Pattern Recognition Letters 33 (2012) 513–523

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

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

Multilabel classification using heterogeneous ensemble of multi-label classifiers

Muhammad Atif Tahir [a,b,*], Josef Kittler [a], Ahmed Bouridane [b]

[a] Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, UK
[b] School of Computing, Engineering and Information Sciences, University of Northumbria, Newcastle upon Tyne NE2 1XE, UK

ARTICLE INFO

Article history:
Received 21 July 2010
Available online 25 November 2011
Communicated by F. Roli

Keywords:
Multilabel classification
Heterogeneous ensemble of multilabel classifiers
Static/dynamic weighting

ABSTRACT

Multilabel classification is a challenging research problem in which each instance may belong to more than one class. Recently, a considerable amount of research has been concerned with the development of "good" multi-label learning methods. Despite the extensive research effort, many scientific challenges posed by e.g. highly imbalanced training sets and correlation among labels remain to be addressed. The aim of this paper is to use a heterogeneous ensemble of multi-label learners to simultaneously tackle both the sample imbalance and label correlation problems. This is different from the existing work in the sense that we are proposing to combine state-of-the-art multi-label methods by ensemble techniques instead of focusing on ensemble techniques within a multi-label learner. The proposed ensemble approach (EML) is applied to six publicly available multi-label data sets from various domains including computer vision, biology and text using several evaluation criteria. We validate the advocated approach experimentally and demonstrate that it yields significant performance gains when compared with state-of-the art multi-label methods.

1. Introduction

A conventional multi-class classification system assigns each instance x a single label l from a set of disjoint labels L. However, in many modern applications such as music categorisation (Li and Ogihara, 2006), text classification (Godbole and Sarawagi, 2004; Zhang and Zhou, 2006), image/video categorisation (Tahir et al., 2009; Dimou et al., 2009), etc., each instance is to be assigned to a subset of labels $Y \subseteq L$. This problem is known as multi-label learning.

There is a considerable amount of research concerned with the development of "good" multi-label learning methods. Despite the extensive research effort, there exist many scientific challenges. They include highly imbalanced training sets, as very limited data is available for some labels, and capturing correlation among classes. Interestingly, most state-of-the-art multi-label methods are designed to focus mainly on the second problem and a very limited effort has been devoted to handling imbalanced data populations. In this paper, we focus on the first problem of multi-label learning, and tackle highly imbalanced data distributions using ensemble of multi-label classifiers.

Ensemble techniques are becoming increasingly important as they have repeatedly demonstrated the ability to improve upon the accuracy of single-classifiers with highly imbalanced data populations (Chawla and Sylvester, 2007; Hsu and Srivastava, 2009). It is well known that an ensemble of classifiers can provide higher accuracy than a single best classifier if the member classifiers are diverse and accurate. Ensembles can be homogeneous, in which every base classifier is constructed using the same algorithm, or heterogeneous in which base classifiers are constructed using different algorithms. In fact, some state-of-the-art multi-label learners use homogeneous or heterogeneous ensemble techniques to improve the overall performance (Cheng and Hullermeier, 2009). Examples include the combination of Logistic Regression and Nearest Neighbour classifiers (Cheng and Hullermeier, 2009), and the binary pairwise *one-vs-one* approach for multi-label classification (Furnkranz et al., 2008).

The aim of this paper is to use heterogeneous ensembles of multi-label learners to improve the performance. This is different from the existing work in the sense that we are proposing to combine state-of-the-art multi-label methods by ensemble techniques instead of mainly focusing on ensemble techniques within a multi-label learner.

Interestingly, another advantage of combining multi-label classifiers is that both the class imbalance and label correlation problems can be tackled simultaneously. The imbalance problem can be handled by using an ensemble of multi-label classifiers where each multi-label method belongs to different adaptation

* Corresponding author at: Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford GU2 7XH, UK. Fax: +44 28 9097 5666.
E-mail addresses: m.tahir@surrey.ac.uk, muhammad.tahir@northumbria.ac.uk (M.A. Tahir), j.kittler@surrey.ac.uk (J. Kittler), Ahmed.Bouridane@northumbria.ac.uk (A. Bouridane).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.10.019

group and therefore provides a potentially more independent and diverse set of predictions. The correlation problem can be solved by using, as base classifiers, state-of-the-art multi-label classifiers that inherently consider correlation among labels.

The proposed ensemble multilabel learning approach (EML) [1] is applied to six publicly available multi-label data sets from different domains (Scene, Yeast, Pascal07, Emotions, Medical and Enron) using 12 different multi-label classification measures. We validate the advocated approach experimentally and demonstrate that it yields significant performance gains when compared with individual state-of-the art multi-label methods.

The paper is organised as follows. In Section 2, we review state-of-the-art multi-label methods. Section 3 discusses the proposed ensemble of multi-label classifiers. Experiments are discussed in Section 4 followed by the results and discussion in Section 5. Section 6 concludes the paper.

2. Related work

The sparse literature on multi-label classification, driven by problems in text classification, bioinformatics, music categorisation, and image/video classification, has recently been evaluated by Tsoumakas et al. (2009). This research can be divided into two different groups: (i) *problem transformation* methods, and (ii) *algorithm adaptation* methods. The problem transformation methods aim to transform a multilabel classification task into one or more single-label classification (Boutell et al., 2004; Tsoumakas and Vlahavas, 2007), or label ranking (Furnkranz et al., 2008) tasks. The algorithm adaptation methods extend traditional classifiers to handle multi-label concepts directly (Zhang and Zhou, 2007; Elisseeff and Weston, 2002; Cheng and Hullermeier, 2009). In this section, we review the state-of-the-art multi-label learners that are used as base classifiers in our ensemble approach namely RaKEL (Tsoumakas and Vlahavas, 2007), Calibrated Label Ranking (CLR) (Furnkranz et al., 2008), Multi-label KNN (MLKNN) (Zhang and Zhou, 2007), Instance Based Logistic Regression (IBLR) (Cheng and Hullermeier, 2009) and Ensemble of Classifier Chains (ECC) (Read et al., 2009).

2.1. RAkEL

Multilabel classification can be reduced to the conventional classification problem by considering each unique set of labels as one of the classes. This approach is referred to as *label powerset* (LP) in the literature. However, this approach leads to a large number of label subsets with the majority of them with a very few examples and it is also computationally expensive. Many approaches have been proposed in the literature to deal with the aforementioned problem (Tsoumakas and Vlahavas, 2007; Read et al., 2008). One state-of-the-art approach is RaKEL (Random k-Labelsets) (Tsoumakas and Vlahavas, 2007) that constructs an ensemble of LP classifiers where each LP classifier is trained using a different small random subset of the set of labels. In order to get near-optimal performance, the parameters of the method (subset size, number of models, threshold, etc.) must be optimised using internal cross validation. However, these parameters are hard to optimise when the number of training samples is insufficient.

2.2. Ensemble of Classifier Chains (ECC)

Multilabel classification can be reduced to the conventional binary classification problem. This approach is referred to as *binary relevance* (BR) learning in the literature. In BR learning, the original data set is divided into $|Y|$ data sets where $Y = \{1,2,\ldots,N\}$ is the finite set of labels. BR learns one binary classifier $h_a : X \rightarrow \{\neg a, a\}$ for each concept $a \in Y$. BR learning is theoretically simple and has a linear complexity with respect to the number of labels. Its assumption of label independence makes it attractive in situations where new examples may not be relevant to any known subset of labels or where label relationships may change over the test data (Read et al., 2009). However, BR learning is criticised for not considering correlations among the labels (Cheng and Hullermeier, 2009; Furnkranz et al., 2008). The work in (Read et al., 2009) is a state-of-the-art BR approach. Their classifier chain (CC) approach only requires a single training iteration like BR and uses labels directly from the training data without any internal classification. Classifiers are linked in a chain where each classifier deals with the BR problem associated with label $y_j \in Y$. However, the order of the chain can clearly have an effect on accuracy. An ensemble ECC framework is used to create different random chain orderings. This method was shown to perform well against BR and other multi-label classifiers.

