Structuring the Output Space in Multi-label Classification by Using Feature Ranking

Stevanche Nikoloski[2,3(✉)], Dragi Kocev[1,2], and Sašo Džeroski[1,2]

[1] Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
{dragi.kocev,saso.dzeroski}@ijs.si
[2] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
stevanche.nikoloski@ijs.si
[3] Teagasc, Environment Soils and Land-Use Department, County Wexford, Ireland

Abstract. Motivated by the increasing interest for the task of multi-label classification (MLC) in recent years, in this study we investigate a new approach for decomposition of the output space with the goal to improve the predictive performance. Namely, the structuring of the output/label space is performed by constructing a label hierarchy and then approaching the MLC task as a task of hierarchical multi-label classification (HMLC). Our approach is as follows. We first perform feature ranking for each of the labels separately and then represent each of the labels with its corresponding feature ranking. The construction of the hierarchy is performed by the (hierarchical) clustering of the feature rankings. To this end, we employ four clustering methods: agglomerative clustering with single linkage, agglomerative clustering with complete linkage, balanced k-means and predictive clustering trees. We then use predictive clustering trees to estimate the influence of the constructed hierarchies, i.e., we compare the predictive performance of models without exploiting the hierarchy and models using hierarchies constructed using label co-occurrences or per label feature rankings. Moreover, we investigate the influence of the hierarchy in the context of single models and ensembles of models. We evaluate the proposed approach across 8 datasets. The results show that the proposed method can yield predictive performance boost across several evaluation measures.

Keywords: Multi-label classification · Hierarchy construction
Feature ranking · Structuring of the label space

1 Introduction

Nowadays, the number of new applications of multi-label learning is steadily increasing, hence, the researchers are very interested to develop novel methods and new ideas related to multi-label classification and structuring of the

A. Appice et al. (Eds.): NFMCP 2017, LNAI 10785, pp. 151–166, 2018.
https://doi.org/10.1007/978-3-319-78680-3_11

label/output space. Multi-label classification (MLC) is the task of learning from data examples where each example can be associated with multiple labels. MLC deals with a label dependencies and relations which is orthogonal with existing traditional methods which take into account label independence and learn independent functions from mapping from input space to the output (label) space. The different application problems include video and image annotations (new movie clips, genres), predicting genes and proteins (functional genomics), classification of a tweets and music into emotions, text classification (web-pages, bookmarks, e-mails,...) and others.

The MLC task is typically approached either by decomposing the MLC problem into multiple single class classification problems (i.e., problem transformation methods) or by modifying the algorithms to consider the multiple classes jointly (i.e., algorithm adaptation methods) [12]. In an extensive experimental study Madjarov et al. [7] show that the landscape of MLC methods is not simple: on some datasets problem transformation methods achieve top performance while on other datasets the algorithm adaptation methods are top performing. Furthermore, the study recommends the use of two algorithms for benchmarking: RF-PCT (Random forests of predictive clustering trees, an algorithm adaptation method) [5] and HOMER (Hierarchy Of Multi-label learnERs, a problem transformation method) [13]. The latter divides the label space into subspaces and then constructs classifiers for each of the subspace (e.g., label power set classifiers). This hints that the best performance might be obtained in between the spectrum of the algorithm adaptation and problem transformation methods. In other words, state-of-the-art MLC performance might be obtained by transforming the original MLC problem into several MLC problems and then learn predictive models (preferably using algorithm adaptation methods).

A crucial step in developing methods for output decomposition for MLC is the creation of the subspaces. More specifically, the goal is to find a dependency structure and consider jointly the labels that are inter-dependent. The construction of the output structure of the labels can be very tedious and expensive process, especially if domain experts are needed to complete the task. Moreover, selection of the representation language of the dependencies can be complicated task on its own. Typically, these dependencies are represented as hierarchies of labels [6]. The hierarchies can then be constructed in a data-driven manner using the descriptive space and/or the label space. This presents automatic and relatively efficient process to obtain the representation of the potential dependencies in the label space.

Madjarov et al. [6] present an extensive study of different data-driven methods for constructing label hierarchies for multi-label classification by using the label co-occurence matrix. More precisely, the hierarchies are constructed using four clustering algorithms, agglomerative clustering with single and complete linkage, balanced k-means and predictive clustering trees applied on the label co-occurrences (see Fig. 1, left table).

Next, Szymansky et al. [11] address the question whether data-driven approach using label co-occurrence graph is significantly better than a random choice

in label space division for multi-label classification as performed by RAkELd. Their results show that in almost all cases data-driven partitioning outperforms the baseline RAkELd in all evaluation measures, but Hamming loss.

