Evaluation of Different Data-Derived Label Hierarchies in Multi-label Classification

Gjorgji Madjarov[1](✉), Ivica Dimitrovski[1], Dejan Gjorgjevikj[1], and Sašo Džeroski[2]

[1] Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University,
Rudgjer Boshkovikj 16, 1000 Skopje, Macedonia
{gjorgji.madjarov,ivica.dimitrovski,dejan.gjorgjevikj}@finki.ukim.mk

[2] Jožef Stefan Institute, Jamova Cesta 39, 1000 Ljubljana, Slovenia
saso.dzeroski@ijs.si

Abstract. Motivated by an increasing number of new applications, the research community is devoting an increasing amount of attention to the task of multi-label classification (MLC). Many different approaches to solving multi-label classification problems have been recently developed. Recent empirical studies have comprehensively evaluated many of these approaches on many datasets using different evaluation measures. The studies have indicated that the predictive performance and efficiency of the approaches could be improved by using data derived (artificial) hierarchies, in the learning and prediction phases. In this paper, we compare different clustering algorithms for constructing the label hierarchies (in a data-driven manner), in multi-label classification. We consider flat label sets and construct the label hierarchies from the label sets that appear in the annotations of the training data by using four different clustering algorithms (balanced k-means, agglomerative clustering with single and complete linkage and predictive clustering trees). The hierarchies are then used in conjunction with global hierarchical multi-label classification (HMC) approaches. The results from the statistical and experimental evaluation reveal that the data-derived label hierarchies used in conjunction with global HMC methods greatly improve the performance of MLC methods. Additionally, multi-branch hierarchies appear much more suitable for the global HMC approaches as compared to the binary hierarchies.

Keywords: Multi-label · Hierarchical · Classification · Clustering

1 Introduction

Multi-label learning is concerned with learning from examples, where each example is associated with multiple labels. Multi-label classification (MLC) has received significant attention in the research community over the past few years, motivated by an increasing number of new applications. The latter include semantic annotation of images and video (news clips, movies clips), functional genomics (predicting gene and protein function), music categorization into emotions, text classification (news articles, web pages, patents, e-mails, bookmarks...), directed marketing and others.

A. Appice et al. (Eds.): NFMCP 2014, LNAI 8983, pp. 19–37, 2015.
DOI: 10.1007/978-3-319-17876-9_2

Madjarov et al. [1] presented an extensive experimental evaluation of the most popular methods for multi-label learning using a wide range of evaluation measures on a variety of datasets. In particular, the authors have experimentally evaluated 12 methods using 16 evaluation measures over 11 benchmark datasets. The results reveal that the best performing methods over all evaluation measures are the Hierarchy Of Multi-label classifiERs (HOMER) [2] and Random Forests of Predictive Clustering Trees for Multi-target Classification (RF-PCTs for MTC) [3], followed by Binary Relevance (BR) [4] and Classifier Chains (CC) [5].

Binary Relevance method addresses the multi-label learning problem by learning one classifier for each class, using all the examples labeled with that class as positive examples and all remaining examples as negative examples. Classifier Chain method involves Q binary classifiers linked along a chain where each classifier deals with the binary relevance problem associated with label $\lambda_i \in \mathcal{L}$, $(1 \leq i \leq Q)$. The feature space of each link in the chain is extended with the 0/1 label associations of all previous links. On the other hand, HOMER transforms the (original, flat) multi-label learning task into a hierarchy of (simpler) multi-label learning tasks, based on a hierarchy of labels derived from the data. The hierarchy is obtained by applying an unsupervised (clustering) approach to the label part of the data that comes from the original MLC problem. For solving the newly defined MLC problems in each node of the hierarchy, HOMER utilizes local BR classifiers. We believe that the better predictive performance and efficiency of the HOMER method as compared to BR and CC in the extensive experimental evaluation [1], is a result of the data derived (artificial) hierarchy, that HOMER defines over the output space of the original MLC problem first, and then uses it in the learning and prediction phases.

In this paper, we experimentally show that structuring the output space (label part) of a flat MLC problem, and using this structure by a classifier that can directly handle HMC problems can improve the predictive performance of a classifier that does not use this structure and directly solves the flat MLC problem. In particular, we derive a hierarchy from the output space of the (original) flat MLC problem using a clustering approach first, and then use a HMC method for solving the newly defined hierarchical multi-label classification problem. To show the improvements that can be achieved by using a data derived structure on the label space, we compare: single PCT [6] for solving classical MLC problems [3], and single PCT for solving HMC problems [7] (both in global settings). Also, we evaluate and analyze the influence of the data-derived label hierarchies, by using four different clustering methods: balanced k-means clustering [2], agglomerative clustering with single and complete linkage [8] and clustering performed by predictive clustering trees for multi-target classification (MTP) [6].

The remainder of this paper is organized as follows. Section 2 defines the tasks of multi-label classification, multi-label ranking and hierarchical multi-label classification. The use of data derived label hierarchies in multi-label classification is presented in Sect. 3. Section 4 describes the multi-label datasets, the evaluation measures and the experimental setup, while Sect. 5 presents and discusses the experimental results. Finally, the conclusions and directions for further work are presented in Sect. 6.

2 Background

In this section, we define the task of multi-label classification and the task of hierarchical multi-label classification.

2.1 The Task of Multi-label Classification (MLC)

Multi-label learning is concerned with learning from examples, where each example is associated with multiple labels. These multiple labels belong to a predefined set of labels. We can distinguish two types of tasks: multi-label classification and multi-label ranking.

In the case of multi-label classification, the goal is to construct a predictive model that will provide a list of relevant labels for a given, previously unseen example. On the other hand, the goal of the task of multi-label ranking is to construct a predictive model that will provide, for each unseen example, a list of preferences (i.e., a ranking) on the labels from the set of possible labels.

The task of multi-label learning is defined as follows [9]:

Given:

- An input space $\mathcal{X}$ that consists of vectors of values of primitive data types (nominal or numeric), i.e., $\forall \mathbf{x_i} \in \mathcal{X}, \mathbf{x_i} = (x_{i_1}, x_{i_2}, ..., x_{i_D})$, where D is the size of the vector (or number of descriptive attributes),
- an output space $\mathcal{Y}$ that is defined as a subset of a finite set of disjoint labels $\mathcal{L} = \{\lambda_1, \lambda_2, ..., \lambda_Q\}$ ($Q > 1$ and $\mathcal{Y} \subseteq \mathcal{L}$)
- a set of examples E, where each example is a pair of a vector and a set from the input and output space respectively, i.e., $E = \{(\mathbf{x_i}, \mathcal{Y}_i) | \mathbf{x_i} \in \mathcal{X}, \mathcal{Y}_i \subset \mathcal{L}, 1 \leq i \leq N\}$ where N is the number of examples of E ($N = |E|$), and
- a quality criterion q, which rewards models with high predictive performance and low computational complexity.