2.3. Calibrated Label Ranking (CLR)

Similar to *one-vs-all* approach in BR learning, the binary pairwise *one-vs-one* approach has also been employed for multi-label classification, therefore requiring $|Y|^2$ classifiers as opposed to $|Y|$. Calibrated Label Ranking (CLR) (Furnkranz et al., 2008) is an efficient pairwise approach for multilabel classification. The key idea in this approach is to introduce an artificial calibration label that, in each example, separates the relevant label from the irrelevant labels. This method was shown to perform well against other multi-label classifiers but mainly on ranking measures.

2.4. Multi-label KNN (MLkNN)

The instance-based approach is also quite popular in multilabel classification. In (Zhang and Zhou, 2007), a lazy learning approach (MLKNN) is proposed. This method is derived from the popular k-Nearest Neighbour (kNN) algorithm. It consists of two main steps. In the first step, for each test instance, its k nearest neighbours in the training set are identified. Next, in the second step, the maximum *a posteriori* probability label set is identified for a test instance based on the statistical information gained from the label sets of these neighbouring instances. This method was shown to perform well in some domains e.g. in predicting the functional classes of genes in the Yeast *Saccharomyces cerevisiae* (Cheng and Hullermeier, 2009).

2.5. Instance Based Logistic Regression (IBLR)

IBLR is also a novel approach to instance-based learning, with the main idea to combine instance-based learning (ILR) and logistic regression (Cheng and Hullermeier, 2009). The key idea is to consider the labels of neighboring instances as "features" of unseen samples and thus reduce ILR to logistic regression. This approach captures interdependencies between labels in multilabel classification.

3. Ensemble of multi-label classifiers (EML)

Let X denote a set of instances and let $Y = \{1,2,\ldots,N\}$ be a set of labels. Given a training set $S = \{(x_1,y_1),\ldots,(x_m,y_m)\}$ where $x_i \in X$ is a single instance and $y_i \subseteq Y$ is the label set associated with x_i, the goal is to design a multi-label classifier H that predicts a set of labels for an unseen example.

[1] A shorter version of this work was published in MCS 2010 (Tahir et al., 2010).

As discussed previously, ensembles methods are well-known for overcoming over-fitting problems especially in highly unbalanced data sets. Ensemble of multi-label classifiers train q multi-label classifiers $H_1, H_2, \ldots, H_q$. Thus, all q models are diverse and able to give different multi-label predictions. For an unseen instance x, each kth individual model (of q models) produces an N-dimensional vector $P_k = [p_{1k}, p_{2k}, \ldots, p_{Nk}]$, where the value p_{bk} is the probability of the bth class label assigned by classifier k being correct.

There are many ways of combining the outputs of these q classifiers. Among them, nontrainable combiners such as MEAN, MAX, MIN are the simplest and most popular way to combine the scores of classifiers with probabilistic outputs (Kuncheva, 2004). These combiners have no extra parameters to be trained. In addition to nontrainable combiners, weighted voting methods also have the potential to make the multiple classifier systems more robust to the choice of individual classifiers. Static and dynamic weighting of classifiers are two well-known approaches to weighting individual classifiers. In the dynamic scheme, the weights assigned to the individual classifiers can change for each test pattern while in the static weighting, the weights are computed for each classifier in the training phase and the weights assigned to each classifier are maintained constant during the classification of the test patterns. In summary, the following five combination methods are evaluated in this paper for multi-label classification.

3.1. Average of probabilities (EML_A)

The fusion by averaging is one of the oldest strategies for decision making and is widely used in different forms in pattern recognition and classification (Kittler et al., 1998). The fusion by averaging will result in an ensemble decision for unseen instance x and label y_b based on

$$\mu_b(x) = \frac{1}{q}\sum_{k=1}^{q} p_{bk}(x) \tag{1}$$

3.2. Average of probabilities and threshold selection via multi-labelledness (EML_T)

It is reported in Fan and Lin (2007) that properly adjusting the decision thresholds (instead of the traditional value of 0.5) can improve the performance of a multi-label classifier. In this paper, the thresholds are adjusted using the method proposed in Read et al. (2009) and described below. It avoids expensive internal cross validation but is only suited for batch classification. Let the sum of probabilities from q models be stored in a vector $W = (\theta_1, \ldots, \theta_N) \in \mathbb{R}^{\mathbb{N}}$ such that $\theta_b = \sum_{k=1}^{q} p_{bk}$. W is then normalised to W^{norm}, which represents a distribution of scores for each label in $[0,1]$. Let X_T be the training set and X_S the test set. A threshold t is then selected using Eq. (2) to choose the final predicted multi-label set Z.

$$t = \arg \min_{\{t \in 0.00, 0.001, \ldots, 1.00\}} |LCard(X_T) - LCard(H_t(X_s))| \tag{2}$$

where LCard (Label Cardinality) is the standard measure of "multi-labelled-ness" (Tsoumakas et al., 2009). It is the average number of labels relevant to each instance and is defined as $LCard(X) = \frac{\sum_{i=1}^{|X|}|E_i|}{|X|}$ where E_i is the actual set of labels for the training set and a predicted set of labels under threshold t for the test set. Eq. (2) measures the difference between the label cardinality of the training set and the predictions made on the test set. It avoids intensive internal cross-validation. Hence, the relevant labels in Z under threshold t represent the final predicted set of labels. It should be clear that the actual test labels are never seen by the presented threshold selection method. The threshold t is calculated using the predicted set of labels only.

3.3. Static weighting by N-Fold Cross Validation (EML_S)

In static weighting, the weights for each classifier are computed in the training phase. In this paper, the weights for each classifier are learnt via N-Fold Cross Validation ($N = 5$) which is the most widely used strategy to learn the weights with Average Precision (see Appendix A) as quality measure of the classifier.

3.4. Dynamic weighting using Dudani rule (EML_D)

In Dudani (1976), a weighted k-NN rule is proposed for classifying new patterns. In this approach, the votes of the k nearest neighbors are weighted by a function of their distance to the test pattern. The main idea is to weight a neighbor with smaller distance more heavily than the one with a greater distance, i.e. the nearest neighbor gets a weight of 1, the furthest neighbor a weight of 0, and the other weights are scaled linearly to take values in the interval $[0,1]$. Dudani's weight can be computed as

$$w_j = \begin{cases} \frac{d_k - d_j}{d_k - d_1} & \text{if } d_k \neq d_1 \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

where d_j denotes the distance of the jth nearest neighbor, d_1 is the distance of the nearest neighbor, and d_k indicates the distance of the furthest neighbor.

This weighting function can be utilized for classifier fusion (Valdovinos and Sánchez, 2009) and is adopted in this paper to learn the weights of multi-label classifiers. The value of k is replaced by the number of models q and for each individual classifier, the q distances of unknown instance x to its nearest neighbor is sorted in an increasing order $(d_1, d_2, \ldots, d_q)$. Thus, Eq. (3), can be rewritten as

$$weight(M_j) = \begin{cases} \frac{d_q - d_j}{d_q - d_1} & \text{if } d_q \neq d_1 \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

where d_1 is the smallest among the q distances of x to its nearest neighbors, and d_q is the largest of those distances. $w(M_j)$ is the weight of multi-label classifier j for instance x.

3.5. Dynamic weighting using Shepard rule (EML_P)

Valdovinos and Sánchez (2009) proposed another weighted function that was based on the work of Shepard (Shepard, 1987) and is adopted in this paper to learn the weights of multi-label classifiers. Shepard argues for a universal perceptual law which states that "the relevance of a previous stimulus for the generalization to a new stimulus is an exponentially decreasing function of its distance in psychological space". This gives the weighted voting function as described below

$$weight(M_j) = e^{-\alpha d_j^{\beta}} \tag{5}$$

where α and β are constants and determine the slope and the power of the exponential decay function respectively. In this work, α and β are set to 1.

