

## Multi-label Learning

Gjorgji Madjarov<sup>a,b</sup>, Dragi Kocev<sup>b</sup>, Dejan Gjorgjevikj<sup>a</sup>, Sašo Džeroski<sup>b</sup>

<sup>a</sup>Faculty of Electrical Engineering and Information Technology, Ss. Cyril and Methodius University, Skopje, Macedonia

<sup>b</sup>Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia

---

### Abstract

#### Keywords:

multi-label learning, multi-label ranking, multi-label classification

---

### 1. Introduction

The traditional problem of single-label classification is concerned with learning from instance, each associated with a single label  $\lambda_i$  from a finite set of disjoint labels  $\mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$ ,  $Q > 1$ . For  $Q > 2$ , the learning problem is referred to as a *multi-class classification*. On the other hand, the task of learning a mapping from an instance  $\mathbf{x} \in \mathcal{X}$  ( $\mathcal{X}$  denotes the domain of instances) to a set of labels  $\mathcal{Y} \subseteq \mathcal{L}$  is referred to as a *multi-label classification*. Thus, in contrast to multi-class classification, alternatives are not assumed to be mutually exclusive such that multiple labels may be associated with a single instance i.e., each instance can be a member of more than one class. The set of labels  $\mathcal{Y}$  are called relevant, while the set  $\mathcal{L} \setminus \mathcal{Y}$  represents irrelevant labels for a given instance.

*Label ranking* studies the problem of learning a mapping from a set of instances to rankings over a finite number of predefined labels. It can be considered a natural generalization of conventional (multi-class) classification, where instead of requesting only a single label (a top label), a ranking of all the labels is performed.

Besides the concept of multi-label classification, the multi-label learning introduces the concept of *multi-label ranking* [1], which is understood as learning a model that the query instance  $\mathbf{x}$  associates both with a ranking of the complete label set and a bipartite partition of this set into relevant and irrelevant labels.

In recent years, many different approaches have been developed to solve the multi-label learning problems. Tsoumakas and Katakis [2] summarize them into two main categories: a) algorithm adaptation methods, and b) problem transformation methods. Algorithm adaptation methods extend specific learning algorithms to handle multi-label data directly. Examples include lazy learning [3] [4] [5], neural networks [6] [7], boosting [8] [9], classification rules [10], etc. Problem transformation methods, on the other hand, transform the multi-label learning problem into one or more single-label classification problems. The single-label classification problems are solved with a commonly used single-label classification approach and the output is transformed back into a multi-label representation via some reverse process. A common approach for problem transformation is to use class binarization methods, i.e. decomposition of the problem into several binary sub-problems that can then be solved using a binary base classifier. The simplest strategies in the multi-label setting are the one-against-all and one-against-one strategies, also referred to as the binary relevance method [2] and pair-wise method [11] [12] respectively.

The issue of learning from multi-label data has recently attracted significant attention from many researchers. They are motivated from an increasing number of new applications, such as semantic annotation of images and video (news clips, movies clips), functional genomics (gene and protein function), music categorization into emotions, text classification (news articles, web pages, patents, emails, bookmarks, ...), directed marketing and others. In the last

few years, several workshops concerning multi-label learning are being organized in order to present the deep impact of these kind of problems.

The work we present in this paper concerns the learning of multi-labeled data. We do not propose a new algorithm; we do not even present a highly non-trivial way of using an existing algorithm. In this paper we compare the predictive performance and the computational complexities of the state of the art algorithms in multi-label learning and provide useful conclusions about them. The contributions of this paper are:

- .
- Providing an extensive experimental comparison of existing state of the art systems for multi-label learning (classification and ranking) for a wide range of evaluation criteria.
- .

Section 2 defines the tasks of multi-label classification and label ranking and surveys the related previous work. The state of the art methods for multi-label learning compared in this paper are presented in Section 3. Section 4 presents the multi-label problems and the experimental setup while the section 5 shows the experimental results that compare the performance of the competing methods. The conclusions are given in Section 5.

## 2. Background

### 2.1. Task description

**Given:**

- An instance space  $X$  that consists of tuples of values of primitive data types (boolean, discrete or continuous), i.e.,  $\forall \mathbf{x}_i \in X, \mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ , where  $D$  is the size of the tuple (or number of descriptive attributes),
- a label space  $\mathcal{L} = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  which is a tuple of  $Q$  discrete variables (with values 0 or 1),
- a set of examples  $E$ , where each example is a pair of tuples from the instance and label space, respectively, i.e.,  $E = \{(\mathbf{x}_i, \mathcal{Y}_i) | \mathbf{x}_i \in X, \mathcal{Y}_i \in \mathcal{L}, 1 \leq i \leq N\}$  and  $N$  is the number of examples of  $E$  ( $N = |E|$ ), and
- a quality criterion  $q$ , which rewards models with high predictive accuracy and low complexity.

The task of multi-label classification formally is defined as follows:

**Find:** a function  $h: X \rightarrow 2^{\mathcal{L}}$  such that  $h$  maximizes  $q$ .

On the other hand the task of label ranking is defined:

**Find:** a function  $f: X \times \mathcal{L} \rightarrow \mathcal{R}$  such that  $f$  maximizes  $q$ .

### 2.2. Related work

In this section, we will give an overview of different methods for solving multi-label learning problems. These methods can be summarized in three main categories: Algorithm adaptation methods, problem transformation methods and ensemble methods. The later as ensemble techniques use base classifiers that belong to the first two methods.

#### 2.2.1. Algorithm adaptation methods

AdaBoost.MH and AdaBoost.MR [8] are two extensions of AdaBoost for multi-label data. While AdaBoost.MH is designed to minimize Hamming loss, AdaBoost.MR is designed to find a hypothesis which places the correct labels at the top of the ranking. A combination of AdaBoost.MH with an algorithm for producing alternating decision trees has been proposed in [9], with the motivation of producing multi-label models that can be understood by humans.

ML-kNN [3] [4] [5] is based on the popular k Nearest Neighbors (kNN) lazy learning algorithm. The first step of the algorithm proposed in [3] is the same as in kNN, i.e., retrieving the k nearest examples. It uses the maximum a posteriori principle in order to determine the label set of the test example, based on prior and posterior probabilities i.e. the frequency of each label within the k nearest neighbors. A hybrid approach of logistic regression and k-nearest

neighbor was proposed by Cheng et al. [13]. Clare et al. [14] adapted the C4.5 algorithm for multi-label data (ML-C4.5). Other decision tree based methods for multi-label classification are predictive clustering trees (PCT) proposed by Blockeel et al. [15].

Neural networks have also been adapted for multi-label classification [6] [7]. BP-MLL [7] is an adaptation of the popular back-propagation algorithm for multi-label learning. The main modification to the algorithm is the introduction of a new error function that takes multiple labels into account. An SVM approach for multi-label classification is proposed in [16].

### 2.2.2. Problem transformation methods

Problem transformation methods can be grouped in three subcategories: Binary relevance methods, label power-set methods and pair-wise methods.

**Binary relevance methods** An extensive bibliography of learning algorithms for problem transformation methods is given by Tsoumakas and Katakis [2]. The simplest strategy in the multi-label setting is the one-against-all strategy also referred to as the binary relevance method (BR) [2]. A method closely related to the BR method is the Classifier Chain method (CC) proposed by Read et al. [17]. Godbole et al. [18] present algorithms which use existing discriminative classification techniques as building blocks to perform multi-label classification.

**Label power-set methods** Second problem transformation method is the label combination method, or label power-set method, (LP), which has been the focus of several recent studies [19] [20] [2]. The basis of this method is to combine entire label sets into atomic (single) labels to form a single-label problem for which the set of possible single labels represents all distinct label subsets in the original multi-label representation. Each  $(x, Y)$  is transformed into  $(x, l)$  where  $l$  is the atomic label representing a distinct label subset. In this way, LP based methods directly take into account label correlations. To solve the problem of the large number of label combinations, Read [21] developed a pruned problem transformation method (PPT), that selects only the transformed labels that occur more than predefined number of times. Another label power-set method is HOMER [22]. It is a computationally efficient multi-label classification method specifically designed for large multi-label dataset.

**Pair-wise methods** Third problem transformation approach to solving the multi-label learning problem by using binary classifiers is pair-wise classification or round robin classification [11] [12]. Its basic idea is to use  $Q * (Q - 1) / 2$  classifiers covering all pairs of labels. Each classifier is trained using the samples of the first label as positive examples and the samples of the second label as negative examples. To combine these classifiers, the pairwise classification method naturally adopts the majority voting algorithm. Given a test example, each classifier delivers a prediction for one of the two labels. This prediction is decoded into a vote for one of the labels. After the evaluation of all  $Q * (Q - 1) / 2$  classifiers, the labels are ordered according to their sum of votes. To predict only the relevant labels for each example a label ranking algorithm is used.

Brinker et al. [1] propose a conceptually new technique for extending the common pair-wise learning approach to the multi-label scenario named Calibrated Label Ranking (CLR). The key idea of calibrated label ranking is to introduce an artificial (calibration) label  $\lambda_0$ , which will represent the split-point between relevant and irrelevant labels.