If the task at hand is multi-label classification, then the goal is to

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

On the other hand, if the task is multi-label ranking, then the goal is to

Find: a function f: $\mathcal{X} \times \mathcal{L} \rightarrow \mathcal{R}$, such that f maximizes q, where $\mathcal{R}$ is the ranking on the labels for a given example.

An extensive bibliography of learning methods for solving multi-label learning problems can be found in [1,4,10,11].

2.2 The Task of Hierarchical Multi-label Classification (HMC)

Hierarchical classification differs from the multi-label classification in the following: the labels are organized in a hierarchy. An example that is labeled with a given label is automatically labeled with all its parent-labels (this is known as the hierarchy constraint). Furthermore, an example can be labeled simultaneously with multiple labels that can follow multiple paths from the root label. This task is called hierarchical multi-label classification (HMC).

Here, the output space $\mathcal{Y}$ is defined with a label hierarchy $(\mathcal{L}, \leq_h)$, where $\mathcal{L}$ is a set of labels and $\leq_h$ is a partial order representing the parent-child relationship ($\forall\ \lambda_1, \lambda_2 \in \mathcal{L} : \lambda_1 \leq_h \lambda_2$ if and only if λ_1 is a parent of λ_2) structured as a tree [9]. Each example from the set of examples E is a pair of a vector and a set from the input and output space respectively, where the set satisfies the hierarchy constraint, i.e., $E = \{(\mathbf{x_i}, \mathcal{Y}_i) | \mathbf{x_i} \in \mathcal{X}, \mathcal{Y}_i \subseteq \mathcal{L}, \lambda \in \mathcal{Y}_i \Rightarrow \forall \lambda' \leq_h \lambda : \lambda' \in \mathcal{Y}_i, 1 \leq i \leq N\}$ where N is the number of examples of E ($N = |E|$). The quality criterion q, rewards models with high predictive performance and low complexity as in the task of multi-label classification.

An extensive bibliography of learning methods for hierarchical classification scattered across different application domains is given by Silla and Freitas [12].

3 The Use of Data Derived Label Hierarchies in Multi-Label Classification

In this study, we suggest to transform the flat multi-label classification problem into a hierarchical multi-label one and solve it by using an approach for HMC [12]. In particular, one should derive a hierarchy from the label part of the original (flat) multi-label classification problem first, and then use this hierarchy to construct hierarchical classification problem that later solves by using a HMC approach [12].

3.1 Generating a Label Hierarchy on a Multi-label Output Space

The process of generating label hierarchies on a multi-label output space is critical for the good performance of the HMC methods on the transformed problems. When we build the hierarchy over the label space, there is only one constraint that we should take care of: the original MLC task should be defined by the leaves of the label hierarchy. In particular, the labels from the original MLC problem represent the leaves of the tree hierarchy (Fig. 1), while the labels that represent the internal nodes of the tree hierarchy are so-called meta-labels (that model the correlation among the original labels).

An example hierarchy of labels generated by using the agglomerative clustering method with single linkage from the *emotions* multi-label classification task (used in the experimental evaluation) is given in Fig. 1. The original label space of the *emotions* dataset has six labels $\{\lambda_1, \lambda_2, ..., \lambda_6\}$ and each example from the dataset originally is labeled with one or more labels. Table 1 shows five examples from the *emotions* dataset with their original labels (third column - *original labels*) and the corresponding hierarchical labels (fourth column - *hierarchical labels*) obtained by using the label hierarchy from Fig. 1 ($\mathcal{HL} = \{\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6\}$). Each example in the transformed, HMC dataset is actually labeled with multiple paths of the hierarchy,

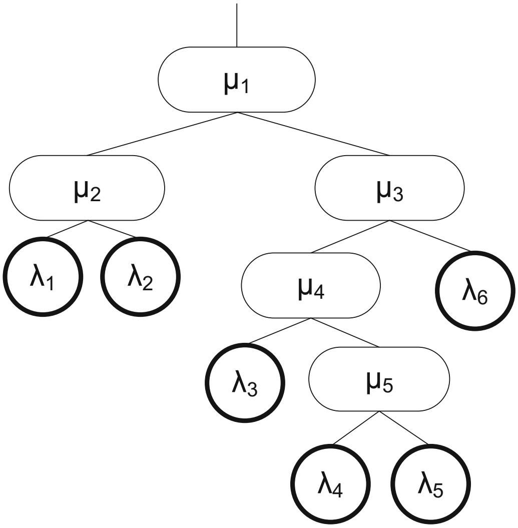


Fig. 1. An example of label hierarchy defined over the flat label space of the *emotions* dataset by using agglomerative clustering method with single linkage (λ_i - original label, μ_i - artificially defined meta-label).

defined from the root to the leaves (represented by the relevant labels for the corresponding example in the original MLC dataset).

In this study, we use four different clustering approaches (two divisive and two agglomerative) for deriving the hierarchy on the output space of the (original) MLC problem:

- balanced k-means clustering approach [2] (divisive approach),
- predictive clustering trees [6] (divisive approach),
- agglomerative clustering by using complete linkage [8], and
- agglomerative clustering by using single linkage [8].

Balanced k-means creates the label hierarchy by partitioning the original labels recursively in a top-down depth-first fashion. The top node of the hierarchy contains all labels. At each node n, $k <= |\mathcal{L}_n|$ child nodes are created. The labels

Table 1. Five examples from the *emotions* dataset with their *original labels* and the corresponding *hierarchical labels* obtained by using the label hierarchy from Fig. 1

example	**features**	**original labels**	**hierarchical labels**
$\mathbf{x_1}$	$x_{1\,1}, x_{1\,2}, \dots, x_{1\,72}$	$\{\boldsymbol{\lambda_1}\}$	$\{\mu_1, \mu_2, \boldsymbol{\lambda_1}\}$
$\mathbf{x_2}$	$x_{2\,1}, x_{2\,2}, \dots, x_{2\,72}$	$\{\boldsymbol{\lambda_3}, \boldsymbol{\lambda_5}\}$	$\{\mu_1, \mu_3, \mu_4, \mu_5, \boldsymbol{\lambda_3}, \boldsymbol{\lambda_5}\}$
$\mathbf{x_3}$	$x_{3\,1}, x_{3\,2}, \dots, x_{3\,72}$	$\{\boldsymbol{\lambda_6}\}$	$\{\mu_1, \mu_3, \boldsymbol{\lambda_6}\}$
$\mathbf{x_4}$	$x_{4\,1}, x_{4\,2}, \dots, x_{4\,72}$	$\{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_6}\}$	$\{\mu_1, \mu_2, \mu_3, \lambda_1, \boldsymbol{\lambda_6}\}$
$\mathbf{x_5}$	$x_{5\,1}, x_{5\,2}, \dots, x_{5\,72}$	$\{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}, \boldsymbol{\lambda_6}\}$	$\{\mu_1, \mu_2, \mu_3, \boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}, \boldsymbol{\lambda_6}\}$

of the current node are distributed (divided) using a clustering method into k disjoint subsets (k meta-labels) with an explicit constraint on the size of each subset, one for each child of the current node.

