, it is not suitable to apply *TSEN* on evaluation of the reduct effects on classifiers for multi-label classification. Nevertheless, the overall advantages against *LPLC* demonstrates that selecting some attributes is avail to multi-label classification.

Herein, we will further elaborate why *TSEN* is more effective than other methods for dealing with multi-label classifications. The most appealing feature of *TSEN* is that it considers the high-order label correlation in an approximate second-order complexity. Unlike *LPLC* or *LLSF* which limits the discussion within second-order, *TSEN* simulates the way that human behaves when there are some related concepts. The decision mechanism for *TSEN* can be described via the following example: suppose we intend to determine whether a picture has the semantics of the sea, the probability of being positive class for sea label is higher if we detect the features describing boat and harbor. On one hand, it works much alike as second-order, since the basic step of ensemble is selectively performed in a pair-wise style. On the other hand, it does not rely on a particular label (which is considered in *LPLC*), and enables the robustness for unknown data.

Although the advantages of *TSEN* over *FRS-SS-LIFT* and *LIFT* is not very impressive, one should admit that *TSEN* makes prediction in its original space. In other words, the benefit from feature mapping is comparable to optimization on attribute distribution. The relative robustness against *MIMLK* and *fRAkEL* suggests that introduction of new parameter does not necessarily improve performance. What is more, a significant superiority over *CDR* suggests that rough set is more preferable for multi-label classification after incorporating with selective ensemble.

To summarize, the proposed model *TSEN* achieves a competitive advantage against a group of multi-label classification methods. Considering that *TSEN* is realized in its original feature space and no nonlinear operations are introduced, the overall performance for *TSEN* against *FRS-SS-LIFT* should be more promising.

5. Conclusion

In this paper, we have employed a novel model for multi-label classification called *TSEN*. The data complexity is well decomposed and label ambiguity is gradually reduced. We have demonstrated that three-way decisions can be more effective if ensemble learning is well integrated. The contributions are two-folds. For the promotion of three-way decisions, we believe that the judgment results with different degrees of uncertainty can be used to refine the existing three-way structure, and the decision-making results in areas with insufficient information should be allowed to have a certain degree of uncertainty. For ensemble learning, we suggest that ensemble strategy can be effective if the intrinsic characterizations of base classifier are considered, given that base classifier itself is capable in representing related concepts. Finally, the performance on publicly available benchmarks demonstrate that proposed model is statistically superior or at least comparable than some state-of-the-art methods.

Much more efforts still remain to be made. Firstly, the data category of multi-label can be numeric, and information loss is inevitable if equivalent relation is directly applied. Secondly, the computational complexity is still flawed, which means it is not applicable if the label space is enormous. For this issue, we intend to integrate with the distributed computing technique. Last but not the least, the assumptions for quality of label side are perfect, and constraints will be softened in future.

Acknowledgements

Authors would like to thank the anonymous reviewers for their constructive comments and valuable suggestions. Special thanks should include Professor Keigo Kimura and Professor Xibei Yang, providing source code for algorithm “fRAkEL” and “FRS-SS-LIFT” respectively. This work is supported by National Key R&D Program of China (Grant No. 213), the National Science Foundation of China (Grant No. 61673301, 61763031, 61563016), and Major Project of Ministry of Public Security (Grant No. 20170004), and the Open Research Funds of State Key Laboratory for Novel Software Technology (Grant No. KFKT2017B22).