Besides majority voting in CLR, Park et al. [23] propose another, more effective voting algorithm named Quick Weighted Voting (QWeighted) for multi-class classification. QWeighted computes the class with the highest accumulated voting mass, while avoiding the evaluation of all possible pairwise classifiers. An adaptation of QWeighted to multi-label learning (QWML) is proposed by Mencia et al. [24].

### 2.2.3. Ensemble methods

Several ensemble approaches have been developed based on the common problem transformation methods. The most well known are the RAKEL system by Tsoumakas et al. [19], ensembles of classifier chains (ECC) [17] and ensemble of pruned sets (EPS) [25]. For  $m$  iterations of the training data, RAKEL draws a random subset of size  $k$  from all labels  $L$  and trains a label power-set classifier using these labels. A simple voting process determines the final classification set. EPS uses pruning to reduce the computational complexity of label power-set methods, and an instance duplication method to reduce error rate as compared to label power-set and other methods. This method proved to be particularly competitive in terms of efficiency. Note that binary methods are occasionally referred to as ensemble methods because they involve multiple binary models. However, none of these models is multi-label itself and therefore we use the term ensemble strictly in the sense of an ensemble of multi-label methods.

### 3. Compared methods

In this section, we briefly introduce the state of the art methods for multi-label learning that are compared in this paper. The predictive performance and the computational complexity of one label power-set method, two binary, two pair-wise, two algorithm adaptation methods and four ensemble methods are compared. One of the ensemble methods is label power-set based, while the other methods are algorithm adaptation based.

#### 3.1. Binary relevance

**Binary Relevance (BR)** [2] is the well known one-against-all strategy. It addresses the multi-label learning problem by learning one classifier for each class, using all the examples labeled with that class as positive examples and all remaining examples as negative examples. At query time, each binary classifier predicts whether its class is relevant for the query example or not, resulting in a set of relevant labels. In the ranking scenario, the labels are ordered according to the probability association of each label from each binary classifier.

The **Chaining method (CC)** [17] involves  $Q$  binary classifiers as in BR. Classifiers are linked along a chain where each classifier deals with the binary relevance problem associated with label  $\lambda_i \in L$ , ( $1 \leq i \leq Q$ ). The feature space of each link in the chain is extended with the 0/1 label associations of all previous links. The ranking and the prediction of the relevant labels in the CC method are the same as in the BR method.

#### 3.2. Pair-wise methods

**Calibrated Label Ranking (CLR)** [23] is a new technique for extending the common pair-wise learning approach to the multi-label scenario. It introduces an artificial (calibration) label  $\lambda_0$ , which represents the split-point between relevant and irrelevant labels. The calibration label  $\lambda_0$  is assumed to be preferred over all irrelevant labels, but all relevant labels are preferred over it. It is represented by the binary relevance classifiers, that are introduced as pair-wise classifiers in the context of pair-wise learning. At prediction time (majority voting is usually used), one will get a ranking over  $Q + 1$  labels (the  $Q$  original labels plus the calibration label  $\lambda_0$ ). CLR is considered a combination of multi-label classification and ranking.

**Quick Weighted voting method (QWML)** proposed by Park et al. [23] is a variant of the CLR method that introduces more effective voting strategy than the majority voting used by the CLR method. Quick Weighted Voting exploits the fact that during voting some classes can be excluded from the set of possible top rank classes early in the process, when it becomes clear that even if they reach the maximal voting mass in the remaining evaluations they can not exceed the current maximum. Pairwise classifiers are selected depending on a voting loss value, which is the number of votes that a class has not received. The voting loss starts with a value of zero and increases monotonically with the number of performed preference evaluations. The class with the current minimal loss is the best candidate for the top ranked class. If all preferences involving this class have been evaluated (and it still has the lowest loss), it can be concluded that no other class can achieve a better ranking. Thus, the QWeighted algorithm always focuses on classes with low voting loss. The adaptation of QWeighted to multi-label learning (QWML) [24] is to repeat the process while all relevant labels are not determined, i.e., until the returned label is the artificial label, which means that all remaining labels will be considered to be irrelevant.

#### 3.3. Label power-set method

**Hierarchy Of Multi-label classifiERs (HOMER)** [22] is a novel algorithm for effective and computationally efficient multi-label learning in domains with a large number of labels. HOMER constructs a hierarchy of multi-label classifiers, each one dealing with a much smaller set of labels compared to  $Q$  (the total number of labels) and a more balanced example distribution. This leads to improved predictive performance and also to linear training and logarithmic testing complexities with respect to  $Q$ . One of the main processes within HOMER is the even distribution of a set of labels into  $k$  disjoint subsets so that similar labels are placed together and dissimilar apart. The best predictive performance is reported utilizing the balanced  $k$  means algorithm proposed by the author. The authors showed that HOMER (using binary relevance as the multi-label classifier at each node) outperforms the BR method in terms of predictive performance and computational complexity. In both methods (HOMER and BR), Naive Bayes is used as a base classifier for the decomposed binary tasks.

### 3.4. Algorithm adaptation methods

**Multi-Label C4.5 (ML-C4.5)** [14] is an adaptation of the well known C4.5 algorithm for multi-label learning. Clare et al. modified the formula of entropy calculation (equation 1) in order to solve multi-label problems. They also allowed multiple labels in the leaves of the tree. The modified entropy sums the entropies for each individual class label.

$$entropy(S) = - \sum_{i=1}^N (p(c_i) \log p(c_i) + q(c_i) \log q(c_i)) \quad (1)$$

where  $S$  is the set of examples,  $p(c_i)$  is the relative frequency of class label  $c_i$  and  $q(c_i) = 1 - p(c_i)$ .

The key property of ML-C4.5 is its computational efficiency. It is among the fastest and the most computationally efficient multi-label classifiers available today. On the other hand, SVMs are among the most powerful classifiers, widely used in classification and regression problems.

**Multi-Label k-Nearest Neighbors (ML-kNN)** [3] is derived from the popular k-Nearest Neighbor (kNN) algorithm. Firstly, for each test instance, its  $k$  nearest neighbors in the training set are identified. Then, according to statistical information gained from the label sets of these neighboring instances, i.e. the number of neighboring instances belonging to each possible class, maximum a posteriori principle is utilized to determine the label set for the test instance.

### 3.5. Ensemble methods

The **Random k-labelsets (RAkEL)** [19] is an ensemble method for multilabel classification. It constructs each member of the ensemble by considering a small random subset of labels and learning a single-label classifier for the prediction of each element in the powerset of this subset. In this way, the proposed algorithm aims to take into account label correlations using single-label classifiers that are applied on subtasks with manageable number of labels and adequate number of examples per label.

**Ensembles of Classifier Chains (ECC)** [17] are ensemble multi-label classification technique that uses CC as a base classifier. ECC trains  $m$  CC classifiers  $C_1, C_2, \dots, C_m$ . Each  $C_k$  is trained with:

- a random chain ordering (of  $\mathcal{L}$ )
- a random subset of  $\mathcal{X}$ .

Hence each  $C_k$  model is likely to be unique and able to give different multi-label predictions. These predictions are summed by label so that each label receives a number of votes. A threshold is used to select the most popular labels which form the final predicted multi-label set.

**Random Forest of ML-C4.5 (RFML-C4.5)** A random forest [26] is an ensemble of trees, where diversity among the predictors is obtained by using bagging, and additionally by changing the feature set during learning. More precisely, at each node in the decision trees, a random subset of the input attributes is taken, and the best feature is selected from this subset. The number of attributes that are retained is given by a function  $f$  of the total number of input attributes  $x$  (e.g.,  $f(x) = 1$ ,  $f(x) = \sqrt{x}$ ,  $f(x) = \lfloor \log_2(x) + 1 \rfloor \dots$ ). The first random forest ensemble method that was included in the comparison is the RFML-C4.5 method (ML-C4.5 is used as a base classifier).

**Random Forest of Predictive Clustering Trees (RF-PCT)** In the predictive clustering trees (PCT) framework [15], a decision tree is seen as a hierarchy of clusters: the top-node corresponds to one cluster containing all data, which is recursively partitioned into smaller clusters while moving down the tree. PCTs can be constructed with a standard top-down induction of decision trees algorithm. The main difference with standard tree learners is that PCTs use a generalized notion of variance to guide the tree construction, and a generalized notion of prototype to determine the value or model stored in the leaf of a tree. Both variance and prototype can be instantiated according to the task at hand. The trees can also easily be combined into ensembles, just like regular trees [27]. The last method that was included in this general comparison of the state of the art multi-label methods is the random forest ensemble method of predictive clustering trees (RF-PCT) [27].

## 4. Experiments

In this section, we present the predictive performance and the computational complexities of the methods explained in the previous section on a number of multi-label classification problems. The problems come from the areas of classification of text, music, images and gene function. Also, we briefly introduce the evaluation measures used in the experiments.