In this work, we use a specific setting from the predictive clustering framework as in [3,13], where the target space is equal to the descriptive space, i.e., the descriptive variables are used to provide descriptions for the obtained clusters. This focuses the predictive clustering setting on the task of clustering instead of classification.

Agglomerative clustering algorithms treat each example as a singleton cluster at the outset and then successively merge pairs of clusters until all clusters have been merged into a single cluster that contains all examples.

The predictive clustering trees and the agglomerative approaches produce binary tree hierarchies, while the balanced k-means clustering approach produces multi-branch tree hierarchies for $k > 2$.

3.2 Solving MLC Problems by Using Classification Approaches for HMC

After the transformation of the original MLC problem into a HMC one, the new HMC problem can be solved by a hierarchical multi-label learning approach. The transformed hierarchical multi-label dataset satisfies the hierarchy constraint (an example that is labeled with a given label is automatically labeled with all its parent-labels).

Figure 2 presents the pseudo-code of the algorithm for solving a MLC problem by using data-derived label hierarchies and a classification approach for HMC. The algorithm first defines the hierarchy, then solves the HMC problem by using a classification approach for HMC. It finally extracts the predictions for the leaves of the hierarchy (that are actually the predictions for the original labels) and evaluates the performance.

E^{train} and E^{test} denote the training and testing examples, while $\mathbf{W}^{train}$ is only the label part (label data) of the training set. Using the label hierarchy derived from the label data, $\mathbf{W}^{train}$ is transformed into new hierarchically organized label data $\mathbf{W}_H^{train}$. E_H^{train} and E_H^{test} denote the corresponding hierarchical multi-label datasets obtained by transforming the original (flat) multi-label datasets (E^{train} and E^{test}) into hierarchical form.

P_H denotes the predictions for the examples of the hierarchical multi-label dataset E_H^{test}, while P denotes the predictions for the original labels. The latter are obtained by extracting the probabilities in the leaves of the label tree from the predictions P_H. The predictions P_H are represented as vectors of probabilities (one vector for one example), where each probability is associated to only one label from the hierarchy (meta-label representing an internal node or original label representing a leaf). Predictions P in the original multi-label scenario can be obtained by using different approaches for transforming the hierarchical multi-label predictions P_H. In this work, we use the simplest approach: only the probabilities for the leaves from the hierarchical predictions P_H are evaluated, while the other probabilities (for the meta-labels) are simply ignored.

```
**procedure** MLCToHMC($E^{train}$ ,$E^{test}$) returns performance
$\mathbf{W}^{train}$ = ExtractLabelSet($E^{train}$);
$\mathbf{W}_H^{train}$ = DefineHierarchy($\mathbf{W}^{train}$);

//transform multi-label dataset to hierarchical multi-label one
$E_H^{train}$ = MLCToHMCTrainDataset($E^{train}$, $\mathbf{W}_H^{train}$);
$E_H^{test}$ = MLCToHMCTestDataset($E^{test}$, $\mathbf{W}_H^{train}$);

//solve transformed hierarchical multi-label problem
//by using approach for HMC
HMCModel = HMCMetod($E_H^{train}$);

//generate HMC predictions
$P_H$ = HMCModel($E_H^{test}$);

//Extract predictions only for the leaves from the HMC predictions $P_H$
$P$ = ExtractLeavesPredictionsFromHMCPredictions($P_H$, $\mathbf{W}_H^{train}$, $\mathbf{W}^{train}$);
**return** EvaluatePredictions(P);
```

Fig. 2. Solving flat MLC problems by using classification approaches for HMC.

3.3 Classification Approaches for HMC

Based on the existing literature, Silla and Freitas [12] propose a unifying framework for hierarchical classification, including a taxonomy of hierarchical classification problems and methods. One of the dimensions along which the hierarchical classification methods differ is the way of using (exploring) the hierarchical label structure in the learning and prediction phases. They reviewed two different approaches that utilize the hierarchy: the top-down (or local) approach that uses local information to create a set of local classifiers and the global (or big-bang) approach.

The recent research shows that learning a single global model for all labels (in the hierarchy) can have some advantages [3,14] over the local approaches. The total size of the global classification model is typically smaller as compared to the total size of all the local models learned by local classifier approaches. Also, in the global classifier approach, a single classification model is built from the training set, taking into account the label hierarchy and relationships. During the prediction phase, each test example is classified using the induced model, in a process that can assign labels to a test example at potentially every level of the hierarchy. Because of that, in this study we compare PCTs for MTP (as flat, global MLC approach) and PCTs for HMC (in a global setting) [3], instead of using local ("per parent node") setting [12] as HOMER does.

Table 2. Description of the benchmark problems in terms of application domain (*domain*), number of training ($\#tr.e.$) and test ($\#t.e.$) examples, the number of features (D), the total number of labels (Q) and label cardinality - average number of labels per example (l_c). The problems are ordered by their overall complexity roughly calculated as $\#tr.e. \times D \times Q$.

	Domain	$\#tr.e.$	$\#t.e.$	D	Q	l_c
Emotions [15]	Multimedia	391	202	72	6	1.87
Scene [16]	Multimedia	1211	1159	294	6	1.07
Yeast [17]	Biology	1500	917	103	14	4.24
Medical [18]	Text	645	333	1449	45	1.25
Enron [19]	Text	1123	579	1001	53	3.38
Corel5k [20]	Multimedia	4500	500	499	374	3.52
Tmc2007 [21]	Text	21519	7077	500	22	2.16
Mediamill [22]	Multimedia	30993	12914	120	101	4.38
Bibtex [23]	Text	4880	2515	1836	159	2.40
Delicious [2]	Text	12920	3185	500	983	19.02
Bookmarks [23]	Text	60000	27856	2150	208	2.03