In this study, we build upon the idea of decomposition of the output space and we present a different approach for data-driven structuring of label space in multi-label classification. Our approach constructs the label hierarchy by clustering the per label feature rankings. Namely, instead of using the original label space consisting of label co-occurrences (see Fig. 1, left table), we calculate a feature importance/ranking scores of the features for each label by using the GENIE3 method for feature importance calculation coupled with the random forest ensemble learning method [1,3] (see Fig. 1, right table).

The obtained structure is then used as the label hierarchy and the MLC task is addressed as hierarchical multi-label classification (HMLC) [5,15]. We thus evaluate whether considering the dependency in the label space can provide better predictive performance than addressing MLC as a flat problem. In other words, we investigate whether considering the MLC task as a hierarchical MLC task can yield better predictive performance. Our approach is illustrated through the example in Fig. 1. The table on the left hand-side shows the construction of the label hierarchy using the label co-occurrence (as performed in [6,11]), while the table on the right hand-side shows our proposed method for constructing the label hierarchy.

Input space					Output space of label co-occurrences					
	BH_LowPeakAmp	BH_LowPeakBPM	BH_HighPeakAmp	…	λ1	λ2	λ3	λ4	λ5	λ6
#1	0.036299	-58.962537	4.698083	…	1	0	0	0	0	0
#2	0.161218	-77.425609	3.09809	…	0	0	1	0	1	1
#3	0.115987	-61.893693	4.478436	…	1	1	1	1	0	0
#4	0.086016	-83.295694	3.786274	…	1	0	1	0	1	1
#5	0.063232	-76.108186	5.911183	…	0	1	0	1	1	1
#6	0.026461	-74.429498	3.046795	…	0	0	1	0	1	1
…	…	…	…	…	…	…	…	…	…	…

Structured label/output space						
	FRank λ1	FRank λ2	FRank λ3	FRank λ4	FRank λ5	FRank λ6
BH_LowPeakAmp	1.369	12.63	22.68	14.06	5.563	1.328
BH_LowPeakBPM	1.588	11.89	26.35	9.177	5.566	0.674
BH_HighPeakAmp	1.433	11.08	44	8.951	19.03	1.479
BH_HighPeakBPM	1.741	7.836	8.206	10.06	8.61	0.561
BH_HighLowRatio	2.169	7.267	6.914	9.166	12.16	0.017
BHSUM1	2.246	5.541	5.494	11.19	14.31	1.058
…	…	…	…	…	…	…

Fig. 1. Excerpt from the original *emotions* dataset showing the output space consists of label co-occurrences (*left* table) and the space consists of ranks of the features for each of the labels, separately (*right* table). The former is obtained with structuring the original label set using feature ranking.

We perform an experimental evaluation using 8 benchmark datasets from different domains: text, image, music and video classification, and gene function prediction. The predictive performance of the methods is assessed using 13 different evaluation measures used in the context of MLC (6 threshold dependent and 7 threshold independent).

The obtained results indicate that using the methods for creating the hierarchies using feature ranking can yield a better predictive performance as compared to the original flat MLC methods without the hierarchy. Moreover, using the hierarchy constructed by structuring of the output space using the feature rankings of the labels gives better predictive performance compared to using the hierarchy obtained using the label co-occurrences.

The reminder of this paper is organized as follows. Section 2 presents the background work, i.e., discussion on the tasks of multi-label classification and hierarchical multi-label classification methods. Then, in Sect. 3, we present the structuring of the output space using feature ranking. In Sect. 4, we show the experimental design. The results obtained from the experiments are presented and discussed in Sect. 5. Finally, Sect. 6 concludes this paper.

2 Background

In this section, we first define the task of multi-label classification and then the task of hierarchical multi-label classification. Multi-label learning considers learning from examples which are associated to more than one label coming from a predefined set of labels containing all possible labels. There are two types of multi-label learning tasks: multi-label classification and multi-label ranking. The main goal of multi-label classification is to create a predictive model that will output a set of relevant labels for a given, previously unseen example. Multi-label ranking, on the other hand, can be understood as learning a model that, for each unseen examples, associates a list of rankings (preferences) on the labels from a given set of possible labels and a bipartite partition of this set into relevant and irrelevant labels. An extensive bibliography of methods for multi-label learning can be found in [7,14] and the references therein.

