A Tutorial on Multi-Label Learning

EVA GIBAJA, Department of Computer Science and Numerical Analysis, University of Córdoba, Spain
SEBASTIÁN VENTURA, Department of Computer Science and Numerical Analysis, University of Córdoba, Spain

Multi-label learning has become a relevant learning paradigm in the last years due to the increasing number of fields where it can be applied and also to the emerging number of techniques that are being developed. This paper presents an up-to-date tutorial about multi-label learning that introduces the paradigm and describes the main contributions developed. Evaluation measures, fields of application, trending topics and resources are also presented.

Categories and Subject Descriptors: F.0 [**Theory of Computation**]: GENERAL

General Terms: Algorithms

Additional Key Words and Phrases: Multi-label learning, ranking, trends

ACM Reference Format:
Eva Gibaja and Sebastián Ventura, 2013. A Tutorial on Multi-Label Learning *ACM Comput. Surv.* 9, 4, Article 39 (March 2010), 39 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Classification is one of the main tasks in data mining. Given a set of training patterns consisting of a set of features and a class associated, the aim of classification is to obtain a model that will be able to assign the proper class to an unknown pattern. This formulation of the problem entails the restriction of *only one label per pattern*; nevertheless, there are more and more classification problems being contemplated today, such as text and sound categorization, semantic scene classification, medical diagnosis or gene and protein function classification, where a pattern can have several labels simultaneously associated. For instance, in the field of semantic scene classification, a picture containing a landscape with both a beach and a mountain could be associated with *beach* and *mountain* categories simultaneously. This type of problem is called *multi-label* in comparison with classical supervised learning (also called *single-label* [Schapire and Singer 2000]). Solving a problem with multi-label data involves news challenges due to the exponential growth of combinations of labels to take into account and also to the computational cost of building and querying the models. Besides, multi-label data usually present features such as high dimensionality, unbalanced data and dependences between labels.

Thus, during the last years the paradigm of *Multi-Label Learning* (MLL) has arisen as a kind of supervised learning and has become a very hot topic. Despite the publica-

This work is supported by the Ministry of Science and Technology project TIN-2011-22408. Author's addresses: E. Gibaja and S. Ventura, Department of Computer Science and Numerical Analysis, University of Córdoba, Spain. Dr. Ventura also belongs to the Computer Sciences Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.

DOI: http://dx.doi.org/10.1145/0000000.0000000

tion of some works compiling the basis of MLL [Tsoumakas and Katakis 2007], [de Carvalho and Freitas 2009], [Tsoumakas et al. 2010a], [Sorower 2010], it is not possible to find an up-to-date tutorial on MLL. The aim of this paper is to cover, among other issues, the formal definition of the problem, the domains where MLL has been applied, an up-to-date summary of the main proposals presented during latest years, evaluation measures and resources. This paper is organized as follows. In Section 2 the MLL problem is formally defined. After that, in section 3, some aspects related to the development and evaluation of MLL models are described. The main approaches developed in the literature are presented in section 4. Next, section 5 describes findings on empirical comparisons between MLL algorithms. Section 6 describes the main domains where MLL has been applied and, finally, new trends in MLL (section 7) and a set of conclusions are presented. The paper includes also an appendix with resources (software, datasets, etc.) for MLL.

2. MULTI-LABEL LEARNING

2.1. MLL settings

According to Read [2010] a multi-label problem has the following settings:

(1) The set of labels is predefined, meaningful and human-interpretable.
(2) The number of labels is limited in scope and not greater than the number of attributes.
(3) Each training example is associated with several labels of the labelset.
(4) The number of attributes may be large, but attribute-reduction strategies can be employed in these cases.
(5) The number of examples may be large.

The two last points are related with the high dimensionality of data, a common feature of many multi-label datasets. It is also worth noting two other features:

(6) Labels may be correlated. As an example, in figure 1 it is shown a graph of the 10 most frequent labels of a multi-label dataset, so-called *imdb* [Read 2010] (more information about imdb dataset is found in the Appendix). This dataset, contains 120919 movie plot text summaries from the imdb database and has 28 labels corresponding with genres (e.g. *comedy*, *action* etc). Node thickness represents prior probability of the label and edge thickness indicates co-occurrence of the two linked labels. It is observed that *talk show* and *war* labels are not related while there are relationships with different strength between the other ones, for instance, *action* and *crime* are more related than *mystery* and *film noir*. These relationships between labels represent additional knowledge that can be explored during the training of the learners to facilitate the learning process.
(7) Data may be unbalanced. This can be seen from two points of view. On one hand, if each particular label is considered, the number of patterns belonging to a certain label may outnumber other labels (inter-class). For instance, figure 1 shows that *comedy* is much more frequent that *film noir*. Besides if the total number of patterns is taken into account, the proportion of positive to negative examples for each class may be alsomples may be associated with the most common labelsets. The figure 2 represents the number of examples per labelset and the *Imbalance Ratio* (IR) showing that the dataset of the example is clearly unbalanced. IR has been computed for each labelset as the quotient between the size of the most frequent labelset and the size of the labelset.

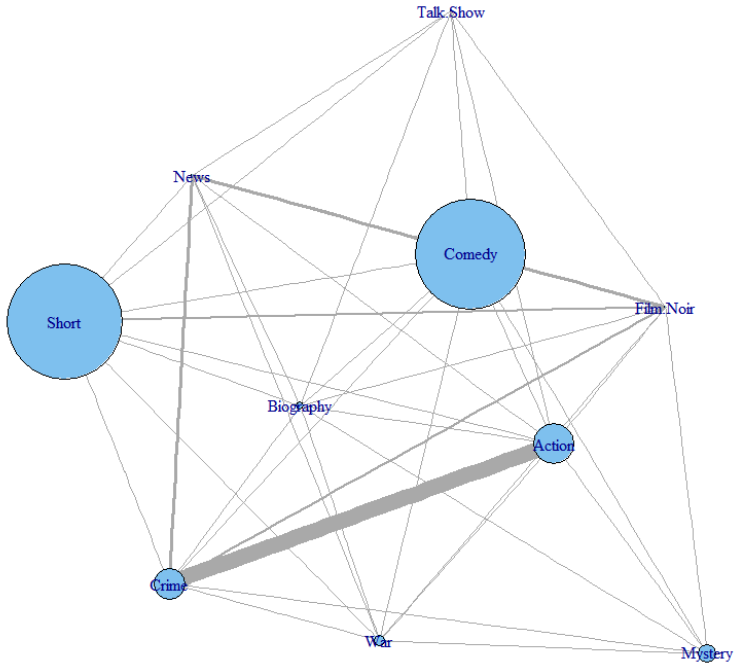

Fig. 1. Co-occurrence graph of labels in imdb dataset

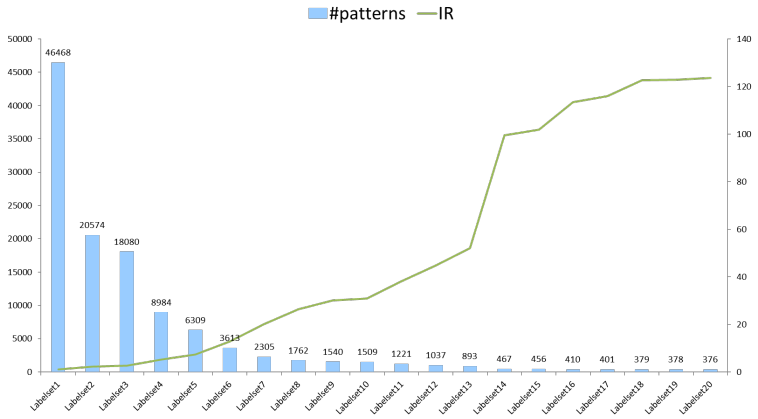

Fig. 2. Examples and IR per labelset in imdb dataset

2.2. A formal definition of MLL

This section is based on the notations defined by Schapire and Singer [2000], Zhang and Zhou [2005] and Brinker et al. [2006]. Let be the following definitions:

— $\mathcal{X}$ a d-dimensional input space of numerical or categorical features.
— $\mathcal{L} = \{\lambda_1, \lambda_2, ..., \lambda_q\}$ an output space of q labels, $q > 1$. Each subset of $\mathcal{L}$ is called *labelset*.
— $(\mathbf{x}, Y)$, where $\mathbf{x} = (x_1, \ldots, x_d) \in \mathcal{X}$, is a d-dimensional instance which has a set of labels associated $Y \subseteq \mathcal{L}$. Label associations can be also represented as a q dimen-

Table I. An example of a single-label vs. a multi-label dataset

EXAMPLE	FEATURES	SINGLE-LABEL BINARY	SINGLE-LABEL MULTI-CLASS	MULTI-LABEL OUTPUT				
		$Y \in \mathcal{L} = \{0,1\}$	$Y \in \mathcal{L} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4,\}$	y_1	y_2	y_3	y_4	$Y \subseteq \mathcal{L} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$
1	$\overline{\mathbf{x_1}}$	1	λ_2	1	1	0	1	$\{\lambda_1, \lambda_2, \lambda_4\}$
2	$\overline{\mathbf{x_2}}$	0	λ_4	0	0	0	1	$\{\lambda_4\}$
3	$\overline{\mathbf{x_3}}$	0	λ_3	0	1	1	1	$\{\lambda_2, \lambda_3, \lambda_4\}$
4	$\overline{\mathbf{x_4}}$	1	λ_1	1	0	1	0	$\{\lambda_1, \lambda_3\}$
5	$\overline{\mathbf{x_5}}$	0	λ_3	0	1	1	0	$\{\lambda_2, \lambda_3\}$
				2	3	3	3	◁COUNT

sional binary vector $\mathbf{y} = (y_1, y_2, \ldots, y_q) = \{0,1\}^q$ where each element is 1 if the label is relevant and 0 otherwise. Table I shows an example of a multi-label dataset compared with a single-label binary one. As it can be observed, in single-label (binary or multi-class) learning $|Y| = 1$.

— $S = \{(\mathbf{x}_i, Y_i) | 1 \leq i \leq m\}$ is a multi-label training set with m examples.

According to [Tsoumakas et al. 2010a], MLL includes two main tasks: *Multi-Label Classification* (MLC) and *Label Ranking* (LR). MLC consists of defining a function $h_{\text{MLC}} : \mathcal{X} \rightarrow 2^{\mathcal{L}}$. Therefore, given an input instance, a multi-label classifier will return a set of relevant labels, Y, being the complement of this set, $\overline{Y}$, the set of irrelevant labels. So, a bipartition of the set of labels into relevant, and irrelevant sets is obtained. Multi-class classification can be seen as a particular case of multi-label classification where $h_{\text{MC}} : \mathcal{X} \rightarrow \mathcal{L}$ while in binary classification $h_{\text{B}} : \mathcal{X} \rightarrow \{0,1\}$.

On the other hand, LR defines a function $f : \mathcal{X} \times \mathcal{L} \rightarrow \mathbb{R}$ that returns an ordering of all the possible labels according to the relevance of labels to a given instance $\mathbf{x}$. Thus label λ_1 is considered to be ranked higher than λ_2 if $f(\mathbf{x}, \lambda_1) > f(\mathbf{x}, \lambda_2)$. A rank function, $\tau_{\mathbf{x}}$, maps the output real value of the classifier to the position of the label in the ranking, $\{1, 2, ..., q\}$. Therefore, if $f(\mathbf{x}, \lambda_1) > f(\mathbf{x}, \lambda_2)$ then $\tau_{\mathbf{x}}(\lambda_1) < \tau_{\mathbf{x}}(\lambda_2)$. The lower the position, the better the position in the ranking is. Finally a third task in MLL, called *Multi-Label Ranking* that can be seen as a generalization of MLC and LR, produces at the same time both a bipartition and a consistent ranking. In other words, if Y is the set of labels associated with an instance, $\mathbf{x}$, and $\lambda_1 \in Y$ and $\lambda_2 \in \overline{Y}$ then a consistent ranking will rank labels in Y higher than labels in $\overline{Y}$, $\tau_{\mathbf{x}}(\lambda_1) < \tau_{\mathbf{x}}(\lambda_2)$. The definition of multi-label classifier from a multi-label ranking model can be derived from the function $f(\mathbf{x}, \lambda) : h(\mathbf{x}) = \{\lambda | f(\mathbf{x}, \lambda) > t(\mathbf{x}), \lambda \in \mathcal{L}\}$, where $t(\mathbf{x})$ is a threshold function. Section 4.4 goes into detail about this topic.

3. EVALUATION OF MULTI-LABEL MODELS

This section includes aspects to consider when a MLL method is being evaluated: evaluation metrics, how to prepare the dataset, statistical tests and complexity.

3.1. Evaluation Metrics

The evaluation of models in MLL needs a special approach because the performance over all labels should be taken into account. Besides, a prediction could be partially correct (some of the labels are correctly predicted), fully wrong (all predictions are wrong) or fully correct (all labels are correctly predicted). In this section the most frequent performance metrics for MLL will be summarized following the taxonomy proposed by Tsoumakas et al. [2010a] which differentiates between two kinds of metrics: *metrics to evaluate bipartitions* and *metrics to evaluate rankings*.

Let the same notations be adopted in the definition of the problem (subsection 2.2), let $T = \{(\mathbf{x}_i, Y_i) | 1 \leq i \leq t\}$ be a multi-label test set with t instances, Y_i and Z_i the sets

of true and predicted labels for an instance. For any predicate, π, $[\![\pi]\!]$ returns 1 if the predicate is true and 0 otherwise and, finally, let $\tau^*_{\mathbf{x}}$ be the true ranking.

3.1.1. Metrics to evaluate bipartitions. Metrics to evaluate bipartitions can be classified into two groups: *label-based* and *example-based*. The former are calculated for each label and then they are averaged across all labels (ignoring relations between labels) while the latter are calculated for each test example and then averaged across the test set.

LABEL-BASED METRICS

Any binary evaluation metric can be used with this type of approach, commonly the precision, recall, accuracy and F1-score. The idea is computing a single-label metric for each label based on the number of true positives (tp), true negatives (tn), false positives (fp) and false negatives (fn). Due to the fact of having several labels per pattern there will be a contingency table for each label, so it is necessary to compute an average value. Two different approaches can be used: *macro* and *micro*. Let B be a binary evaluation measure, the macro approach computes one metric for each label and then the values are averaged over all the categories (see equation 1) while the micro approach considers predictions from all instances together (aggregating the values of all the contingency tables) and then calculates the measure across all labels (see equation 2).

$$B_{macro} = \frac{1}{q}\sum_{i=1}^{q} B(tp_i, fp_i, tn_i, fn_i) \tag{1}$$

$$B_{micro} = B(\sum_{i=1}^{q} tp_i, \sum_{i=1}^{q} fp_i, \sum_{i=1}^{q} tn_i, \sum_{i=1}^{q} fn_i) \tag{2}$$

These two types of averaging are informative and there is no general agreement about using a macro or micro approach. According to [Yang 1999] and [Yang and Liu 1999], macro averaged scores give equal weight to every category, regardless of its frequency (per-category averaging) and is more influenced by the performance on rare categories. On the other hand, micro averaged scores give equal weight to every example (per-example averaging) and tend to be dominated by the performance in most common categories. In the same vein, Pestian et al. [2007] pointed that macro approach would be better when the system is required to perform consistently across all classes regardless the frequency of the class (i.e. in problems where distribution of training samples across categories is skewed) while the micro approach may be better if the density of the class is important.

INSTANCE/EXAMPLE-BASED METRICS

0/1 subset accuracy [Zhu et al. 2005]. This metric, also called *classification accuracy* or *exact match ratio*, computes the percentage of instances whose predicted labels are exactly the same as their corresponding set of ground-truth labels (see equation 3). As an exact match between the predicted and the true sets of labels is needed, it does not distinguish between *completely incorrect* and *partially correct* predictions being a very strict evaluation measure.

$$0/1 \; subset \; accuracy = \frac{1}{t}\sum_{i=1}^{t} [\![Z_i = Y_i]\!] \tag{3}$$

Hamming Loss [Schapire and Singer 1999]. Evaluates how many times, on average, an example-label pair is misclassified. This metric takes into account both prediction

errors (an incorrect label is predicted) and omission errors (a correct label is not predicted) normalized over total number of classes and total number of examples. The lower the value the better the performance of the classifier is. The expression of this metric is in equation 4 where Δ stands for the symmetric difference of two sets, the $1/q$ factor is used to obtain a normalized value in [0, 1].

$$Hamming\ loss = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{q}|Z_i \Delta Y_i| \tag{4}$$

In MLL it is also common to use a group of example-based metrics from the *information retrieval* (IR) area [Godbole and Sarawagi 2004]: *Recall* (see equation 5) is the fraction of predicted correct labels of the actual labels while *precision* (see equation 6) is the proportion of labels correctly classified of the predicted positive labels, averaged over all instances.

$$recall = \frac{1}{t}\sum_{i=1}^{t}\frac{|Z_i \cap Y_i|}{|Y_i|} \tag{5}$$

$$precision = \frac{1}{t}\sum_{i=1}^{t}\frac{|Z_i \cap Y_i|}{|Z_i|} \tag{6}$$

The *accuracy* (see equation 7) is the proportion of label values correctly classified of the total number (predicted and actual) of labels for that instance averaged over all instances.

$$accuracy = \frac{1}{t}\sum_{i=1}^{t}\frac{|Z_i \cap Y_i|}{|Z_i \cup Y_i|} \tag{7}$$

Finally, the *F1-Score* or *harmonic mean* that combines precision and recall is defined in equation 8.

$$F1 - score = \frac{1}{t}\sum_{i=1}^{t}\frac{2|Z_i \cap Y_i|}{|Z_i| + |Y_i|} \tag{8}$$

3.1.2. Metrics to evaluate rankings. All the metrics detailed below can be also considered label-based metrics as they are firstly computed for each test example and then they are averaged across the test set.