### 4.1. Evaluation measures

Performance evaluation for multi-label learning systems differs from that of classical single-label learning systems. In any multi-label experiment, it is essential to include multiple and contrasting measures because of the additional degrees of freedom that the multi-label setting introduces. In our experiments we used various evaluation measures that have been proposed in [28]. They are grouped in two separate groups: *bipartitions-based* and *rankings-based* with respect to the ground truth of multi-label data. Some of the measures that evaluate bipartitions called *example-based* are calculated based on the average differences of the actual and the predicted sets of labels over all examples of the evaluation data set. Others, called *label-based* evaluation measures decompose the evaluation process into separate evaluations for each label, which they subsequently average over all labels. In our experiments we used six example-based evaluation measures (Hamming loss, accuracy, precision, recall, F1 and subset accuracy), six label-based (micro precision, micro recall, micro F1, macro precision, macro recall and macro F1) and four ranking-based measures (one-error, coverage, ranking loss and average precision). Note that most models predict a numerical value for each label; the label is predicted to be present if that value exceeds some threshold  $t$ . For the example-based and label-based evaluation measures, the performance of the model directly depends on  $t$ . To compare the methods according to the threshold-dependent measures (example and label based measures) we applied a threshold calibration method by choosing the threshold that minimizes the difference in label cardinality between the training data and the predictions for the test data [17].

### 4.2. Datasets

Eleven different multi-label classification problems were addressed in our experiments. The predictive performance in terms of the metrics defined above and the training and testing times were recorded for every method for each classification problem. The problems considered in the experiments include:

1. image classification: scene [29] and corel5k [30];
2. gene function classification: yeast [16];
3. text classification: enron [31], medical <sup>1</sup>, bibtex [32], delicious [22], bookmarks [32] and tmc2007 [33];
4. music classification: emotions [34];
5. video classification: mediamill [35]

The complete description of the datasets in terms of the number of training ( $\#tr.e.$ ) and test ( $\#t.e.$ ) examples, the number of features ( $D$ ), the total number of labels ( $Q$ ) and label cardinality ( $l_c$ ) [2] are shown in Table 1.

We strived to include a considerable variety and scale of multi-label datasets. In total we use eleven datasets, with dimensions ranging from 6 to 983 labels, and from less than 1,000 examples to almost 80,000. The datasets are roughly ordered by complexity ( $\#tr.e. \times D \times Q$ ).

### 4.3. Experimental setup

The training and the testing of the compared methods was performed using the MULAN <sup>2</sup> library under the machine learning framework Weka [36] except for the CC, ECC and RF-PCT methods. For the first two methods we used the MEKA <sup>3</sup> extension for the WEKA framework, while the third method (RF-PCT) was evaluated under the predictive clustering framework CLUS <sup>4</sup>. All experiments were performed on a server with an Intel Xeon processor at 2.50GHz on 64GB of RAM with the Fedora 14 operating system.

<sup>1</sup><http://www.cs.waikato.ac.nz/~jmr30/>

<sup>2</sup><http://mulan.sourceforge.net/>

<sup>3</sup><http://meka.sourceforge.net/>

<sup>4</sup><http://dtai.cs.kuleuven.be/clus/>

Table 1: Dataset description.

	#tr.e.	#t.e.	$D$	$Q$	$l_c$
<b>emotions</b>	391	202	72	6	1.87
<b>scene</b>	1211	1159	294	6	1.07
<b>yeast</b>	1500	917	103	14	4.24
<b>medical</b>	645	333	1449	45	1.25
<b>enron</b>	1123	579	1001	53	3.38
<b>corel5k</b>	4500	500	499	374	3.52
<b>tmc2007</b>	21519	7077	500	22	2.16
<b>mediamill</b>	30993	12914	120	101	4.38
<b>bibtex</b>	4880	2515	1836	159	2.40
<b>delicious</b>	12920	3185	500	983	19.02
<b>bookmarks</b>	60000	27856	2150	208	2.03

#### 4.3.1. Base classifiers

The LIBSVM library [37], and in particular SVMs with a radial basis kernel, were used for solving the partial binary classification problems for all datasets in all problem transformation methods and the ensemble methods ECC and RAKEL. The kernel parameter  $\gamma$  and the penalty  $C$  for the datasets for each method were determined by 10-fold cross validation using only the training sets. The exception to this is the ensemble method RAKEL where the kernel parameter  $\gamma$  and the penalty  $C$  were determined by 5-fold cross validation for the tmc2007 and mediamill datasets because of its computational complexity. The values  $2^{-15}, 2^{-13}, \dots, 2^1, 2^3$  were considered for  $\gamma$  and  $2^{-5}, 2^{-3}, \dots, 2^{13}, 2^{15}$  for the penalty  $C$ . After determining the best parameters values for each method on every dataset the classifiers were trained using all available training examples and were evaluated by recognizing all test examples from the corresponding dataset.

#### 4.3.2. Methods parameters

For the ensemble methods based on decision trees (RFML-C4.5 and RF-PCT), the number of models (classifiers) used in the ensemble was 100. This value is proposed in the literature closed to ensembles of decision trees. The number of models in the ECC method was set to 10 as proposed by the original authors. On the other hand, the number of models in RAKEL was set to  $\min(2 * Q, 100)$  ( $Q$  is the number of labels) for all datasets, except for the mediamill, delicious and bookmarks datasets, where this parameter was 10 as a result of the memory constraints.

The RAKEL method requires one additional parameter over the ensemble size: the size of the labelsets  $k$ . For each dataset, this parameter was set to half of the number of labels ( $Q/2$ ). Previous work has shown this to be a reasonable choice, since it provides a balance between computational complexity and predictive performance [19][17].

The ML-C4.5 method uses subtree raising with a pruning confidence of 0.25 as a post pruning strategy in all classification problems. The minimal number of examples in the leaves in each model of the RFML-C4.5 was set to 10. The number of neighbors in the ML-kNN method for each dataset was determined from the values 6 to 20 with step 2 for which the best results were obtained. HOMER also requires one additional parameter to be configured: number of clusters. For this parameter five different values (2-6) were considered in the experiments. These values were used by the original authors [22]. The best obtained results are presented in the Results subsection.

## 5. Results

Tables 1 to 18 give the performance of each method on each of the datasets measured in terms of the sixteen performance measures, training and testing speed. The first column of the tables lists the method, while the remaining columns show the performance of each method for every dataset. Tables 1 to 6 show the predictive performance in terms of the example based measures, Tables 7 to 12 in terms of the label bases measures, while the ranking performance are presented in Tables 13 to 16. The training and testing times of each method on each of the datasets

measured in seconds, are given in Tables 17 and 18. The best results per dataset in these tables are shown in boldface. DNF in the result tables indicates that the experiment Did Not Finish within one week under the available resources.

To assess whether the overall differences in performance across the ten different approaches are statistically significant, we employed the corrected Friedman test [38] and the post hoc Nemenyi test [39] as recommended by Demsar [40]. The Friedman test is a non-parametric test for multiple hypotheses testing. It ranks the algorithms according to their performance for each dataset separately, thus the best performing algorithm gets the rank of 1, the second best the rank of 2, etc. In case of ties, it assigns average ranks. Then, the Friedman test compares the average ranks of the algorithms and calculates the Friedman statistic  $\chi_F^2$ , distributed according to the  $\chi_F^2$  distribution with  $k - 1$  degrees of freedom ( $k$  being the number of algorithms). Iman and Davenport [41] show that the Friedman statistic is undesirably conservative and derive a corrected F-statistic that is distributed according to the F-distribution with  $k - 1$  and  $(k - 1) \times (N - 1)$  degrees of freedom ( $k$  being the number of algorithms and  $N$  being the number of datasets).

If a statistically significant difference in the performance is detected, then we can proceed with a post hoc test. The Nemenyi test is used to compare all the classifiers to each other. In this procedure, the performance of two classifiers is significantly different if their average ranks differ more than some critical distance. The critical distance depends on the number of algorithms, the number of datasets and the critical value (for a given significance level -  $p$ ) that is based on the Studentized range statistic and can be found in statistical textbooks.

We present the result from the Nemenyi post hoc test with average rank diagrams as suggested by Demsar [40]. These are given on Figures 6 to 6. The ranks are depicted on the axis, in such a manner that the best ranking algorithms are at the right-most side of the diagram. The algorithms that do not differ significantly (at the significance level of  $p = 0.05$ ) are connected with a line.