The task of multi-label learning can be defined as follows [5]. The input space $\mathcal{X}$ consists of vectors of values of nominal or numeric data types i.e., $\forall x_i \in \mathcal{X}, x_i = (x_{i1}, x_{i2}, \ldots x_{iD})$, where D is a number of descriptive attributes. The output space $\mathcal{Y}$ consists of a subset of a finite set of disjoint labels $\mathcal{L} = \{\lambda_1, \lambda_2, \ldots, \lambda_Q\}$ (Q > 1 and $\mathcal{Y} \subseteq \mathcal{L}$). Given this, each example is a pair of a vector and a set from the input and output space, respectively. All of the examples then form the set of examples (i.e., the dataset) E. The goal is then to find a function $h : \mathcal{X} \rightarrow 2^{\mathcal{L}}$ such that from the input space assigns a set of labels to each example.

The main difference between multi-label classification and hierarchical multi-label classification (HMLC) is that in the latter the labels from the label space are organized into a hierarchy. A given example labeled with a given label it is also labeled with all its parent labels (known as the hierarchy constraint). Furthermore, an example can be labeled with multiple labels, simultaneously. That means a several paths can be followed from the root node in order to arrive at a given label.

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 parent-child relationship structured as a tree ($\forall \lambda_1, \lambda_2 \in \mathcal{L} : \lambda_1 \leq_h \lambda_2$ if and only if λ_1 is a parent of λ_2) [5]. 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., $E = \{(x_i, \mathcal{Y}_i) | x_i \in \mathcal{X}, \mathcal{Y} \subseteq \mathcal{L}, \lambda \in \mathcal{Y}_i \Rightarrow \forall \lambda' \leq_h \lambda : \lambda' \in \mathcal{Y}_i, 1 \leq i \leq N\}$, where N is a number of examples in E. Same conditions as in multi-label classification should be satisfied for the quality criterion q (high predictive performance and

low computational cost). In [9], an extensive bibliography is given, where the HMLC task is presented across different application domains.

3 Structuring of Label Spaces Using Feature Ranking

In this section, we explain our method for structuring the label space using feature ranking and we describe the different clustering algorithms used in this work. Our proposed method for label space structuring is outlined in procedure *StructuringLabelSpaceFR* in Table 1. First, we take the original training dataset D^{train} and using random forest method with GENIE3 feature importance, we create feature rankings for each label separately. We then construct a dataset D^{ranks} consisting of the feature rankings. Next, we obtain a hierarchy using one of the clustering algorithms described bellow. The hierarchy is then used to preprocess the datasets and obtain their hierarchical variants D_H^{train} and D_H^{test}. At the end, we learn the HMLC predictive models.

Table 1. The algorithm for structuring the label space using feature rankings per label.

procedure StructuringLabelSpaceFR(D^{train}, D^{test}) **returns** performance
1: *// create feature importance (.fimp) file with Random forest (GENIE3)*
2: FimpPath = CreateFimp(D^{train});
3: *// Create new arff with feature ranks from fimp file*
4: D^{ranks} = CreateArffFromFimp(FimpPath);
5: **hierarchy** = Clustering(D^{ranks});
6: *//transform multi-label dataset to hierarchical multi-label one*
7: D_H^{train} = MLC2HMC(D^{train}, **hierarchy**);
8: D_H^{test} = MLC2HMC(D^{test}, **hierarchy**);
9: *//solve transformed hierarchical multi-label problem by using approach for HMC*
10: HMCModel = HMCMethod(D_H^{train});
11: *//generate HMC predictions using CLUS platform*
12: **predictions** = HMCModel(D_H^{test});
13: *//Extract predictions only for the leaves from the HMC predictions*
14: P = ExtractLeavesPredictionsFromHMCPredictions(**predictions**);
15: **return** Evaluate(P)

In our approach, described in a procedure *StructuringLabelSpaceFR* (Table 1), we can see that additional step, compare to the algorithm given by Madjarov et al. [6], is the function *CreateFimp* at line 4, which increases the theoretical complexity of the procedure. According to the dimensionality of the space which is going to be clustered using the function *Clustering* at line 5, one dimension in the space consists of label co-occurrences is the number of examples (instances) which means that in case of more complex datasets with large number of examples, the clustering procedure will take more of the time in order to create a hierarchy. From the other side, the procedure of creating the hierarchy

using feature rankings has a dimension which depends of the feature space cardinality. Typically, the feature space cardinality is much smaller than the number of examples. It means that clustering of the rankings will finish faster than clustering of the label co-occurrences for datasets with large number of examples but small number of features, which is a case in most of the benchmarks datasets available. Consequently, although we have additional function in our procedure of structuring of the output space, for more complex datasets with high number of examples and smaller number of features, the clustering procedure, i.e., the hierarchy creation will be completed in a reasonable time, thus compensating for obtaining the feature rankings.