4. Experiments

4.1. Datasets

We experimented with six multi-label datasets from a variety of domains. Table 1 shows certain standard statistics of these datasets. The image datasets "scene" and 'pascal07' are concerned with semantic indexing of images of still scenes (Boutell et al., 2004) and objects respectively. The "yeast" data set contains functional classes of genes of Yeast *S. cerevisiae* (Tsoumakas and Vlahavas, 2007;

Table 1
Standard and multilabel statistics for the data sets used in the experiments.

Datasets	Domain	Samples	Features	Labels	LCard
Enron	Text	1702	1001	53	3.38
Medical	Text	978	1449	45	1.25
Scene	Vision	2407	294	6	1.07
Pascal07	Vision	9963	500	20	1.44
Yeast	Biology	2417	103	14	4.24
Emotions	Music	593	72	6	1.87

Cheng and Hullermeier, 2009). The "enron" is a subset of the Enron email corpus (Read et al., 2008). The "emotions" consists of 100 songs from each of the following seven different genres: Classical, Reggae, Rock, Pop, Hip-Hop, Techno and Jazz. The collection was created from 233 musical albums choosing three songs from each album (Trohidis et al., 2008). The Medical dataset (NLP, 2007; Read et al., 2008) is composed of documents with a free-text summary of patient symptom histories and prognoses, which are used to predict insurance codes. All reported results are estimated from 5×2-fold cross validation and the paired t-test is then used to determine their significance under a value of 0.05.

4.2. Features

Publicly available feature vectors are used for all datasets[2] except Pascal07. For Scene, the feature vector consists of 294 dimensions computed as spatial colour moments in the *LUV* space. The image is divided into 49 blocks using a 7×7 grid. The first and second moments (mean and variance) are computed for each band. The end result is a $49 \times 2 \times 3 = 294$ dimensional feature vector. For Mediamill, a set of MPEG-7 features (totaling 320), namely colour structure and layout, edge histogram, homogeneous texture and colour information are extracted. For Pascal07, the OpponentSIFT descriptor, which gave the best performance in the concept detection problem for Pascal VOC 07 (van de Sande et al., 2010), is computed for densely sampled image regions. The descriptors are clustered using the *k*-means algorithm and form codebooks of 500 clusters each. An image is then represented as a histogram of occurrences of clusters in the image. A detailed description about feature vectors of Yeast, Scene, Mediamill, Emotions, Medical and Enron can be found in (Boutell et al., 2004; Dimou et al., 2009; Tsoumakas and Vlahavas, 2007; Read et al., 2008; Trohidis et al., 2008; NLP, 2007).

4.3. Evaluation measures

A multi-label classification requires different evaluation measures than traditional single-label classification. The details are given in Appendix A. These measures can be categorised into three groups: example based, label-based and ranking-based. In this paper, 12 different evaluation measures are used to compare the proposed approach. These measures include Hamming Loss, Accuracy, F_1, and Classification Accuracy from the example-based category, and Micro/Macro F_1/AUC from the label-based group. Additionally, we use One-error, Coverage, Ranking Loss and Average Precision from the ranking-based group.

4.4. Benchmark methods

The proposed EML method is compared with the state-of-the-art multi-label classifiers discussed in Section 2: RaKEL (Tsoumakas and Vlahavas, 2007), ECC (Read et al., 2009), CLR (Furnkranz et al., 2008), MLKNN (Zhang and Zhou, 2007), and, IBLR (Cheng and Hullermeier, 2009). Since all these multi-label classifiers are quite diverse, they are selected as base classifiers in the proposed EML method. MLKNN and IBLR are from the algorithm adaptation group while ECC, RaKEL and CLR are from the problem transformation group. Further, C4.5 is used as a base classifier in RaKEL while Linear SVM is used as a base classifier in ECC and CLR. For the training of MLkNN, IBLR, CLR and RAkEL, the Mulan[1] open-source library in Java for multi-label classification is used. For the training of ECC, the MEKA[3] open-source library is used with the default parameters. Both libraries are an extension of WEKA (Witten and Frank, 2005). All multi-label classifiers are trained using default parameters which are also the best reported parameters, e.g. the number of neighbours is 10 for IBLR and MLkNN; the number of iterations is 10 for ECC.

The multi-label classifiers are compared with the following variants of the proposed method.

- EML_A: An ensemble of multilabel classifiers using the MEAN rule. It should be noted that we have also tried several other rules such as MAX, MIN and only the best is reported here.
- EML_T: Same as EML_A but threshold is selected using threshold selection method discussed in Section 3.
- EML_S: An ensemble of multi-label classifiers with weights for each classifier learnt by using static weighting. Thresholds are then selected using the proposed threshold selection method.
- EML_D: An ensemble of multi-label classifiers with weights for each classifier learnt dynamically by using Dudani's rule as discussed in Section 3. The thresholds are then selected using the proposed threshold selection method.
- EML_P: An ensemble of multi-label classifiers with weights for each classifier learnt dynamically by using Shepard's rule as discussed in Section 3. The thresholds are then selected using the proposed threshold selection method.

5. Results and discussion

In this section, we discuss the results obtained using EML for the multilabel datasets. These results are summarised as follows:

- Tables 2–7 show the comparison of the various variants of EML with the state-of-the-art multilabel classifiers for the Yeast, Pascal07, Emotions, Medical, Enron and Scene respectively. Overall, when the individual multi-label classifiers are compared with each other, ECC exhibits very good performance in the majority of examples and label-based measures that require a predicted set of labels from an unseen example i.e. either 0 or 1. For example, ECC consistently performs well on Accuracy, Fmeasure, Micro/Macro F_1 in all except Enron/Medical. This may be explained by the fact that this method uses an ensemble framework to create different random chain orderings along with the threshold selection method described in Section 3.2. This can help the correct identification of labels.
- When the individual multi-label classifiers are compared with each other for the example-based ranking measures (One-Error, Coverage, Ranking Loss, Average Precision) and the label-based ranking measures (Micro/Macro AUC), it is hard to pick a multi-label method that can perform consistently well. For example, IBLR demonstrates very good performance in Yeast and Scene but does not deliver similar superiority on the other data sets. Similarly, CLR demonstrates very good performance on Pascal07, Emotions and Enron but does not deliver similar superiority on the other data sets. It is also observed that RAkEL performs very well on data sets from text domain (Enron and Medical).

[2] http://www.mlkd.csd.auth.fr/multilabel.html.

[3] http://www.cs.waikato.ac.nz/jmr30/software.

Table 2
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for Yeast. For each evaluation criterion, ↓ indicates "the smaller the better" while ↑ indicates "the higher the better". ∗ means significantly better than all other methods except those which are marked as +. Bold values indicate the best performance among all including ensemble classifiers while underscore values indicate the best performance among Individual Classifiers.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming Loss ↓	0.198	0.198	0.229	0.211	0.216	**0.193***	0.199	0.198	0.197	0.198
Accuracy ↑	0.499	0.510	0.475	0.491	0.529	0.540	0.549	0.549	**0.553***	0.552$^+$
Fmeasure ↑	0.635	0.644	0.621	0.633	0.659	0.671	0.681$^+$	0.681$^+$	**0.683***	0.682$^+$
ClassAcc ↑	0.165	0.186	0.102	0.134	0.183	0.196$^+$	0.188	0.188	**0.200***	0.196
Micro F_1 ↑	0.633	0.641	0.615	0.629	0.649	0.665	0.672$^+$	0.672$^+$	**0.674***	0.672$^+$
Macro F_1 ↑	0.352	0.375	0.400	0.390	0.415	0.399	0.486$^+$	0.486$^+$	**0.489***	0.488
Micro AUC ↑	0.835	0.838	0.792	0.815	0.805	**0.846***	0.842	0.842	0.843	0.843
Macro AUC ↑	0.668	0.685	0.628	0.656	0.652	**0.705***	0.693	0.693	0.695	0.693
One-error ↓	0.238	0.237	0.279	0.239	0.259	0.227$^+$	0.227$^+$	0.226$^+$	**0.224***	0.225$^+$
Coverage ↓	6.370	6.331	7.641	6.632	7.086	6.241$^+$	6.241$^+$	6.236$^+$	**6.173***	6.190$^+$
Ranking Loss ↓	0.173	0.170	0.223	0.181	0.214	0.162$^+$	0.162$^+$	0.162$^+$	**0.161***	0.162$^+$
AvgPrecision ↑	0.756	0.759	0.710	0.750	0.735	0.768$^+$	0.768$^+$	0.768$^+$	**0.769***	0.768$^+$
# Wins (Ind)	1/12	8/12	0/12	0/12	4/12	–		–	–	–
# Wins (All)	0/12	0/12	0/12	0/12	0/12	3/12	0/12	0/12	9/12	0/12