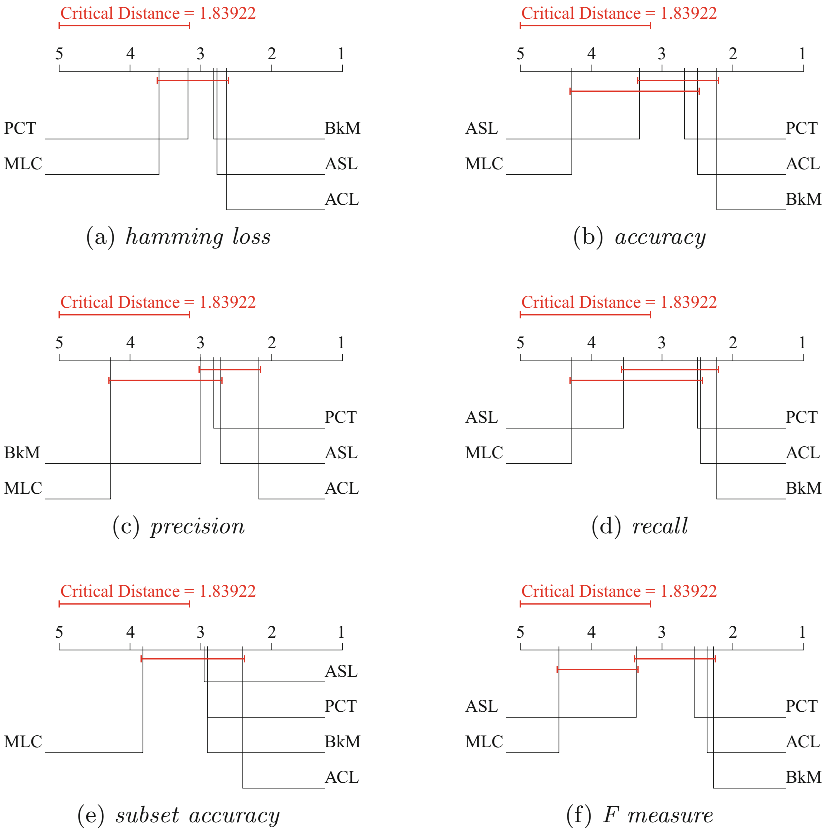
4 Experimental Design

4.1 Datasets and Evaluation Measures

We use 11 multi-label classification benchmark problems used in previous studies and evaluations of methods for multi-label learning. Table 2 presents the basic statistics of the datasets. The datasets come from the domain of text categorization, multimedia and biology and pre-divided into training and testing parts as used by other researchers.

In any multi-label experiment, it is essential to include multiple and contrasting measures because of the additional degrees of freedom that the multi-label setting introduces. In our experiments, we used various evaluation measures that have been suggested by Tsoumakas et al. [11] In particular, we used 12 *bipartitions-based* evaluation measures: six *example-based* evaluation measures (*hamming loss*, *accuracy*, *precision*, *recall*, *F measure* and *subset accuracy*) and six *label-based* evaluation measures (*micro precision*, *micro recall*, *micro* F_1, *macro precision*, *macro recall* and *macro* F_1). Note that these evaluation measures require predictions stating that a given label is present or not (binary 1/0 predictions). However, most predictive models predict a numerical value for each label and the label is predicted as present if that numerical value exceeds some pre-defined threshold τ. The performance of the predictive model thus directly depends on the selection of an appropriate value of τ.

Also, we used four *ranking-based* evaluation measures (*one-error*, *coverage*, *ranking loss* and *average precision*) that compare the predicted ranking of the



(a) *hamming loss*

(b) *accuracy*

(c) *precision*

(d) *recall*

(e) *subset accuracy*

(f) *F measure*

Fig. 3. The critical diagrams for the example-based evaluation measures: The results from the Nemenyi post-hoc test at 0.05 significance leve.

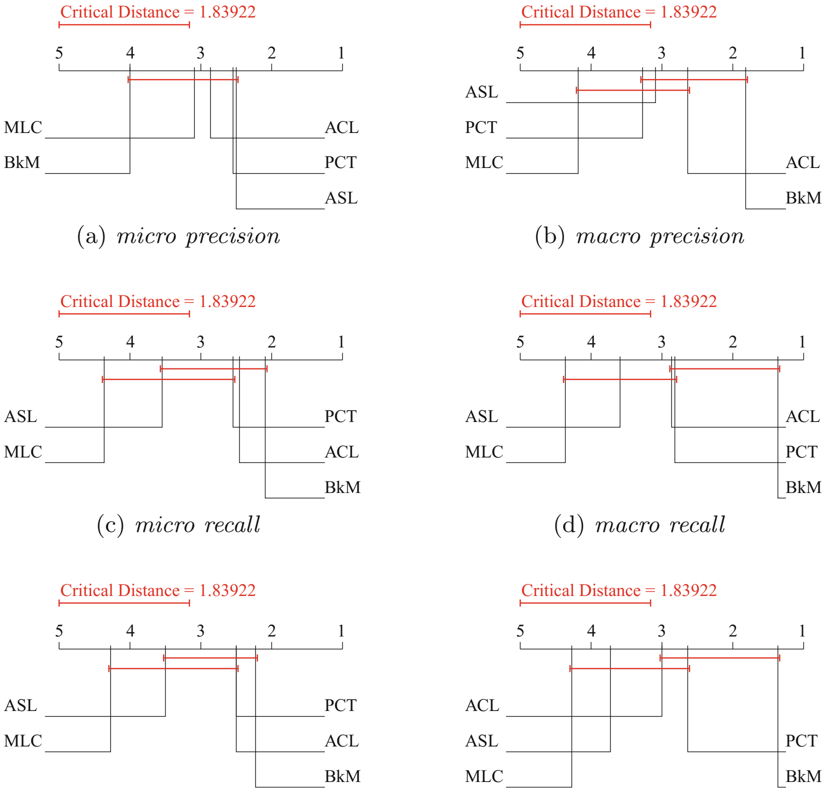
labels with the ground truth ranking. A detailed description of the evaluation measures is given in Appendix A.

4.2 Experimental Setup

The comparison of the multi-label learning methods was performed using the CLUS[1] system for predictive clustering. All experiments were performed on a server with an Intel Xeon processor at 2.5 GHz and 64 GB of RAM with the Fedora 14 operating system. We used the default settings of CLUS to learn the single PCT approaches (PCTs for MTP - as flat MLC approach, and PCTs for HMC). The threshold τ for the *bipartitions-based* evaluation measures was set to 0.5 for all compared methods.

The balanced k-means clustering method requires to be configured the number of clusters k in each node of the hierarchy. For this parameter, five different

[1] http://clus.sourceforge.net.



(a) *micro precision*

(b) *macro precision*

(c) *micro recall*

(d) *macro recall*

(e) *micro* F_1

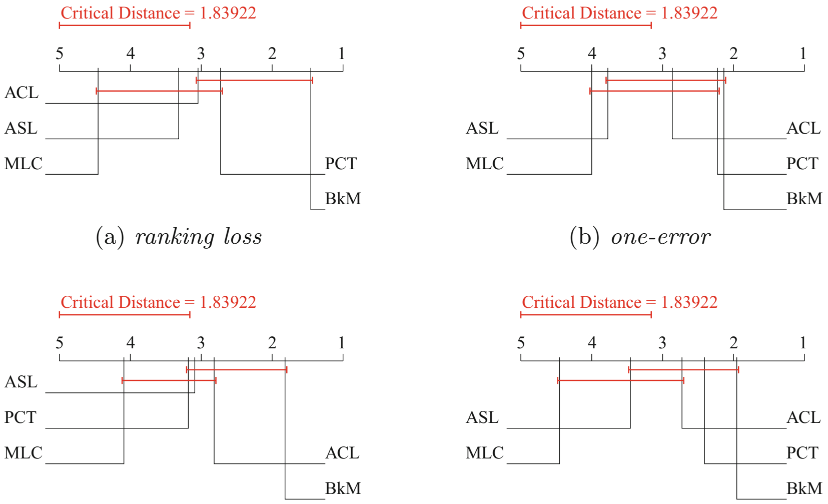
(f) *macro* F_1

Fig. 4. The critical diagrams for the label-based evaluation measures: The results from the Nemenyi post-hoc test at 0.05 significance level.

values (2–6) were considered in the cross-validation phase [2]. After determining the best value of k on every dataset (via cross-validation on the training dataset), the PCT for HMC was trained using all available training examples and was evaluated by recognizing all test examples from the corresponding dataset. The values of the parameter k are 3 for most of the datasets, 2 for the *emotions* dataset, 5 for the *yeast* dataset, and 4 for the *enron* and *delicious* datasets. Also, for the balanced k-means and the agglomerative methods, Euclidean distance was used as a distance measure.