We next describe the procedures for obtaining the feature rankings. Random forests as ensemble method for predictive modeling are originally proposed by Breiman [1]. The empirical analysis of their use as feature ranking methods has been studied by Verikas et al. [16]. The random forests are constructed by first performing bootstrap sampling on the data and then building a decision tree for each bootstrap sample. The decision trees are constructed by taking the best split at each level, from a randomly selected feature subset.

Huynh-Thu et al. [3] propose to use the reduction of the variance in the output space at each test node in the tree (the resulting algorithm is named GENIE3). Namely, the variables that reduce the variance of the output more are, consequently, more important than the ones that reduce the variance less. Hence, for each descriptive variable we measure the reduction of variance it produces when selected as splitting variable. If a variable is never selected as splitting variable then its importance will be 0.

The GENIE3 algorithm has been heavily evaluated for single-target regression tasks (e.g., for gene regulatory network reconstruction). The basic idea adopted for future ranking is the same of that proposed in GENIE3, but we use random forest of predictive clustering trees (PCTs) for building the ensemble. The result is a feature ranking algorithm that works for different types of structure output prediction tasks (including MLC and HMLC).

Furthermore, we discuss the different clustering methods used to obtain the hierarchies of the labels. For achieving a good performance of the HMLC methods, it is critical to generate label hierarchies that more closely capture the relations among the labels. The only constraint when building the hierarchy is that we should take care about the leaves of the label hierarchies. They need to define the original MLC task. In particular, the labels from the original MLC problem represent the leaves of the label hierarchy, while the labels in internal nodes of the tree are so-called meta-labels. Meta-labels model the potential relations among the original labels.

For obtaining the hierarchies, we use four different clustering methods (two agglomerative and two divisive):

- agglomerative clustering with single linkage;
- agglomerative clustering with complete linkage;
- balanced k-means clustering (*divisive*) and
- predictive clustering trees (*divisive*).

Agglomerative clustering algorithms consider each example as separate cluster at the beginning and then iteratively merge pairs of clusters based on their distance metric (linkage). If we use the maximal distance of two examples from the clusters C_1 and C_2, then this type of agglomerative clustering is using *complete* linkage, i.e., $\max\{dist(c_1, c_2) : c_1 \in C_1, c_2 \in C_2\}$. If we use the minimal distance between two clusters, then the agglomerative clustering approach is with *single* linkage i.e., $\min\{dist(c_1, c_2) : c_1 \in C_1, c_2 \in C_2\}$.

Balanced k-means is top-down approach for clustering. First, all labels from the label space $\mathcal{L}$ are in one common cluster at the top node of the hierarchy. Then, the procedure consecutively divides (splits) this cluster into k disjoint sub-clusters ($k < |\mathcal{L}_n|$) using k-means clustering. The division also is concerned with the number of examples in each cluster: the algorithm outputs clusters with approximately equal size [13]. The procedure recursively is repeated on each sub-cluster (meta-label) until we have n different clusters consisting of one label from the label space $\mathcal{L}$. In other words, our label space $\mathcal{L}$ is covered by leaves of the hierarchy obtained by the balanced k-means clustering approach.

We also use predictive clustering trees to construct the label hierarchies. More specifically, the setting from the predictive clustering framework used in this work is based on treating the target space as descriptive space, i.e., the target space is also a descriptive space. Descriptive/target variables are used to provide descriptions for the obtained clusters. Here, the focus is using predictive clustering framework on the task of clustering instead of predictive modelling [2,4]. The obtained hierarchies using agglomerative clustering (single and complete linkage) and using predictive clustering trees for *emotions* dataset are shown in Fig. 2.

We next present the predictive clustering trees (PCTs) - the modelling framework we used throughout this work. PCTs are a generalization of decision trees towards the tasks of predicting structured outputs, including both MLC and HMLC. In order to apply PCTs to the task of HMLC, Vens et al. [15] define the variance and the prototype as follows. First, the set of labels for each example is represented as a vector of binary components. If the example belongs to the class c_i then the i'th component of the vector is 1 and 0, otherwise. The variance of a set of examples E is thus defined as follows:

$$Var(E) = \frac{1}{|E|} \cdot \sum_{i=1}^{|E|} dist(\Gamma_i, \overline{\Gamma})^2 \tag{1}$$

where $\overline{\Gamma} = \frac{1}{|E|} \cdot \sum_{i=1}^{|E|} \Gamma_i$.