### 5.1. Results on the example based measures

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
<b>emotions</b>	0.257	0.256	0.257	0.254	0.361	0.247	0.267	0.294	0.282	0.281	0.198	<b>0.189</b>
<b>scene</b>	0.079	0.082	0.080	0.081	0.082	0.141	0.129	0.099	<b>0.077</b>	0.085	0.116	0.094
<b>yeast</b>	<b>0.190</b>	0.193	<b>0.190</b>	0.191	0.207	0.234	0.219	0.198	0.192	0.207	0.205	0.197
<b>medical</b>	0.077	0.077	0.017	<b>0.012</b>	<b>0.012</b>	0.013	0.023	0.017	<b>0.012</b>	0.014	0.022	0.014
<b>enron</b>	<b>0.045</b>	0.064	0.048	0.048	0.051	0.053	0.058	0.051	<b>0.045</b>	0.049	0.047	0.046
<b>corel5k</b>	0.017	0.017	0.012	0.012	0.012	0.010	0.009	0.009	0.009	0.009	0.009	<b>0.007</b>
<b>tmc2007</b>	0.013	0.013	0.014	0.014	0.015	0.093	0.075	0.058	0.021	0.026	0.037	<b>0.011</b>
<b>mediamill</b>	0.032	0.032	0.043	0.043	0.038	0.044	0.034	0.031	0.035	0.035	0.030	<b>0.029</b>
<b>bibtex</b>	<b>0.012</b>	<b>0.012</b>	<b>0.012</b>	<b>0.012</b>	0.014	0.016	0.014	0.014	DNF	0.013	0.014	0.013
<b>delicious</b>	<b>0.018</b>	<b>0.018</b>	DNF	DNF	0.022	0.019	0.019	<b>0.018</b>	DNF	DNF	<b>0.018</b>	<b>0.018</b>
<b>bookmarks</b>	DNF	DNF	DNF	DNF	DNF	<b>0.009</b>	<b>0.009</b>	<b>0.009</b>	DNF	DNF	<b>0.009</b>	<b>0.009</b>

Figure 1: The performance of the multi-label learning approaches in terms of the Hamming loss measure

### 5.2. Results on the label based measures

### 5.3. Results on the ranking based measures

### 5.4. Training and testing times

## 6. Conclusions

## References

- [1] K. Brinker, J. Fürnkranz, E. Hüllermeier, A unified model for multilabel classification and ranking, in: Proceeding of the 2006 conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29 – September 1, 2006, Riva del Garda, Italy, IOS Press,



	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
<b>emotions</b>	0.361	0.356	0.361	0.373	0.471	<b>0.536</b>	0.448	0.319	0.419	0.432	0.488	0.519
<b>scene</b>	0.689	0.723	0.686	0.683	0.717	0.569	0.538	0.629	0.734	<b>0.735</b>	0.388	0.541
<b>yeast</b>	0.520	0.527	0.524	0.523	<b>0.559</b>	0.480	0.440	0.492	0.531	0.546	0.453	0.478
<b>medical</b>	0.206	0.211	0.656	0.658	0.713	<b>0.730</b>	0.228	0.528	0.673	0.611	0.250	0.591
<b>enron</b>	0.446	0.334	0.459	0.388	<b>0.478</b>	0.418	0.196	0.319	0.428	0.462	0.374	0.416
<b>corel5k</b>	0.030	0.030	0.195	0.195	0.179	0.002	0.000	0.014	0.000	0.001	0.005	<b>0.215</b>
<b>tmc2007</b>	0.891	0.899	0.889	0.889	0.888	0.110	0.436	0.574	0.852	0.808	0.663	<b>0.914</b>
<b>mediamill</b>	0.403	0.390	0.095	0.095	0.413	0.052	0.354	0.421	0.337	0.349	0.423	<b>0.441</b>
<b>bibtex</b>	0.348	<b>0.352</b>	0.334	0.338	0.330	0.108	0.046	0.129	DNF	0.186	0.060	0.166
<b>delicious</b>	0.136	0.137	DNF	DNF	<b>0.207</b>	0.001	0.001	0.102	DNF	DNF	0.151	0.146
<b>bookmarks</b>	DNF	DNF	DNF	DNF	DNF	<b>0.237</b>	0.133	0.202	DNF	DNF	0.176	0.204

Figure 2: The performance of the multi-label learning approaches in terms of the accuracy measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
<b>emotions</b>	0.550	0.551	0.538	0.548	0.509	0.606	0.577	0.502	0.564	0.580	0.625	<b>0.644</b>
<b>scene</b>	0.718	0.758	0.714	0.711	0.746	0.592	0.565	0.661	0.768	<b>0.770</b>	0.403	0.565
<b>yeast</b>	0.722	0.727	0.719	0.718	0.663	0.620	0.705	0.732	0.715	0.667	0.738	<b>0.744</b>
<b>medical</b>	0.211	0.217	0.695	0.697	0.762	<b>0.797</b>	0.285	0.575	0.730	0.662	0.284	0.635
<b>enron</b>	0.703	0.464	0.650	0.624	0.616	0.623	0.415	0.587	0.708	0.652	0.690	<b>0.709</b>
<b>corel5k</b>	0.042	0.042	0.329	0.326	0.317	0.005	0.000	0.035	0.000	0.002	0.018	<b>0.514</b>
<b>tmc2007</b>	0.941	0.944	0.937	0.937	0.926	0.146	0.659	0.738	0.928	0.872	0.874	<b>0.977</b>
<b>mediamill</b>	0.731	0.741	0.201	0.203	0.597	0.056	0.694	0.724	0.705	0.690	0.765	<b>0.772</b>
<b>bibtex</b>	<b>0.515</b>	0.508	0.488	0.496	0.472	0.123	0.140	0.254	DNF	0.324	0.159	0.292
<b>delicious</b>	0.443	0.399	DNF	DNF	0.369	0.001	0.001	0.424	DNF	DNF	0.472	<b>0.512</b>
<b>bookmarks</b>	DNF	DNF	DNF	DNF	DNF	<b>0.271</b>	0.133	0.218	DNF	DNF	0.182	0.218

Figure 3: The performance of the multi-label learning approaches in terms of the precision measure

Amsterdam, The Netherlands, The Netherlands, 2006, pp. 489–493.

URL <http://portal.acm.org/citation.cfm?id=1567016.1567123>

- [2] G. Tsoumakas, I. Katakis, Multi Label Classification: An Overview, *International Journal of Data Warehouse and Mining* 3 (3) (2007) 1–13.
- [3] M. L. Zhang, Z. H. Zhou, MI-knn: A lazy learning approach to multi-label learning, *Pattern Recognition* 40 (7) (2007) 2038–2048.
- [4] A. Wiczorkowska, P. Synak, Z. Ras, Multi-label classification of emotions in music, in: M. Klopotek, S. Wierzchon, K. Trojanowski (Eds.), *Intelligent Information Processing and Web Mining*, Vol. 35 of *Advances in Intelligent and Soft Computing*, Springer Berlin / Heidelberg, 2006, pp. 307–315.
- [5] E. Spyromitros, G. Tsoumakas, I. Vlahavas, An empirical study of lazy multilabel classification algorithms, in: *Proceedings of the 5th Hellenic conference on Artificial Intelligence: Theories, Models and Applications*, SETN '08, Springer-Verlag, Berlin, Heidelberg, 2008, pp.

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	0.409	0.397	0.410	0.429	<b>0.775</b>	0.703	0.534	0.377	0.491	0.533	0.545	0.582
scene	0.711	0.726	0.712	0.709	0.744	0.582	0.539	0.655	0.740	<b>0.771</b>	0.388	0.541
yeast	0.591	0.600	0.601	0.600	<b>0.714</b>	0.608	0.490	0.549	0.615	0.673	0.491	0.523
medical	0.735	0.754	0.795	<b>0.801</b>	0.760	0.740	0.227	0.547	0.679	0.642	0.251	0.599
enron	0.497	0.507	0.557	0.453	<b>0.610</b>	0.487	0.229	0.358	0.469	0.560	0.398	0.452
corel5k	0.055	0.056	<b>0.264</b>	0.264	0.250	0.002	0.000	0.014	0.000	0.001	0.005	0.215
tmc2007	0.928	0.934	0.929	0.929	<b>0.943</b>	0.111	0.478	0.664	0.880	0.903	0.677	0.920
mediamill	0.450	0.424	0.101	0.101	<b>0.563</b>	0.052	0.379	0.470	0.353	0.372	0.456	0.476
bibtex	0.373	0.378	0.364	0.366	<b>0.389</b>	0.111	0.046	0.132	DNF	0.187	0.060	0.167
delicious	0.155	0.157	DNF	DNF	<b>0.303</b>	0.001	0.001	0.112	DNF	DNF	0.176	0.160
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.244</b>	0.137	0.207	DNF	DNF	0.181	0.208

Figure 4: The performance of the multi-label learning approaches in terms of the recall measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	0.469	0.461	0.465	0.481	0.614	<b>0.651</b>	0.554	0.431	0.525	0.556	0.583	0.611
scene	0.714	0.742	0.713	0.710	0.745	0.587	0.551	0.658	0.754	<b>0.771</b>	0.395	0.553
yeast	0.650	0.657	0.655	0.654	<b>0.687</b>	0.614	0.578	0.628	0.661	0.670	0.589	0.614
medical	0.328	0.337	0.742	0.745	0.761	<b>0.768</b>	0.253	0.560	0.704	0.652	0.267	0.616
enron	0.582	0.484	0.600	0.525	<b>0.613</b>	0.546	0.295	0.445	0.564	0.602	0.505	0.552
corel5k	0.047	0.048	0.293	0.292	0.280	0.003	0.000	0.021	0.000	0.001	0.008	<b>0.303</b>
tmc2007	0.934	0.939	0.933	0.933	0.934	0.126	0.554	0.699	0.904	0.887	0.763	<b>0.948</b>
mediamill	0.557	0.539	0.134	0.135	0.579	0.054	0.490	0.570	0.471	0.483	0.572	<b>0.589</b>
bibtex	0.433	<b>0.434</b>	0.417	0.421	0.426	0.117	0.069	0.174	DNF	0.237	0.087	0.212
delicious	0.230	0.225	DNF	DNF	<b>0.343</b>	0.001	0.001	0.017	DNF	DNF	0.256	0.244
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.257</b>	0.135	0.213	DNF	DNF	0.181	0.213

Figure 5: The performance of the multi-label learning approaches in terms of the F1 measure

401–406.