4.3 Statistical Evaluation

To assess whether the overall differences in performance across the five different approaches are statistically significant, we employed the corrected Friedman test [24] and the post-hoc Nemenyi test [25] as recommended by Demšar [26].



(a) *ranking loss*

(b) *one-error*

(c) *coverage*

(d) *average precision*

Fig. 5. The critical diagrams for the ranking-based evaluation measures: The results from the Nemenyi post-hoc test at 0.05 significance level.

If a statistically significant difference in the performance is detected, then next step is a post-hoc test to detect between which algorithms those differences appear. The Nemenyi test is used to compare all the classifiers to each other. In this procedure, the performance of two classifiers is significantly different if their average ranks differ by more than some critical distance. The critical distance depends on the number of algorithms, the number of datasets and the critical value (for a given significance level - p) that is based on the Studentized range statistic and can be found in statistical textbooks (e.g., see [27]).

We present the results from the Nemenyi post-hoc test with average rank diagrams [26]. These are given in Figs. 3, 4 and 5. A critical diagram contains an enumerated axis on which the average ranks of the algorithms are drawn. The algorithms are depicted along the axis in such a manner, that the best ranking ones are at the right-most side of the diagram. The lines for the average ranks of the algorithms that do not differ significantly (at the significance level of $p = 0.05$) are connected with a line.

5 Results and Discussion

In this section, we present the results from the experimental evaluation. For each type of evaluation measure, we present and discuss the critical diagrams from the tests for statistical significance. The complete results over all evaluation measures are given in Appendix B. We have compared five different method:

- PCTs for MTP, that don't use a hierarchy for solving the original MLC problem (labeled as *MLC*)

Table 3. The predictive performances of PCTs for MLC obtained on the original (flat) MLC problems and PCTs for HMC obtained on the transformed (newly) defined HMC problems by using four different clustering approaches (balanced *k*-means, predictive clustering trees, and agglomerative clustering with complete and single linkage) along 16 performance evaluation measures on all datasets.