In other words, the variance $Var(E)$ in (1) represents the average squared distance between each example's class vector (Γ_i) and the mean class vector of the set ($\overline{\Gamma}$). When we talk about HMC, then the similarity at higher levels of the hierarchy are more important than the similarity at lower levels. This is reflected with the distance term used in (1), which is weighted Euclidean distance:

$$dist(\Gamma_1, \Gamma_2) = \sqrt{\sum_{s=1}^{|\Gamma|} \theta(c_s) \cdot (\Gamma_{1,s} - \Gamma_{2,s})^2}$$

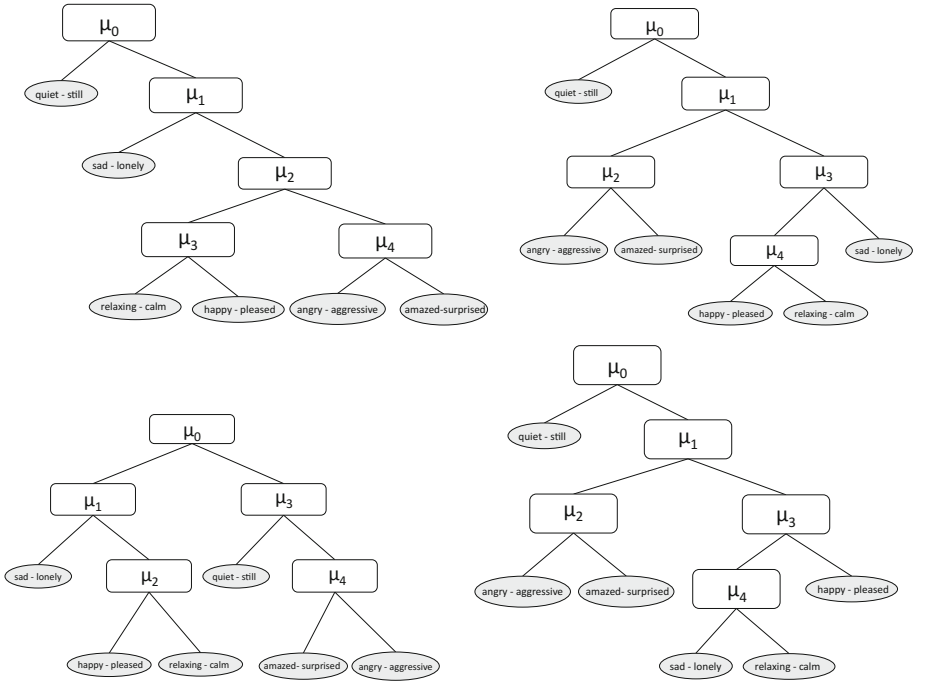

Fig. 2. Hierarchies obtained using agglomerative single (top-left), agglomerative complete (top-right), balanced K-means clustering (bottom - left) and PCTs (bottom - right) clustering methods for *emotions* dataset.

where $\Gamma_{i,s}$ is the s'th component of the class vector Γ_i of the instance E_i, $|\Gamma|$ is the size of the class vector, and the class weights $\theta(c) = \theta_0 \cdot \text{avg}_j\{\theta(p_j(c))\}$, where $p_j(c)$ is j'th parent of the class c and $0 < \theta_0 < 1$. The class weights $\theta(c)$ decrease with the depth of the class in the hierarchy thus making the differences in the lower parts of the hierarchy less influential to the overall score.

Random forests of PCTs for HMLC are considered in the same way as the random forest of PCTs for MLC. In the case of HMLC, the ensemble is a set of PCTs for HMLC. A new example is classified by taking a majority vote from the combined predictions of the member classifiers. The prediction of the random forest ensemble of PCTs for HMLC follows the hierarchy constraint (if the example is labeled with a given label then is automatically labeled with all its ancestor-labels).

4 Experimental Design

The aim of our study is to address the following questions:

(i) Whether feature ranking on the label (output) space in the MLC task can be used to construct good label hierarchies?

(ii) Which clustering method yields better hierarchy?
(iii) How this scales from single model to ensemble of models?
(iv) Can we achieve better predictive models with using a hierarchies obtained by structuring the feature ranking or co-occurrences space?

In order to answer the above questions, we use eight multi-label classification benchmark problems from different domains. We have 3 datasets from text classification, 4 datasets from multimedia, includes movie clips and genres classification and 1 dataset from biology. All datasets are predefined by other researchers (typically the data owners) and divided into train and test subsets. The basic information and statistics about these datasets are given in Table 2.

Table 2. Statistics of used benchmark tasks in terms of application domain (*domain*), number of training examples (*#tr.e*), testing examples (*#t.e*), number of descriptors (D), total number of labels (L) and number of labels per example.