- [6] K. Crammer, Y. Singer, A family of additive online algorithms for category ranking, *J. Mach. Learn. Res.* 3 (2003) 1025–1058.
- [7] M. L. Zhang, Z. H. Zhou, Multi-label neural networks with applications to functional genomics and text categorization, *IEEE Transactions on Knowledge and Data Engineering* 18 (10) (2006) 1338–1351.
- [8] R. E. Schapire, Y. Singer, Boostexter: A boosting-based system for text categorization, *Machine Learning* 39 (2000) 135–168.
- [9] F. De Comit , R. Gilleron, M. Tommasi, Learning multi-label alternating decision trees from texts and data, in: *Proceedings of the 3rd international conference on Machine learning and data mining in pattern recognition, MLDM'03*, Springer-Verlag, Berlin, Heidelberg, 2003, pp. 35–49.
- [10] F. A. Thabtah, P. Cowling, Y. Peng, MMAC: A New Multi-class, Multi-label Associative Classification Approach, in: *Proceedings of the 4th*

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	0.129	0.124	0.144	0.149	0.163	0.277	0.223	0.084	0.208	0.168	0.272	<b>0.307</b>
scene	0.639	0.685	0.633	0.630	0.661	0.533	0.509	0.573	<b>0.694</b>	0.665	0.372	0.518
yeast	0.190	<b>0.239</b>	0.195	0.192	0.213	0.158	0.152	0.159	0.201	0.215	0.129	0.152
medical	0.000	0.000	0.486	0.480	0.610	<b>0.646</b>	0.177	0.462	0.607	0.526	0.216	0.538
enron	<b>0.149</b>	0.000	0.117	0.097	0.145	0.140	0.002	0.062	0.136	0.131	0.124	0.131
corel5k	0.000	0.000	0.010	0.012	0.002	0.000	0.000	0.000	0.000	0.001	0.008	<b>0.303</b>
tmc2007	0.772	0.787	0.767	0.768	0.765	0.078	0.215	0.305	0.734	0.608	0.421	<b>0.816</b>
mediamill	0.080	0.080	0.044	0.044	0.053	0.049	0.065	0.110	0.060	0.065	0.104	<b>0.122</b>
bibtex	0.194	<b>0.202</b>	0.183	0.186	0.165	0.095	0.004	0.056	DNF	0.109	0.011	0.098
delicious	0.004	0.006	DNF	DNF	0.001	0.001	0.001	0.003	DNF	DNF	<b>0.018</b>	0.007
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.209</b>	0.129	0.187	DNF	DNF	0.167	0.189

Figure 6: The performance of the multi-label learning approaches in terms of the subset accuracy measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	0.684	0.698	0.685	0.680	0.471	0.607	0.607	0.584	0.586	0.579	<b>0.783</b>	<b>0.783</b>
scene	0.843	0.814	0.835	0.832	0.804	0.619	0.512	0.691	0.831	0.773	<b>0.960</b>	0.930
yeast	0.733	0.726	0.729	0.727	0.647	0.618	0.698	0.736	0.720	0.662	0.747	<b>0.755</b>
medical	0.225	0.229	0.669	0.667	0.807	0.796	0.826	0.807	0.881	0.834	0.884	<b>0.885</b>
enron	0.721	0.492	0.652	0.687	0.597	0.613	0.601	0.684	0.743	0.642	<b>0.768</b>	0.738
corel5k	0.061	0.061	0.338	0.339	0.308	0.160	0.000	0.730	0.000	0.333	0.750	<b>1.000</b>
tmc2007	0.947	0.948	0.940	0.941	0.922	0.940	0.689	0.757	0.938	0.869	0.963	<b>0.992</b>
mediamill	0.742	0.753	0.582	0.580	0.569	0.597	0.743	0.739	0.725	0.708	0.788	<b>0.798</b>
bibtex	0.753	0.744	0.734	0.736	0.547	0.359	<b>1.000</b>	0.819	DNF	0.948	0.940	0.957
delicious	0.658	0.660	DNF	DNF	0.396	0.000	0.000	0.651	DNF	DNF	0.589	<b>0.695</b>
bookmarks	DNF	DNF	DNF	DNF	DNF	0.632	<b>0.947</b>	0.850	DNF	DNF	0.878	0.895

Figure 7: The performance of the multi-label learning approaches in terms of the micro precision measure

IEEE International Conference on Data Mining, ICDM '04, 2004, pp. 217–224.

[11] J. Fürnkranz, Round robin classification, *J. Mach. Learn. Res.* 2 (2002) 721–747.

[12] T.-F. Wu, C.-J. Lin, R. C. Weng, Probability estimates for multi-class classification by pairwise coupling, *J. Mach. Learn. Res.* 5 (2004) 975–1005.

[13] W. Cheng, E. Hullermeier, Combining instance-based learning and logistic regression for multilabel classification, *Machine Learning* 76 (2009) 211–225.

[14] A. Clare, R. D. King, Knowledge Discovery in Multi-label Phenotype Data, *Lecture Notes in Computer Science* 2168.

[15] H. Blockeel, L. D. Raedt, J. Ramon, Top-down induction of clustering trees, in: *Proceedings of the Fifteenth International Conference on Machine Learning, ICML '98*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1998, pp. 55–63.

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.406	0.393	0.409	0.431	<b>0.782</b>	0.712	0.539	0.376	0.489	0.531	0.551	0.589
scene	0.694	0.708	0.695	0.692	0.727	0.570	0.521	0.634	0.721	<b>0.751</b>	0.572	0.523
yeast	0.587	0.588	0.595	0.595	<b>0.702</b>	0.603	0.492	0.543	0.602	0.655	0.491	0.521
medical	0.725	0.739	0.782	<b>0.787</b>	0.742	0.720	0.227	0.522	0.600	0.624	0.237	0.569
enron	0.464	0.472	0.532	0.438	<b>0.585</b>	0.440	0.246	0.353	0.435	0.532	0.366	0.422
corel5k	0.057	0.057	<b>0.258</b>	<b>0.258</b>	0.248	0.002	0.000	0.015	0.000	0.001	0.005	0.219
tmc2007	0.917	0.924	0.920	0.920	<b>0.932</b>	0.073	0.454	0.621	0.847	0.869	0.651	0.902
mediamill	0.415	0.385	0.066	0.066	<b>0.537</b>	0.004	0.351	0.432	0.315	0.333	0.418	0.435
bibtex	0.328	0.335	0.322	0.328	<b>0.353</b>	0.053	0.057	0.118	DNF	0.142	0.066	0.131
delicious	0.143	0.144	DNF	DNF	<b>0.297</b>	0.000	0.000	0.101	DNF	DNF	0.174	0.151
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.170</b>	0.135	0.135	DNF	DNF	0.112	0.136

Figure 8: The performance of the multi-label learning approaches in terms of the micro recall measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.509	0.503	0.512	0.528	0.588	0.655	0.571	0.457	0.533	0.554	0.647	0.672
scene	0.761	0.757	0.758	0.756	0.764	0.593	0.516	0.661	<b>0.772</b>	0.762	0.717	0.669
yeast	0.652	0.650	0.655	0.654	<b>0.673</b>	0.610	0.577	0.625	0.656	0.658	0.593	0.617
medical	0.343	0.350	0.721	0.722	<b>0.773</b>	0.756	0.356	0.634	0.714	0.714	0.374	0.693
enron	0.564	0.482	0.585	0.535	<b>0.591</b>	0.512	0.349	0.466	0.548	0.582	0.496	0.537
corel5k	0.059	0.059	0.293	0.293	0.275	0.004	0.000	0.030	0.000	0.002	0.010	<b>0.359</b>
tmc2007	0.932	0.936	0.930	0.930	0.927	0.135	0.547	0.682	0.890	0.869	0.777	<b>0.945</b>
mediamill	0.533	0.509	0.118	0.119	0.553	0.007	0.477	0.545	0.440	0.453	0.546	<b>0.563</b>
bibtex	0.457	<b>0.462</b>	0.448	0.454	0.429	0.093	0.108	0.206	DNF	0.247	0.123	0.230
delicious	0.234	0.236	DNF	DNF	<b>0.339</b>	0.000	0.000	0.175	DNF	DNF	0.269	0.248
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.268</b>	0.236	0.232	DNF	DNF	0.199	0.236

Figure 9: The performance of the multi-label learning approaches in terms of the micro F1 measure

- [16] A. Elisseeff, J. Weston, A Kernel Method for Multi-Labelled Classification, in: Annual ACM Conference on Research and Development in Information Retrieval, 2005, pp. 274–281.
- [17] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, in: W. Buntine, M. Grobelnik, D. Mladenic, J. Shawe-Taylor (Eds.), Machine Learning and Knowledge Discovery in Databases, Vol. 5782 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2009, pp. 254–269.
- [18] S. Godbole, S. Sarawagi, Discriminative Methods for Multi-labeled Classification, 2004, pp. 22–30.  
URL <http://www.springerlink.com/content/maa4ag38jd3pwr0>
- [19] G. Tsoumakas, I. Vlahavas, Random k-labelsets: An ensemble method for multilabel classification, in: Proceedings of the 18th European conference on Machine Learning, ECML '07, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 406–417.