References

[1] E. Gibaja, S. Ventura, A tutorial on multilabel learning, ACM Comput. Surv. 47 (3) (2015) 1–38.
[2] M.L. Zhang, Z.H. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (8) (2014) 1819–1837.
[3] M.L. Zhang, Z.H. Zhou, ML-KNN: a lazy learning approach to multi-label learning, Pattern Recognit. 40 (7) (2007) 2038–2048.
[4] Q.Y. Wu, M.K. Tan, H.J. Song, J. Chen, M.K. Ng, ML-FOREST: a multi-label tree ensemble method for multi-label classification, IEEE Trans. Knowl. Data Eng. 28 (10) (2016) 2665–2680.
[5] W.J. Chen, Y.H. Shao, C.N. Li, N.Y. Deng, MLTSVM: a novel twin support vector machine to multi-label learning, Pattern Recognit. 52 (2016) 61–74.
[6] F. Tai, H.T. Lin, Multilabel classification with principal label space transformation, Neural Comput. 24 (9) (2012) 2508–2542.
[7] Y. Zhang, Z.H. Zhou, Multilabel dimensionality reduction via dependence maximization, ACM Trans. Knowl. Discov. Data 4 (3) (2010) 1–21.
[8] M.L. Zhang, L. Wu, LIFT: multi-label learning with label-specific features, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 107–120.
[9] Z.J. Lin, G.G. Ding, M.Q. Hu, J.M. Wang, Multi-label classification via feature-aware implicit label space encoding, in: International Conference on International Conference on Machine Learning, 2014, pp. 325–333.
[10] S.P. Xu, X.B. Yang, H.L. Yu, D.J. Yu, J.Y. Yang, E.C.C. Tsang, Multi-label learning with label-specific feature reduction, Knowl.-Based Syst. 104 (2016) 52–61.
[11] J. Huang, G.R. Li, Q.M. Huang, X.D. Wu, Learning label-specific features and class-dependent labels for multi-label classification, IEEE Trans. Knowl. Data Eng. 28 (12) (2016) 3309–3323.
[12] C. Gentile, F. Orabona, On multilabel classification and ranking with bandit feedback, J. Mach. Learn. Res. 15 (2014) 2451–2487.
[13] J. Huang, G.R. Li, S.H. Wang, Z. Xue, Q.M. Huang, Multi-label classification by exploiting local positive and negative pairwise label correlation, Neurocomputing 257 (2017) 164–174.
[14] S.J. Huang, Z.H. Zhou, Multi-label learning by exploiting label correlations locally, in: Twenty-Sixth AAAI Conference on Artificial Intelligence, 2012, pp. 949–955.
[15] H.Y. Lo, S.D. Lin, H.M. Wang, Generalized k-labelsets ensemble for multi-label and cost-sensitive classification, IEEE Trans. Knowl. Data Eng. 26 (7) (2014) 1679–1691.
[16] G. Tsoumakas, I. Katakis, I. Vlahavas, Random k-labelsets for multilabel classification, IEEE Trans. Knowl. Data Eng. 23 (7) (2011) 1079–1089.
[17] S. Kanj, F. Abdallah, T. Denoeux, Evidential Multi-Label Classification Using the Random k-Label Sets Approach, Springer, Berlin, Heidelberg, 2012.
[18] Y.P. Wu, H.T. Lin, Progressive random k-labelsets for cost-sensitive multi-label classification, Mach. Learn. 106 (5) (2017) 671–694.
[19] W. Weng, Y.J. Lin, S.X. Wu, Y.W. Li, Y. Kang, Multi-label learning based on label-specific features and local pairwise label correlation, Neurocomputing 273 (2017) 385–394.
[20] E.L. Menía, F. Janssen, Learning rules for multi-label classification: a stacking and a separate-and-conquer approach, Mach. Learn. 105 (1) (2016) 1–50.
[21] Z.H. Zhou, J. Wu, W. Tang, Ensembling neural networks: many could be better than all, Artif. Intell. 137 (1–2) (2002) 239–263.
[22] C. Lin, W.Q. Chen, C. Qiu, Y.F. Wu, S. Krishnan, Q. Zou, LibD3C: ensemble classifiers with a clustering and dynamic selection strategy, Neurocomputing 123 (2014) 424–435.