Dataset	Domain	*#tr.e*	*#t.e*	D	L	l_c
emotions	multimedia	391	202	72	6	1.87
scene	multimedia	1211	1159	294	6	1.07
yeast	biology	1500	917	103	14	4.24
tmc2007	text	21519	7077	500	22	2.16
medical	text	645	333	1449	45	1.25
enron	text	1123	579	1001	53	3.38
mediamill	multimedia	30993	12914	120	101	4.38
corel5k	multimedia	4500	500	499	374	3.52

In our experiments, we use 13 different evaluation measures, as presented in [7,14]. These are divided into two groups: 6 threshold dependent/example based measures (*hamming loss, accuracy, precision, recall*, F_1 *score*) and 7 threshold independent measures out of which three ranking-based (*one-error, coverage* and *ranking-loss*) and four areas under ROC and PRC curves (*AUROC, AUPRC, wAUPRC* and *pooledAUPRC*). The threshold independent measures are typically used in HMLC and they do not require a (pre)selection of thresholds and calculating a prediction [15]. All of the above measures offer different viewpoints on the results from the experimental evaluation.

Hamming loss is an example-based evaluation measure that evaluate how many times a pair of example and its label are misclassified. *One-error* is a ranking-based evaluation measure that evaluates how many times the top-ranked label does not exist in a set of relevant labels of the example. *Coverage* evaluates how far, on average, we need to go down the list of label ranks in order to cover all relevant labels of given example. *Ranking loss* evaluates the average fraction of the label pairs that are reversely ordered for the given example. Precision and recall are very important measures defined for binary classification tasks with

classes of positive and negative examples. *Precision* is a proportion of positive prediction that are correct, and *recall* is the proportion of positive examples that correctly predicted as positive. F_1 *score* is the harmonic mean between precision and recall. *Accuracy* for each instance is defined as the proportion of correctly predicted labels over total number of labels for that instance. Overall accuracy is the average across all instances. A precision-recall curve (PR curve) is a curve that represent the precision as a function of its recall. *AUPRC* (area under the PR curve) is the area between the PR curve and the recall axis. *wAUPRC* evaluates the weighted average of the areas under the individual (per class) PR-curves. If choosing some threshold, we transform the multi-label problem into binary problems with considering binary classifier as a couple (instance, class) and predicting whether that instance belongs to that class, we can obtain PR curves that differ depend of the varying the threshold. The area under the average PR curve (from all different threshold curves) is called *pooledAUPRC*. From the other side, if we consider the space of true positive rates (sensitivity) versus false positive rates (fall-out) then the curve considers the sensitivity as a function of the fall-out is called ROC-curve. The are under this ROC-curve is the evaluation measure called *AUROC*.

The majority of our experiments are performed using the CLUS software package (https://sourceforge.net/projects/clus/), which implements the predictive clustering framework, including PCTs, random forests of PCTs and feature ranking [5,10]. A hierarchical tree defined by the used clustering methods in HMLC setting are defined as tree shaped hierarchies. We use the same values for k in balanced k-means clustering algorithm, as suggested in [7].

For obtaining a hierarchy using the agglomerative clustering method we use the R software package (function *agnes()* from the *cluster* package. For more info, see https://stat.ethz.ch/R-manual/R-devel/library/cluster/html/agnes.html). We use the MATLAB software package to create hierarchies with balanced k-means clustering which is based on Hungarian (Munkres') assignment algorithm to assign the examples to the clusters [8]. We use Euclidean distance metric in all our algorithms that require distance. Moreover, for random forest for feature ranking we use GENIE3 as a feature importance method based on variable selection with ensembles of PCTs [3].

In order to make a comparative analysis with the results obtained by the study by Madjarov et al. [6], we repeated their experiments on the same experimental setting with the experiments we perform for feature ranking.

5 Results

In this section, we present the obtained results from the experiments we performed using our novel proposed method for structuring the output space. In our study, as an output space, we consider the space consisting of label co-occurrences (as presented by Madjarov et al. [6]) and the space consisting of feature ranks for each label, respectively. We compare the following methods for hierarchy construction:

- flat MLC problem without considering a hierarchy in the label space (*FlatMLC*);
- agglomerative clustering with single linkage (*AggSingle*);
- agglomerative clustering with complete linkage (*AggComplete*)
- balanced k-means clustering (*BKmeans*)
- clustering using predictive clustering trees (*ClusPCTs*).