The *One-error* [Schapire and Singer 2000] measure evaluates how many times the top-ranked label was not in the set of possible labels so the lower the value the better the metric is. It measures the probability of not getting even one of the labels correct. A priori this metric is not a good metric for multi-label classification because it only takes into account the top ranked label. Note that for single-label the one-error is equivalent to ordinal error. The expression of this metric is shown in equation 9, $\arg$ function returns a label, $\lambda \in \mathcal{L}$.

$$one - error = \frac{1}{t}\sum_{i=1}^{t}[\![\arg\min_{\lambda \in \mathcal{L}} \tau_i(\lambda) \notin Y_i]\!] \tag{9}$$

Coverage [Schapire and Singer 2000]. This metric (see equation 10) measures the average depth in the ranking in order to cover all the labels associated with an instance. The smaller the value, the better the performance is. While one-error only takes into account the performance for the top-ranked label, the coverage measures the perfor-

mance for all the possible labels.

$$coverage = \frac{1}{t}\sum_{i=1}^{t}\max_{\lambda\in Y_i}\tau_i(\lambda) - 1 \tag{10}$$

Ranking loss [Schapire and Singer 1999]. Evaluates the average fraction of pairs of labels that are misordered for the instance. The lower the value of the metric the better the performance. The goal is to obtain a small number of misorderings so that the labels in Y are ranked above the ones in $\overline{Y}$. The $|E|$ is called in [Crammer and Singer 2003] and [Park and Fürnkranz 2008] *error-set-size*.

$$ranking\ loss = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i||\overline{Y_i}|}|E| where$$

$$E = \left\{(\lambda,\lambda')|\tau_i(\lambda) > \tau_i(\lambda'), (\lambda,\lambda') \in Y_i \times \overline{Y_i}\right\} \tag{11}$$

IsError [Loza and Fürnkranz 2010] [Crammer and Singer 2003]. It measures whether the induced ranking is perfect or not. It returns 0 if the ranking is perfect and 1 otherwise, irrespective of how wrong the ranking is. This metric has the same meaning as $0/1$ subset accuracy described in [Zhu et al. 2005], but applied to ranking.

$$is\ error = \frac{1}{t}\sum_{i=1}^{t}[\![\sum_{\lambda\in\mathcal{L}}|\tau_i^*(\lambda) - \tau_i(\lambda)| \neq 0]\!] \tag{12}$$

Average precision [Schapire and Singer 2000]. Coverage and one-error are not complete metrics for multi-label classification because it is possible to have a good coverage while having high one-error values. The average precision evaluates the average fraction of labels ranked above a particular label, $\lambda \in Y$, which actually are in Y. The performance is perfect when the value of the metric is 1, the bigger the value of the metric, the better the performance is.

$$avg.\ precision = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i|}\sum_{\lambda\in Y_i}\frac{|\{\lambda' \in Y_i|\tau_i(\lambda') \leq \tau_i(\lambda)\}|}{\tau_i(\lambda)} \tag{13}$$

Margin loss [Loza and Fürnkranz 2010]. This metric returns the number of positions between the worst ranked positive and the best ranked negative classes. This measure is related to the number of wrongly ranked classes.

$$margin\ loss = \frac{1}{t}\sum_{i=1}^{t}\max(0, \max\{\tau(\lambda)|\lambda \in Y_i\} - \min\{\tau(\lambda')|\lambda' \notin Y_i\}) \tag{14}$$

Ranking error [Park and Fürnkranz 2008]. Returns the normalised sum of squared position differences for each label in the predicted and true ranking. It is 0 for a ranking which is identical to the true ranking and 1 for a completely reversed ranking.

$$ranking\ error = \frac{1}{t}\sum_{i=1}^{t}\sum_{\lambda\in\mathcal{L}}|\tau_i^*(\lambda) - \tau_i(\lambda)|^2 \tag{15}$$

3.2. Partitioning datasets

In supervised learning, two of the more frequent evaluation techniques are the *holdout* method that splits the dataset into a training and a test set, and the *cross-validation* which is used when the training data is limited and splits the dataset into a number of disjoint subsets of the same size. In both techniques, partitions should have

approximately the same data distribution of the original distribution, this is known as *stratified partition*. In MLL it is not clear how stratification should be carried out. Most works have used the default train/test partitions of the dataset or the random version of holdout and cross-validation methods, and literature about multi-label stratification is sparse. The work of Sechidis et al. [2011] can be referenced in which two stratification methods are proposed. The first one splits the partitions considering the different combinations of labels. This approach is impractical for datasets where the number of distinct labelsets is too large. The second proposal is a relaxed interpretation whose aim is to maintain the distribution of positive and negative examples of each label. Their experiments conclude that both approximations are better than the random sampling.

3.3. Statistical tests

In order to compare the performance of several multi-label classifiers, a two-step procedure for classical single-label classification recommended in [Demšar 2006] has been used. It consists of applying a Friedman test with the null hypothesis that all learners have equal performance and, if the null-hypothesis is rejected, a post-hoc test is carried out. Two main post-hoc tests have been applied: in [Cheng and Hüllermeier 2009], [Cherman et al. 2012] and [Madjarov et al. 2012], a Nemenyi test that compared learners in a pairwise way and, in [Ávila et al. 2011], a Bonferroni-Dunn post-hoc test. This test compares not in a pairwise way, but with a control algorithm (the one that obtains the lower ranking value in the Friedman test). According to Demšar [2006], if the target is only testing whether a newly proposed method is better than the existing ones, the power of the post-hoc test is much greater when all classifiers are compared only to a control classifier and not in a pairwise way. For pairwise comparison of two classifiers, the non-parametric Wilcoxon signed-rank test (a better alternative to the paired t-test when the performance scores are not normally distributed) has been used in [Vens et al. 2008] and [Yang and Gopal 2012]. The null hypothesis is that both methods are equally effective; the alternative is that one of the methods is better. Finally, Yang and Gopal [2012] recommended the Wilcoxon test instead the Friedman test when the number of datasets is 10 or less because the latter requires a relatively large number of datasets for meaningful testing.

3.4. Complexity

According to [Tsoumakas et al. 2008], the high dimensionality of the label space may challenge the efficiency of MLL methods in two ways. On the one hand, the computational cost of training a multi-label classifier may be affected by the number of labels. There are simple algorithms (e.g. Binary Relevance) with linear complexity with respect to q, but there are also more advanced methods whose complexity is worse (e.g. Ranking by Pairwise Comparison). Secondly, the classification stage can also be influenced by the number of classifiers and can be quite time-consuming especially in classification problems with large numbers of labels. Another important factor to consider related to high dimensionality is the memory requirements. All of these factors have to be taken into account when developing a new MLL and make the development of a time and space efficiency analysis necessary. In [Sorower 2010] a complexity summary of the main MLL methods can be found.

4. MULTI-LABEL LEARNING METHODS

Nowadays, most authors agree with the taxonomy presented in Tsoumakas et al. [2010a] that differentiates between two main approaches in solving MLL problems: *problem transformation methods* and *algorithm adaptation methods*. The former

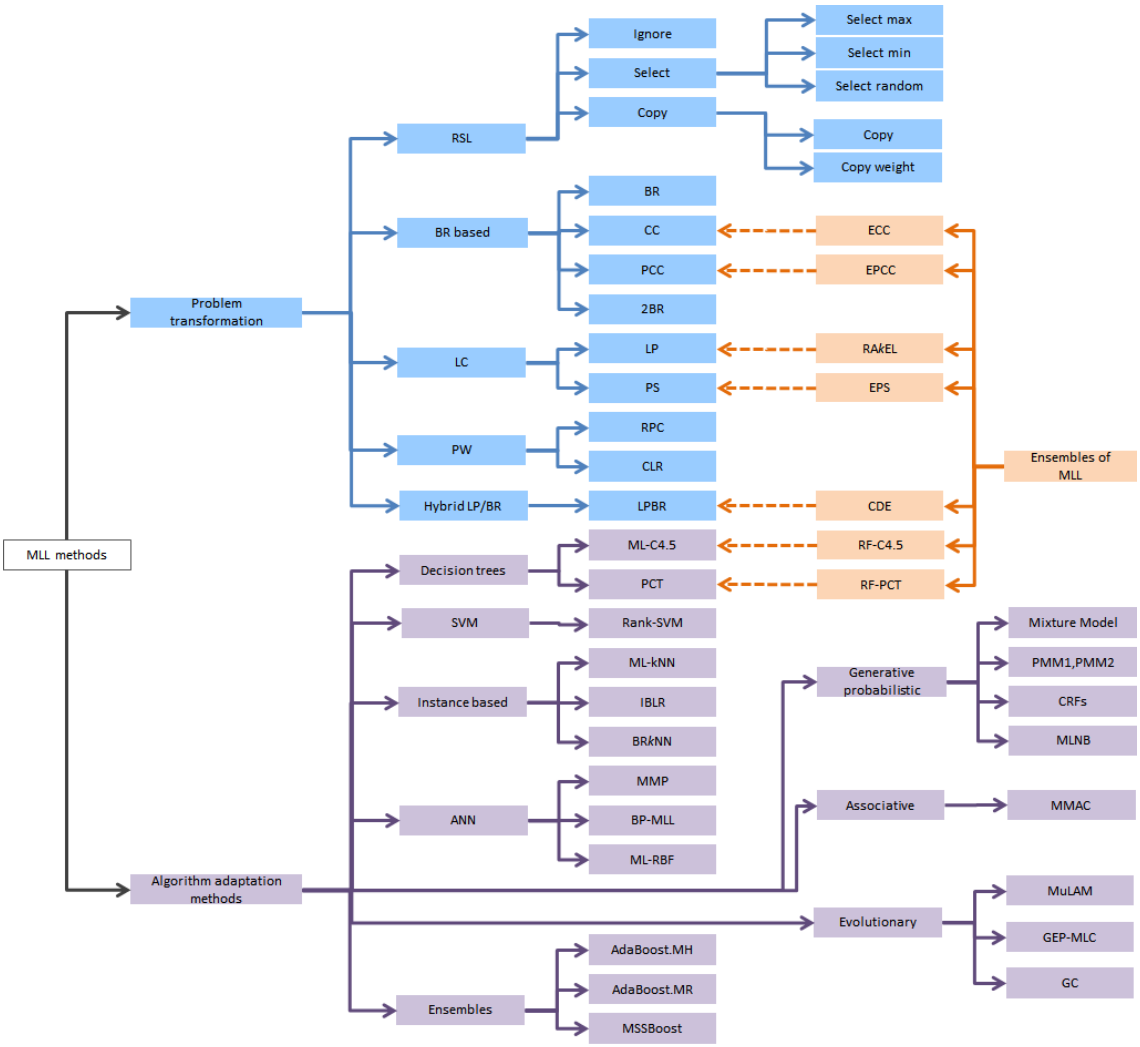

Fig. 3. Taxonomy of MLL methods

transforms the multi-label problem into one or more single-label ones that are applied a single-label classification algorithm, while the latter consists of extending a single-label algorithm in order to directly deal with multi-label data. It is worth noting that the problem transformation approach is algorithm independent. A taxonomy with the proposals described in this paper is found in figure 3.

4.1. Problem transformation methods

4.1.1. Ranking Via Single-Label Learning methods. The approach dubbed *ranking via single-label learning* (SLR) transforms a multi-label dataset into a single-label one and then uses a single-label classifier which is able to produce a score (e.g. probability) for each label in order to obtain a ranking. Thus the label with the highest probability will be ranked first, and so on. A straight transformation is the one called *ignore* (see

Table II. *Ignore* transformation of the dataset in table I

EXAMPLE	$Y \subseteq \mathcal{L}$
2	λ_4

Table III. *Select* transformation of the dataset in table I

(a) Select-max

EXAMPLE	$Y \subseteq \mathcal{L}$
1	λ_2
2	λ_4
3	λ_2
4	λ_3
5	λ_2

(b) Select-min

EXAMPLE	$Y \subseteq \mathcal{L}$
1	λ_1
2	λ_4
3	λ_4
4	λ_1
5	λ_3

(c) Select-random

EXAMPLE	$Y \subseteq \mathcal{L}$
1	λ_1
2	λ_4
3	λ_3
4	λ_1
5	λ_2

Table IV. *Copy* and *copy-weight* transformation of the dataset in table I

EXAMPLE	$Y \subseteq \mathcal{L}$	WEIGHT
1	λ_1	0.33
1	λ_2	0.33
1	λ_4	0.33
2	λ_4	1.00
3	λ_2	0.33
3	λ_3	0.33
3	λ_4	0.33
4	λ_1	0.50
4	λ_3	0.50
5	λ_2	0.50
5	λ_3	0.50

Table II) which consists of ignoring all multi-label instances. It is a very simple idea, but useless due to its great drawback: the loss of information.

Another simple transformation is selecting one of the labels of those instances with more than one label, this method is called *select* and it also produces some information loss. Depending on the method used to select the label of the instance it can be distinguished between the *select-max* (the most frequent label), *select-min* (the less frequent label) and *select-random* (a random selection) methods (see table III).

The latest simple transformation method is called *copy* (see table IV). It consists of transforming every multi-label instance into several ones, one per label. It is also possible to weight examples by $\frac{1}{|Y|}$, in which case it is called *copy-weight*. This last method does not produce information loss but it increases the number of patterns and may complicate modelling decision boundaries [Read 2010].

4.1.2. Binary Relevance methods. The *Binary Relevance* (BR) method generates one binary dataset for each label where positive patterns are those predicting the label and the rest are considered to be negative patterns (an example is shown in table V). Once an unknown pattern is presented to the model, the output will be the set of positive classes predicted. This approach is similar to the *one-versus-all* (OVA) approach employed to solve multi-class problems with binary classifiers with the difference that in multi-class problems an instance has only one label associated. In [Zhou et al. 2012a] three problems of BR are described. The first one is the fact that BR assumes labels are independent so it ignores correlations and interdependences between labels and this is not always true. According to Read et al. [2011], due to this information loss, BR pre-

Table V. *BR* transformation of the dataset in table I

EXAMPLE	λ_1 vs. rest
1	true
2	false
3	false
4	true
5	false

EXAMPLE	λ_2 vs. rest
1	true
2	false
3	true
4	false
5	true

EXAMPLE	λ_3 vs. rest
1	false
2	false
3	true
4	true
5	true

EXAMPLE	λ_4 vs. rest
1	true
2	true
3	true
4	false
5	false

Table VI. *LP* transformation of the dataset in table I

EXAMPLE	$Y \subseteq \mathcal{L}$
1	$\lambda_{1,2,4}$
2	λ_4
3	$\lambda_{2,3,4}$
4	$\lambda_{1,3}$
5	$\lambda_{2,3}$

dictive performance can decrease and also in [Tsoumakas et al. 2009] it is highlighted that BR may fail to predict label combinations or rankings of labels. The second one is the problem of sample imbalance that may occur after the BR transformation. It leads to induce binary classifiers from datasets where negative examples tend to outnumber the positive ones. The last problem is related to the high dimensionality of labels that may increase the sample imbalance and can also increase the number of classifiers to be trained. Despite these drawbacks, BR is simple and reversible (the original dataset can be recovered). In [Read et al. 2011] the main advantages of BR are highlighted. Firstly, its low computational complexity as compared with other methods and the fact that BR scales linearly with the number of labels. Secondly, since labels are independent they can be added and removed without affecting the rest of the model. This makes it applicable to an evolving or dynamic scenario and offers the opportunity of parallel implementation.

4.1.3. Label Powerset methods. The *Label Powerset* (LP) approach, also called *Label Combination* (LC) in [Read et al. 2011], generates a new class for each possible combination of labels and then solves the problem as a single-label multi-class one (an example can be seen in table VI). When a new unknown instance is presented, LP outputs a class, which is actually a set of labels in the original dataset. This approach is effective and simple and is also able to model label correlations in the training data. Nevertheless, after the transformation it is possible to have limited training examples for many new classes (the less frequent combinations), producing a sample imbalance issue. Besides, this approach only takes into account the distinct labelsets in the training set, so it cannot predict unseen labelsets which may be also lead to a tendency to over-fit the training data [Read et al. 2011]. Finally, another problem is the potentially large number of classes which must be dealt with. This number is upper bounded by $min(m, 2^q)$, thus in the worst-case scenario the complexity is exponential with the number of labels. This is the reason why LP typically works well if the original labelset is small, but quickly deteriorates for larger labelsets [Cheng and Hüllermeier 2009].