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.721	0.581	0.677	0.660	0.464	0.602	0.628	0.518	0.547	0.531	<b>0.828</b>	0.802
scene	0.844	0.817	0.835	0.832	0.807	0.635	0.682	0.784	0.835	0.785	<b>0.963</b>	0.919
yeast	0.628	0.602	0.614	0.614	0.471	0.377	0.479	0.600	0.480	0.391	0.533	<b>0.674</b>
medical	<b>0.399</b>	0.391	0.288	0.285	0.287	0.263	0.018	0.267	0.269	0.266	0.190	0.269
enron	0.258	<b>0.260</b>	0.205	0.242	0.241	0.142	0.023	0.170	0.222	0.249	0.245	0.233
corel5k	0.052	0.053	0.059	0.059	0.044	0.004	0.000	0.031	0.000	0.001	0.007	<b>0.211</b>
tmc2007	0.972	0.972	0.964	0.965	0.954	0.925	0.386	0.780	0.973	0.938	0.994	<b>0.997</b>
mediamill	0.112	0.144	0.140	0.133	0.107	0.046	0.401	0.308	0.025	0.037	0.397	<b>0.441</b>
bibtex	0.528	<b>0.539</b>	0.503	0.490	0.391	0.128	0.006	0.192	DNF	0.121	0.080	0.127
delicious	0.299	0.303	DNF	DNF	0.154	0.000	0.000	0.134	DNF	DNF	<b>0.422</b>	0.293
bookmarks	DNF	DNF	DNF	DNF	DNF	0.292	0.018	0.414	DNF	DNF	0.388	<b>0.522</b>

Figure 10: The performance of the multi-label learning approaches in terms of the macro precision measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.378	0.364	0.381	0.398	<b>0.775</b>	0.702	0.533	0.334	0.462	0.508	0.532	0.569
scene	0.703	0.716	0.704	0.701	0.734	0.573	0.529	0.647	0.727	<b>0.757</b>	0.381	0.533
yeast	0.355	0.357	0.361	0.361	<b>0.466</b>	0.375	0.269	0.308	0.352	0.388	0.257	0.286
medical	0.423	<b>0.428</b>	0.307	0.324	0.282	0.249	0.022	0.163	0.183	0.179	0.040	0.176
enron	0.120	0.146	0.139	0.120	<b>0.163</b>	0.107	0.030	0.075	0.097	0.129	0.082	0.100
corel5k	0.023	0.023	0.039	0.039	<b>0.041</b>	0.005	0.000	0.006	0.000	0.001	0.001	0.037
tmc2007	0.915	<b>0.924</b>	0.914	0.914	0.897	0.085	0.235	0.418	0.739	0.772	0.297	0.769
mediamill	0.049	0.044	0.028	0.028	0.074	0.002	0.029	<b>0.088</b>	0.020	0.023	0.065	0.080
bibtex	0.250	<b>0.257</b>	0.236	0.238	0.247	0.034	0.006	0.049	DNF	0.044	0.013	0.043
delicious	0.072	0.075	DNF	DNF	<b>0.103</b>	0.000	0.000	0.039	DNF	DNF	0.092	0.060
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.098</b>	0.016	0.070	DNF	DNF	0.048	0.072

Figure 11: The performance of the multi-label learning approaches in terms of the macro recall measure

- [20] J. Read, B. Pfahringer, G. Holmes, Multi-label classification using ensembles of pruned sets, in: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, IEEE Computer Society, Washington, DC, USA, 2008, pp. 995–1000.
- [21] J. Read, A Pruned Problem Transformation Method for Multi-label classification, in: Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008), 2008, pp. 143–150.
- [22] G. Tsoumakas, I. Katakis, I. Vlahavas, Effective and Efficient Multilabel Classification in Domains with Large Number of Labels, in: Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data, 2008.
- [23] S.-H. Park, J. Fürnkranz, Efficient pairwise classification, in: J. Kok, J. Koronacki, R. Mantaras, S. Matwin, D. Mladenic, A. Skowron (Eds.), Machine Learning: ECML 2007, Vol. 4701 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, pp. 658–665.
- [24] E. L. Mencía, S.-H. Park, J. Fürnkranz, Efficient voting prediction for pairwise multilabel classification, Neurocomputing 73 (2010) 1164–

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.440	0.420	0.443	0.458	0.570	0.630	0.568	0.385	0.488	0.500	0.620	<b>0.650</b>
scene	0.765	0.762	0.762	0.759	0.768	0.596	0.593	0.692	<b>0.777</b>	0.770	0.514	0.658
yeast	0.392	0.390	0.392	0.394	<b>0.447</b>	0.370	0.293	0.336	0.359	0.350	0.283	0.322
medical	0.361	<b>0.371</b>	0.281	0.286	0.282	0.250	0.020	0.192	0.210	0.203	0.058	0.207
enron	0.143	0.153	0.149	0.143	<b>0.167</b>	0.115	0.026	0.087	0.115	0.140	0.102	0.122
corel5k	0.021	0.021	0.042	0.042	0.036	0.008	0.000	0.010	0.000	0.001	0.001	<b>0.056</b>
tmc2007	0.942	<b>0.947</b>	0.938	0.938	0.924	0.124	0.263	0.493	0.826	0.834	0.371	0.857
mediamill	0.056	0.052	0.037	0.037	0.073	0.003	0.031	<b>0.113</b>	0.019	0.022	0.088	0.112
bibtex	0.307	<b>0.316</b>	0.291	0.292	0.266	0.045	0.006	0.065	DNF	0.052	0.016	0.055
delicious	0.096	0.100	DNF	DNF	0.103	0.000	0.000	0.051	DNF	DNF	<b>0.142</b>	0.083
bookmarks	DNF	DNF	DNF	DNF	DNF	<b>0.119</b>	0.017	0.096	DNF	DNF	0.065	0.101

Figure 12: The performance of the multi-label learning approaches in terms of the macro F1 measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.386	0.376	0.391	0.391	0.411	0.347	0.386	0.406	0.396	0.426	0.277	<b>0.262</b>
scene	<b>0.180</b>	0.204	0.190	0.193	0.216	0.394	0.389	0.242	0.197	0.213	0.232	0.210
yeast	0.236	0.268	<b>0.229</b>	0.233	0.248	0.312	0.264	0.234	0.254	0.249	0.250	0.248
medical	0.135	<b>0.123</b>	0.168	0.165	0.216	0.198	0.612	0.279	0.312	0.315	0.243	0.174
enron	0.237	0.238	0.231	0.269	0.314	0.309	0.392	0.280	0.290	0.247	<b>0.219</b>	0.221
corel5k	0.660	0.674	0.588	0.592	0.652	0.762	0.776	0.706	0.758	0.992	0.644	<b>0.003</b>
tmc2007	0.029	0.026	0.033	0.033	0.050	0.145	0.306	0.190	0.047	0.052	0.071	<b>0.006</b>
mediamill	0.188	0.193	0.586	0.560	0.219	0.194	0.220	0.182	0.234	0.242	0.171	<b>0.159</b>
bibtex	0.346	<b>0.342</b>	0.388	0.380	0.466	0.529	0.783	0.576	DNF	0.666	0.544	0.433
delicious	0.354	0.367	DNF	DNF	0.509	0.411	0.592	0.416	DNF	DNF	0.368	<b>0.332</b>
bookmarks	DNF	DNF	DNF	DNF	DNF	0.643	0.817	0.639	DNF	DNF	0.607	<b>0.541</b>

Figure 13: The performance of the multi-label learning approaches in terms of the one-error measure

1176.

- [25] J. Read, B. Pfahringer, G. Holmes, Multi-label Classification Using Ensembles of Pruned Sets, in: ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Vol. 0, IEEE Computer Society, 2008, pp. 995–1000. doi:10.1109/ICDM.2008.74.  
URL <http://www.cs.waikato.ac.nz/~jmr30/papers/icdm08-eps-short.pdf>
- [26] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.
- [27] D. Koccev, C. Vens, J. Struyf, S. Džeroski, Ensembles of multi-objective decision trees, in: Proceedings of the 18th European conference on Machine Learning, ECML '07, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 624–631.
- [28] G. Tsoumakas, I. Katakis, I. Vlahavas, Mining multi-label data, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer US, 2010, pp. 667–685.