	HammingLoss	Accuracy	Precision	Recall	Fmeasure	SubsetAccuracy	MicroPrecision	MicroRecall	MicroF1	MacroPrecision	MacroRecall	MacroF1	OneError	Coverage	RankingLoss	AvgPrecision
emotions																
no hierarchy (flat MLC)	0.267	0.448	0.577	0.534	0.554	0.223	0.607	0.539	0.571	0.628	0.533	0.568	0.386	2	0.27	0.713
balanced-k-means - HMC	0.274	0.419	0.587	0.501	0.54	0.144	0.602	0.496	0.544	0.644	0.499	0.522	0.391	2	0.247	0.731
agglomerative (complete) - HMC	0.266	0.441	0.616	0.518	0.563	0.173	0.619	0.501	0.554	0.645	0.493	0.518	0.386	2	0.253	0.73
agglomerative (single) - HMC	0.266	0.441	0.616	0.518	0.563	0.173	0.619	0.501	0.554	0.645	0.493	0.518	0.386	2	0.253	0.73
PCTs - HMC	0.269	0.416	0.611	0.458	0.524	0.163	0.629	0.446	0.522	0.627	0.422	0.471	0.361	2	0.25	0.742
scene																
no hierarchy (flat MLC)	0.129	0.538	0.565	0.539	0.552	0.509	0.692	0.521	0.594	0.682	0.529	0.592	0.389	1	0.174	0.75
balanced-k-means - HMC	0.142	0.523	0.547	0.538	0.542	0.483	0.63	0.527	0.574	0.629	0.538	0.578	0.413	1	0.202	0.728
agglomerative (complete) - HMC	0.149	0.418	0.439	0.425	0.432	0.39	0.636	0.413	0.501	0.638	0.418	0.501	0.449	1	0.224	0.699
agglomerative (single) - HMC	0.149	0.418	0.439	0.425	0.432	0.39	0.636	0.413	0.501	0.638	0.418	0.501	0.449	1	0.224	0.699
PCTs - HMC	0.155	0.504	0.528	0.514	0.521	0.469	0.582	0.506	0.541	0.593	0.509	0.547	0.447	1	0.227	0.701
yeast																
no hierarchy (flat MLC)	0.219	0.44	0.705	0.49	0.578	0.153	0.699	0.492	0.577	0.479	0.269	0.293	0.264	7	0.2	0.725
balanced-k-means - HMC	0.216	0.469	0.68	0.549	0.607	0.138	0.68	0.545	0.605	0.445	0.308	0.327	0.256	7	0.196	0.73
agglomerative (complete) - HMC	0.217	0.456	0.69	0.521	0.594	0.144	0.69	0.519	0.592	0.459	0.289	0.307	0.265	7	0.198	0.728
agglomerative (single) - HMC	0.217	0.456	0.69	0.521	0.594	0.144	0.69	0.519	0.592	0.459	0.289	0.307	0.265	7	0.198	0.728
PCTs - HMC	0.217	0.457	0.687	0.524	0.595	0.147	0.687	0.522	0.593	0.46	0.292	0.314	0.265	7	0.197	0.727
medical																
no hierarchy (flat MLC)	0.023	0.228	0.285	0.228	0.253	0.177	0.826	0.227	0.356	0.018	0.022	0.02	0.613	5	0.104	0.522
balanced-k-means - HMC	0.014	0.665	0.721	0.692	0.706	0.58	0.812	0.66	0.728	0.306	0.254	0.27	0.213	3	0.054	0.801
agglomerative (complete) - HMC	0.013	0.698	0.76	0.717	0.738	0.616	0.821	0.682	0.745	0.277	0.226	0.24	0.219	3	0.048	0.819
agglomerative (single) - HMC	0.013	0.677	0.736	0.693	0.714	0.601	0.829	0.663	0.737	0.262	0.223	0.235	0.225	3	0.045	0.819
PCTs - HMC	0.013	0.676	0.739	0.695	0.716	0.592	0.838	0.667	0.743	0.251	0.203	0.219	0.219	3	0.045	0.819
enron																
no hierarchy (flat MLC)	0.058	0.196	0.415	0.229	0.295	0.002	0.602	0.247	0.35	0.023	0.03	0.026	0.392	15	0.114	0.547
balanced-k-means - HMC	0.052	0.37	0.61	0.412	0.492	0.097	0.646	0.386	0.483	0.101	0.077	0.082	0.28	13	0.094	0.642
agglomerative (complete) - HMC	0.051	0.4	0.643	0.454	0.532	0.102	0.642	0.427	0.513	0.1	0.08	0.084	0.244	14	0.098	0.647
agglomerative (single) - HMC	0.051	0.357	0.693	0.38	0.491	0.097	0.689	0.345	0.459	0.088	0.056	0.061	0.264	13	0.097	0.644
PCTs - HMC	0.051	0.397	0.65	0.445	0.528	0.105	0.651	0.417	0.508	0.087	0.076	0.078	0.25	14	0.098	0.643
corel5k																
no hierarchy (flat MLC)	0.009	0	0	0	0	0	0	0	0	0	0	0	0.777	116	0.139	0.208
balanced-k-means - HMC	0.009	0.021	0.061	0.022	0.032	0.002	0.52	0.022	0.042	0.016	0.004	0.006	0.71	115	0.132	0.253
agglomerative (complete) - HMC	0.011	0.058	0.193	0.059	0.09	0.004	0.217	0.059	0.093	0.007	0.004	0.003	0.778	121	0.152	0.202
agglomerative (single) - HMC	0.011	0.058	0.193	0.059	0.09	0.004	0.215	0.059	0.093	0.007	0.004	0.003	0.782	122	0.155	0.195
PCTs - HMC	0.009	0.021	0.064	0.022	0.033	0.002	0.603	0.023	0.045	0.014	0.003	0.004	0.71	116	0.133	0.254
tmc2007																
no hierarchy (flat MLC)	0.075	0.436	0.659	0.478	0.554	0.215	0.689	0.454	0.547	0.386	0.235	0.263	0.307	5	0.100	0.700
balanced-k-means - HMC	0.067	0.515	0.688	0.604	0.643	0.253	0.704	0.563	0.625	0.735	0.341	0.409	0.246	3	0.066	0.774
agglomerative (complete) - HMC	0.069	0.498	0.692	0.572	0.626	0.245	0.708	0.527	0.605	0.562	0.291	0.351	0.26	4	0.073	0.76
agglomerative (single) - HMC	0.068	0.501	0.699	0.571	0.628	0.25	0.717	0.524	0.605	0.629	0.283	0.344	0.247	4	0.071	0.767
PCTs - HMC	0.101	0.559	0.746	0.703	0.723	0.184	0.742	0.625	0.678	0.675	0.358	0.418	0.084	12	0.055	0.835
mediamill																
no hierarchy (flat MLC)	0.034	0.354	0.694	0.379	0.49	0.065	0.743	0.351	0.477	0.04	0.029	0.031	0.22	20	0.063	0.654
balanced-k-means - HMC	0.033	0.386	0.716	0.427	0.535	0.082	0.733	0.393	0.512	0.217	0.054	0.07	0.197	19	0.058	0.684
agglomerative (complete) - HMC	0.033	0.383	0.724	0.419	0.531	0.087	0.746	0.382	0.506	0.103	0.039	0.046	0.191	19	0.059	0.684
agglomerative (single) - HMC	0.033	0.383	0.723	0.417	0.529	0.086	0.75	0.379	0.504	0.138	0.04	0.049	0.192	19	0.059	0.683
PCTs - HMC	0.033	0.387	0.715	0.429	0.536	0.084	0.738	0.392	0.512	0.128	0.046	0.058	0.19	20	0.06	0.683
bibtex																
no hierarchy (flat MLC)	0.014	0.046	0.140	0.046	0.069	0.004	1	0.057	0.108	0.006	0.006	0.006	0.783	59	0.256	0.212
balanced-k-means - HMC	0.015	0.243	0.368	0.290	0.324	0.113	0.550	0.259	0.352	0.296	0.174	0.202	0.449	30	0.105	0.491
agglomerative (complete) - HMC	0.014	0.198	0.343	0.202	0.255	0.111	0.8	0.158	0.263	0.086	0.053	0.06	0.524	36	0.147	0.396
agglomerative (single) - HMC	0.014	0.175	0.289	0.183	0.225	0.103	0.749	0.145	0.243	0.079	0.044	0.052	0.589	46	0.19	0.341
PCTs - HMC	0.014	0.197	0.328	0.204	0.251	0.117	0.796	0.161	0.268	0.082	0.056	0.062	0.541	36.93	0.152	0.388
delicious																
no hierarchy (flat MLC)	0.019	0.001	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000	0.000	0.000	0.592	692	0.172	0.206
balanced-k-means - HMC	0.018	0.118	0.429	0.132	0.201	0.007	0.621	0.120	0.201	0.162	0.049	0.062	0.386	548	0.121	0.336
agglomerative (complete) - HMC	0.018	0.109	0.425	0.121	0.188	0.006	0.618	0.113	0.191	0.116	0.033	0.043	0.396	555	0.123	0.326
agglomerative (single) - HMC	0.019	0.074	0.354	0.081	0.132	0.003	0.59	0.077	0.136	0.064	0.018	0.022	0.44	559	0.131	0.293
PCTs - HMC	0.019	0.097	0.376	0.107	0.167	0.002	0.609	0.101	0.173	0.066	0.029	0.034	0.418	554	0.128	0.316
bookmarks																
no hierarchy (flat MLC)	0.009	0.133	0.133	0.137	0.135	0.129	0.947	0.076	0.141	0.018	0.016	0.017	0.817	74	0.258	0.213
balanced-k-means - HMC	0.009	0.205	0.224	0.211	0.217	0.188	0.776	0.139	0.236	0.299	0.071	0.097	0.651	50	0.169	0.370
agglomerative (complete) - HMC	0.009	0.179	0.191	0.183	0.187	0.167	0.831	0.112	0.197	0.122	0.034	0.041	0.699	53	0.182	0.326
agglomerative (single) - HMC	0.009	0.16	0.163	0.165	0.164	0.153	0.875	0.097	0.175	0.103	0.026	0.03	0.729	58	0.2	0.302
PCTs - HMC	0.009	0.177	0.185	0.181	0.183	0.167	0.846	0.11	0.195	0.116	0.036	0.044	0.699	56	0.193	0.328

– PCTs for HMC, that use data-derived label hierarchies, defined by:
  - balanced k-means clustering approach (labeled as BkM)
  - agglomerative clustering by using complete linkage (labeled as ACL)
  - agglomerative clustering by using single linkage (labeled as ASL)
  - predictive clustering trees (labeled as PCT)

The results of the statistical evaluation are given in Figs. 3, 4 and 5, while the complete results are given in Table 3. Considering the results from the statistical evaluation, we can make several conclusions. The first conclusion that draws our attention is that PCTs for HMC outperform PCTs for MLC on all datasets and on all evaluation measures. The differences in predictive performance are rarely significant at the significance level of 0.05, but the PCTs for HMC (that use balanced k-means clustering approach) are significantly better than PCTs for MLC on 12 out of 16 evaluation measures (*accuracy*, *recall*, *F measure micro recall*, *micro* F_1, *macro precision*, *macro recall*, *macro* F_1, *one-error*, *coverage*, *ranking loss* and *average precision*).