Table 3
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for Pascal07.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming Loss ↓	0.068	0.068	0.079	0.076	0.085	**0.065***	0.078	0.078	0.080	0.080
Accuracy ↑	0.126	0.149	0.194	0.240	0.339	0.280	0.358	**0.359***	0.354	0.353
Fmeasure ↑	0.153	0.179	0.246	0.306	0.427	0.334	0.461	**0.462***	0.457	0.456
ClassAcc ↑	0.071	0.085	0.095	0.127	**0.175***	0.168	0.169$^+$	0.170$^+$	0.166	0.165
Micro F_1 ↑	0.226	0.255	0.296	0.356	0.414	0.395	0.453	**0.454***	0.447	0.445
Macro F_1 ↑	0.070	0.083	0.134	0.249	0.301	0.171	0.337	**0.338**	0.333	0.325
Micro AUC ↑	0.800	0.826	0.734	0.851	0.775	**0.865***	0.864	0.864	0.860	0.851
Macro AUC ↑	0.699	0.759	0.647	0.807	0.722	**0.821***	0.820	0.820	0.814	0.798
One-error ↓	0.559	0.557	0.611	0.526	0.533	0.499	0.499	**0.497***	0.504	0.505
Coverage ↓	5.344	4.714	6.671	3.973	5.640	3.737	3.737	**3.728***	3.831	4.102
Ranking Loss ↓	0.212	0.187	0.272	0.154	0.225	0.143	0.143	**0.142***	0.147	0.158
AvgPrecision ↑	0.511	0.525	0.463	0.569	0.537	0.589	0.589	**0.590***	0.582	0.576
# Wins (Ind)	1/12	1/12	0/12	6/12	5/12	–	–	–	–	
# Wins (All)	0/12	0/12	0/12	0/12	1/12	3/12	0/12	8/12	0/12	0/12

Table 4
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for emotions.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming Loss ↓	0.204	0.201	0.231	0.205	0.204	**0.185***	0.199	0.200	0.199	0.200
Accuracy ↑	0.512	0.523	0.482	0.522	0.564	**0.579***	0.568	0.567	0.568	0.568
Fmeasure ↑	0.625	0.630	0.599	0.635	0.681$^+$	0.694	0.701$^+$	0.700$^+$	**0.702***	0.699$^+$
ClassAcc ↑	0.261	0.288	0.227	0.267	0.304	**0.326***	0.272	0.272	0.271	0.272
Micro F_1 ↑	0.644	0.656	0.616	0.660	0.676	**0.697***	0.680	0.680	0.680	0.679
Macro F_1 ↑	0.608	0.632	0.603	0.647	0.663	**0.673***	0.656	0.656	0.657	0.656
Micro AUC ↑	0.844	0.851	0.811	0.844	0.828	**0.871***	0.855	0.855	0.855	0.851
Macro AUC ↑	0.820	0.832	0.793	0.827	0.821	**0.855***	0.848	0.848	0.849	0.845
One-error ↓	0.284	0.279	0.327	0.271	0.275	0.249$^+$	0.249$^+$	**0.248***	0.251$^+$	0.254$^+$
Coverage ↓	1.83	1.77	2.02	1.73	1.90	**1.68***	**1.68***	**1.68***	**1.68***	1.694
Ranking Loss ↓	0.170	0.164	0.205	0.154	0.181	**0.143***	**0.143***	**0.143***	0.144$^+$	0.147
AvgPrecision ↑	0.791	0.798	0.762	0.807	0.795	**0.818***	**0.818***	**0.818***	0.817$^+$	0.815
# Wins (Ind)	0/12	3/12	0/12	4/12	5/12	–	–	–	–	
# Wins (All)	0/12	0/12	0/12	0/12	0/12	10/12	3/12	4/12	2/12	0/12

- Tables 2–7 show that ensemble of multi-label classifiers delivers significant performance gains for almost all measures. However, the performance of different variants of EML may vary among the data sets as described below.
  - For ranking-based measures, all variants of the proposed ensemble methods have very similar performance and are not significantly better than each other in all but Pascal07 where static weighting technique has the best performance.
  - It is observed that the weights learnt dynamically for each test sample from Shepard approach (EML_P) are superior than weights from Dudani's approach (EML_D). This can be explained by the fact that Dudani's rule assigns a weight of 0 to the classifier with the largest distance and thus enforces sparsity and may lead to loss of information. On the other hand, Shepard's rule weight is an exponentially decreasing function of d.

Table 5
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for medical.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming loss ↓	0.016	0.026	0.011	0.014	0.012	**0.010***	**0.010***	0.011	0.012	0.014
Accuracy ↑	0.518	0.502	0.733	0.658	0.760	0.773	**0.777***	**0.777***	0.744	0.681
Fmeasure ↑	0.550	0.576	0.774	0.716	0.817$^+$	0.815	**0.822***	**0.822***	0.794	0.739
ClassAcc ↑	0.451	0.378	0.647	0.541	0.646	**0.685***	**0.685***	**0.685***	0.643	0.565
Micro F_1 ↑	0.637	0.552	0.801	0.739	0.791	**0.817***	0.815	**0.817***	0.788	0.747
Macro F_1 ↑	0.178	0.202	**0.361***	0.339$^+$	0.352$^+$	0.340	0.298	0.300	0.295	0.293
Micro AUC ↑	0.955	0.913	0.909	0.972	0.949	0.976	0.976	**0.977***	0.974	0.971
Macro AUC ↑	0.579	0.531	0.551	**0.676**	0.586	0.675$^+$	0.674$^+$	0.675$^+$	0.656	0.636
One-error ↓	0.266	0.414	0.195	0.208	0.171	0.151$^+$	0.151$^+$	**0.150***	0.151$^+$	0.153$^+$
Coverage ↓	2.65	4.58	4.70	1.77	2.92	1.54$^+$	1.54$^+$	1.53$^+$	**1.52***	1.70
Ranking loss ↓	0.044	0.084	0.085	0.027	0.048	**0.023***	**0.023***	**0.023***	**0.023***	0.026
AvgPrecision ↑	0.793	0.686	0.815	0.849	0.862	0.887$^+$	0.887$^+$	0.888$^+$	**0.888***	0.885
# Wins (Ind)	0/11	0/11	4/11	4/11	4/11	–	–	–	–	
# Wins (All)	0/11	0/11	1/11	1/11	0/11	4/11	5/11	7/11	3/11	0/11

Table 6
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for enron.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming loss ↓	0.054	0.064	0.051	0.065	0.061	**0.048***	0.052	0.051	0.054	0.056
Accuracy ↑	0.295	0.291	0.418	0.386	0.400	**0.459***	0.456$^+$	0.457$^+$	0.446	0.413
Fmeasure ↑	0.414	0.411	0.559	0.541	0.556	0.604	0.613$^+$	**0.614***	0.601	0.578
ClassAcc ↑	0.052	0.062	0.109	0.071	0.062	**0.118***	0.066	0.066	0.074	0.038
Micro F_1 ↑	0.444	0.421	0.546	0.523	0.531	0.586	0.595$^+$	**0.596***	0.579	0.564
Macro F_1 ↑	0.073	0.126	0.146	**0.222***	0.198	0.153	0.207	0.208	0.202	0.193
Micro AUC ↑	0.894	0.871	0.804	0.901	0.858	**0.914***	0.912	0.912	0.901	0.900
Macro AUC ↑	0.584	0.579	0.565	0.681	0.631	**0.688**	0.676	0.677	0.617	0.626
One-error ↓	0.332	0.469	0.315	0.323	0.329	0.240	0.240$^+$	**0.239***	0.250	0.247$^+$
Coverage ↓	13.80	16.05	26.10	12.68	20.34	12.04	12.04	**12.03***	12.29	12.58
Ranking loss ↓	0.098	0.120	0.213	0.089	0.150	**0.079***	**0.079***	**0.079***	0.081	0.084
AvgPrecision ↑	0.610	0.564	0.590	0.630	0.614	**0.686***	**0.686***	**0.686***	0.677	0.675
# Wins (Ind)	0/11	0/11	6/11	6/11	0/11	–	–	–	–	
# Wins (All)	0/11	0/11	0/11	1/11	0/11	7/11	2/11	6/11	0/11	0/11

Table 7
Comparison of the proposed ensemble method (EML) with the state-of-the-art multi-label classifiers for scene.