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	2.307	2.317	2.386	2.807	2.634	2.069	2.356	2.490	2.465	2.619	<b>1.801</b>	1.827
scene	<b>0.399</b>	0.417	0.423	0.631	0.739	0.945	0.964	0.569	0.635	0.625	0.495	0.461
yeast	6.330	6.439	6.286	8.659	7.285	7.105	6.705	6.414	7.983	7.153	6.276	<b>6.179</b>
medical	1.610	<b>1.471</b>	2.036	1.832	5.324	3.033	5.813	2.844	8.520	7.994	1.889	1.619
enron	12.530	12.437	<b>11.763</b>	22.746	24.190	17.010	14.920	13.181	30.509	27.760	12.485	12.074
corel5k	104.800	105.428	91.506	206.880	250.800	279.900	115.676	113.046	340.398	348.160	110.356	<b>2.540</b>
tmc2007	1.311	1.302	1.363	2.796	2.369	2.671	4.572	2.155	2.498	2.494	1.416	<b>1.219</b>
mediamill	20.481	20.333	24.247	28.982	47.046	22.096	20.456	18.719	56.617	58.865	<b>16.868</b>	16.926
bibtex	20.926	21.078	<b>18.540</b>	57.343	65.626	58.016	58.599	56.266	DNF	87.841	32.580	25.854
delicious	530.126	537.388	DNF	DNF	933.956	620.155	691.622	589.898	DNF	DNF	624.572	<b>504.999</b>
bookmarks	DNF	DNF	DNF	DNF	DNF	58.353	73.780	54.528	DNF	DNF	40.903	<b>34.185</b>

Figure 14: The performance of the multi-label learning approaches in terms of the coverage measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAkEL	ECC	RFML-C4.5	RF-PCT
emotions	0.246	0.245	0.264	0.331	0.297	0.210	0.270	0.283	0.281	0.310	0.153	<b>0.151</b>
scene	<b>0.060</b>	0.064	0.065	0.103	0.119	0.169	0.174	0.093	0.104	0.103	0.079	0.072
yeast	0.164	0.170	<b>0.163</b>	0.296	0.205	0.225	0.199	0.172	0.259	0.224	0.173	0.167
medical	0.021	<b>0.019</b>	0.028	0.027	0.090	0.048	0.104	0.045	0.159	0.152	0.028	0.024
enron	0.084	0.083	<b>0.078</b>	0.177	0.183	0.120	0.114	0.093	0.283	0.238	0.083	0.079
corel5k	0.117	0.118	0.100	0.245	0.352	0.479	0.140	0.130	0.673	0.749	0.122	<b>0.000</b>
tmc2007	<b>0.003</b>	<b>0.003</b>	0.005	0.039	0.028	0.043	0.100	0.031	0.031	0.032	0.007	0.006
mediamill	0.061	0.062	0.092	0.101	0.177	0.073	0.063	0.055	0.236	0.258	<b>0.047</b>	<b>0.047</b>
bibtex	0.068	0.067	<b>0.065</b>	0.207	0.255	0.260	0.255	0.217	DNF	0.394	0.126	0.093
delicious	0.114	0.117	DNF	DNF	0.379	0.174	0.172	0.129	DNF	DNF	0.140	<b>0.106</b>
bookmarks	DNF	DNF	DNF	DNF	DNF	0.194	0.258	0.181	DNF	DNF	0.129	<b>0.104</b>

Figure 15: The performance of the multi-label learning approaches in terms of the ranking loss measure

- [29] M. R. Boutell, J. Luo, X. Shen, C. M. Brown, Learning multi-label scene classification, Pattern Recognition 37 (9) (2004) 1757–1771.
- [30] P. Duygulu, K. Barnard, J. de Freitas, D. Forsyth, Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, in: Computer Vision ECCV 2002, Vol. 2353 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2002, pp. 349–354.
- [31] B. Klimt, Y. Yang, The Enron Corpus: A New Dataset for Email Classification Research, in: Machine Learning: ECML 2004, 2004, pp. 217–226.
- [32] I. Katakis, G. Tsoumakas, I. Vlahavas, Multilabel Text Classification for Automated Tag Suggestion, in: Proc. ECML/PKDD 2008 Discovery Challenge, 2008.
- [33] A. Srivastava, B. Zane-Ulman, Discovering recurring anomalies in text reports regarding complex space systems, in: Proc. of the IEEE Aerospace Conference, Morgan Kaufmann, 2005, pp. 55–63.
- [34] K. Trohidis, G. Tsoumakas, G. Kalliris, I. Vlahavas, Multilabel Classification of Music into Emotions, in: Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008), Philadelphia, PA, USA, 2008, 2008, pp. 320–330.
- [35] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, A. W. M. Smeulders, The challenge problem for automated detection of 101 semantic concepts in multimedia, in: Proc. 14th Annual ACM Intl Conf. on Multimedia, ACM, 2006, pp. 421–430.
- [36] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, The weka data mining software: an update, SIGKDD Explorations

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	0.257	0.256	0.257	0.254	<b>0.361</b>	0.247	0.267	0.294	0.282	0.281	0.198	0.189
scene	<b>0.893</b>	0.881	0.886	0.864	0.848	0.751	0.745	0.851	0.862	0.856	0.862	0.874
yeast	0.768	0.755	<b>0.768</b>	0.698	0.740	0.706	0.724	0.758	0.715	0.734	0.749	0.757
medical	0.896	<b>0.901</b>	0.864	0.862	0.786	0.823	0.522	0.784	0.676	0.684	0.817	0.868
enron	0.693	0.695	<b>0.699</b>	0.604	0.604	0.629	0.546	0.635	0.522	0.576	0.680	0.698
corel5k	0.303	0.293	0.352	0.311	0.222	0.196	0.208	0.266	0.088	0.014	0.314	<b>0.997</b>
tmc2007	0.978	0.981	0.972	0.938	0.945	0.842	0.700	0.844	0.939	0.935	0.945	<b>0.996</b>
mediamill	0.686	0.672	0.450	0.492	0.583	0.669	0.654	0.703	0.492	0.453	0.728	<b>0.737</b>
bibtex	0.597	<b>0.599</b>	0.579	0.498	0.407	0.392	0.212	0.349	DNF	0.228	0.418	0.525
delicious	0.351	0.343	DNF	DNF	0.231	0.321	0.206	0.326	DNF	DNF	0.359	<b>0.395</b>
bookmarks	DNF	DNF	DNF	DNF	DNF	0.378	0.213	0.381	DNF	DNF	0.423	<b>0.480</b>

Figure 16: The performance of the multi-label learning approaches in terms of the average precision measure

	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	4.0	6.0	10.0	10.0	4.0	0.3	<b>0.1</b>	0.4	5.0	4.9	1.2	2.5
scene	71.0	99.0	195.0	195.0	68.0	8.0	<b>2.0</b>	14.0	79.0	319.0	10.0	50.0
yeast	145.0	206.0	672.0	672.0	101.0	14.0	<b>1.5</b>	8.2	157.0	497.0	19.0	169.0
medical	18.0	28.0	40.0	40.0	16.0	3.0	<b>0.6</b>	1.0	82.0	103.0	7.0	15.0
enron	318.0	440.0	971.0	971.0	158.0	15.0	<b>1.1</b>	6.0	493.0	1467.0	25.0	41.0
corel5k	926.0	1225.0	2388.0	2388.0	771.0	369.0	<b>30.0</b>	389.0	3380.0	20073.0	385.0	1355.0
tmc2007	42645.0	46704.0	52427.0	52427.0	31300.0	469.0	<b>11.5</b>	737.0	102394.0	92169.0	460.0	31600.0
mediamill	85468.0	100435.0	260156.0	260156.0	78195.0	2030.0	<b>440.0</b>	1094.0	33554.0	188957.0	4056.0	62915.0
bibtex	11013.0	12434.0	13424.0	13424.0	2896.0	566.0	<b>16.4</b>	124.0	DNF	29578.0	645.0	802.0
delicious	57053.0	84903.0	DNF	DNF	21218.0	2738.0	<b>70.0</b>	236.0	DNF	DNF	21776.0	7562.0
bookmarks	DNF	DNF	DNF	DNF	DNF	4039.0	<b>965.0</b>	15990.0	DNF	DNF	5602.0	13420.0

Figure 17: The performance of the multi-label learning approaches in terms of the training time measures in seconds

11 (2009) 10–18.

[37] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (2001).

[38] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, *Annals of Mathematical Statistics* (11) (1940) 86–92.

[39] P. B. Nemenyi, Distribution-free multiple comparisons, in: PhD Thesis, Princeton University, Princeton, NY, USA, 1963.

[40] J. Demsar, Statistical comparisons of classifiers over multiple data sets, *Journal of Machine Learning Research* (7) (2006) 1–30.

[41] R. L. Iman, J. M. Davenport, Approximations of the critical region of the friedman statistic, *Communications in Statistics* (1980) 571–595.

## Appendix A. Evaluation measures

In this section we present the measures that are used in the experiments for evaluation the predictive performance of the compared methods.