The *RAndom k-labELsets* method (RAkEL) [Tsoumakas and Vlahavas 2007], [Tsoumakas et al. 2010b] constructs an ensemble of LP classifiers. Each classifier is trained with a random subset of k labels. Thus RAkEL, as LP, is able to deal with label correlations but avoiding the problems of LP related to the computational cost and class imbalance when the number of labels is high. During

Table VII. *RPC* transformation of the dataset in table I

EXAMPLE	λ_1 *vs.* λ_2
3	false
4	true
5	false

EXAMPLE	λ_1 *vs.* λ_3
1	true
3	false
5	false

EXAMPLE	λ_1 *vs.* λ_4
2	false
3	false
4	true

EXAMPLE	λ_2 *vs.* λ_3
1	true
4	false

EXAMPLE	λ_2 *vs.* λ_4
2	false
5	true

EXAMPLE	λ_3 *vs.* λ_4
1	false
2	false
4	true
5	true

classification, when an unknown instance is presented, the response of the classifiers is averaged per label, and after that a threshold is used to assign the labelset. Besides it is able to predict unseen labelsets. It is worth highlighting that although RAkEL is independent of the multi-label algorithm, authors recommend using methods heavily influenced by the specific set of labels such as LP or Pruned Sets (see below) while they recommend against BR and ML-kNN (this last one is heavily influenced by the feature space). Regarding the base classifier, authors performed a study concluding that, in general, C4.5 and Support Vector Machines (SVMs) perform better than naive Bayes (NB). Experiments showed the improvement of RAkEL over BR and LP. Nevertheless, its build time increases by a factor of approximately ten each time k is doubled [Read et al. 2008].

Pruned Problem Transformation (PPT) or *Pruned Sets* (PS) [Read et al. 2008] extends LP transformation but it tries to avoid its problems, related to complexity and unbalanced data, by pruning examples with less frequent labelsets (under a user-defined threshold). This reduces complexity by focusing on the most important combinations of labels. To compensate for such information loss, it reintroduces the pruned example. The *Ensemble of Pruned Sets* (EPS) algorithm [Read et al. 2008] constructs a number of PS by sampling the training sets (i.e. bootstrap). Given a new instance, the final response is obtained by a voting schema and a threshold that allow EPS to form new combinations of labels. The experiments showed that PS performed best in an ensemble scheme (EPS), EPS outperformed LP and RAkEL and proved to be particularly competitive in terms of efficiency.

4.1.4. Pairwise methods (PW). The *Ranking by Pairwise Comparison* (RPC) [Hüllermeier et al. 2008] approach transforms a dataset with q classes into $q(q-1)/2$ binary datasets, one per each pair of labels, and a binary classifier is built for each dataset. Each dataset, $\lambda_i vs. \lambda_j$, contains the patterns labelled with at least one of the two labels, but not both, being a pattern true if λ_i is true and false otherwise (an example is shown in table VII). This approach is similar to the *one-versus-one* (OVO) for multi-class problems. Given a new instance, all models are invoked and a ranking is obtained by the counting votes for each label. The main drawback is the space complexity and the need to query all the generated (q^2) binary models at runtime. According to Read et al. [2011], this quadratic complexity in terms of the number of labels makes RPC very sensitive to large q and usually intractable for large problems.

It is also worth mentioning *Calibrated Label Ranking* (CLR) [Brinker et al. 2006] [Fürnkranz et al. 2008] that extends RPC by means of an additional virtual or calibration label, λ_0, introduced to the original dataset. This virtual label can be interpreted as a split point between relevant and irrelevant labels. Thus the final ranking will include the virtual label that acts as a split point for relevant and non-relevant labels obtaining a consistent ranking and bipartition. The transformation is built by adding to the RPC transformation (in table VII) a new dataset for each label, λ_i, corresponding to the pair $\lambda_i vs. \lambda_0$. Each new dataset uses all the examples so when the label λ_i is true, the virtual label is considered false and vice versa. An ex-

Table VIII. Datasets added to *RPC* to obtain *CLR* transformation of the dataset in table I

EXAMPLE	$\lambda_1 vs. \lambda_0$
1	true
2	false
3	false
4	true
5	false

EXAMPLE	$\lambda_2 vs. \lambda_0$
1	true
2	false
3	true
4	false
5	true

EXAMPLE	$\lambda_3 vs. \lambda_0$
1	false
2	false
3	true
4	true
5	true

EXAMPLE	$\lambda_4 vs. \lambda_0$
1	true
2	true
3	true
4	false
5	false

Table IX. *CC* transformation of the dataset in table I. Chain $\lambda_1, \lambda_2, \lambda_3, \lambda_4$

EXAMPLE	$\lambda_1 vs. rest$
1	true
2	false
3	false
4	true
5	false

EXAMPLE$\cup\lambda_1$	$\lambda_2 vs. rest$
1 true	true
2 false	false
3 false	true
4 true	false
5 false	true

EXAMPLE$\cup\lambda_1 \cup \lambda_2$	$\lambda_3 vs. rest$
1 true true	false
2 false false	false
3 false true	true
4 true false	true
5 false true	true

EXAMPLE$\cup\lambda_1 \cup \lambda_2 \cup \lambda_3$	$\lambda_4 vs. rest$
1 true true false	true
2 false false false	true
3 false true true	true
4 true false true	false
5 false true true	false

ample of the datasets corresponding to the pairwise comparisons of the virtual label added to the RPC transformation is shown in table VIII. Experiments carried out in the fields of text categorization and gene analysis concluded that CLR outperformed the BR approach [Fürnkranz et al. 2008]. Nevertheless, the space complexity of the model is similar to RPC but needs to query $q^2 + q$ binary models. Alternatives to decrease the complexity of the voting process have been proposed in [Madjarov et al. 2011] and [Loza et al. 2009].

4.1.5. Transformations for Identifying Label Dependences. The *Classifier Chains* (CC) model [Read et al. 2011] generates q binary classifiers, but they are linked in such a way that the feature space of each link in the chain is extended with the labels associations of all previous links (see table IX). Thus CC overcomes the label independence assumption of BR whilst overcoming the worst-case computational complexity of LP (exponential with the number of labels). On one hand, when labels are independent, CC will tend to function similarly to BR while, on the other hand, given the presence of label correlations, despite not being optimal, it will tend to function like LP. As the order of the chain itself can influence the performance, authors proposed using an *Ensemble of Classifier Chains* (ECC) that trains a set of CC classifiers with a random chain ordering and a random subset of training patterns. The sum of votes for each label is computed and normalised, the output of the classifier being those labels that exceed a threshold. A Bayes optimal way of forming classifier chains based in probability theory, dubbed *Probabilistic Classifier Chains* (PCC), was described in [Dembczyński et al. 2010]. It tests all possible chain orderings and predicts $\arg\max_{Y \subseteq \mathcal{L}} P(Y|\mathbf{x})$. Despite obtaining better accuracy than CC, as PCC has to look at each of 2^q possible combinations at prediction stage, the applicability of the algorithm is only advisable to data sets with a small to moderate number ($q \leq 15$).

To overcome the problems of the independence assumption of the BR model, some authors have proposed *2BR*, which basically consists of applying BR twice. During the first step a BR classifier is learned and the second BR step implements a meta-learning stage. There will be a binary meta-learner for each label to be learnt. The input will be the output of all the BR classifiers in the first step plus the desired output. It follows the philosophy of *stacking* proposed by [Wolpert 1992] and maintains the linear time complexity with respect to the number of labels in the dataset. In [Tsoumakas et al. 2009], during training, predictions of the base-level models only participated on

Table X. Frequency counts of labels λ_i and λ_j

	λ_j	$\neg\lambda_j$	total
λ_i	a	b	a+b
$\neg\lambda_i$	c	d	c+d
total	a+c	b+d	a+b+c+d

those labels whose absolute value of the ϕ coefficient was greater or equal to a certain threshold $t, 0 \leq t \leq 1$. Given two labels, λ_i and λ_j, and the frequency counts of their co-occurrences (see table X) the ϕ coefficient is obtained by formula 16. Experiments showed that the pruning substantially improved computational cost while maintained or improved predictive performance.

$$\phi = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}} \tag{16}$$

In [Tenenboim-Chekina et al. 2010] LPBR is proposed which tries to find the optimal trade-off between the simplicity of BR and the complexity of the LP, and to find a balance between independence assumption of BR and few examples for many labels of LP datasets. It manages this target by combinations of rounds of LP and BR. A first round with BR is applied and the most dependent labels are clustered into a new label and a new multi-label classification model is induced. LP is applied within the groups of dependent labels (applied to a group with a limited, potentially small number of labels) while BR is applied to the independent groups of labels. The approach where the dependence between labels is computed by using the χ^2 score (see formula 17) is called *ChiDep*.

$$\chi^2 = \frac{(ad - bc)^2(a + b + c + d)}{(a+b)(c+d)(b+d)(a+c)} \tag{17}$$

To improve performance, a method to develop a *ChiDep Ensemble* (CDE) is proposed. A large number of random labelsets partitions are generated (i.e. 10000), and a score based on the χ^2 score of all label pairs in each partition is computed. The top high-scoring partitions are selected as members of the ensemble. Experiments showed a predictive performance higher than BR and LP, nevertheless both ChiDep's and CDE's train times were relatively long and approximate to that of RAkEL and 2BR.

4.2. Algorithm adaptation methods

Almost all classical paradigms in classical or single-label classification have been revisited in order to be adapted to multi-label data.

4.2.1. Decision trees. Decision tree methods have been mainly used in the field of genomics due to the interpretability of its outputs and in hierarchical [Blockeel et al. 2006] [Vens et al. 2008] and ensemble settings [Madjarov et al. 2012]. It is worth citing ML-C4.5, the adaptation of the popular C4.5 [Quinlan 1993] carried out by Clare and King [2001]. Multiple labels in the tree's leaves were allowed and the entropy definition was adapted in order to take into account how much information was needed to describe what classes a certain pattern belonged to (see formula 18). So, given q classes, not only the probability of membership, $p(\lambda)$, is considered but also the probability of not membership, $1-p(\lambda)$. These probabilities are measured in terms of relative frequency. When the algorithm checks whether is better pruning a branch and replacing it by a leaf, the most frequent set of classes in the branch is found (rather than the best single class) and how many items have this set of classes.

$$entropy_{ML}(S) = \sum_{i=1}^{q} P(\lambda_i)\log(P(\lambda_i)) + (1 - P(\lambda_i))\log(1 - P(\lambda_i)) \quad (18)$$

Predictive Clustering Trees (PCT) [Blockeel et al. 1998] consider a decision tree as a hierarchy of clusters where data is partitioned in a top-down strategy by minimizing the variance. The leaves represent the clusters and are labelled with its cluster's prototype (prediction). Unlike standard decision trees, the variance and the prototype functions are treated as parameters. Particularly, in MLL the variance function is computed as the sum of the Gini indices [Breiman et al. 1984] of the variables from the target tuple and the prototype function returns a vector with probabilities for each label [Madjarov et al. 2012]. It has been used in hierarchical multi-label learning [Vens et al. 2008] (see section 7) and obtains competitive performance as base classifier on random forest ensembles (see section 5).

4.2.2. Support Vector Machines. Single-label SVMs have been widely used in MLL by applying an OVA approach [Gonçalves and Quaresma 2004] [Boutell et al. 2004].The algorithm adaptation approach has also been used. In [Elisseeff and Weston 2001] the authors proposed a SVM ranking based algorithm called *Rank-SVM* that improved performance over BR with SVMs. A set of q linear classifiers, $\{h_j(\mathbf{x}) = \langle w_j, \mathbf{x}\rangle + b_j = w_j^T \cdot \mathbf{x} + b_j | 1 \leq j \leq q\}$, each with weight vector, w_j, and bias, b_j, are defined. They are optimized to minimize the empirical ranking loss with quadratic programming and use kernel trick to manage non-linearity. The learning multi-label margin on the whole training set (formula 19) considers its capability to properly rank every relevant-irrelevant label pair for each training example, $(\mathbf{x}_i, Y_i)$, in the training set, S. The boundary for each pair of relevant-irrelevant labels corresponds to the hyperplane $\langle w_j - w_k, \mathbf{x}_i\rangle + b_j - b_k$. Improvements of this method have been presented in [Jiang et al. 2008] and [Xu 2012].

$$\min_{(\mathbf{x}_i, Y_i) \in S} \min_{(y_j, y_k) \in Y_i \times \overline{Y_i}} \frac{\langle w_j - w_k, \mathbf{x}_i\rangle + b_j - b_k}{\parallel w_j - w_k \parallel} \quad (19)$$

4.2.3. Instance based algorithms. As far as we know, the first multi-label lazy learning algorithm is *multi-label k-nearest neighbour* (ML-kNN) proposed by Zhang and Zhou [2005]. Given an unknown instance, $\mathbf{x}$, the algorithm first determines $N = \{(\mathbf{x}_i, Y_i) | 1 \leq i \leq k\}$, the set of k nearest neighbours and obtains a membership counting vector, $\mathbf{c} = (c_1, \ldots c_q)) | c_j = \sum_{(\mathbf{x}_i, Y_i) \in N} [\![\lambda_j \in Y_i]\!]$, that stores, for each label, the number of examples in the neighbourhood of $\mathbf{x}$. Then, based on prior and posterior probabilities for the frequency of each label within the k nearest neighbours, it identifies the set of labels to be associated to the unseen instance by using the *maximum a posteriori* (MAP) principle (formula 4.2.3).

$$y_j = \begin{cases} 1 & if\ P(c_j | y_j = 1)P(y_j = 1) \geq P(c_j | y_j = 0)P(y_j = 0) \\ 0 & otherwise \end{cases}$$

Cheng and Hüllermeier [2009] proposed *Instance Based Learning by Logistic Regression* (IBLR), an approach that combines instance-based learning (IBL) and logistic regression. The key idea is to consider labels of neighbour instances as features of unseen samples and to reduce IBL to logistic regression. This approach is able to capture interdependences between labels that are reflected by the sign and magnitude of the regression coefficients improving upon ML-kNN. Experiments showed IBLR outperformed the predictive accuracy of LP, MLkNN and BR with kNN as base classifier.

Together with ML-kNN it can be considered state-of-the-art in MLL by instance based approaches. Finally, in [Spyromitros et al. 2008], BR*k*NN was described, equivalent to using BR with kNN as the base classifier, but much faster because instead computing q times the k nearest neighbours, it searches the k nearest neighbours only once.

4.2.4. Neural Networks. Crammer and Singer [2003] proposed *Multi-label Multi-class Perceptron algorithm* (MMP). Just as in BR, one perceptron is used for each label and the prediction is calculated via the inner products. Nevertheless, instead of learning the relevance of each class independently, MMP is incrementally trained to produce a real-valued relevance score that ranks relevant labels above the irrelevant ones. So that the performance of the whole ensemble is considered to update each individual perceptron. Studies have demonstrated it is efficient, competitive and suitable for solving large-scale multi-label problems [Loza and Fürnkranz 2007].

Later, Zhang and Zhou [2006] developed *Backpropagation for Multilabel Learning* (BP-MLL) an adaptation of the traditional multilayer feed-forward neural network to multi-label data. The net is trained with gradient descendent and error backpropagation with an error function closely related to the ranking loss that took into account the multi-label data (see formula 20).

$$E = \sum_{i=1}^{m} \frac{1}{|Y_i||\overline{Y_i}|} \sum_{(j,k)\in Y_i \times \overline{Y_i}} exp(-(o_j^i - o_k^i)) \tag{20}$$

where $o_j^i - o_k^i$ measures the difference between the outputs of the network on one label belonging to the i-th pattern and one label not belonging to it. The network architecture has three layers. The input layer consists of d units corresponding each one to one dimension in the input space. The output layer has q units whose output will be used for label ranking (i.e. the labels belonging to an instance should be ranked higher than those not belonging to it). The hidden feed forward layer is fully connected with the input and output layers with weights. Experimental results showed competitive performance in genomics and text categorization domains with a computational cost according to neural networks methods.

Finally, in [Zhang 2009] *Multi Label Radial Basis Function* (ML-RBF), an approach inspired by the well-known RBF method, was presented. The input corresponds to a d-dimensional feature vector. It consists of two layers of neurons: in the first layer, each hidden neuron (basis function) is associated with a prototype vector while each output neuron corresponds to a possible class. The network is trained by means of a two-stage procedure. Firstly, basis functions in the hidden layer are learnt by performing k-means clustering on instances of each possible class. So that the centroids of the clustered groups will constitute the prototype vectors of the first-layer basis functions. After that, the weights of the second-layer are optimized through minimizing a sum-of-squares error function. It is worth noting that each output neuron is connected with all basis functions corresponding to the prototype vectors of all possible classes. Therefore, the correlations between different classes is addressed both in training and test.