	MLkNN	IBLR	RAkEL	CLR	ECC	EML_A	EML_T	EML_S	EML_P	EML_D
Hamming loss ↓	0.092	0.089	0.107	0.113	0.098	**0.080***	0.088	0.088	0.088	0.089
Accuracy ↑	0.644	0.659	0.609	0.600	0.706	**0.728***	0.715	0.715	0.718	0.719
Fmeasure ↑	0.665	0.677	0.639	0.653	0.742	0.756$^+$	0.755$^+$	0.755$^+$	0.757$^+$	**0.758***
ClassAcc ↑	0.604	0.622	0.549	0.500	0.634	**0.673***	0.634	0.634	0.638	0.643
Micro F_1 ↑	0.714	0.723	0.680	0.683	0.729	**0.769***	0.753	0.753	0.754	0.752
Macro F_1 ↑	0.718	0.729	0.689	0.692	0.738	**0.777***	0.655	0.654	0.653	0.649
Micro AUC ↑	0.937	0.941	0.902	0.923	0.917	**0.956***	0.950	0.950	0.950	0.945
Macro AUC ↑	0.927	0.936	0.894	0.917	0.915	**0.950***	0.944	0.944	0.944	0.939
One-error ↓	0.242	0.235	0.286	0.255	0.253	0.206$^+$	0.206$^+$	0.206$^+$	**0.205***	0.208$^+$
Coverage ↓	0.504	0.491	0.645	0.501	0.560	**0.401***	**0.401***	**0.401***	0.403$^+$	0.418
Ranking loss ↓	0.083	0.081	0.112	0.083	0.094	**0.064***	**0.064***	**0.064***	**0.064***	0.067
AvgPrecision ↑	0.856	0.860	0.825	0.850	0.845	**0.881***	**0.881***	**0.881***	**0.881***	0.878
# Wins (Ind)	0/12	7/12	0/12	0/12	5/12	–	–	–	–	
# Wins (All)	3/12	0/12	0/12	0/12	0/12	10/12	3/12	3/12	3/12	1/12

- For yeast, Shepard's dynamic weighting technique gives the best performance in the majority of evaluation measures with as high as 15% increase in Macro F_1 when compared with the best individual multi-label classifier. Overall, EML_P and EML_A rank first in nine and three evaluation measures in this data set, respectively.
- For Pascal07 and medical, the static weighting technique gives the best performance in the majority of evaluation measures resulting in a performance increase of 11% and 9% increase in Macro/Micro F_1 for pascal07, respectively, and 7% increase in classification accuracy for medical when compared with the best individual multi-label classifier. Overall, EML_S ranks first in 8 and 7 evaluation measures for pascal07 and medical respectively. These results are quite interesting in a sense that for some measures such as Accuracy and Fmeasure in pascal07, only one multi-label classifier (ECC) is strong while other multi-label classifiers have quite poor performance. However, the presented ensemble techniques are able to improve over the best individual classifier.

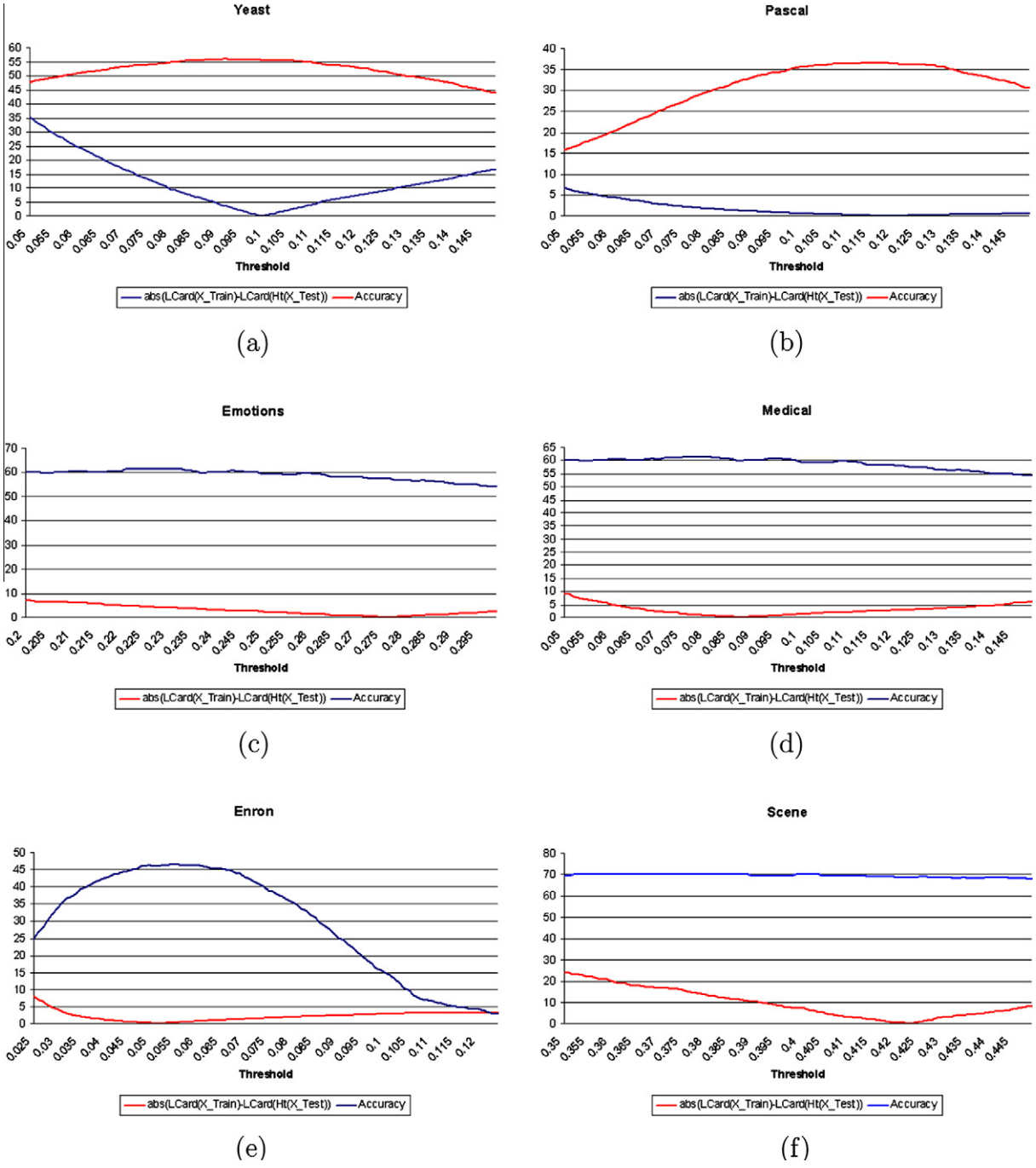

Fig. 1. Threshold t vs $\{|LCard(X_T) - LCard(H_t(X_s))|, \text{Accuracy}\}$.