	BR	CC	CLR	QWML	HOMER	ML-C4.5	PCT	ML-kNN	RAKEL	ECC	RFML-C4.5	RF-PCT
emotions	1.0	1.0	3.0	2.0	1.0	<b>0.0</b>	<b>0.0</b>	0.4	2.0	6.6	0.1	0.3
scene	25.0	25.0	87.0	40.0	21.0	1.0	<b>0.0</b>	14.0	72.0	168.0	2.0	1.0
yeast	23.0	25.0	153.0	64.0	17.0	0.1	<b>0.0</b>	5.0	70.0	158.0	0.5	0.2
medical	4.0	6.0	90.0	25.0	1.5	0.1	<b>0.0</b>	0.2	24.0	46.0	0.5	0.5
enron	50.0	53.0	634.0	174.0	22.0	0.2	<b>0.0</b>	3.0	153.0	696.0	1.0	1.0
corel5k	25.0	31.0	2161.0	119.0	14.0	<b>1.0</b>	<b>1.0</b>	45.0	3613.0	2077.0	1.8	2.5
tmc2007	927.0	891.0	3282.0	1543.0	730.0	1.7	<b>0.0</b>	230.0	10985.0	10865.0	3.4	2.8
mediamill	6152.0	6125.0	76385.0	20317.0	6079.0	<b>1.0</b>	<b>1.0</b>	477.0	39001.0	50183.0	8.0	4.0
bibtex	654.0	661.0	16733.0	4710.0	155.0	6.5	<b>0.0</b>	64.0	DNF	10756.0	12.0	18.0
delicious	2045.0	1872.0	DNF	DNF	816.0	19.0	<b>10.0</b>	55.0	DNF	DNF	32.0	48.0
bookmarks	DNF	DNF	DNF	DNF	DNF	21.0	<b>15.0</b>	4084.0	DNF	DNF	28.0	52.0

Figure 18: The performance of the multi-label learning approaches in terms of the testing time measures in seconds

#### Appendix A.1. Example based measures

1. Hamming loss evaluates how many times an example-label pair is misclassified, i.e., throughout a label not belonging to the instance is predicted or a label belonging to the instance is not predicted. The smaller the value of  $hamming\_loss(h)$ , the better the performance. The performance is perfect when  $hamming\_loss(h) = 0$ . This metric is defined as:

$$hamming\_loss(h) = \frac{1}{N} \sum_{i=1}^N \frac{1}{Q} |h(\mathbf{x}_i) \Delta \mathcal{Y}_i| \quad (A.1)$$

where  $\Delta$  stands for the symmetric difference between two sets,  $N$  is the number of examples and  $Q$  is the total number of possible class labels.

2. Accuracy is measured by the Hamming score which symmetrically measures how close  $h(\mathbf{x}_i)$  is to  $\mathcal{Y}_i$  for an example  $\mathbf{x}_i$ . Accuracy is micro-averaged across all examples.

$$accuracy(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i) \cup \mathcal{Y}_i|} \quad (A.2)$$

3. Precision in multi-label setting is defined as:

$$precision(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|\mathcal{Y}_i|} \quad (A.3)$$

4. Recall is defined as:

$$recall(h) = \frac{1}{N} \sum_{i=1}^N \frac{|h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i)|} \quad (A.4)$$

where  $\Delta$  stands for the symmetric difference between two sets,  $N$  is the number of examples and  $Q$  is the total number of possible class labels.

5.  $F_1$  score is the harmonic mean between precision and recall and in the multi-labeled classification setting is defined as:

$$F_1 = \frac{1}{N} \sum_{i=1}^N \frac{2 \times |h(\mathbf{x}_i) \cap \mathcal{Y}_i|}{|h(\mathbf{x}_i)| + |\mathcal{Y}_i|} \quad (A.5)$$

$F_1$  is example based metric and its value is an average over all examples in the dataset.  $F_1$  reaches its best value at 1 and worst score at 0.

6. Subset accuracy or classification accuracy is defined as follows:

$$subset\_accuracy(h) = \frac{1}{N} \sum_{i=1}^N I(h(\mathbf{x}_i) = \mathcal{Y}_i) \quad (\text{A.6})$$

where  $I(true) = 1$  and  $I(false) = 0$ . This is a very strict evaluation measure as it requires the predicted set of labels to be an exact match of the true set of labels.

#### Appendix A.2. Label based measures

1. Macro precision (precision averaged across all labels) is defined as:

$$macro\_precision = \frac{1}{Q} \sum_{j=1}^Q \frac{tp_j}{tp_j + fp_j} \quad (\text{A.7})$$

where  $tp_j, fp_j$  are the number of true positives and false positives after binary evaluation for the label  $\lambda_j$ .

2. Macro recall (recall averaged across all labels) is defined as:

$$macro\_recall = \frac{1}{Q} \sum_{j=1}^Q \frac{tp_j}{tp_j + fn_j} \quad (\text{A.8})$$

where  $tp_j, fp_j$  are defined as for the macro precision and  $fn_j$  is the number of false negatives after binary evaluation for the label  $\lambda_j$ .

3. Macro  $F_1$  is the harmonic mean between precision and recall where the average is calculated per label and then averaged across all labels. If  $p_j$  and  $r_j$  are the precision and recall for all  $\lambda_j \in h(\mathbf{x}_i)$  from  $\lambda_j \in \mathcal{Y}_i$ , the macro  $F_1$  is

$$macro\_F_1 = \frac{1}{Q} \sum_{j=1}^Q \frac{2 \times p_j \times r_j}{p_j + r_j} \quad (\text{A.9})$$

4. Micro precision (precision averaged over all the example/label pairs) is defined as:

$$micro\_precision = \frac{\sum_{j=1}^Q tp_j}{\sum_{j=1}^Q tp_j + \sum_{j=1}^Q fp_j} \quad (\text{A.10})$$

where  $tp_j, fp_j$  are defined as for macro precision.

5. Micro recall (recall averaged over all the example/label pairs) is defined as:

$$micro\_recall = \frac{\sum_{j=1}^Q tp_j}{\sum_{j=1}^Q tp_j + \sum_{j=1}^Q fn_j} \quad (\text{A.11})$$

where  $tp_j$  and  $fn_j$  are defined as for macro recall.

6. Micro  $F_1$  is the harmonic mean between micro precision and micro recall. Micro  $F_1$  is defined as:

$$micro\_F_1 = \frac{2 \times micro\_precision \times micro\_recall}{micro\_precision + micro\_recall} \quad (\text{A.12})$$

### Appendix A.3. Ranking based measures

1. One error evaluates how many times the top-ranked label is not in the set of relevant labels of the example. The metric  $one\_error(f)$  takes values between 0 and 1. The smaller the value of  $one\_error(f)$ , the better the performance. This evaluation metric is defined as:

$$one\_error(f) = \frac{1}{N} \sum_{i=1}^N \left\| \left[ \arg \max_{\lambda \in \mathcal{Y}} f(\mathbf{x}_i, \lambda) \right] \notin \mathcal{Y}_i \right\| \quad (\text{A.13})$$

where  $\lambda \in L = \{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  and  $\llbracket \pi \rrbracket$  equals 1 if  $\pi$  holds and 0 otherwise for any predicate  $\pi$ . Note that, for single-label classification problems, the One Error is identical to ordinary classification error.

2. Coverage evaluates how far, on average we need to go down the list of ranked labels in order to cover all the relevant labels of the instance. The smaller the value of  $coverage(f)$ , the better the performance.

$$coverage(f) = \frac{1}{N} \sum_{i=1}^N \max_{\lambda \in \mathcal{Y}_i} rank_f(\mathbf{x}_i, \lambda) - 1 \quad (\text{A.14})$$

where  $rank_f(\mathbf{x}_i, \lambda)$  maps the outputs of  $f(\mathbf{x}_i, \lambda)$  for any  $\lambda \in \mathcal{L}$  to  $\{\lambda_1, \lambda_2, \dots, \lambda_Q\}$  so that  $f(\mathbf{x}_i, \lambda_m) > f(\mathbf{x}_i, \lambda_n)$  implies  $rank_f(\mathbf{x}_i, \lambda_m) < rank_f(\mathbf{x}_i, \lambda_n)$ . The smallest possible value for  $coverage(f)$  is  $l_c$ , i.e., the label cardinality of the given dataset.

3. Ranking loss evaluates the average fraction of label pairs that are reversely ordered for the particular example given by:

$$ranking\_loss(f) = \frac{1}{N} \sum_{i=1}^N \frac{|D_i|}{|\mathcal{Y}_i| |\bar{\mathcal{Y}}_i|} \quad (\text{A.15})$$

where  $D_i = \{(\lambda_m, \lambda_n) | f(\mathbf{x}_i, \lambda_m) \leq f(\mathbf{x}_i, \lambda_n), (\lambda_m, \lambda_n) \in \mathcal{Y}_i \times \bar{\mathcal{Y}}_i\}$ , while  $\bar{\mathcal{Y}}$  denotes the complementary set of  $\mathcal{Y}$  in  $\mathcal{L}$ . The smaller the value of  $ranking\_loss(f)$ , the better the performance, so the performance is perfect when  $ranking\_loss(f) = 0$ .

4. Average Precision is the average fraction of labels ranked above an actual label  $\lambda \in \mathcal{Y}_i$  that actually are in  $\mathcal{Y}_i$ . The performance is perfect when  $avg\_precision(f) = 1$ ; the larger the value of  $avg\_precision(f)$ , the better the performance. This metric is defined as:

$$avg\_precision(f) = \frac{1}{N} \sum_{i=1}^N \frac{1}{|\mathcal{Y}_i|} \sum_{\lambda \in \mathcal{Y}_i} \frac{|\mathcal{L}_i|}{rank_f(\mathbf{x}_i, \lambda)} \quad (\text{A.16})$$

where  $\mathcal{L}_i = \{\lambda' | rank_f(\mathbf{x}_i, \lambda') \leq rank_f(\mathbf{x}_i, \lambda), \lambda' \in \mathcal{Y}_i\}$  and  $rank_f(\mathbf{x}_i, \lambda)$  is defined as in coverage above.