4.2.5. Generative and probabilistic models. Many of the approaches for multi-label document classification mainly rely on discriminative modelling techniques; nevertheless, some generative models have also been devised. In [McCallum 1999] a probabilistic generative model for text classification was presented. It assumes that, associated with each individual label is a word distribution $P(w|\lambda)$, for all words in the vocabulary, $\mathcal{V} = \{w_1, \ldots, w_d\}$. So, a document is generated by a mixture of these word distributions, with mixture weights, $\gamma^Y = (\gamma^Y_{\lambda_1}, \ldots, \gamma^Y_{\lambda_q})$. Given a document, $\mathbf{x}$, it is associated

a labelset, Z, according to formula 21.

$$Z = \arg\max_{Y \subseteq \mathcal{L}} P(Y) \prod_{w \in \mathbf{x}} \sum_{\lambda \in Y} \gamma_\lambda^Y P(w|\lambda) \tag{21}$$

where $P(Y)$ and $P(w|\lambda)$ are estimated from the training set and γ_λ^Y is the mixture weight of label λ in mixture weight distribution γ^Y estimated with expectation-maximization (EM) [Dempster et al. 1977]. Later Ueda and Saito [2002a] presented PMM1 and PMM2, two probabilistic generative *Parametric Mixture Models*. The basic assumption under PMMs is that multi-labelled text has a mixture of characteristic words appearing in single-labelled text that belong to each category of the multi-categories. As the described generative models are based on text frequencies in documents, they are specific for text domains.

The use of *Conditional Random Fields* (CRFs) [Lafferty et al. 2001] has been proposed in [Ghamrawi and McCallum 2005] with two multi-label graphical models for classification, that parametrise label co-occurrences and in [Shotton et al. 2009] that used CRFs to incorporate different low-level image features. Finally, in [Zhang et al. 2009] a method called *Multi-label Naive Bayes* (MLNB) was presented that adapted the NB classifier to deal with multi-label instances. Using the Bayesian rule and adopting the assumption of class conditional independence among features, as classic naive Bayes, given a test instance, x, the MAP estimate of one is computed as in formula 4.2.5. The density of the features variables, conditioned on the class values, follows a Gaussian Distribution $g(x_k, \mu_k^{jb}, \sigma_k^{jb}), 1 \leq k \leq d$.

$$y_j = \begin{cases} 1 & if\ P(y_j = 1)\exp(\phi_1) \geq P(y_j = 0)\exp(\phi_0) \\ 0 & otherwise \end{cases} \quad where\ \phi_b = -\sum_{k=1}^{d} \frac{\left(x_k - \mu_k^{jb}\right)^2}{2{\sigma_k^{jb}}^2} - \sum_{k=1}^{d} \ln \sigma_k^b$$

4.2.6. Associative classification. Associative classification integrates association rule and classification. One of the first approaches to MLL was *multi-class, multi-label associative classification* (MMAC) [Thabtah et al. 2004]. It scanned the training data first to discover and generate an initial set of classification rules by association rule mining. Next, an iterative process learnt rules from the remaining unclassified instances, until no further frequent items were left. Finally, the rules sets derived at each iteration were merged to form a multi-label classifier. Other associative approaches are found in [Thabtah and Cowling 2007] and [Rak et al. 2008].

4.2.7. Evolutionary approaches. Bio-inspired approaches have also been used to solve multi-label problems. To the best of our knowledge the first one, called *Multi-Label Ant-Miner* (MuLAM), was proposed by Chan and Freitas [2006]. It was an extension of the ant colony-based Ant-Miner algorithm [Parpinelli et al. 2002]. The rule representation allowed more than one predicted class in the rule consequent and each ant was able to discover a set of rules, at least one rule and at most a rule for each class. In [Ávila et al. 2011] GEP-MLC, an evolutionary approach to find discriminant functions was proposed. Later, the same authors proposed GC [Ávila et al. 2010], another evolutionary approach to build classification rules, a model more interpretable than discriminant functions. Both proposals obtained results competitive with the state-of-the-art in MLL.

4.2.8. Ensembles. Schapire and Singer [1999; 2000] proposed a set of boosting algorithms for text categorization adapted to the multi-label case that were based in the popular AdaBoost [Freund and Schapire 1997], namely *AdaBoost.MH* and *AdaBoost.MR*. The aim of AdaBoost.MH is to minimise Hamming loss and maintains

a set of weights not only over the training examples, but also over labels. During training, weights related to labels and examples difficult to classify are increased. This algorithm carries out a reduction of the multi-label problem mapping each example, $(\mathbf{x_i}, Y_i)$, to q binary examples, one for label in the dataset and then binary AdaBoost is applied to the binary data. The output of each weak learner is a hypothesis $h : \mathcal{X} \times \mathcal{L} \rightarrow \mathbb{R}$, being the sign of the output interpreted as a prediction on the relevance of the label and the magnitude interpreted as a measure on the confidence of the prediction. In the more standard setting of AdaBoost.MH, the predictions of the weak hypothesis are forced to be either -1 or $+1$. Given a ensemble of T base classifiers, the final output is $f(\mathbf{x}, \lambda) = \sum_{i=1}^{T} \alpha_t h_t(\mathbf{x}, \lambda)$. A discussion on the values of α_t can be found in [Schapire and Singer 2000]. On the other hand, AdaBoost.MR operates in a similar way. Nevertheless, as its aim is to minimise the ranking loss, it has to take into account all labels misorderings, thus the set of weights is maintained for each instance and pair of labels. Some works based on these popular algorithms can be found in [Sebastiani et al. 2000], [Nardiello et al. 2003] and [De Comité et al. 2003].

Several ensemble-based approaches that work at feature and data level have been presented. An example is a boosting-type algorithm, called *model-shared subspace boosting* (MSSBoost) [Yan et al. 2007], where each model was learnt from random feature subspace and bootstrap data sampling. It exploited the label space redundancy by allowing base models to be shared across multiple labels.

4.3. Ensembles of MLL methods

Madjarov et al. [2012] consider ensemble methods whose base classifiers are multi-label techniques are a special group different from both problem transformation and algorithm adaptation because they are developed on top of these two approaches. Therefore RAkEL, EPS, ECC, EPCC and CDE (section 4.1) can be considered problem transformation ensembles. Regarding algorithm adaptation ensembles, it is worth citing *Random forest of predictive clustering tress* (RF-PCT) in [Kocev et al. 2007] [Kocev 2012] and *Random forest of ML-C4.5* (RF-C4.5) [Madjarov et al. 2012], two ensemble methods that used PCT [Blockeel et al. 1998] and ML-C4.5 trees as base classifiers respectively. The diversity of base classifiers was obtained by bagging and changing the feature set during learning (i.e. a random subset of features was considered to select the best split attribute). In this tutorial we consider ensembles of MLL methods are different from ensembles on section 4.2.8 as the latter, one way or another, decompose the problem into a set of binary single-label ones while ensembles of MLL methods use directly multi-label data and have, as base classifier, a multi-label learner.

4.4. Thresholding strategies

Many of the described algorithms output a score (e.g. probability, ensemble votes, etc.) or a ranking but obtaining a bipartition is needed.As has been mentioned in section 2.2 a bipartition of labels can be obtained from a ranking by means of a threshold. The simplest approach is using a predefined threshold, t, and given a new instance a label will be considered relevant if its score is greater than t. This value can be user defined (i.e. 0.5) or previously tuned by a validation set (inefficient). This approach is called *one threshold* (OT) in [Ioannou et al. 2010].

Another simple approach is *RCut* [Yang 2001], a *ranking-based* strategy that assigns the t top ranked categories where $1 \leq t \leq q$ can be either specified by the user (a common value is the label cardinality of the dataset [Tang et al. 2009]) or tuned by means of a validation set. Note that when $t = 1$ the output is single-label. According to [Montejo-Ráez and Ureña López 2006] it is not a good approach because classes that were refused by the binary classifier can be selected and equally, some classes

that were found positive by the classifier may be discarded. Yang [2001] also noted that it tends to over fit and to be unstable across datasets.

5. EXPERIMENTAL COMPARISONS OF MLL METHODS

Despite the number of proposals for MLL, the development of exhaustive experimental comparative studies to get a better understanding of different MLL algorithms is still an open topic. It is worth citing an extensive comparison with significance statistical test involving 12 MLL methods, 16 evaluation measures and 11 benchmark datasets with different scale and from different domains (i.e. biology, multimedia and text). The best overall methods were RF-PCT and HOMER (see HOMER in section 7), followed by BR and CC. Regarding the base classifiers, SVMs and decision trees (i.e. J48 and PCT) were used. As SVMs are able to exploit the information of all the features they performed better on datasets with a large number of features but small number of examples, while tree-based methods performed better with larger datasets. ML-kNN performed poor across all evaluation measures. Considering efficiency, tree-based methods were faster in train and prediction time. In [Chekina et al. 2011] a meta-learning approach[1] based on datasets' metafeatures was used to recommend the best MLL algorithm to be used over a certain domain was proposed. The study involved 11 algorithms, 12 datasets and 18 measures. HOMER, BR, ECC and EPS obtained the best predictive performance results and a set of measures specific for multi-labelled data were found to be relevant for recommending an algorithm (e.g. number of labels, label cardinality of the training set, average examples per class, number of unconditionally dependent label pairs etc.) Results of those studies shed some light on which algorithm to select or which algorithms must be taken into account when developing a new one. The final decision will depend on the problem to face and the requirements to satisfy: efficiency, flexibility, predictive performance, interpretability of the model, etc.

6. APPLICATIONS OF MLL

Although the term multi-label was not yet used, first works in MLL were related to text categorization, [Yang 1999], [McCallum 1999], [Schapire and Singer 2000]. Later many other applications, mainly related to bioinformatics and classification of multimedia, arose. Recently MLL has been applied to an increasing number of new applications. A summary of the main fields of applications of MLL is described below.

6.1. Text categorization

The problem of text categorization basically consists of assigning a set of predefined categories to documents in order to make certain tasks faster and cheaper. Since a document can belong simultaneously to more than one category it can be considered a multi-label problem. Document classification has been applied to many domains. For example, in [Loza and Fürnkranz 2008], [Loza and Fürnkranz 2010] the authors studied the problem of assigning documents of the EUR-Lex database of legal documents (treaties, legislation, case-law, legislative proposals, etc.) of the European Union to a few of 4000 possible labels. Due to the fact that the number of web documents is increasing day by day, another domain is the automatic categorization of web documents [Rubin et al. 2012], [Ueda and Saito 2002a]. Automatic document categorization has also been applied in the fields of news [Schapire and Singer 2000], research papers [Nguyen et al. 2005] and economics [Vogrincic and Bosnic 2011]. It is worth to highlighting other applications and fields of interest where the manual categorization of documents to facilitate IR needs a lot of time and electronic and human resources:

[1]Only one algorithm was recommended but, meta-learning for recommendation of algorithms may be also a multi-label problem

— *Document indexing* is the task of assigning a document a set of keywords from a controlled vocabulary (thesaurus or ontology) in order to describe its content. In [Lauser and Hotho 2003] MLL was applied to an extensive document base maintained by the Food and Agriculture Organization (FAO) of the United Nations (UN).
— *Tag suggestion* is the automated process of suggesting useful and informative tags or keywords to an emerging object based on historical information. Examples are found in [Song et al. 2011] and [Katakis et al. 2008].
— *Medical coding* is the process of transforming information contained in patient medical records into standard predefined codes (an example is the ICD-9-CM). As each document can be assigned to one or more codes the problem can be considered from MLL perspective [Yan et al. 2010].
— *IR from narrative clinical text*. In [Spat et al. 2008] MLL was used to classify clinical texts into a set of categories referring to medical fields (*surgery*, *radiology*, etc.).
— *Economic activities classification* [Ciarelli et al. 2009]. This task consists of finding a correspondence between a contract, which contains a description of the business activities of one company, and a set of standard categories.
— *Patent classification*. As a patent document may be associated with several categories the problem can be considered multi-label [Cong and Tong 2008].
— *E-mail filtering*. Yearwood et al. [2010] applied MLL to obtain phishing profiles from emails where profiles were generated based on the predictions of the classifier.
— *Classifying news sentences into multiple emotion categories* may be used to design intelligent interfaces. In [Bhowmick et al. 2010] a set of news sentences were categorized according to the emotions triggered on readers (*disgust*, *happyness*, etc.).
— *Aeronautics reports*. The Aviation Safety Reporting System database (ASRS) to detect anomalies contains reports submitted by the flight crews regarding events that took place during a flight. As a report may belong to multiple classes, in [Oza et al. 2009] the problem has been solved using MLL.
— *Query categorization*. In applications such as Internet portals and search engines it is useful to categorize the user search queries into a set of relevant classes to deliver to users content and ads that are relevant to their interests [Tang et al. 2009].

6.2. Multimedia

With the exponential growth of digital multimedia resources whose manual annotation requires great effort, MLL techniques have been applied including many types of resources such as images, videos and sound. The main applications are listed below.

— *Automatic image and video annotation*. An image can have simultaneously associated several tags making this problem a multi-label one [Wang et al. 2008], [Tahir et al. 2009]. This application is also called semantic scene classification. On the other hand, video annotation consists on assigning several semantic concepts to a video [Dimou et al. 2009], [Wang et al. 2010].
— *Face verification* consists of determining whether different images correspond to the same person. In order to tackle this target, in [Kumar et al. 2009] a set of MLL classifiers return the presence of certain visual features or traits in the picture.
— *Object recognition* is the automatic detection, recognition, and segmentation of object classes in pictures. Therefore, given a picture, the system is able to automatically find semantic regions, each labelled with an object class [Shotton et al. 2009].
— *Detection of emotions in music* is a MLL problem where songs are classified simultaneously in several categories [Trohidis et al. 2008] and [Ma et al. 2009]. It has applications such as music recommendation systems or music therapy.
— *Speech emotion classification*. This problem consists of inferring affective states (such as emotions, mental states, attitudes, etc.) from non-verbal expressions in

speech. These affective states can occur simultaneously. It has applications on fields such as human-computer and human-robot interfaces, and public speaking skills assessment. It has been solved with MLL in [Sobol-Shikler and Robinson 2010].
— *Music metadata extraction* is a MLL problem consisting of automatically extracting perceptive information such as genre, mood or main instruments from acoustic signals [Pachet and Roy 2009].

6.3. Biology

In the field of biology it is worth noting the following problems where MLL has been successfully applied.

— *Gene function prediction* consists of assigning functions for unknown genes based on diverse large-scale data. Each gene may be associated with not one, but a set of functional classes, e.g. in the Yeast dataset the gene *YAL062w* belongs to several different function classes like metabolism, transcription, and protein synthesis [Elisseeff and Weston 2001], [Barutcuoglu et al. 2006], [Skabar et al. 2006], [Zhang and Zhou 2006].
— *Protein function prediction*. As proteins often have multiple functions, MLL has been applied in [Diplaris et al. 2005] and [Chan and Freitas 2006].
— *Protein sub-cellular multi-location*. The localizations of a protein in a cell are important functional attributes. As proteins may simultaneously exist at, or move between, two or more different subcellular locations, MLL has been applied in [Yang and Lu 2006] and [Chou et al. 2011].
— *Predicting proteins 3D structures*. Proteins fold into a specific 3D structure that determines their functions in the cell. In [Duwairi and Kassawneh 2008] a multilabel classifier is developed where there are two or more class labels to be predicted, and a hierarchical classifier is able to predict the structural classes and folds of proteins simultaneously, as in the natural hierarchy of proteins itself.

6.4. Chemical analysis

The main applications of MLL in the field of chemical data analysis are detailed next.

— *Drug discovery*. In [Kawai and Takahashi 2009] MLL has been used to identify drugs that have two or more different biological actions (e.g. dual action antihypertensive drugs).
— *Vision Based Metal Spectral Analysis*. A spectrum could contain emissions from multiple elements. Thus in [Ukwatta and Samarabandu 2009] spectral images are processed in real-time in order to detect contaminants in machine lubricants.
— *Adverse Drug Reactions* (ADRs). MLL has been applied to predict reactions given a set of drugs, and for identifying the most likely drugs responsible for given reactions [Mammadov et al. 2007].

6.5. Social network mining

The application of MLL to classification problems in social networks has become a new area of interest. The following applications can be highlighted:

— *Collective behaviour learning*. It consists of inferring behaviour or preferences of unobserved individuals. Here, behaviour can include actions as: joining a group, connecting with someone, clicking on an ad, becoming interested in certain topics, etc. This problem is dealt with in [Tang and Liu 2009] from an MLL perspective.
— *Social networking advertising*. In [Krohn-Grimberghe et al. 2012] an MLL approach is developed for predicting a list of items ranked in order to make recommendations.

— *Automatic annotation*. In [Peters et al. 2010] the task of assigning labels for images when users and images are connected through multiple relations (e. g. authorship, friendship, etc.) is addressed.

6.6. Other applications

Finally, other fields of applications where MLL has been applied are enumerated.

— *Tagging of Learning Objects* (LO). A learning object can be defined as a minimal content unit that intends to teach something and can be reused on different platforms. Since an LO can be multiple tagged, MLL has been used in [López et al. 2012].
— *Direct Marketing*. On the contrary to mass marketing, which advertises indiscriminately, direct marketing is a process which identifies potential buyers of a certain product in order to become the target for promotions. As one customer can be the target of several products, MLL has applied in in [Zhang et al. 2006].
— *Medical Diagnosis*. In clinical data, a case has many symptoms and they may be associated with more than one syndrome, hence medical diagnosis can be solved by MLL techniques [Shao et al. 2010]. Besides, in [Huang et al. 2008] MLL was proposed to simultaneously segment several significant tissue regions in digitized uterine cervix images. Finally, in [Abb 2013] MLL was applied to the problem of classifying dermoscopy images of skin lesions (skin lesions often contain several pattern lesions).