Table 8
Performance of multi-label classifiers when the threshold is selected using Micro F_1 and 3-Fold Cross Validation.

Micro F_1 ↑	$MLkNN_{Micro}$	$IBLR_{Micro}$	$RAkEL_{Micro}$	CLR_{Micro}	ECC_{Micro}	$EML_{T_{Micro}}$
Emotions	0.670	0.656	0.617	0.686	0.684	**0.703***
Enron	0.521	0.488	0.564	0.521	0.526	**0.597***
Medical	0.680	0.560	0.804$^+$	0.762	0.786	**0.810***
Pascal	0.396	0.398	0.337	0.420	0.414	**0.451***
Scene	0.735	0.746	0.685	0.724	0.726	**0.770***
Yeast	0.661	0.664	0.625	0.645	0.648	**0.677***
# Wins (Ind)	0/6	2/6	2/6	2/6	0/6	–
# Wins (All)	0/6	0/6	0/6	0/6	0/6	6/6

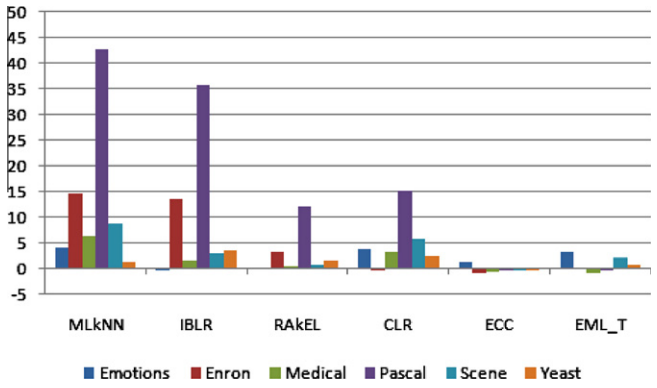

Fig. 2. Performance gains obtained by multi-label classifiers when the threshold is optimized using Micro F_1 and the performance is evaluated using Micro F_1.

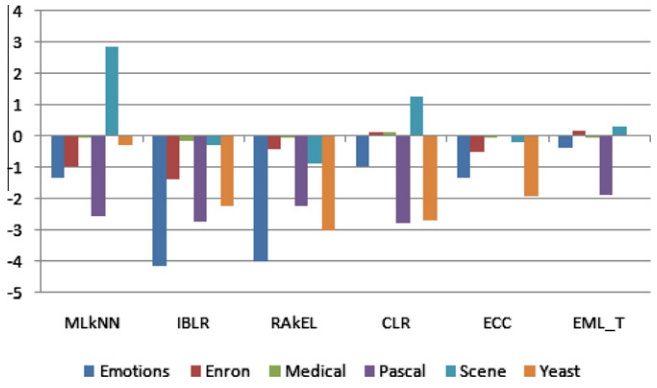

Fig. 3. Difference in performance obtained by multi-label classifiers when the threshold is optimized using Micro F_1 and the performance is evaluated using Hamming loss.

- It is also observed that the performance of Macro F_1 is significantly lower in medical and enron when using the advocated technique. This can be explained by the fact that these data sets contain as few as one example for some categories, and since Macro F_1 gives an equal weight to every category, misclassification of these few samples can significantly drop the performance. The poor performance in these data sets can also be explained by the fact that an ensemble of classifiers can provide higher accuracy than a single best classifier if the member classifiers are diverse and accurate. In case of medical, enron and pascal07, some base classifiers e.g. MLkNN and IBLR have extremely low performance when Macro F_1 is used as an evaluation criterion. These poor classifiers can also overwhelm correct predictions of good classifiers.
- For emotions, scene and enron, the simple average of the ensemble classifiers (EML_A) gives the best performance in the majority of the evaluation measures. Overall, EML_A ranks first in 10, 10 and 7 evaluation measures for emotions, scene and enron respectively. The superiority of the AVG rule especially in emotions and scene can be explained by the fact that the performance of the individual classifiers is more or less similar.

- In order to demonstrate the effectiveness of the threshold selection method discussed in Section 3.2, Fig. 1 shows the graphs for different values of threshold t in the X-axis and two curves in the Y-axis ($|LCard(X_T) - LCard(H_t(X_s))|$, Accuracy) for the various data sets. It should be noted that the accuracy curve is plotted using the actual test labels for the sake of demonstration. For clarity, only a limited range of t is shown as the optimal value lies in this range. For example, for yeast, only values with $t < 0.145$ are shown here as the optimal value lies between $t = 0.05$ and $t = 0.145$. It is clear from these graphs that the threshold selection method is able to deliver a near-optimal value of accuracy. The optimal value of accuracy for yeast is 56.04 under a threshold $t = 0.095$. In contrast, the best value of accuracy obtained by Eq. (2) is 55.70 with $t = 0.099$, which is very close to the optimal one. Similarly, for pascal07, the optimal value of accuracy is 36.65 for threshold $t = 0.117$. In contrast, the best value of accuracy obtained by Eq. (2) is 36.49 under $t = 0.119$, which is again very close to the optimal one. In summary, these graphs clearly show the merit of the presented threshold selection method, as this simple approach attains near-optimal values without expensive internal cross validation.

5.1. Effect of threshold selection using N-Fold Cross Validation on multi-label classifiers

As described previously, out of total 12 evaluation measures, six measures depend on predicted labels and appropriate tuning of thresholds. Fan and Lin (2007) pointed out that the performance can be improved if threshold is optimized using N-Fold Cross Validation of training data and by directly using evaluation criterion such as Micro F_1. In this section, we will evaluate the performance of multi-label classifiers when threshold is optimized using N-Fold CV. Let us assume that the objective is to optimize threshold using Micro F_1. The threshold selection method works as follows: the training set is randomly partitioned into $N = 3$ disjoint subsets of approximately equal size. Next, the cross-validation is performed N different times. In each run, $N - 1$ subsets are used for training and the remaining subset for testing. Once all the confidences are made available, threshold that gives the best Micro F_1 is chosen as the classification threshold. It should be noted that this threshold method is applied on individual multi-label classifier. The same procedure is also applied to Ensemble of Multi-label classifiers $EML_{T_{Micro}}$. $EML_{T_{Micro}}$ is the same as EML_T but the threshold is optimized using Micro F_1 instead of the threshold selection method described by Read et al. (2009).

Table 8 shows the performance of multi-label classifiers while Fig. 2 shows the performance gains when the threshold is

Table 9
Performance of multi-label classifiers using Hamming Loss when the threshold is selected using Micro F_1 and 3-Fold Cross Validation.

Hamming Loss ↓	$MLkNN_{Micro}$	$IBLR_{Micro}$	$RAkEL_{Micro}$	CLR_{Micro}	ECC_{Micro}	$EML_{T_{Micro}}$
Emotions	0.215	0.233	0.261	0.213	0.214	**0.202***
Enron	0.063	0.077	0.055	0.064	0.066	**0.050***
Medical	0.017	0.028	**0.011***	0.013	0.012	0.011$^+$
Pascal	0.091	0.093	0.099	0.102	**0.085***	0.095
Scene	0.097	0.092	0.114	0.102	0.100	**0.085***
Yeast	0.220	0.216	0.252	0.231	0.231	**0.205***
# Wins (Ind)	0/6	2/6	2/6	2/6	0/6	–
# Wins (All)	0/6	0/6	0/6	0/6	1/6	5/6