Since we have two different models (single PCTs model and random forest of PCTs) and two different structured output spaces, we show separately the results for single PCTs (Fig. 3) and random forest of PCTs (Fig. 4). In order to distinguish between using either single tree or random forest of PCTs and different methods of structuring the output space (label co-occurrences and feature rankings), we use prefixes (*PCT-* and *RF-*) and suffixes (*-CO* and *-FR*) before and after the hierarchy construction method name, respectively. For example, *RF-AggComplete-CO* refers to the agglomerative clustering method with complete linkage of the output space of label co-occurrences using random forest of PCTs for model creation. Then, *PCT-ClusPCTs-FR* refers to the clustering method with PCTs of the output space consists of feature rankings per label using single PCTs for model creation, etc.

Observing the results obtained using single PCTs (Fig. 3), we can note that there is no clear winner across all evaluation measures and datasets. In the case of threshold independent measures, such as *AUPRC, AUROC, wAUPRC* and *pooledAUPRC*, we can see that hierarchies created using clustering of the output space consisting of feature rankings perform the best for enron, emotions, mediamill and yeast datasets. Considering the scene and corel5k datasets, we can observe that they perform the best according to *AUROC, AUPRC* and *pooledAUPRC*, but not for *wAUPRC*. *PCT-BKmeans-FR* outperforms the other algorithms for hierarchy creation in the emotions dataset according to the most of the evaluation measures but not according to one-error. Moreover, the hierarchies created clustering the feature rankings outperform the other algorithms considering the ML performance measures (*ML F1 measure, ML accuracy, ML precision* and *ML recall*) in 5 out of the 8 datasets.

Generally, structuring the output space consisting of feature rankings for each label yields better predictive performance compared to the structuring the output space consisting of label co-occurrences considering most of the evaluation measures in almost all datasets. For the corel5k dataset only, we can see that both have similar performance. If we consider medical and tmc2007 datasets, we can see that structuring the output space does not improve the performance as compared to the flat MLC task, where there is no hierarchy considered. All in all, we can conclude that using the hierarchies, the predictive performance can be improved.

The results obtained when random forests are used as predictive models are given in Fig. 4. These results present a different situation as compared to the results obtained when single PCTs are used as predictive models. First of all, the predictive performance is improved as compared to the single PCTs for large majority of the cases. Most notably, the performance for the threshold