7. TRENDING CHALLENGES

New trends as scalability, incremental, concept drift and semi-supervised. multi-label data may be expensive to obtain. No label may not imply negative example for one label. As has been shown, MLL is a trending learning paradigm and new unsolved issues continue arising such as:

— *Dimensionality Reduction*. One common feature of multi-label data is the high number of attributes and labels that may influence the efficiency and effectiveness of the algorithms. The *dimensionality reduction of the input space* tries to reduce the number of attributes under two distinct points of view: feature selection (removing irrelevant or redundant features) and feature extraction (compressing dependent variables into a smaller number of predictors). The former has been used in [Yang and Pedersen 1997], [Trohidis et al. 2008], [Chen et al. 2007], [Zhang et al. 2009] while the latter has been used in [Yu et al. 2005] and [Zhang and Zhou 2010]. On the other hand, strategies for the *reduction of the label space* have been also applied. It can be cited *Hierarchy Of Multilabel classifiERs* (HOMER) [Tsoumakas et al. 2008] which generates a tree of classifiers and whose complexity depends on q and other proposals as *Compressed Sensing* (CS) [Hsu et al. 2009] and *Principal Label Space Transformation* (PLST) [Tai and Lin 2010].
— *Label Dependence*. In MLL problems, when the number of labels is high, even moderate, the complexity would become exponential due to the possible combinations of labels. To cope with this issue, correlations between labels could be explored. This issue has been tackled in [Dembczyński et al. 2012] where two types of label dependency in multi-labelled data were identified: conditional (dependent on a particular instance) and unconditional (independent of a certain instance). Particular approaches are found in [Qi et al. 2007], [Zhu et al. 2005], [Ji et al. 2010] and [Zhang and Zhang 2010].
— *Active learning*. There are many applications where data is unlabelled or labelling data is expensive or impractical. The aim of *Active learning* (AL) strategies is to iteratively rank the unlabelled examples in terms of how useful they would be in order to propose only the top-ranked ones to human annotators. In the framework

of MLL, active learning becomes complex, mainly in text and image classification domains, because the effort of assigning several labels and the correlations between labels. A straightforward way is to apply a BR transformation and to solve q binary problems independently [Brinker 2006]. Other approaches take into account label relationships to reduce human effort during the labelling process [Qi et al. 2009]. In [Esuli and Sebastiani 2009] a unique ranking of examples that combines the outputs of the individual binary classifiers is proposed.

— *Multi-instance multi-label learning* (MIML). *Multi-instance learning* (MIL) is a variation on supervised learning, where the task is to find a function $h_{MIL} : 2^{\mathcal{X}} \rightarrow \{0, 1\}$ from patterns, (X, y), consisting of a set of vectors (a bag of instances), $X \subseteq \mathcal{X}$, each labelled as positive or negative, $y \in \{0, 1\}$. While multi-label concerns the ambiguity in the output space, multi-instance concerns the ambiguity of the input space. In *Multi-Instance Multi-Label learning* (MIML) a training example is described by multiple instances and is associated to not one, but a set of labels. So formally, the task of MIML is learning a function $h_{MIML} : 2^{\mathcal{X}} \rightarrow 2^{\mathcal{L}}$ from patterns (X, Y) where $X \subseteq \mathcal{X}$ and $Y \subseteq \mathcal{L}$. Some proposals can be found in [Zhou and Zhang 2006] and [Zhang and Zhou 2007]. As in many MLL problems, labels can be related only to different parts of the object, developing learners on a MIML setting may reduce noise and improve performance [Zhou et al. 2012b]. Two issues are identified as crucial when dealing a MLL problem with a MIML setting. The first one is modelling the connections between instances and labels of a sample, and the second one is exploiting relationships between labels [He et al. 2012]. These two problems are not usually tackled together as the models may become complex and difficult to be solved.

— *Multi-view learning*. The MI, ML and MIML settings only deal with situations where data comes from a single feature set (single-view). Nevertheless there are real world applications in which one object usually has different representations in the form of multiple views (multi-view). Multi-view and multi-label learning have been successfully integrated in [Fang and Zhang 2012]. In addition, multi-view and active learning can be effectively integrated [Wang and Zhou 2008](e.g. recommending for examples in which different views predict different labels). So the combination of multi-view and unsupervised learning could be used to reduce the annotation effort on multi-label learning.

— *Multi-task learning* (MTL). It is a setting in which several tasks that share a common representation are learnt together. This setting is slightly different to MLL (e.g. the input space can be the same, but not the set of instances [Evgeniou and Pontil 2004]) but connections between these two approaches have not been deeply studied. MTL has been integrated with multi-view learning [Zhang and Huan 2012] leading us to think that relationships between multi-label and multi-view learning may offer interesting research opportunities.

— *Hierarchical multi-label classification* (HMC), in contrast to *flat classification*, is another important challenge. In this kind of problem examples can be associated with multiple labels and labels are organized in a hierarchical structure and the classifier should take relationships among categories into account. The hierarchy may have the structure of a tree or a directed acyclic graph (DAG), where a child category may have more than one parent category. Some approaches are found in [Barutcuoglu et al. 2006] and [Vens et al. 2008].

8. CONCLUDING REMARKS

This paper has presented an up-to-date tutorial with a description of the MLL framework and the main areas of application. The main proposals developed have been discussed, including new challenging issues. The paper has also described methodologi-

cal aspects for the evaluation of the models: performance metrics, partitioning datasets and significance tests. Finally, the main resources (datasets, repositories, bibliography and software) for MLL learning have been summarized. We can conclude that MLL has been applied, and demonstrated to be useful, time-saving and effort-saving, in numerous fields such as text, image and video annotation, detection of emotions in music, medical diagnosis, gene and protein function prediction. Its domains of application are still increasing (e.g. speech emotion recognition or social network mining). Moreover, in MLL researchers have found a challenging application, since datasets with a great number of instances, features and labels are widely available. In addition MLL involves facing challenges as relationships between labels, high dimensionality of data, efficiency and even the integration with other learning settings such as multi-instance learning, semi-supervised learning etc. All of these factors make MLL a trending research area within the machine learning and data mining disciplines.

ACKNOWLEDGMENTS

This work is supported by the Ministry of Science and Technology project TIN-2011-22408.

APPENDIX. RESOURCES

The aim of this section is providing a useful list of benchmark datasets, metrics for caracterizing multi-label data and software for MLL. Besides, table XI summarizes other interesting resources on MLL such as bibliographic, research teams, tutorials, workshops, books, special issues and PhD. theses.

A.1. Benchmark datasets

A.1.1. Metrics about datasets. Before carrying out experiments over multi-label data it is important to measure some characteristics of the data set that can influence the performance of the developed proposals. Let $S = \{(\mathbf{x}_i, Y_i)|1 \leq i \leq m\}$ be a multi-label dataset of m instances.

The *label cardinality* (LCard) and *label density* (LDen) [Tsoumakas and Katakis 2007] metrics measure how multi-labelled a dataset is. Cardinality is the average number of labels per pattern (see equation 22) while density is the cardinality divided by the total number of labels and it is used to compare datasets with different numbers of labels (see equation 23).

$$LCard(S) = \frac{1}{m}\sum_{i=1}^{m}|Y_i| \tag{22}$$

$$LDen(S) = \frac{1}{m}\sum_{i=1}^{m}\frac{|Y_i|}{q} = \frac{LCard(S)}{q} \tag{23}$$

In [Tsoumakas et al. 2010b] and [Zhang and Zhang 2010] the *distinct labelsets* (DL) is described as the number of different label combinations in the dataset.

$$DL(S) = |Y \subseteq \mathcal{L}|\exists(\mathbf{x}, Y) \in S| \tag{24}$$

In [Read 2008] and [Zhang and Zhang 2010] the *proportion of distinct labelsets* (PDL) is defined as the number of distinct label subsets relative to the total number of examples.

$$PDL(S) = \frac{DL(S)}{m} \tag{25}$$

The *diversity* [Tsoumakas et al. 2010b] is defined as the percentage of the bound of labels sets that the distinct represents (that is really in the dataset). In [Read 2010]

Table XI. MLL resources

REVIEWS	
Multi-label classification: an overview	[Tsoumakas and Katakis 2007]
Mining Multi-label Data	[Tsoumakas et al. 2010a]
A Tutorial on Multi-label Classification Techniques	[de Carvalho and Freitas 2009]
A Literature Survey on Algorithms for Multi-label Learning	[Sorower 2010]
BIBLIOGRAPHIC COMPILATIONS	
Multilabel Classification - 413 articles	[KDIS 2021]
MLKD - 161 articles	[MLKD 2012]
TUTORIALS	
Multi-label classification (TAMIDA2010)	[Larrañaga 2010]
Learning from multi-label data (ECML/PKDD 2009)	[Tsoumakas et al. 2009]
Advances in Multi-label Classification	[Read 2011]
WORKSHOPS	
1st International Workshop on Learning from Multi-Label Data (MLD'09) ECML/PKDD 2009	[MLD 2009]
2nd International Workshop on Learning from Multi-Label Data (MLD'10) ICML 2010	[MLD 2010]
Extreme Classification: Multi-Class & Multi-Label Learning with Millions of Categories NIPS 2013	[NIP 2013]
The First International Workshop on Learning with Weak Supervision (LAWS'12) ACML 2012	[LAW 2012]
WEB SITES OF RESEARCH TEAMS	
Computational Intelligence group	http://www.uni-marburg.de/fb12/kebi
Jožef Stefan Institute	http://kt.ijs.si/area
Jesse Read's site	http://www.tsc.uc3m.es/ jesse/
KDIS group	http://www.uco.es/grupos/kdis/kdiswiki
Knowledge Engineering group	http://www.ke.tu-darmstadt.de/
KU Leuven Machine Learning group	http://dtai.cs.kuleuven.be/ml/about/
LAMBDA group	http://lamda.nju.edu.cn/MainPage.ashx
Min-Ling Zhang's site	http://cse.seu.edu.cn/people/zhangml/Resources.htm
MLKD group	http://mlkd.csd.auth.gr/multilabel.html
LABIC laboratory	http://computer.njnu.edu.cn/Lab/LABIC/LABIC_index.html
SPECIAL ISSUES AND BOOKS	
Machine Learning. Special Issue on Learning from Multi-Label Data	[ML2 2012]
Multi-Label Dimensionality Reduction	[Sun et al. 2013]
PHD. THESES	
Machine learning and data mining for yeast functional genomics	[Clare 2003]
Large Margin Multiclass Learning: Models and Algorithms	[Aiolli 2004]
Multilabel Classification over Category Taxonomies	[Cai 2008]
Scalable Multi-label Classification	[Read 2010]
Learning with Limited Supervision by Input and Output Coding	[Zhang 2012]
Ensembles for predicting structured outputs	[Kocev 2012]
Modelos de aprendizaje basados en programación genética para Clasificación Multietiqueta	[Ávila 2013]

another two measures are introduced that provide information about the uniformity of the labelling scheme: the first one, the *proportion of unique label combinations* (PUniq) (see equation 26), is the proportion of labelsets which are unique across the total number of examples and the second one, PMax, which represents the proportion of examples associated with the most frequently occurring labelset. It is computed as in equation 27 where *count* is the frequency of Y as a labelset in S.

$$PUniq(S) = \frac{|Y \subseteq \mathcal{L}|!\exists \mathbf{x} : (\mathbf{x}, Y) \in S|}{m} \tag{26}$$

$$PMax(S) = max_{Y \subseteq \mathcal{L}} \frac{count(Y, S)}{m} \quad (27)$$

According to Read [2010], high PUniq indicates irregular labelling and when PMax is also high the data presents *label skew*. In multi-label this means that a relatively high number of examples are associated with the most common labelsets while a relatively high number of examples are associated with infrequent labelsets. Label skew becomes class imbalance when each label is considered separately as a binary problem. A complete summary of meta-features used to characterize datasets can be found in [Chekina et al. 2011].

A.1.2. Benchmark datasets. In this section the main datasets that have been used in MLL are presented. Table XII summarizes their main characteristics[2]. Items in the table are ordered according a somewhat rough overall complexity measure used by Read [2010] and Madjarov et al. [2012], consisting in the product of $instances \times labels \times features$. The double line separates large datasets.

— FLAGS [Gonçalves et al. 2013]. This small dataset contains details of various nations and their flags. The multi-label classification task is to predict the 7 colors that appear on the flags (e.g. *red*, *green*, *yellow*, etc.). It has 194 instances and 19 attributes about area, population, presence of triangles, religion of the country, etc. In [Gonçalves et al. 2013] it was been used for testing an evolutionary algorithm that optimizes labelling ordering in CC transformation.
— EMOTIONS [Trohidis et al. 2008]. Also called MUSIC [Read 2010], it is a small dataset to classify music into the emotions that it evokes according to the Tellegen-Watson-Clark model of mood: *amazed-surprised*, *happy-pleased*, *relaxing-clam*, *quiet-still*, *sad-lonely*, and *angry-aggressive*. It consists of 593 songs, 6 labels and 72 features falling into two categories: rhythmic and timbre. It has been used in research for detecting emotions into music [Trohidis et al. 2008], [Ma et al. 2009], and intensively as a benchmark.
— BIRDS [Briggs et al. 2013]. The goal of this dataset is to predict the set of bird species that are present given a ten-second audio clip. The full dataset consists of 645 audio recordings and 19 species of birds that may be simultaneously vocalizing. It was used in a conference competition [Briggs et al. 2013].
— YEAST [Elisseeff and Weston 2001] contains 103 numeric features about microarray expressions and phylogenetic profiles for 2417 yeast genes. Each gene is annotated with a subset of 14 functional categories (e.g. *metabolism*, *energy*, etc.) from the top level of the functional catalog (FunCat[3]). It has been used in protein and gen function classification [Clare and King 2001], [Diplaris et al. 2005], and intensively as benchmark. It is also a benchmark on HMC [Blockeel et al. 2006], [Barutcuoglu et al. 2006].
— SCENE [Boutell et al. 2004] has 2407 images annotated with up to 6 concepts (e.g. *beach*, *mountain*, etc.). Each one is described with 294 visual numeric features corresponding to spatial colour moments in the LUV space. It is relatively small but widely used as benchmark and for semantic scene classification [Boutell et al. 2004].
— PLANT and HUMAN [Xu 2013a] are two datasets used to predict the subcellular locations of proteins according to their sequences. Some multiplex proteins can simultaneously exist at, or move between, two or more different location sites (e.g. *nucleus*, *golgi apparatus*, *mitochondrion*, etc.). They contain 978 and 3106 protein sequences for plant and human species respectively with 440 numeric features (20

[2]They can be downloaded from http://www.uco.es/grupos/kdis/kdiswiki/index.php/Resources#Download_links
[3]http://mips.gsf.de/proj/yeast/catalogues/funcat/

amino acid, 20 pseudo-amino acid, and 400 dipetide compositions). There are 12 positions or labels for human and 14 for plant.

— COMPUTER AUDITION LAB 500 (CAL500) [Turnbull et al. 2008] is a dataset composed of 502 popular Western songs, represented by 68 acoustic features, each of which has been manually annotated by at least three human annotators, who employed a vocabulary of 174 tags concerning semantic concepts. These tags span 6 semantic categories: instrumentation, vocal characteristics, genres, emotions, acoustic quality of the song, and usage terms (e.g. I would like listen to this song while *driving*). It has been used as benchmark and in semantic annotation and retrieval of music [Turnbull et al. 2008].

— GENBASE [Diplaris et al. 2005] is a dataset for protein function classification. Each of the 662 instances is a protein chain represented using a motif sequence vocabulary of fixed size. Thus, each sequence is encoded as a binary array where each bit is 1 if the corresponding motif is present and 0 otherwise. Each label identifies the functional family of the sequence. It has been mainly used as a benchmark dataset.

— MEDICAL [Pestian et al. 2007] is based on the data made available during the Computational Medicine Center's 2007 Medical Natural Language Processing Challenge[4]. It consists of 978 clinical free text radiology reports labelled ICD-9-CM disease codes. The dataset has 45 codes and 1149 binary attributes representing if a certain term is or not in the report. It has been used as a benchmark dataset.

— SLASHDOT [Read 2010] is a collection of 3782 texts, mined from Slashdot[5], labelled with 22 subject categories (e.g. *enternaiment*, *interviews*, *games*, etc.). Each document has 1079 binary features that represent the presence of a term. It has been used as benchmark and in the field of multi-label data streams [Read et al. 2010].

— ENRON [Read et al. 2008] is a subset of the Enron email text corpus[6]. It is based on a collection of emails exchanged between the Enron Corporation employees, which were made available during a legal investigation. It contains 1702 emails that were categorized into 53 topic categories, such as *company strategy*, *humor* and *legal advice*. It has been mainly used as benchmark dataset.

— LANGUAGELOG (LANGLOG) [Read 2010] is a dataset with 1460 instances and 1004 binary features and it was compiled from the Language Log Forum[7], which discusses various topics relating to language (primarily English). It has 75 topics (e.g. *prepositions*, *punctuation*, *relative_clauses*, etc.) and has been used as benchmark.