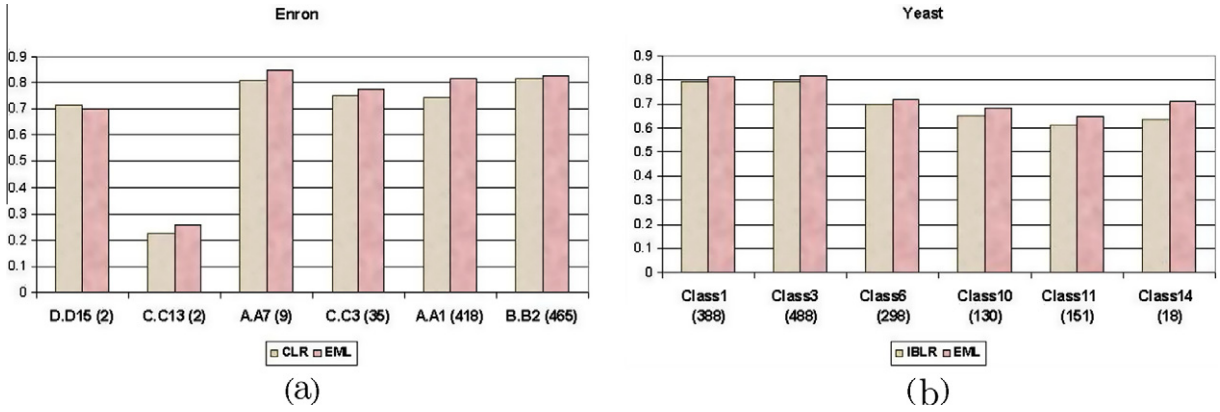

Fig. 4. Performance of individual concepts using AUC. The numbers in bracket shows the total number of samples belong to that concept. (a) Enron. (b) Yeast.

optimised using MicroF_1. As expected, significant performance gains are achieved by individual multi-label classifiers when the threshold is optimised using Micro F_1 for MLkNN$_{Micro}$, IBLR$_{Micro}$, RAkEL$_{Micro}$ and CLR$_{Micro}$. Nevertheless, the proposed ensemble of multi-label classifiers (EML$_{T_{Micro}}$) is still significantly better in all data sets (Table 8). Furthermore, since both ECC and EML$_{T_{Micro}}$, already benefit from the threshold selection method, very limited gains are observed in both ECC and EML$_{T_{Micro}}$ (in fact in some datasets, a slight drop in performance), which again shows the merit of the threshold selection method described in Section 3.2, as this simple approach attains near-optimal values without expensive internal cross validation. However, optimizing the threshold using one measure e.g. Micro F_1 will not in general optimize other evaluation measures as pointed out by Lewis (1995). Fig. 3 supports this argument and clearly indicates a drop in performance in the majority of data sets and multi-label classifiers. But despite that, EML$_{T_{Micro}}$ is still significantly better in all except ECC for pascal data set and RakEL for medical data set (Table 9).

5.2. Discussion

The results presented in this paper show the merit of combining multi-label classifiers to overcome over-fitting and improve the accuracy of individual classifiers. EML$_A$ improves the results consistently for ranking based measures and some label-based measures (Micro/Macro AUC) where performance is evaluated directly using scores/probabilites. When compared with a non-trainable ensemble of multi-label classifiers (EML$_A$), complex EML methods such as static/dynamic weighting have limited or no improvement in measures where predicted sets of labels are not required. For measures that require predicted sets of labels, performance vary among different variants of ensemble multi-label classifiers but it is still consistently significantly better when compared with the individual methods and across the majority of evaluation measures. However, this performance gain is achieved at the expense of inherent computational complexity of ensemble techniques since several multi-label classifiers need to be trained separately. The easiest solution is to use parallel computing techniques to improve the efficiency since all base classifiers can be trained independently.

In order to show the effectiveness of the proposed approach in some highly unbalanced concepts, Fig. 4 shows the performance using the Area Under the ROC Curve (AUC) for some highly unbalanced categories in enron and yeast, respectively. The graph clearly indicates that the presented approach (EML$_A$) has significantly improved the performance in the majority of the highly unbalanced categories. For example, there is an increase of approximatively 3% in performance in categories such as C.C13/A.A7 in Enron. Similarly, there is an increase of up to 8% in highly unbalanced category (Class14) in yeast.

To the best of our knowledge, this is the first study that aims to combine the output of various multi-label classifiers. In this paper, we advocate ensemble techniques employing the nontrainable AVG rule, static/dynamic weighting methods and an automatic threshold selection method. Since, multi-label classifiers inherently are computationally intensive and data is highly imbalanced, it opens new research challenges as how to use other combination techniques efficiently such as trainable combiners (Fuzzy Integral) or class indifferent combiners (Decision Templates and Dempster–Shafer Combination). The other interesting research issue that needs to be investigated is how to select the base classifiers in EML since different combinations of base classifiers may perform differently for specific problem domains.

6. Conclusion

In this paper, heterogeneous ensemble of multi-label learners is proposed to simultaneously tackle both class imbalance and class correlation problems. For multi-label classification, this idea is especially appealing, as ensemble methods are well-known for overcoming over-fitting problems and improving the performance of individual classifiers. Five ensemble techniques have been investigated and then applied to six publicly available multi-label data sets using several evaluation criteria. It has been shown that the presented approach provides a very accurate and efficient solution when compared with the state-of-the-art multi-label methods.

Appendix A

As discussed in Section 4, multi-label classification requires different evaluation measures than those used in traditional single-label classification. In this appendix, we describe in detail the most commonly used multi-label measures. These measures can be categorised into three groups: example based, label-based and ranking-based.

Let X denote a set of images (instances) and let $Y = \{1,2,\ldots,N\}$ be a set of labels. Given a test set $S = \{(x_1,y_1),\ldots,(x_m,y_m)\}$ where $x_i \in X$ is a single instance and $y_i \subseteq Y$ is the label set associated with x_i, the goal is to design a multi-label classifier that predicts a set of labels from an unseen example. Let Z_i be the set of labels predicted by a multilabel classifier for sample x_i. In most cases, the output of the learning system is a function $f : X \times Y \to \mathbb{R}$ which ranks labels according to $f(x_i,\cdot)$ so that the label l_1 is considered to be ranked higher than label l_2 if $f(x_i,l_1) > f(x_i,l_2)$. As in (Schapire and Singer, 2000), the *rank* of a given label l for instance x_i under f is denoted

by $rank_f(x_i,l)$. Formally, $rank_f(x_i,\cdot)$ is a one-to-one mapping onto Y such that if $f(x_i,l_1) > f(x_i,l_2)$, then $rank_f(x_i,l_1) < rank_f(x_i,l_2)$.

A.1. Example-based measures

Hamming loss computes the percentage of labels that are misclassified, i.e. relevant labels that are not predicted or irrelevant labels that are predicted. The *hamming loss* (Schapire and Singer, 2000) is defined as

$$Hamming\ loss = \frac{1}{m}\sum_{i=1}^{m}\frac{|y_i \Delta Z_i|}{|Y|} \tag{A-1}$$

where, Δ stands for the symmetric difference of two sets (XOR in boolean logic). The smaller the value of Hamming Loss, the better is the performance.

F-measure is the harmonic mean between precision (the percentage of predicted labels that are relevant) and recall (the percentage of relevant labels that are predicted) and is defined as

$$FMeasure = \frac{1}{m}\sum_{i=1}^{m}\frac{2|y_i \cap Z_i|}{|Z_i| + |y_i|} \tag{A-2}$$

Accuracy is measured by the Hamming score which symmetrically measures how close Y is to Z (Godbole and Sarawagi, 2004) and is defined as

$$Accuracy = \frac{1}{m}\sum_{i=1}^{m}\frac{|y_i \cap Z_i|}{|y_i \cup Z_i|} \tag{A-3}$$

Classification accuracy is a very strict evaluation measure that requires the actual set of labels to be an exact match of the predicted set of actuals (Tsoumakas et al., 2009) and is defined as

$$Classification\ Accuracy = \frac{1}{m}\sum_{i=1}^{m} I(y_i = Z_i) \tag{A-4}$$

where $I(\text{true}) = 1$ and $I(\text{false}) = 0$.