[23] L. Rokach, A. Schclar, E. Itach, Ensemble methods for multi-label classification, Expert Syst. Appl. 41 (16) (2014) 7507–7523.
[24] J. Lee, D.W. Kim, Feature selection for multi-label classification using multivariate mutual information, Pattern Recognit. Lett. 34 (3) (2013) 349–357.
[25] Y.J. Lin, Q.H. Hu, J.H. Liu, J.K. Chen, J. Duan, Multi-label feature selection based on neighborhood mutual information, Appl. Soft Comput. 38 (2016) 244–256.
[26] H. Li, D.Y. Li, Y.H. Zhai, S.G. Wang, J. Zhang, A novel attribute reduction approach for multi-label data based on rough set theory, Inf. Sci. 367–368 (2016) 827–847.
[27] H.C. Peng, F.H. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[28] Y.J. Lin, J.H. Liu, J.H. Liu, J. Duan, Multi-label feature selection based on max-dependency and min-redundancy, Neurocomputing 168 (C) (2015) 92–103.
[29] J. Lee, D.W. Kim, Memetic feature selection algorithm for multi-label classification, Inf. Sci. 293 (293) (2015) 80–96.
[30] J. Lee, D.W. Kim, SCLS: multi-label feature selection based on scalable criterion for large label set, Pattern Recognit. (2017) 342–352.
[31] F. Li, D.Q. Miao, W. Pedrycz, Granular multi-label feature selection based on mutual information, Pattern Recognit. 67 (2017) 410–423.
[32] T. Denoeux, Z. Younes, F. Abdallah, Representing uncertainty on set-valued variables using belief functions, Artif. Intell. 174 (7–8) (2010) 479–499.
[33] M. Everingham, L. Gool, C.K. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[34] X. Geng, R.Z. Ji, Label distribution learning, IEEE Trans. Knowl. Data Eng. 28 (7) (2014) 1734–1748.
[35] B.B. Gao, C. Xing, C.W. Xie, J. Wu, X. Geng, Deep label distribution learning with label ambiguity, IEEE Trans. Image Process. 26 (6) (2017) 2825–2838.
[36] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. (1982) 341–356.
[37] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, 1992.
[38] H. Li, D.Y. Li, Y.H. Zhai, S.G. Wang, J. Zhang, A variable precision attribute reduction approach in multilabel decision tables, Sci. World J. 2014 (2014) 1–7.
[39] Y. Yu, W. Pedrycz, D.Q. Miao, Neighborhood rough sets based multi-label classification for automatic image annotation, Int. J. Approx. Reason. 54 (9) (2013) 1373–1387.
[40] S. Vluymans, C. Cornelis, F. Herrera, Y. Saeys, Multi-label classification using a fuzzy rough neighborhood consensus, Inf. Sci. 433 (2018) 96–114.
[41] Y.Y. Yao, Three-way decision: an interpretation of rules in rough set theory, in: International Conference on Rough Sets and Knowledge Technology, 2009, pp. 642–649.
[42] Y.Y. Yao, The superiority of three-way decisions in probabilistic rough set models, Inf. Sci. 181 (6) (2011) 1080–1096.
[43] B.Q. Hu, H. Wong, K.F.C. Yiu, On two novel types of three-way decisions in three-way decision spaces, Int. J. Approx. Reason. 82 (2017) 285–306.
[44] J. Qian, C.Y. Dang, X.D. Yue, N. Zhang, Attribute reduction for sequential three-way decisions under dynamic granulation, Int. J. Approx. Reason. 85 (2017) 196–216.
[45] X.N. Li, H.Q. Yi, Y.H. She, B.Z. Sun, Generalized three-way decision models based on subset evaluation, Int. J. Approx. Reason. 83 (C) (2017) 142–159.
[46] B.Z. Sun, W.M. Ma, B.J. Li, X.N. Li, Three-way decisions approach to multiple attribute group decision making with linguistic information-based decision-theoretic rough fuzzy set, Int. J. Approx. Reason. 93 (2018) 424–442.
[47] J. Read, B. Pfahringer, G. Holmes, Classifier chains for multi-label classification, Mach. Learn. 85 (2011) 333–359.
[48] G. Madjarov, D. Gjorgjevikj, S. Deroski, Two stage architecture for multi-label learning, Pattern Recognit. 45 (3) (2012) 1019–1034.
[49] W.S. Kai, H.L. Chong, Addressing class-imbalance in multi-label learning via two-stage multi-label hypernetwork, Neurocomputing 266 (29) (2017) 375–389.
[50] F.J. Ren, L. Wang, Sentiment analysis of text based on three-way decisions, J. Intell. Fuzzy Syst. 33 (1) (2017) 245–254.
[51] Y.Y. Yao, Y. Zhao, J. Wang, On reduct construction algorithms, in: Rough Sets & Knowledge Technology Proceedings, vol. 4062, 2006, pp. 100–117.
[52] Y.Y. Yao, S.K. Wong, A decision theoretic framework for approximating concepts, Int. J. Man-Mach. Stud. 37 (6) (1992) 793–809.
[53] Y.W. Guo, L.C. Jiao, S. Wang, S. Wang, F. Liu, K.X. Rong, T. Xiong, A novel dynamic rough subspace based selective ensemble, Pattern Recognit. 48 (5) (2014) 1638–1652.
[54] G.Y. Wang, X.A. Ma, H. Yu, Monotonic uncertainty measures for attribute reduction in probabilistic rough set model, Int. J. Approx. Reason. 59 (C) (2015) 41–67.
[55] X.Y. Zhang, D.Q. Miao, Three-way attribute reducts, Int. J. Approx. Reason. 88 (2017) 401–434.
[56] J. Ar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (1) (2006) 1–30.
[57] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek, I. Vlahavas, Mulan: a Java library for multi-label learning, J. Mach. Learn. Res. 12 (7) (2012) 2411–2414.
[58] J. Read, P. Reutemann, B. Pfahringer, G. Holmes, MEKA: a multi-label/multi-target extension to WEKA, J. Mach. Learn. Res. 17 (1) (2016) 667–671.
[59] K. Kimura, M. Kudo, L. Sun, S. Koujaku, Fast random k-labelsets for large-scale multi-label classification, in: International Conference on Pattern Recognition, 2017, pp. 438–443.
[60] G. Tsoumakas, I. Vlahavas, Random k-labelsets: an ensemble method for multilabel classification, in: European Conference on Machine Learning, ECML, 2007, pp. 406–417.
[61] S. Godbole, S. Sarawagi, Discriminative methods for multi-labeled classification, in: Advances in Knowledge Discovery and Data Mining, in: Lecture Notes in Computer Science, vol. 3056, 2004, pp. 22–30.
[62] R.E. Schapire, Y. Singer, Improved boosting algorithms using confidence-rated predictions, Mach. Learn. 37 (1999) 297–336.
[63] R.E. Schapire, Y. Singer, BoosTexter: A Boosting-Based Systemfor Text Categorization, Kluwer Academic Publishers, 2000.