— REUTERS [Lewis et al. 2005]. The Reuters-RCV1 dataset is a well-known benchmark for text classification methods [Sebastiani 2002], [Rubin et al. 2012]. Lewis, et al. [Lewis et al. 2005] made some corrections to the RCV1 dataset resulting a new dataset called RCV1-v2. It has 5 subsets each one with 6000 news articles assigned into one or more of 101 topics. In [Read 2010] the attribute space was reduced to 500. Reuters-21578 [Hettich and Bay 1999] is other Reuters dataset commonly used in MLL [Godbole and Sarawagi 2004], consisting of a set of 21578 news stories that appeared on the Reuters news wire in 1987. Is has been also used as benchmark of MLL algorithms and in HMC [Cesa-Bianchi et al. 2006].

— 20 NEWSGROUP (20NG) [Lang 2008] is a compilation of 19300 posts to 20 different newsgroups. The names of the groups are the labels 22 possible labels (e.g. *rec.motorcycles*, *sci.electronics*, etc.) and binary attributes represent the presence or absence of a word in the post. Around 1000 posts are available for each of group. It has been mainly used as benchmark dataset [Read 2010].

[4]http://www.computationalmedicine.org/challenge/
[5]http://slashdot.org
[6]http://www.cs.cmu.edu/ enron/
[7]http://languagelog.ldc.upenn.edu/nll

— MEDIAMILL [Snoek et al. 2006]. Is a multimedia dataset for generic video indexing which was extracted from the TRECVID 2005/2006 benchmark[8]. This benchmark dataset contains 85 hours of international broadcast news data categorized into 101 semantic concepts (e.g. *car*, *golf*, *bird*, etc.), and each video instance is represented as a numeric vector of 120 features including visual and textual information. It has been also used in the field of multi-label data streams [Read et al. 2010].
— ECCV2002 OR COREL5K [Duygulu et al. 2002] is based on 5000 Corel images, 4500 of which are used for training and the rest 500 for testing. Images were segmented and then only regions larger than a threshold were clustered into 499 blobs using k-means, which are the features used to describe the image. JMLR2003 OR COREL16K [Barnard et al. 2003] is derived from ECCV2002 by eliminating infrequent labels. It is a popular benchmark for image annotation and retrieval [Nasierding and Kouzani 2010] and has been also used in MIML [Zhou and Zhang 2006] .
— BIBTEX [Katakis et al. 2008] is based on the data of the ECML/PKDD 2008 discovery challenge. This benchmark dataset contains 7395 bibtex entries from the BibSonomy[9] social bookmark and publication sharing system, annotated with a subset of the tags assigned by users (e.g. *statistics*, *quantum*, *data mining* etc.). The title and abstract of entries were used to construct features using the boolean bag-of-words model. It as been also used for text classification [Katakis et al. 2008].
— YAHOO! [Ueda and Saito 2002a] is a dataset to categorize web pages and consists of 11 of the 14 top-level categories of the Yahoo! directory. About 30-45% of the pages were multi-labelled over the 11 text classification problems. Attributes represent the presence of a word in the document. It has been used as benchmark and in the field of text categorization [Ueda and Saito 2002a], [Rubin et al. 2012].
— IMDB [Read 2010] contains 120919 movie plot text summaries from the Internet Movie Database [10], labelled with one or more genres (e.g. *comedy*, *western*, *documentary*, etc). The dataset has 120919 instances and 1001 binary attributes that follow the binary bag-of-words model. It has been used as benchmark and also in the field of multi-label data streams [Read et al. 2012].
— DELICIOUS [Tsoumakas et al. 2008] contains textual data from the Delicious[11] website along with their tags (e.g. *academia*, *airline*, *algorithm*, etc.). In [Read 2010] it is highlighted that this dataset is a modified tagging problem where the label space was not predefined prior to labelling and the size of the label space, 983 labels, is greater than the size of the input space, 16105 instances. Attributes are binary and represent the presence of a word in the document. It has been used in works focused in its great number of labels [Zhang et al. 2010].
— EUR-LEX [Loza and Fürnkranz 2008]. The EUR-Lex text collection contains 19348 documents on European Union law (e.g. treaties, legislation, legislative proposals) which are indexed according to several orthogonal categorization schemes to allow for multiple search facilities. The most important categorization is provided by the EUROVOC descriptors, which form a topic hierarchy with almost 4000 categories regarding different aspects of European law. 'Attributes are numeric and represent the TF-IDF measure for each term in the document. It has been used in the domain of document classification [Rubin et al. 2012] and due to its dimensionality, to test MLL algorithms designed for large datasets [Loza and Fürnkranz 2008].
— TMC2007 [Srivastava and Zane-Ulman 2005]. The SIAM Text Mining Competition (TMC) 2007 dataset is a subset of the Aviation Safety Reporting System (ASRS)

[8]http://www-nlpir.nist.gov/projects/trecvid/
[9]http://www.bibsonomy.org/
[10]http://www.imdb.com/
[11]https://delicious.com/

dataset. This benchmark contains 28596 aviation safety free text form reports that the flight crews submit after completion of each flight regarding events that took place during a flight. The goal is to label the documents with respect to what types of problems they describe. The dataset has 49060 binary attributes corresponding to the presence of terms in the collection. The safety reports are provided with 22 labels, each of them representing a problem type that appears during flights. In [Tsoumakas et al. 2010b] a χ^2 feature ranking method was used separately for each label and the top 500 features based on their maximum rank over all labels were selected. Text representation follows the boolean bag-of-words model.

— BOOKMARKS [Katakis et al. 2008] is based on the data of the ECML/PKDD 2008 discovery challenge and contains metadata for bookmark entries from the Bibsonomy system such as the URL of the web page, an URL hash, a description of the web page, etc. It has 208 labels and 2150 binary attributes that represent terms in the document. It was used in a extensive experimental comparison of MLL methods [Madjarov et al. 2012] and some of them were not able to finish with this dataset.

A.1.3. ML dataset formats. Many of the MLL benchmark datasets are available in Mulan or Meka frameworks, both based in the *arff* format [Hall et al. 2009]. Mulan [Tsoumakas et al. 2011] datasets use two files. The first one is an *arff* file where labels should be nominal attributes with 0 or 1 values. The second one is an XML file where labels are defined and allows representing hierarchical relationships among them. An example of a Mulan dataset with 3 features and 4 labels is presented in table XIII. On the other hand, Meka [Read 2012] datasets are also in *arff* and use one attribute for each target or label. The dataset options (like the -C option for the number of labels, q) can be included either in the *@relation* tag of the *arff* file or in the command line. Meka allows also the train/test split percentage to be stored in the *@relation* name where a colon (:) is used to separate the dataset name and the option. An example of a Meka dataset with 3 features and 4 labels is presented in table XIII.

A.2. Software

This section will be focused in software that implements MLL baseline methods. Table XIV summarizes the features of the main APIs and software packages. Firstly it can be cited Mulan [Tsoumakas et al. 2011]. It is an open-source Java API for multi-label classification that is built on the top of Weka [Hall et al. 2009] and implements many transformation methods like BR, LP, copy methods, EPS or CLR and multi-label algorithms like ML-kNN, RAkEL, HOMER or BP-MLL. It also provides support for the evaluation of the models, a number of multi-label datasets and offers methods for basic feature selection. Another framework is Meka [Read 2012], a multi-label extension for the Weka framework that also provides an open-source Java implementation of the PS and CC methods. Written in Java, it is also compatible with Mulan and provides support for development, running and evaluation of multi-label and multi-target (multi-class outputs instead binary outputs) classifiers.

On the LAMDA research group's website (see table XI) several Matlab packages are provided with the implementation of the BP-MLL, InsDif or ML-kNN among others. From the Min-Ling Zhang's website (see table XI)the code of several multi-label models, such as ML-kNN, Rank-SVM, BP-MLL or ML-RBF, can be also downloaded. LIBSVM [Chang and Lin 2011] is a library for support vector machines that allows the handling of multi-label data. Specifically, the LP and BR transformation methods are implemented.

Some data mining software suites include any multi-label capabilities. Thus, ORANGE [Laboratory 2013] includes a multi-target add-on with methods such as BR, CC or ECC and SCIKIT-LEARN [Pedregosa et al. 2011] provides the RPC, BR and ECOC ap-

Table XII. ML datasets. In the feat. column *n*, *b*, *nb*, and *c* mean numeric, binary, numeric with binary values, and categorical attributes respectively

DATASET	DOMAIN	INST.	FEAT.	q	CARD.	DENS.	DIST.	DOWNLOAD
Flags	images(toy)	194	9c+10n	7	3.392	0.485	54	[Tsoumakas et al. 2011] [Newman et al. 1998]
Emotions	music	593	72n	6	1.869	0.311	27	[Tsoumakas et al. 2011]
Birds	audio	645	2c+258n	19	1.059	0.053	133	[Tsoumakas et al. 2011]
Yeast	biology	2417	103n	14	4.237	0.303	198	[Tsoumakas et al. 2011]
Scene	images	2407	294n	6	1.074	0.179	15	[Tsoumakas et al. 2011]
Plant	biology	97	440n	12	1.078	0.089	32	[Xu 2013b]
CAL500	music	502	68n	174	26.044	0.150	502	[Tsoumakas et al. 2011]
Human	biology	3106	440n	14	1.185	0.084	85	[Xu 2013b]
Genbase	biology	662	1186b	27	1.252	0.046	32	[Tsoumakas et al. 2011]
Medical	text	978	1449b	45	1.245	0.028	94	[Tsoumakas et al. 2011]
Slashdot	text	3782	1079nb	22	1.18	0.053	156	[Read 2012]
Enron	text	1702	1001b	53	3.378	0.064	753	[Tsoumakas et al. 2011] [Read 2012]
LangLog	text	1460	1004nb	75	1.180	0.015	304	[Read 2012]
Tmc2007-500	text	28596	500b	22	2.219	0.100	1172	[Tsoumakas et al. 2011]
20ng	text	19299	1006nb	20	1.028	0.051	55	[Read 2012]
Mediamill	video	43907	120n	101	4.376	0.043	6555	[Tsoumakas et al. 2011]
Corel5k	images	5000	499b	374	3.522	0.009	3175	[Tsoumakas et al. 2011]
Corel16k(10samples)	images	13811	500b	161	2.867	0.018	4937	[Tsoumakas et al. 2011]
Bibtex	text	7395	1836b	159	2.402	0.015	2856	[Tsoumakas et al. 2011]
Yahoo(Health)	text(web)	9205	30605nb	32	1.644	0.051	335	[Ueda and Saito 2002b]
Yahoo(Arts)	text(web)	7484	23146nb	26	1.653	0.063	599	[Ueda and Saito 2002b]
Yahoo(Business)	text(web)	11214	21924nb	30	1.598	0.053	233	[Ueda and Saito 2002b]
Yahoo(Reference)	text(web)	8027	39679nb	33	1.174	0.035	275	[Ueda and Saito 2002b]
Yahoo(Science)	text(web)	6428	37187nb	40	1.449	0.036	457	[Ueda and Saito 2002b]
IMDB	text	120919	1001nb	28	2.00	0.071	4503	[Read 2012]
Yahoo(Education)	text(web)	12030	27534nb	33	1.463	0.044	511	[Ueda and Saito 2002b]
Yahoo(Entertainment)	text(web)	12730	32001nb	21	1.413	0.067	337	[Ueda and Saito 2002b]
Yahoo(Recreation)	text(web)	12828	30324nb	22	1.428	0.064	530	[Ueda and Saito 2002b]
Yahoo(Computers)	text(web)	12444	34096nb	33	1.507	0.045	428	[Ueda and Saito 2002b]
Delicious	text(web)	16105	500b	983	19.02	0.019	15806	[Tsoumakas et al. 2011]
Yahoo(Social)	text(web)	12111	52350nb	39	1.279	0.032	361	[Ueda and Saito 2002b]
Yahoo(Society)	text(web)	14512	31802nb	27	1.670	0.061	1054	[Ueda and Saito 2002b]
EUR-Lex(subjectmatters)	text	19348	5000n	201	2.213	0.011	2504	[Tsoumakas et al. 2011]
Rcv1v2(subset4)	text	6000	47229n	101	2.484	0.025	816	[Tsoumakas et al. 2011]
Rcv1v2(subset5)	text	6000	47235n	101	2.642	0.026	946	[Tsoumakas et al. 2011]
Rcv1v2(subset1)	text	6000	47236n	101	2.880	0.029	1028	[Tsoumakas et al. 2011]
Rcv1v2(subset2)	text	6000	47236n	101	2.634	0.026	954	[Tsoumakas et al. 2011]
Rcv1v2(subset3)	text	6000	47236n	101	2.614	0.026	939	[Tsoumakas et al. 2011]
Tmc2007	text	28596	49060b	22	2.158	0.098	1341	[Tsoumakas et al. 2011]
Bookmarks	text	87856	2150b	208	2.028	0.01	18716	[Tsoumakas et al. 2011]
EUR-Lex(directorycodes)	text	19348	5000n	412	1.292	0.003	1615	[Tsoumakas et al. 2011] [Loza and Fürnkranz 2013]
EUR-Lex(eurovocdescript.)	text	19348	5000n	3993	5.31	0.001	16467	[Tsoumakas et al. 2011][Loza and Fürnkranz 2013]

proaches. All the material (including the repository of manipulated datasets and software for dataset characteristics extraction) used by Chekina et al. [2011] for dataset characteristics extraction is also available[12]. HMC software and datasets developed by the Declarative Languages and Artificial Intelligence group of the Katholieke Universiteit Leuven and the Department of Knowledge Technologies of the Jožef Stefan Institute are available[13].

[12]http://www.ise.bgu.ac.il/faculty/liorr/lena/meta.html

[13]http://dtai.cs.kuleuven.be/clus/ and http://kt.ijs.si/DragiKocev/PhD/resources/doku.php?id=hmc_classification

Table XIII. Examples of a multi-label dataset with 3 features and 4 labels in Mulan and Meka formats

(a) Mulan format

ARFF FILE

```
@relation MultiLabelExample
@attribute feature1 numeric
@attribute feature2 numeric
@attribute feature3 numeric
@attribute label1 0, 1
@attribute label2 0, 1
@attribute label3 0, 1
@attribute label4 0, 1
@data
4.1,2.9,3.7,0,0,1,1
```

XML FILE

```
<labels xmlns="http://mulan.sourceforge.net/labels">
<label name="label1"></label>
<label name="label2"></label>
<label name="label3"></label>
<label name="label4"></label>
</labels>
```

(b) Meka format

ARFF FILE

```
@relation 'Example_Dataset: -C 4 -split-percentage 50'

@attribute label1 0,1
@attribute label2 0,1
@attribute label3 0,1
@attribute label4 0,1
@attribute feature1 numeric
@attribute feature2 numeric
@attribute feature3 numeric
@data
0,0,1,1,4.1,2.9,3.7
```

Table XIV. Main features of software packages for MLL

	MULAN	MEKA	LAMBDA GR.	ZHANG'S SITE
RSL	Copy, ignore, select	RT		
BR-BASED	BR	BR, BRq		
LC	LP, PS, EPS	LP(LC), PS, EPS		
PW	CLR			
LABEL DEPENDENCES	CC, ECC	CC, ECC, PCC, MCC(Montecarlo)	ML-LOC	Bayesian Networks
LAZY	BRkNN, IBLR, MLkNN		MLkNN	MLkNN
ANN	BPMLL, MMP		BPMLL	BPMLL, ML-RBF
SVMS				RankSVM
META-LEARNERS	RAkEL, HOMER, LPBR(Subset Learner), AdaBoost.MH, 2BR	BaggingML, 2BR(MBR), EnsembleML, RandomSubspaceML		
OTHER	Hierarchical HMC	Majority labelset, Conditional Dependence Network (CDN), unsupervised (EM)	InsDiff, MIML, WELL, Multimodal MIML	Naive BayesML, LIFT, MIML
UTILS	Dimensionality Reduction, ConverterLibSVM, ConverterCLUS, Iterative Stratification, Statistics of data, metrics, thresholding	Wrapper Mulan	Dimensionality reduction	
MULTI-TARGET		CC, CR, Nearest Set Replacement (NSR), EnsembleMT, BaggingMT		
GUI	No	Yes	No	No
LANGUAJE	Java	Java	Matlab	Matlab
LICENSE	GNU GPL	GNU GPL	Free for academic purpose	Free for academic purpose

REFERENCES

2009. 1st International Workshop on Learning from Multi-Label Data (MLD'09). http://lpis.csd.auth.gr/workshops/mld09/mld09.pdf. (September 2009).

2010. 2nd International Workshop on Learning from Multi-Label Data (MLD'10). http://cse.seu.edu.cn/conf/MLD10/files/MLD'10.pdf. (June 2010).

2012. The First International Workshop on Learning with Weak Supervision (LAWS'12). http://cse.seu.edu.cn/conf/LAWS12/files/LAWS'12.pdf. (November 2012).

2012. Machine Learning. Special Issue on Learning from Multi-Label Data. (2012).

2013. Extreme Classification: Multi-Class & Multi-Label Learning with Millions of Categories. http://nips.cc/Conferences/2013/Program/event.php?ID=3707. (2013).

2013. Pattern classification of dermoscopy images: A perceptually uniform model. *Pattern Recognition* 46, 1 (2013), 86 – 97.

Fabio Aiolli. 2004. *Large Margin Multiclass Learning: Models and Algorithms*. Ph.D. Dissertation. Università degli Studi di Pisa.