PCTs for HMC that use balanced k-means clustering for deriving the label hierarchies outperform PCTs for HMC that use agglomerative clustering with single and complete linkage and PCTs on all evaluation measures except on *hamming loss*, *precision*, *subset accuracy* and *micro precision*. All clustering approaches that produce binary hierarchies (agglomerative clustering with single and complete linkage and PCTs) show similar results and there is no statistical significant difference between their predictive performance.

Considering the complete results that are given in Table 3 we can see that the highest improvement of utilizing the data-derived hierarchies is obtained on *delicious* dataset, as a result of the largest number of labels and the largest label cardinality (average number of labels per example). A large number of labels and large label cardinality yields a larger hierarchy that emphasizes the relations between labels, and improves the process of learning and prediction. PCTs for MLC outperform PCTs for HMC only on the *scene* and *emotions* datasets, which was a result of the smallest number of labels and label cardinality that have these two datasets.

Finally, multi-branch hierarchy (defined by balanced k-means clustering) is more suitable for the global HMC approach as compared to the binary hierarchies defined by agglomerative clustering with single and complete linkage and PCTs, especially on datasets with higher number of labels and higher label cardinality.

6 Conclusions and Further Work

In this paper, we have investigated the use of label hierarchies in multi-label classification, constructed in a data-driven manner. We consider flat label-sets and construct label hierarchies from the label sets that appear in the annotations of the training data by using clustering approaches based on balanced k-means clustering, agglomerative clustering with single and complete linkage, and clustering performed by PCTs. The hierarchies are then used in conjunction with hierarchical multi-label classification approaches in the hope of achieving better multi-label classification.

In particular, we investigate and evaluate the utility of four different data-derived label hierarchies in the context of predictive clustering trees for HMC. The experimental results clearly show that the use of the hierarchy results in improved performance and the more balanced hierarchy offers better representation of the label relationships.

The label hierarchies used in PCTs for HMC greatly improve the performance of PCTs for MTP (as used for MLC): The results show improvement in performance on all evaluation measures considered. Multi-branch hierarchy (defined by balanced k-means clustering) appears much more suitable for the global HMC approach (PCTs for HMC) as compared to the binary hierarchies defined by agglomerative clustering with single and complete linkage and PCTs. It outperforms binary hierarchies on datasets with higher number of labels and this improvement is especially emphasized on the *delicious* dataset, as a result of the higher label cardinality that this dataset has in comparison to the other evaluated datasets.

The final recommendation considering the performance of the evaluated methods is that we should use data-derived label hierarchies. We should transform the original (flat) multi-label classification problem into hierarchical multi-label one by using more balanced hierarchies, and solve the newly defined hierarchical classification problem by a classifier that can directly handle HMC problems.

We plan to extend the experimental evaluation in this study by using different local approaches (as the approaches *per node* and *per parent node*) for solving the HMC problem. We plan to consider other MLC approaches, local and global, for use in conjunction with the label hierarchies.

A final direction for further work might be the comparison of hierarchies constructed by humans and hierarchies generated in a data-driven fashion. For HMC problems, we can consider the MLC task defined by the leaves of the provided label hierarchy. We can then construct label hierarchies automatically, as described above, and compare these hierarchies (and their utility) to the originally provided label hierarchy.

Acknowledgements. We would like to acknowledge the support of the European Commission through the project MAESTRA - Learning from Massive, Incompletely annotated, and Structured Data (Grant number ICT-2013-612944). Also, we would like to acknowledge the support of the Faculty of Computer Science and Engineering at the "Ss. Cyril and Methodius" University.

A Evaluation Measures

In this section, we present the measures that are used to evaluate the predictive performance of the compared methods in our experiments. In the definitions below, $\mathcal{Y}_i$ denotes the set of true labels of example $\mathbf{x_i}$ and $h(\mathbf{x_i})$ denotes the set of predicted labels for the same examples. All definitions refer to the multi-label setting.

A.1 Example Based Measures

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

$$hamming\_loss(h) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{Q} |h(\mathbf{x_i}) \Delta \mathcal{Y}_i| \tag{1}$$

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

Accuracy for a single example $\mathbf{x_i}$ is defined by the Jaccard similarity coefficients between the label sets $h(\mathbf{x_i})$ and $\mathcal{Y}_i$. Accuracy is micro-averaged across all examples.

$$accuracy(h) = \frac{1}{N} \sum_{i=1}^{N} \frac{|h(\mathbf{x_i}) \bigcap \mathcal{Y}_i|}{|h(\mathbf{x_i}) \bigcup \mathcal{Y}_i|} \tag{2}$$

Precision is defined as:

$$precision(h) = \frac{1}{N} \sum_{i=1}^{N} \frac{|h(\mathbf{x_i}) \bigcap \mathcal{Y}_i|}{|h(\mathbf{x_i})|} \tag{3}$$

Recall is defined as:

$$recall(h) = \frac{1}{N} \sum_{i=1}^{N} \frac{|h(\mathbf{x_i}) \bigcap \mathcal{Y}_i|}{|\mathcal{Y}_i|} \tag{4}$$

F_1 **score** is the harmonic mean between precision and recall and is defined as:

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

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

Subset Accuracy or classification accuracy is defined as follows:

$$subset\_accuracy(h) = \frac{1}{N} \sum_{i=1}^{N} I(h(\mathbf{x_i}) = \mathcal{Y}_i) \tag{6}$$

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

A.2 Label Based Measures

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

$$macro\_precision = \frac{1}{Q}\sum_{j=1}^{Q}\frac{tp_j}{tp_j + fp_j} \tag{7}$$

where tp_j, fp_j are the number of true positives and false positives for the label λ_j considered as a binary class.

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

$$macro\_recall = \frac{1}{Q}\sum_{j=1}^{Q}\frac{tp_j}{tp_j + fn_j} \tag{8}$$

where tp_j, fp_j are defined as for the macro precision and fn_j is the number of false negatives for the label λ_j considered as a binary class.

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

$$macro\_F_1 = \frac{1}{Q}\sum_{j=1}^{Q}\frac{2 \times p_j \times r_j}{p_j + r_j} \tag{9}$$

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

$$micro\_precision = \frac{\sum_{j=1}^{Q} tp_j}{\sum_{j=1}^{Q} tp_j + \sum_{j=1}^{Q} fp_j} \tag{10}$$

where tp_j, fp_j are defined as for macro precision.