A.2. Label-based measures

These measures are used to assess the average performance of a binary classifier over multiple categories. In this paper, two different measures for binary evaluation are used, namely the area under the ROC curve (AUC) and F_1 along with two averaging operations, i.e. macro-averaging and micro-averaging (Yang, 1997). These two averaging measures are frequently used in Information Retrieval tasks for all labels. Consider a binary evaluation measure $B(tp,fn,fp,tn)$ that is calculated based on the number of true positives (tp), false negatives (fn), false positives (fp) and true negatives (tn) (Tsoumakas et al., 2009). Let tp_l, fn_l, fp_l and tn_l be the number of true positives, false negatives, false positives and true negatives after the binary evaluation of a label l which are then used to calculate Micro/Macro AUC and F_1. The micro-averaged and macro-averaged versions of B are calculated as follows:

$$B_{macro} = \frac{1}{|Y|}\sum_{i=1}^{|Y|} B(tp_l,fn_l,fp_l,tn_l) \tag{A- 5}$$

$$B_{micro} = B\left(\sum_{i=1}^{|Y|} tp_l, \sum_{i=1}^{|Y|} fn_l, \sum_{i=1}^{|Y|} fp_l, \sum_{i=1}^{|Y|} tn_l\right) \tag{A-6}$$

A.3. Ranking-based measures

One-error measure evaluates how many times the top-ranked label was not in the set of possible labels. For single label classification problems, the one-error is similar to ordinary error. The smaller the value of one-error, the better is the performance. The *one-error* is defined as

$$one-error(f) = \frac{1}{m}\sum_{i=1}^{m} H(x_i) \tag{A-7}$$

where

$$H(x_i) = \begin{cases} 0, & \text{if } argmax_{l\in Y} f(x_i,l) \in y_i \\ 1, & \text{otherwise} \end{cases} \tag{A-8}$$

Coverage measure assesses the performance of a system for all the possible labels of samples. For single-label classification problems, the coverage is the average rank of the correct label and is zero if the system does not make any classification errors. The *coverage* is defined as

$$Coverage = \frac{1}{m}\sum_{i=1}^{m} max_{l\in y_i} rank_f(x_i,l) - 1 \tag{A-9}$$

Ranking Loss evaluates the average fraction of label pairs that are reversely ordered for an instance. Let $\overline{y_i}$ be the complementary set of y_i in Y. The ranking loss is

$$RLoss = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{|y_i||\overline{y_i}|}|R(x_i)| \tag{A-10}$$

where $R(x_i) = \{(l_1,l_2)|f(x_i,l_1) \leqslant f(x_i,l_2),(l_1,l_2) \in y_i \times \overline{y_i}\}$. The smaller the value of ranking loss, the better is the performance.

Average Precision is frequently used in information retrieval systems to evaluate the image ranking performance in query retrieval. Nevertheless, it is used here to measure the effectiveness of the label rankings. In other words, this measure evaluates the average fraction of labels ranked above a particular label $l \in y_i$ which actually are in y_i. The higher the value of average precision, the better is the performance. The *average-precision* is defined as

$$AP_{multi-label} = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{|y_i|}P(x_i) \tag{A-11}$$

where

$$P(x_i) = \sum_{l\in y_i}\frac{|\{l'|rank_f(xi,l') \leqslant rank_f(x_i,l), l' \in y_i\}|}{rank_f(x_i,l)} \tag{A-12}$$

References

Boutell, M.R., Luo, J., Shen, X., Brown, C.M., 2004. Learning multi-label scene classification. Pattern Recognit. 37 (9), 1757–1771.

Chawla, N.V., Sylvester, J.C., 2007. Exploiting diversity in ensembles: Improving the performance on unbalanced datasets. In: Proceedings of Multiple Classifier Systems.

Cheng, W., Hullermeier, E., 2009. Combining instance-based learning and logistic regression for multilabel classification. Mach. Learn. 76 (2–3), 211–225.

Dimou, A., Tsoumakas, G., Mezaris, V., Kompatsiaris, I., Vlahavas, I., 2009. An empirical study of multi-label learning methods for video annotation. In: Proceedings of the 7th International Workshop on CBMI, Chania, Greece.

Dudani, S.A., 1976. The distance weighted k-nearest neighbor rule. IEEE Trans. Syst. Man Cybernat. (6), 325–327.

Elisseeff, A., Weston, J., 2002. A kernel method for multi-labelled classification. In: Advances in NIPS, vol. 14.

Fan, R.E., Lin, C.J., 2007. A study on threshold selection for multi-label classification. Tech. rep., National Taiwan University. <http://www.csie.ntu.edu.tw/cjlin/papers/threshold.pdf>.

Furnkranz, J., Hullermeier, E., Menca, E.L., Brinker, K., 2008. Multilabel classification via calibrated label ranking. Mach. Learn. 23 (2), 133–153.

Godbole, S., Sarawagi, S., 2004. Discriminative methods for multi-labeled classification. Adv. Knowledge Discovery Data Min., 22–30.

Hsu, K.-W., Srivastava, J., 2009. An empirical study of applying ensembles of heterogeneous classifiers on imperfect data. In: PAKDD Workshops'09, pp. 28–39.

Kittler, J., Hatef, M., Duin, R.P.W., Matas, J., 1998. On combining classifiers. IEEE Trans. Pattern Anal. Machine Intell. 20 (3), 226–239.

Kuncheva, L.I., 2004. Combining Pattern Classifiers. Wiley.

Lewis, D.D., 1995. Evaluating and optimizing autonomous text classification systems. In: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '95, pp. 246–254.
Li, T., Ogihara, M., 2006. Toward intelligent music information retrieval. IEEE Trans. Multimedia 8 (3), 564–574.
NLP, 2007. Computational medical center: Medical NLP challenge. <http://www.computationalmedicine.org/challenge/index.php>.
Read, J., Pfahringer, B., Holmes, G., 2008. Multi-label classification using ensembles of pruned sets. In: Proceedings of the ICDM.
Read, J., Pfahringer, B., Holmes, G., Frank, E., 2009. Classifier chains for multi-label classification. In: Proceedings of the ECML.
Schapire, R.E., Singer, Y., 2000. Boostexter: A boosting based system for text categorization. Mach. Learn. 39 (2–3), 135–168.
Shepard, R.N., 1987. Toward a universal law of generalization for psychological science. Science 237, 1317–1323.
Tahir, M.A., Kittler, J., Mikolajczyk, K., Yan, F., 2010. Improving multilabel classification performance by using ensemble of multi-label classifiers. In: MCS'10, pp. 11–21.
Tahir, M.A., Kittler, J., Yan, F., Mikolajczyk, K., 2009. Kernel discriminant analysis using triangular kernel for semantic scene classification. In: Proceedings of the 7th International Workshop on CBMI, IEEE, Crete, Greece.
Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I., 2008. Multilabel classification of music into emotions. In: 9th International Conference on Music Information Retrieval (ISMIR 2008), Philadelphia, PA, USA, pp. 325–330.
Tsoumakas, G., Katakis, I., Vlahavas, I., 2009. Data Mining and Knowledge Discovery Handbook, 2nd ed. Springer, Ch. Mining Multilabel Data.
Tsoumakas, G., Vlahavas, I., 2007. Random k-labelsets: An ensemble method for multilabel classification. In: Proceedings of the ECML, Warsaw, Poland.
Valdovinos, R.M., Sánchez, J.S., 2009. Combining multiple classifiers with dynamic weighted voting. In: 4th International Conference on Hybrid Artificial Intelligence Systems.
van de Sande, K.E.A., Gevers, T., Snoek, C.G.M., 2010. Evaluating color descriptors for object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 32 (9), 1582–1596.
Witten, I.H., Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann.
Yang, Y., 1997. An evaluation of statistical approaches to text categorization. J. Inform. Retrieval 1, 67–88.
Zhang, M.L., Zhou, Z.H., 2006. Multilabel neural networks with applications to functional genomics and text categorization. IEEE Trans. Knowledge Data Eng. 18 (10), 1338–1351.
Zhang, M.L., Zhou, Z.H., 2007. ML-KNN: A lazy learning approach to multi-label learning. Pattern Recognit. 40 (7), 2038–2048.