J.L. Ávila. 2013. *Modelos de aprendizaje basados en programación genética para clasificación multietiqueta*. Ph.D. Dissertation. University of Córdoba.

J. Ávila, E. Gibaja, and S. Ventura. 2010. Evolving Multi-label Classification Rules with Gene Expression Programming: A Preliminary Study. In *Hybrid Artificial Intelligence Systems*. Lecture Notes in Computer Science, Vol. 6077. Springer Berlin / Heidelberg, Chapter 2, 9–16.

J. L. Ávila, E. L. Gibaja, A. Zafra, and S. Ventura. 2011. A Gene Expression Programming Algorithm for Multi-Label Classification. *Journal of Multiple-Valued Logic and Soft Computing* 17, 2-3 (2011), 183–206.

Kobus Barnard, Pinar Duygulu, Nando de Freitas, David Forsyth, David Blei, and Michael I. Jordan. 2003. Matching Words and Pictures. *Journal of Machine Learning Research* 3 (2003), 1107–1135.

Zafer Barutcuoglu, Robert E. Schapire, and Olga G. Troyanskaya. 2006. Hierarchical multi-label prediction of gene function. *Bioinformatics (Oxford, England)* 22, 7 (April 2006), 830–836.

Plaban K. Bhowmick, Anupam Basu, Pabitra Mitra, and Abhisek Prasad. 2010. Sentence Level News Emotion Analysis in Fuzzy Multi-label Classification Framework. *Research in Computer Science, special issue:Natural Language Processing and its Applications* 46 (2010), 143–154.

Hendrik Blockeel, Luc De Raedt, and Jan Ramon. 1998. 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, 55–63.

H. Blockeel, L. Schietgat, J. Struyf, S. Džeroski, and A. Clare. 2006. Decision trees for hierarchical multilabel classification: A case study in functional genomics. *Lecture Notes in Computer Science* 4213 (2006), 18–29.

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

Leo Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees*. Wadsworth.

Forrest Briggs, Raviv Raich, Konstantinos Eftaxias, Zhong Lei, , and Yonghong Huang. 2013. The ninth annual MLSP competition: Overview. In *Proceedings of the 2013 IEEE International Workshop on Machine Learning for Signal Processing*.

Klaus Brinker. 2006. On Active Learning in Multi-label Classification. In *From Data and Information Analysis to Knowledge Engineering*, Myra Spiliopoulou, Rudolf Kruse, Christian Borgelt, Andreas Nürnberger, and Wolfgang Gaul (Eds.). Springer Berlin Heidelberg, 206–213.

Klaus Brinker, Johannes Fürnkranz, and Eyke Hüllermeier. 2006. A Unified Model for Multilabel Classification and Ranking. In *17th European Conference on Artificial Intelligence*. IOS Press, Amsterdam, The Netherlands, 489–493.

Lijuan Cai. 2008. *Multilabel Classification over Category Taxonomies*. Ph.D. Dissertation. Brown University.

Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. 2006. Hierarchical classification: combining Bayes with SVM. In *Proceedings of the Twenty-Third International Conference on Machine Learning (ICML 2006)*. 177–184.

Allen Chan and Alex A. Freitas. 2006. A new ant colony algorithm for multi-label classification with applications in bioinfomatics. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM Press, New York, NY, USA, 27–34.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27. Issue 3. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

L. Chekina, L. Rokach, and B. Shapira. 2011. Meta-learning for Selecting a Multi-label Classification Algorithm. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*. 220 –227.

Weizhu Chen, Jun Yan, Benyu Zhang, Zheng Chen, and Qiang Yang. 2007. Document Transformation for Multi-label Feature Selection in Text Categorization, In IEEE International Conference on Data Mining. *Data Mining, IEEE International Conference on* (2007), 451–456.

Weiwei Cheng and Eyke Hüllermeier. 2009. Combining instance-based learning and logistic regression for multilabel classification. *Mach. Learn.* 76(2-3) (Sept. 2009), 211–225.

Everton Alvares Cherman, Jean Metz, and Maria Carolina Monard. 2012. Incorporating label dependency into the binary relevance framework for multi-label classification. *Expert Syst. Appl.* 39, 2 (2012), 1647–1655.

Kuo-Chen Chou, Zhi-Cheng Wu, and Xuan Xiao. 2011. iLoc-Euk: A Multi-Label Classifier for Predicting the Subcellular Localization of Singleplex and Multiplex Eukaryotic Proteins. *PLoS ONE* 6, 3 (03 2011), e18258.

Patrick Marques Ciarelli, Elias Oliveira, Claudine Badue, and Alberto Ferreira De Souza. 2009. Multi-Label Text Categorization Using a Probabilistic Neural Network. *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)* 1 (2009), 133–144.

Amanda Clare. 2003. *Machine learning and data mining for yeast functional genomics*. Ph.D. Dissertation. University of Wales, Aberystwyth.

Amanda Clare and Ross D. King. 2001. Knowledge Discovery in Multi-label Phenotype Data. *Lecture Notes in Computer Science* 2168 (2001), 42 – 53.

He Cong and Loh H. Tong. 2008. Grouping of TRIZ Inventive Principles to facilitate automatic patent classification. *Expert Systems with Applications* 34, 1 (Jan. 2008), 788–795.

Koby Crammer and Yoram Singer. 2003. A family of additive online algorithms for category ranking. *J. Mach. Learn. Res.* 3 (March 2003), 1025–1058.

André de Carvalho and Alex Freitas. 2009. A Tutorial on Multi-label Classification Techniques. In *Foundations of Computational Intelligence Volume 5*. Studies in Computational Intelligence, Vol. 205. Springer Berlin / Heidelberg, Chapter 8, 177–195.

Francesco De Comité, Rémi Gilleron, and Marc Tommasi. 2003. 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, 35–49.

Krzysztof Dembczyński, Weiwei Cheng, and Eyke Hüllermeier. 2010. Bayes Optimal Multilabel Classification via Probabilistic Classifier Chains. In *ICML*. 279–286.

Krzysztof Dembczyński, Willem Waegeman, Weiwei Cheng, and Eyke Hüllermeier. 2012. On label dependence and loss minimization in multi-label classification. *Machine Learning* 88 (2012), 5–45. Issue 1.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistics Society -B* 39(1) (1977), 1–38.

Janez Demšar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research* 7 (2006), 1–30.

Anastasios Dimou, Grigorios Tsoumakas, Vasileios Mezaris, Ioannis Kompatsiaris, and Ioannis Vlahavas. 2009. An Empirical Study of Multi-label Learning Methods for Video Annotation. *Content-Based Multimedia Indexing, International Workshop on* 0 (2009), 19–24.

Sotiris Diplaris, Grigorios Tsoumakas, Pericles Mitkas, and Ioannis Vlahavas. 2005. Protein Classification with Multiple Algorithms, In Proceedings of the 10th Panhellenic Conference on Informatics (PCI 2005). *Advances in Informatics* (November 2005), 448–456.

R. Duwairi and A. Kassawneh. 2008. A framework for predicting proteins 3D structures. In *IEEE/ACS International Conference on Computer Systems and Application*. Washington, DC, USA, 37 –44.

Pinar Duygulu, Kobus Barnard, Nando de Freitas, and David Forsyth. 2002. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Seventh European Conference on Computer Vision*. IV:97–112, 2002.

Andre Elisseeff and Jason Weston. 2001. Kernel methods for Multi-labelled classification and Categorical regression problems. In *In Advances in Neural Information Processing Systems 14*. MIT Press, 681–687.

Andrea Esuli and Fabrizio Sebastiani. 2009. Active Learning Strategies for Multi-Label Text Classification. In *Advances in Information Retrieval*, Mohand Boughanem, Catherine Berrut, Josiane Mothe, and Chantal Soule-Dupuy (Eds.). Lecture Notes in Computer Science, Vol. 5478. Springer Berlin / Heidelberg, Berlin, Heidelberg, Chapter 12, 102–113.

Theodoros Evgeniou and Massimiliano Pontil. 2004. Regularized Multi–task Learning. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04)*. ACM, 109–117.

Zheng Fang and Zhongfei (Mark) Zhang. 2012. Simultaneously Combining Multi-view Multi-label Learning with Maximum Margin Classification.. In *ICDM*. IEEE Computer Society.

Yoav Freund and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* 55 (August 1997), 119–139. Issue 1.

Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza mencía, and Klaus Brinker. 2008. Multilabel classification via calibrated label ranking. *Machine Learning* 73(2) (2008), 133 – 153.

Nadia Ghamrawi and Andrew McCallum. 2005. Collective multi-label classification. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 195–200.

Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative Methods for Multi-Labeled Classification. In *In Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 22–30.

Eduardo Corrêa Gonçalves, Alexandre Plastino, and Alex Alves Freitas. 2013. A Genetic Algorithm for Optimizing the Label Ordering in Multi-label Classifier Chains. In *IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI)*. 469–476.

Teresa Gonçalves and Paulo Quaresma. 2004. Using IR Techniques to Improve Automated Text Classification, In Proceedings of the 9th International Conference on Applications of Natural Language to Information Systems. *Natural Language Processing and Information Systems* (June 2004), 374–379.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations* 11(1) (2009).

Jianjun He, Hong Gu, and Zhelong Wang. 2012. Multi-instance multi-label learning based on Gaussian process with application to visual mobile robot navigation. *Information Sciences* 190, 0 (2012), 162 – 177.

S. Hettich and S. D Bay. 1999. The UCI KDD Archive. http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html. (1999).

Daniel Hsu, Sham Kakade, John Langford, and Tong Zhang. 2009. Multi-Label Prediction via Compressed Sensing. In *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta (Eds.). 772–780.

Xiaolei Huang, Wei Wang, Zhiyun Xue, Sameer Antani, L. Rodney Long, and Jose Jeronimo. 2008. Tissue classification using cluster features for lesion detection in digital cervigrams. In *Proc. SPIE Medical Imaging*. 2008–6914.

Eyke Hüllermeier, Johannes Fürnkranz, Weiwei Cheng, and Klaus Brinker. 2008. Label Ranking by Learning Pairwise Preferences. *Artificial Intelligence* 172 (2008), 1897–1916.

Marios Ioannou, George Sakkas, Grigorios Tsoumakas, and Ioannis P. Vlahavas. 2010. Obtaining Bipartitions from Score Vectors for Multi-Label Classification. In *ICTAI (1)*. 409–416.

Shuiwang Ji, Lei Tang, Shipeng Yu, and Jieping Ye. 2010. A shared-subspace learning framework for multi-label classification. *ACM Trans. Knowl. Discov. Data* 4, 2 (May 2010), 1–29.

Aiwen Jiang, Chunheng Wang, and Yuanping Zhu. 2008. Calibrated Rank-SVM for multi-label image categorization, In Neural Networks, 2008. IJCNN 2008. IEEE International Joint Conference on. *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on* (June 2008), 1450–1455.

Ioannis Katakis, Grigorios Tsoumakas, and Ioannis Vlahavas. 2008. Multilabel Text Classification for Automated Tag Suggestion. In *Proceedings of the ECML/PKDD 2008 Discovery Challenge*.

Kentaro Kawai and Yoshimasa Takahashi. 2009. Identification of the Dual Action Antihypertensive Drugs Using TFS-Based Support Vector Machines. *Chem-Bio Informatics Journal* 4 (2009), 44–51.

KDIS. 2021. Multilabel Classification library. http://www.citeulike.org/group/4310. (2021).

Dragi Kocev. 2012. *Ensembles for predicting structured outputs*. Ph.D. Dissertation. Jŏzef Stefan International Postgraduate School.

Dragi Kocev, Celine Vens, Jan Struyf, and Sašo Džeroski. 2007. Ensembles of Multi-Objective Decision Trees. In *Proceedings of the 18th European conference on Machine Learning (ECML '07)*. Springer-Verlag, Berlin, Heidelberg, 624–631.

Artus Krohn-Grimberghe, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2012. Multi-relational matrix factorization using bayesian personalized ranking for social network data. In *Proceedings of the fifth ACM international conference on Web search and data mining (WSDM '12)*. ACM, New York, NY, USA, 173–182.

Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. 2009. Attribute and Simile Classifiers for Face Verification. In *In IEEE International Conference on Computer Vision (ICCV)*.

FRI UL Bioinformatics Laboratory. 2013. Orange Multitarget add-on for Orange data mining software package. http://pypi.python.org/pypi/Orange-Multitarget. (2013).

J. Lafferty, A. McCallum, and F.C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labelling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*. 282289.

K. Lang. 2008. The 20 newsgroup dataset. http://people.csail.mit.edu/jrennie/20Newsgroups/. (2008).

Pedro Larrañaga. 2010. Multi-label classification. http://www.dynamopro.org/IMG/pdf/tamida2010-larranaga.pdf. (2010).

Boris Lauser and Andreas Hotho. 2003. Automatic Multi-label Subject Indexing in a Multilingual Environment. In *ECDL*. 140–151.

David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2005. RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research* 5 (2005), 361–397.

Vivian F. López, Fernando de la Prieta, Mitsunori Ogihara, and Ding Ding Wong. 2012. A model for multi-label classification and ranking of learning objects. *Expert Systems with Applications* 39, 10 (2012), 8878 – 8884.

Eneldo Loza and Johannes Fürnkranz. 2007. An Evaluation of Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In *Proceedings of the LWA 2007: Lernen - Wissen - Adaption*, Alexander Hinneburg (Ed.). Halle, Germany, 126–132.

Eneldo Loza and Johannes Fürnkranz. 2008. Efficient Pairwise Multilabel Classification for Large-Scale Problems in the Legal Domain. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD-2008)*. Springer-Verlag, 50–65.

Eneldo Loza and Johannes Fürnkranz. 2010. Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In *Semantic Processing of Legal Texts*, Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia (Eds.). Lecture Notes in Computer Science, Vol. 6036. Springer Berlin / Heidelberg, Berlin, Heidelberg, Chapter 11, 192–215.

Eneldo Loza and Johannes Fürnkranz. 2013. The EUR-Lex Dataset. http://www.ke.tu-darmstadt.de/resources/eurlex. (2013).

Eneldo Loza, Sang-Hyeun Park, and Johannes Fürnkranz. 2009. Efficient voting prediction for pairwise multilabel classification. In *Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN)*.

Aiyesha Ma, Ishwar Sethi, and Nilesh Patel. 2009. Multimedia Content Tagging Using Multilabel Decision Tree. In *11th IEEE International Symposium on Multimedia, 2009. ISM '09*. 606–611.

Gjorgji Madjarov, Dejan Gjorgjevikj, and Sašo Džeroski. 2011. Dual layer voting method for efficient multi-label classification. In *Proceedings of the 5th Iberian conference on Pattern recognition and image analysis (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics))*, Vol. 6669 LNCS. 232–239.

Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj, and Sašo Džeroski. 2012. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition* 45, 9 (2012), 3084 – 3104.

M. A. Mammadov, A. M. Rubinov, and J. Yearwood. 2007. The study of drug-reaction relationships using global optimization techniques. *Optimization Methods Software* 22 (Feb. 2007), 99–126.

Andrew Kachites McCallum. 1999. Multi-label text classification with a mixture model trained by EM. In *AAAI 99 Workshop on Text Learning*.

MLKD. 2012. MLKD - with tag multilabel library. http://www.citeulike.org/group/7105/tag/multilabel. (2012).

Arturo Montejo-Ráez and Luis Ureña López. 2006. Selection Strategies for Multi-label Text Categorization. In *Advances in Natural Language Processing*, Tapio Salakoski, Filip Ginter, Sampo Pyysalo, and Tapio Pahikkala (Eds.). Lecture Notes in Computer Science, Vol. 4139. Springer Berlin / Heidelberg, Berlin, Heidelberg, Chapter 58, 585–592.

Pio Nardiello, Fabrizio Sebastiani, and Alessandro Sperduti. 2003. Discretizing Continuous Attributes in AdaBoost for Text Categorization. In *Advances in Information Retrieval*, Fabrizio Sebastiani (Ed.). Lecture Notes in Computer Science, Vol. 2633. Springer Berlin / Heidelberg, Chapter 23, 78.

Gulisong Nasierding and Abbas Z. Kouzani. 2010. Empirical Study of Multi-label Classification Methods for Image Annotation and Retrieval. 617–622.

D. Newman, S. Hettich, C. Blake, and C. Merz. 1998. UCI Repository of Machine Learning Databases. http://www.ics.uci.edu/m̃learn/MLRepository.html.. (1998).

Cao D. Nguyen, Tran A. Dung, and Tru H. Cao. 2005. Text Classification for DAG-Structured Categories. In *Advances in Knowledge Discovery and Data Mining*, Tu B. Ho, David Cheung, and Huan Liu (Eds.). Lecture Notes in Computer Science, Vol. 3518. Springer Berlin / Heidelberg, Berlin, Heidelberg, Chapter 36, 1–18.

N. Oza, J. P. Castle, and J. Stutz. 2009. Classification of Aeronautics System Health and Safety Documents. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 39, 6 (Nov. 2009), 670–680.

F. Pachet and P. Roy. 2009. Improving Multilabel Analysis of Music Titles: A Large-Scale Validation of the Correction Approach. *IEEE Transactions on Audio, Speech, and Language Processing* 17, 2 (Feb. 2009), 335–343.