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

$$micro\_recall = \frac{\sum_{j=1}^{Q} tp_j}{\sum_{j=1}^{Q} tp_j + \sum_{j=1}^{Q} fn_j} \tag{11}$$

where tp_j and fn_j are defined as for macro recall.

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

$$micro\_F_1 = \frac{2 \times micro\_precision \times micro\_recall}{micro\_precision + micro\_recall} \tag{12}$$

A.3 Ranking Based Measures

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

$$one\_error(f) = \frac{1}{N}\sum_{i=1}^{N} \left[\left[\arg\max_{\lambda \in \mathcal{Y}} f(\mathbf{x_i}, \lambda) \right] \notin \mathcal{Y}_i \right] \tag{13}$$

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

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

$$coverage(f) = \frac{1}{N}\sum_{i=1}^{N} \max_{\lambda \in \mathcal{Y}_i} rank_f(\mathbf{x_i}, \lambda) - 1 \tag{14}$$

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

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

$$ranking\ loss(f) = \frac{1}{N}\sum_{i=1}^{N} \frac{|D_i|}{|\mathcal{Y}_i|\,|\bar{\mathcal{Y}}_i|} \tag{15}$$

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

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

$$avg\_precision(f) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{|\mathcal{Y}_i|} \sum_{\lambda \in \mathcal{Y}_i} \frac{|\mathcal{L}_i|}{rank_f(\mathbf{x_i}, \lambda)} \tag{16}$$

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

B Complete Results from the Experimental Evaluation

In this section, we present the results from the experimental evaluation. Table 3 shows the predictive performance of the compared methods. First column of the tables describes the methods used for defining the hierarchies, while the other columns show the predictive performance of the compared methods and hierarchies in terms of the 16 performance evaluation measures.

References

1. Madjarov, G., Kocev, D., Gjorgjevikj, D., Dzeroski, S.: An extensive experimental comparison of methods for multi-label learning. Pattern Recogn. **45**(9), 3084–3104 (2012)
2. Tsoumakas, G., Katakis, I., Vlahavas, I.: Effective and efficient multilabel classification in domains with large number of labels. In: Proceedings of the ECML/PKDD Workshop on Mining Multidimensional Data, pp. 30–44 (2008)
3. Kocev, D.: Ensembles for predicting structured outputs. Ph.D. thesis, IPS Jožef Stefan, Ljubljana, Slovenia (2011)
4. Tsoumakas, G., Katakis, I.: Multi label classification: an overview. Int. J. Data Warehouse Min. **3**(3), 1–13 (2007)
5. Mencía, E.L., Park, S.H., Fürnkranz, J.: Efficient voting prediction for pairwise multilabel classification. Neurocomputing **73**, 1164–1176 (2010)
6. Blockeel, H., Raedt, L.D., Ramon, J.: Top-down induction of clustering trees. In: Proceedings of the 15th International Conference on Machine Learning, pp. 55–63 (1998)
7. Vens, C., Struyf, J., Schietgat, L., Džeroski, S., Blockeel, H.: Decision trees for hierarchical multi-label classification. Mach. Learn. **73**(2), 185–214 (2008)
8. Manning, C.D., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2009)
9. Kocev, D., Vens, C., Struyf, J., Džeroski, S.: Tree ensembles for predicting structured outputs. Pattern Recogn. **46**(3), 817–833 (2013)
10. de Carvalho, A.C.P.L.F., Freitas, A.A.: A tutorial on multi-label classification techniques. In: Abraham, A., Hassanien, A.-E., Snášel, V. (eds.) Foundations of Comput. Intel. Vol. 5. SCI, vol. 205, pp. 177–195. Springer, Heidelberg (2009)
11. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining multi-label data. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 667–685. Springer, Heidelberg (2010)
12. Silla Jr., C.N., Freitas, A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Dis. **22**, 31–72 (2011)
13. Dimitrovski, I., Kocev, D., Loskovska, S., Džeroski, S.: Fast and scalable image retrieval using predictive clustering trees. In: Fürnkranz, J., Hüllermeier, E., Higuchi, T. (eds.) DS 2013. LNCS, vol. 8140, pp. 33–48. Springer, Heidelberg (2013)
14. Levatić, J., Kocev, D., Džeroski, S.: The use of the label hierarchy in HMC improves performance: a case study in predicting community structure in ecology. In: Proceedings of the Workshop on New Frontiers in Mining Complex Patterns held in Conjunction with ECML/PKDD2013, pp. 189–201 (2013)
15. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.: Multilabel classification of music into emotions. In: Proceedings of the 9th International Conference on Music Information Retrieval, pp. 320–330 (2008)

16. Boutell, M.R., Luo, J., Shen, X., Brown, C.M.: Learning multi-label scene classification. Pattern Recogn. **37**(9), 1757–1771 (2004)
17. Elisseeff, A., Weston, J.: A kernel method for multi-labelled classification. In: Proceedings of the Annual ACM Conference on Research and Development in Information Retrieval, pp. 274–281 (2005)
18. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009, Part II. LNCS, vol. 5782, pp. 254–269. Springer, Heidelberg (2009)
19. Klimt, B., Yang, Y.: The enron corpus: a new dataset for email classification research. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) ECML 2004. LNCS (LNAI), vol. 3201, pp. 217–226. Springer, Heidelberg (2004)
20. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D.: Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 97–112. Springer, Heidelberg (2002)
21. Srivastava, A., Zane-Ulman, B.: Discovering recurring anomalies in text reports regarding complex space systems. In: Proceedings of the IEEE Aerospace Conference, pp. 55–63 (2005)
22. Snoek, C.G.M., Worring, M., van Gemert, J.C., Geusebroek, J.M., Smeulders, A.W.M.: The challenge problem for automated detection of 101 semantic concepts in multimedia. In: Proceedings of the 14th Annual ACM International Conference on Multimedia, pp. 421–430 (2006)
23. Katakis, I., Tsoumakas, G., Vlahavas, I.: Multilabel text classification for automated tag suggestion. In: Proceedings of the ECML/PKDD Discovery Challenge (2008)
24. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. Ann. Math. Stat. **11**, 86–92 (1940)
25. Nemenyi, P.B.: Distribution-free multiple comparisons. Ph.D. thesis, Princeton University (1963)
26. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
27. Pearson, E.S., Hartley, H.O.: Biometrika Tables for Statisticians, vol. 1. Cambridge University Press, Cambridge (1966)

http://www.springer.com/978-3-319-17875-2

New Frontiers in Mining Complex Patterns
Third International Workshop, NFMCP 2014, Held in
Conjunction with ECML-PKDD 2014, Nancy, France,
September 19, 2014, Revised Selected Papers
Appice, A.; Ceci, M.; Loglisci, C.; Manco, G.; Masciari, E.;
Rás, Z.W. (Eds.)

2015, XII, 211 p. 61 illus., Softcover
ISBN: 978-3-319-17875-2