Sang-Hyeun Park and Johannes Fürnkranz. 2008. *Multi-Label Classification with Label Constraints*. Technical Report. Knowledge Engineering Group, TU Darmstadt.

R.S. Parpinelli, H.S. Lopes, and A.A Freitas. 2002. Data Mining with an Ant Colony Optimization Algorithm. *EEE Trans. On Evolutionary Computation* 6(4) (2002), 321–332.

Fabian Pedregosa, Gal Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. https://pypi.python.org/pypi/Orange-Multitarget, *Journal of Machine Learning Research* 12 (2011), 2825–2830.

J.P. Pestian, C. Brew, P.M. Matykiewicz, D.J. Hovermale, N. Johnson, K.B. Cohen, and W. Duch. 2007. A shared task involving multi-label classification of clinical free text. In *Proceedings of ACL BioNLP*. Prague.

S. Peters, L. Denoyer, and P. Gallinari. 2010. Iterative Annotation of Multi-relational Social Networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*. 96 –103.

Guo J. Qi, Xian S. Hua, Yong Rui, Jinhui Tang, Tao Mei, and Hong J. Zhang. 2007. Correlative multi-label video annotation. In *Proceedings of the 15th international conference on Multimedia (MULTIMEDIA '07)*. ACM, New York, NY, USA, 17–26.

Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, and Hong-Jiang Zhang. 2009. Two-Dimensional Multi-label Active Learning with an Efficient Online Adaptation Model for Image Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 10 (Oct. 2009), 1880–1897.

J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

R. Rak, L. Kurgan, and M. Reformat. 2008. A tree-projection-based algorithm for multi-label recurrent-item associative-classification rule generation. *Data & Knowledge Engineering* 64, 1 (Jan. 2008), 171–197.

Jesse Read. 2008. A Pruned Problem Transformation Method for Multi-label Classification. In *Proceedings of the NZ Computer Science Research Student Conference*.

Jesse Read. 2010. *Scalable Multi-label Classification*. Ph.D. Dissertation. University of Waikato.

Jesse Read. 2011. Advances in Multi-label Classification. http://users.ics.aalto.fi/jesse/talks/Charla-Malaga.pdf. (2011).

Jesse Read. 2012. MEKA: A Multi-label Extension to WEKA. http://meka.sourceforge.net/. (2012).

Jesse Read, Albert Bifet, Geoffrey Holmes, and Bernhard Pfahringer. 2010. *Efficient multi-label classification for evolving data streams*. Technical Report. University of Waikato, Department of Computer Science.

Jesse Read, Albert Bifet, Geoff Holmes, and Bernhard Pfahringer. 2012. Scalable and efficient multi-label classification for evolving data streams. *Machine Learning* 88 (2012), 243–272. Issue 1.

Jesse Read, Bernhard Pfahringer, and Geoff Holmes. 2008. Multi-label Classification Using Ensembles of Pruned Sets, In ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining. *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on* 0 (2008), 995–1000.

J. Read, B. Pfahringer, G. Holmes, and E. Frank. 2011. Classifier chains for multi-label classification. *Machine Learning* 85, 3 (2011), 1–27.

Timothy Rubin, America Chambers, Padhraic Smyth, and Mark Steyvers. 2012. Statistical topic models for multi-label document classification. *Machine Learning* 88 (2012), 157–208. Issue 1.

Robert E. Schapire and Yoram Singer. 1999. Improved Boosting Algorithms Using Confidence-rated Predictions. *Machine Learning* 37(3) (1999), 297 – 336.

Robert E. Schapire and Yoram Singer. 2000. BoosTexter: A Boosting-based System for Text Categorization. *Machine Learning* 39, 2/3 (2000), 135–168.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.* 34 (March 2002), 1–47. Issue 1.

Fabrizio Sebastiani, Alessandro Sperduti, and Nicola Valdambrini. 2000. An improved boosting algorithm and its application to text categorization. In *Proceedings of the ninth international conference on Information and knowledge management (CIKM '00)*. ACM, New York, NY, USA, 78–85.

Konstantinos Sechidis, Grigorios Tsoumakas, and Ioannis P. Vlahavas. 2011. On the Stratification of Multi-label Data. In *Proceedings of the Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD (part III) (Lecture Notes in Computer Science)*, Vol. 6913. Springer, Athens, Greece, 145–158.

Huan Shao, GuoZheng Li, GuoPing Liu, and YiQin Wang. 2010. Symptom selection for multi-label data of inquiry diagnosis in traditional Chinese medicine. *Science China Information Sciences* 1 (2010), 1–13.

Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. 2009. TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context. *Int. J. Comput. Vision* 81 (Jan. 2009), 2–23.

Andrew Skabar, Dennis Wollersheim, and Tim Whitfort. 2006. Multi-label classification of gene function using MLPs. In *Proceedings of the International Joint Conference on Neural Networks*. 2234–2240.

C.G.M. Snoek, M. Worring, J.C. van Gemert, J.-M. Geusebroek, and A.W.M. Smeulders. 2006. The Challenge Problem for Automated Detection of 101 Semantic Concepts in Multimedia. In *Proceedings of ACM Multimedia*. Santa Barbara, USA, 421–430.

Tal Sobol-Shikler and Peter Robinson. 2010. Classification of Complex Information: Inference of Co-Occurring Affective States from Their Expressions in Speech. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 7 (July 2010), 1284–1297.

Yang Song, Lu Zhang, and C. Lee Giles. 2011. Automatic tag recommendation algorithms for social recommender systems. *ACM Trans. Web* 5, 1 (Feb. 2011), 4:1–4:31.

Mohammad S Sorower. 2010. *A Literature Survey on Algorithms for Multi-label Learning*. Technical Report. Oregon State University. http://people.oregonstate.edu/~sorowerm/pdf/Qual-Multilabel-Shahed-CompleteVersion.pdf

Stephan Spat, Bruno Cadonna, Ivo Rakovac, Christian Gütl, Hubert Leitner, Günther Stark, and Peter Beck. 2008. Enhanced Information Retrieval from Narrative German-language Clinical Text Documents using Automated Document Classification. In *eHealth Beyond the Horizon - Get IT There, Proceedings of MIE2008, The XXIst International Congress of the European Federation for Medical Informatics*. Göteborg, Sweden, 473–478.

E. Spyromitros, G. Tsoumakas, and Ioannis Vlahavas. 2008. An Empirical Study of Lazy Multilabel Classification Algorithms, In SETN '08: Proceedings of the 5th Hellenic conference on Artificial Intelligence. *Artificial Intelligence: Theories, Models and Applications* (2008), 401–406.

A. Srivastava and B. Zane-Ulman. 2005. Discovering recurring anomalies in text reports regarding complex space systems. In *Proceedings of the 2005 IEEE Aerospace Conference*.

Liang Sun, Shuiwang Ji, and Jieping Ye. 2013. *Multi-Label Dimensionality Reduction*. Chapman & Hall/CRC Machine Learning & Pattern Recognition.

Muhammad A. Tahir, Josef Kittler, Fei Yan, and Krystian Mikolajczyk. 2009. Kernel Discriminant Analysis Using Triangular Kernel for Semantic Scene Classification. *Content-Based Multimedia Indexing, International Workshop on* 0 (2009), 1–6.

Farbound Tai and Hsuan-Tien Lin. 2010. Multi-Label Classification with Principal Label Space Transformation. In *2nd International Workshop on Learning from Multi-Label Data (MLD'10)*. 45–52.

Lei Tang and Huan Liu. 2009. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '09)*. ACM, New York, NY, USA, 817–826.

Lei Tang, Suju Rajan, and Vijay K. Narayanan. 2009. Large scale multi-label classification via metalabeler. In *18th international conference on World wide web*. New York, NY, USA, 211–220.

Lena Tenenboim-Chekina, Lior Rokach, and Bracha Shapira. 2010. Identification of Label Dependencies for Multi-Label Classification. In *2nd International Workshop on Learning from Multi-Label Data (MLD'10)*. 53–60.

F. A. Thabtah, P. Cowling, Y. Peng, R. Rastogi, K. Morik, M. Bramer, and X. Wu. 2004. MMAC: A new multi-class, multi-label associative classification approach, In 4th IEEE International Conference on Data Mining, ICDM 2004. *Proceedings - Fourth IEEE International Conference on Data Mining, ICDM 2004* (2004), 217–224.

F. A. Thabtah and P. I. Cowling. 2007. A greedy classification algorithm based on association rule. *Appl. Soft Comput.* 7, 3 (June 2007), 1102–1111.

K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. 2008. Multi-label Classification of Music into Emotions. In *Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008), Philadelphia, PA, USA, 2008*.

G. Tsoumakas, A. Dimou, E. Spyromitros, V. Mezaris, I. Kompatsiaris, and I. Vlahavas. 2009. Correlation-Based Pruning of Stacked Binary Relevance Models for Multi- Label Learning. In *1st International Workshop on Learning from Multi-Label Data (MLD'09)*. Bled, Slovenia, 101–116.

G. Tsoumakas and Katakis. 2007. Multi Label Classification: An Overview. *International Journal of Data Warehousing and Mining* 3, 3 (2007), 1–13.

G. Tsoumakas, I. Katakis, and I. Vlahavas. 2008. Effective and Efficient Multilabel Classification in Domains with Large Number of Labels. In *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*.

G. Tsoumakas, I. Katakis, and I. Vlahavas. 2010a. *Data Mining and Knowledge Discovery Handbook, Part 6*. Springer, Chapter Mining Multi-label Data, 667–685.

G. Tsoumakas, I. Katakis, and I. Vlahavas. 2010b. Random k-Labelsets for Multi-Label Classification. *IEEE Transactions on Knowledge and Data Engineering* 23(7) (2010), 1079–1089.

Grigorios Tsoumakas, Eleftherios Spyromitros-Xioufis, Jozef Vilcek, and Ioannis Vlahavas. 2011. Mulan: A Java Library for Multi-Label Learning. *Journal of Machine Learning Research* 12 (2011), 2411–2414.

G. Tsoumakas and I. Vlahavas. 2007. Random k-labelsets: An ensemble method for multilabel classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 4701 LNAI (2007), 406–417.

Grigorios Tsoumakas, Min Ling Zhang, and Zhi-Hua Zhou. 2009. Learning from Multi-label Data. ECML/PKDD'09. (September 2009).

D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. 2008. Semantic Annotation and Retrieval of Music and Sound Effects. *Audio, Speech, and Language Processing, IEEE Transactions on* 16(2) (2008), 467 –476.

Naonori Ueda and Kazumi Saito. 2002a. Parametric Mixture Models for Multi-Labeled Text. In *Neural Information Processing Systems (NIPS)*. 721–728.

Naonori Ueda and Kazumi Saito. 2002b. Yahoo dataset. http://www.kecl.ntt.co.jp/as/members/ueda/yahoo.tar.gz. (2002).

E. Ukwatta and J. Samarabandu. 2009. Vision Based Metal Spectral Analysis Using Multi-label Classification. In *Canadian Conference on Computer and Robot Vision (CRV '09)*. 132 –139.

Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. 2008. Decision trees for hierarchical multi-label classification. *Machine Learning* 73, 2 (Nov. 2008), 185–214.

Sergeja Vogrincic and Zoran Bosnic. 2011. Ontology-based multi-label classification of economic articles. *Comput. Sci. Inf. Syst.* 8, 1 (2011), 101–119.

Jingdong Wang, Yinghai Zhao, Xiuqing Wu, and Xian-Sheng Hua. 2010. A transductive multi-label learning approach for video concept detection. *Pattern Recognition* 44 (July 2010), 2274–2286.

Mei Wang, Xiangdong Zhou, and Tat S. Chua. 2008. Automatic image annotation via local multi-label classification. In *Proceedings of the 2008 international conference on Content-based image and video retrieval (CIVR '08)*. ACM, New York, NY, USA, 17–26.

Wei Wang and Zhi-Hua Zhou. 2008. On Multi-view Active Learning and the Combination with Semi-supervised Learning. In *Proceedings of the 25th International Conference on Machine Learning (ICML '08)*. New York, NY, USA, 1152–1159.

David H. Wolpert. 1992. Stacked Generalization. *Neural Networks* 5 (1992), 241–259.

Jianhua Xu. 2012. An efficient multi-label support vector machine with a zero label. *Expert Systems with Applications* 39, 5 (2012), 4796 – 4804.

Jianhua Xu. 2013a. Fast multi-label core vector machine. *Pattern Recognition* 46, 3 (2013), 885 – 898. DOI:http://dx.doi.org/10.1016/j.patcog.2012.09.003

Jianhua Xu. 2013b. Laboratory of Intelligent computation. http://computer.njnu.edu.cn/Lab/LABIC/LABIC_Software.html. (June 2013).

Rong Yan, Jelena Tesic, and John R. Smith. 2007. Model-shared subspace boosting for multi-label classification. In *13th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, USA, 834–843.

Yan Yan, Glenn Fung, Jennifer G. Dy, and Romer Rosales. 2010. Medical coding classification by leveraging inter-code relationships. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '10)*. ACM, New York, NY, USA, 193–202.

Yiming Yang. 1999. An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval* 1 (May 1999), 69–90. Issue 1-2.

Yiming Yang. 2001. A study of thresholding strategies for text categorization. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '01)*. ACM, New York, NY, USA, 137–145.

Yiming Yang and Siddharth Gopal. 2012. Multilabel classification with meta-level features in a learning-to-rank framework. *Machine Learning* 88 (2012), 47–68. Issue 1.

Y. Yang and X. Liu. 1999. A re-examination of text categorization methods. In *22nd Annual International SIGIR*. Berkley, 42–49.

Yang Yang and Bao-Liang Lu. 2006. Prediction of Protein Subcellular Multi-locations with a Min-Max Modular Support Vector Machine. In *Advances in Neural Networks - ISNN 2006*, Jun Wang, Zhang Yi, Jacek Zurada, Bao-Liang Lu, and Hujun Yin (Eds.). Lecture Notes in Computer Science, Vol. 3973. Springer Berlin / Heidelberg, Berlin, Heidelberg, Chapter 98, 667–673.

Yiming Yang and Jan O. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning (ICML '97)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 412–420.

John Yearwood, Musa Mammadov, and Arunava Banerjee. 2010. Profiling Phishing Emails Based on Hyperlink Information. In *2010 International Conference on Advances in Social Networks Analysis and Mining*. IEEE, 120–127.

Kai Yu, Shipeng Yu, and Volker Tresp. 2005. Multi-label informed latent semantic indexing. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, New York, NY, USA, 258–265.

Jintao Zhang and Jun Huan. 2012. Inductive Multi-task Learning with Multiple View Data. In *18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 543–551.

Min-Ling Zhang. 2009. Ml-rbf: RBF Neural Networks for Multi-Label Learning. *Neural Processing Letters* 29, 2 (April 2009), 61–74.

Min-Ling Zhang, José M. Peña, and Victor Robles. 2009. Feature selection for multi-label naive Bayes classification. *Information Sciences* 179, 19 (Sept. 2009), 3218–3229.

Min L. Zhang and Kun Zhang. 2010. Multi-label learning by exploiting label dependency. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '10)*. ACM, New York, NY, USA, 999–1008.

Min-Ling Zhang and Zhi-Hua Zhou. 2005. A k-Nearest Neighbor Based Algorithm for Multi-label Classification, In Proceedings of the IEEE International Conference on Granular Computing (GrC). *IEEE International Conference on Granular Computing* 2 (July 2005), 718–721.

Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization. *IEEE Transactions on Knowledge and Data Engineering* 18, 10 (Oct. 2006), 1338–1351.

Min L. Zhang and Zhi H. Zhou. 2007. Multi-Label Learning by Instance Differentiation. In *AAAI*. 669–674.

Xiatian Zhang, Quan Yuan, Shiwan Zhao, Wei Fan, Wentao Zheng, and Zhong Wang. 2010. Multi-label classification without the multi-label cost. In *10th SIAM International Conference on Data Mining*.

Yi Zhang. 2012. *Learning with Limited Supervision by Input and Output Coding*. Ph.D. Dissertation. Carnegie Mellon University.

Yi Zhang, Samuel Burer, W. Nick Street, Kristin Bennett, and Emilio Parrado-hern. 2006. Ensemble Pruning Via Semi-definite Programming. *Journal of Machine Learning Research* 7 (2006), 1315–1338.

Yin Zhang and Zhi H. Zhou. 2010. Multilabel dimensionality reduction via dependence maximization. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 4 (Oct. 2010), paper 14.

Tianyi Zhou, Dacheng Tao, and Xindong Wu. 2012a. Compressed labeling on distilled labelsets for multi-label learning. *Machine Learning* 88 (Jan. 2012), 69–126.

Zhi H. Zhou and Min L. Zhang. 2006. Multi-Instance Multi-Label Learning with Application to Scene Classification. In *NIPS*, Bernhard Schölkopf, John C. Platt, and Thomas Hoffman (Eds.). 1609–1616.

Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. 2012b. Multi-instance multi-label learning. *Artificial Intelligence* 176, 1 (2012), 2291 – 2320.

Shenghuo Zhu, Xiang Ji, Wei Xu, and Yihong Gong. 2005. Multi-labelled classification using maximum entropy method. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM Press, New York, NY, USA, 274–281.

Received February 2007; revised March 2009; accepted June 2009