PCTs	Hamming loss	Average precision	Coverage	ML Accuracy	ML F1 measure	ML Precision	ML Recall	One Error	Ranking Loss	AUROC	AUPRC	wAUPRC	pooled AUPRC
ENRON													
PCT-FlatMLC	0.071	0.538	40.513	0.360	0.467	0.489	0.502	0.444	0.151	0.130	0.585	0.353	0.416
PCT-AggSingle-FR	0.071	0.595	39.630	0.380	0.485	0.503	0.527	0.383	0.104	0.142	0.598	0.367	0.428
PCT-AggComplete-FR	0.072	0.565	39.703	0.371	0.478	0.486	0.530	0.382	0.192	0.148	0.601	0.370	0.433
PCT-BKmeans-FR	0.072	0.466	39.665	0.374	0.480	0.489	0.527	0.501	0.341	0.142	0.593	0.358	0.419
PCT-ClusterPCTs-FR	0.072	0.554	39.472	0.354	0.459	0.471	0.499	0.382	0.194	0.142	0.590	0.354	0.418
PCT-AggSingle-CO	0.067	0.482	36.858	0.374	0.475	0.488	0.520	0.458	0.356	0.137	0.591	0.362	0.421
PCT-AggComplete-CO	0.066	0.471	37.104	0.364	0.463	0.485	0.504	0.453	0.350	0.128	0.580	0.359	0.419
PCT-BKmeans-CO	0.068	0.541	37.879	0.364	0.472	0.493	0.522	0.323	0.222	0.131	0.586	0.357	0.413
PCT-ClusterPCTs-CO	0.072	0.588	40.473	0.374	0.476	0.487	0.517	0.396	0.108	0.142	0.594	0.366	0.424
EMOTIONS													
PCT-FlatMLC	0.292	0.669	4.431	0.460	0.541	0.550	0.582	0.450	0.335	0.516	0.680	0.509	0.524
PCT-AggSingle-FR	0.304	0.666	4.569	0.421	0.502	0.514	0.549	0.490	0.317	0.487	0.661	0.480	0.503
PCT-AggComplete-FR	0.296	0.672	4.574	0.442	0.528	0.551	0.559	0.470	0.314	0.507	0.679	0.508	0.517
PCT-BKmeans-FR	0.266	0.717	4.173	0.507	0.589	0.597	0.634	0.401	0.265	0.552	0.714	0.558	0.563
PCT-ClusterPCTs-FR	0.292	0.702	4.569	0.438	0.529	0.554	0.564	0.386	0.291	0.505	0.670	0.509	0.517
PCT-AggSingle-CO	0.307	0.670	4.460	0.439	0.525	0.528	0.576	0.450	0.345	0.496	0.664	0.495	0.510
PCT-AggComplete-CO	0.307	0.670	4.460	0.439	0.525	0.528	0.576	0.450	0.345	0.496	0.664	0.495	0.510
PCT-BKmeans-CO	0.312	0.640	4.698	0.414	0.507	0.541	0.535	0.485	0.357	0.496	0.653	0.491	0.505
PCT-ClusterPCTs-CO	0.297	0.681	4.639	0.440	0.516	0.535	0.535	0.446	0.323	0.489	0.664	0.491	0.502
MEDICAL													
PCT-FlatMLC	0.014	0.795	11.447	0.724	0.766	0.759	0.809	0.204	0.104	0.321	0.686	0.672	0.702
PCT-AggSingle-FR	0.015	0.794	12.874	0.706	0.741	0.742	0.771	0.216	0.082	0.320	0.685	0.646	0.682
PCT-AggComplete-FR	0.015	0.785	12.207	0.721	0.759	0.758	0.791	0.222	0.115	0.325	0.690	0.665	0.692
PCT-BKmeans-FR	0.015	0.771	12.616	0.710	0.750	0.751	0.786	0.219	0.125	0.320	0.689	0.648	0.687
PCT-ClusterPCTs-FR	0.015	0.787	11.832	0.727	0.767	0.771	0.803	0.231	0.087	0.314	0.696	0.670	0.699
PCT-AggSingle-CO	0.016	0.761	12.258	0.694	0.733	0.726	0.777	0.264	0.133	0.315	0.684	0.645	0.687
PCT-AggComplete-CO	0.016	0.763	12.640	0.694	0.734	0.733	0.773	0.240	0.141	0.294	0.662	0.638	0.676
PCT-BKmeans-CO	0.015	0.795	12.003	0.716	0.757	0.753	0.797	0.198	0.078	0.340	0.695	0.652	0.691
PCT-ClusterPCTs-CO	0.016	0.795	12.003	0.707	0.747	0.751	0.783	0.228	0.063	0.298	0.678	0.658	0.686
MEDIAMILL													
PCT-FlatMLC	0.052	0.472	77.282	0.356	0.476	0.491	0.551	0.445	0.247	0.089	0.571	0.339	0.440
PCT-AggSingle-FR	0.052	0.584	76.868	0.353	0.474	0.495	0.549	0.318	0.105	0.087	0.570	0.350	0.439
PCT-AggComplete-FR	0.052	0.610	76.795	0.358	0.478	0.498	0.553	0.313	0.083	0.089	0.570	0.353	0.443
PCT-BKmeans-FR	0.053	0.509	76.514	0.357	0.477	0.493	0.554	0.394	0.118	0.093	0.575	0.347	0.441
PCT-ClusterPCTs-FR	0.052	0.604	76.004	0.360	0.479	0.499	0.552	0.351	0.071	0.088	0.574	0.352	0.443
PCT-AggSingle-CO	0.053	?	73.362	0.341	0.452	0.478	0.516	0.440	0.291	0.087	0.562	0.345	0.429
PCT-AggComplete-CO	0.055	?	72.275	0.339	0.450	0.474	0.513	0.516	0.321	0.081	0.564	0.337	0.428
PCT-BKmeans-CO	0.054	?	70.465	0.349	0.463	0.479	0.537	0.471	0.273	0.090	0.571	0.339	0.434
PCT-ClusterPCTs-CO	0.051	?	78.356	0.343	0.455	0.480	0.516	0.267	0.156	0.088	0.569	0.339	0.428
SCENE													
PCT-FlatMLC	0.263	0.636	4.537	0.271	0.288	0.289	0.302	0.686	0.183	0.193	0.530	0.255	0.907
PCT-AggSingle-FR	0.251	0.491	4.215	0.311	0.333	0.332	0.360	0.669	0.475	0.183	0.479	0.282	0.903
PCT-AggComplete-FR	0.247	0.658	4.595	0.304	0.333	0.351	0.347	0.628	0.166	0.191	0.494	0.265	0.907
PCT-BKmeans-FR	0.237	0.688	4.157	0.342	0.371	0.372	0.397	0.587	0.151	0.196	0.546	0.291	0.906
PCT-ClusterPCTs-FR	0.247	0.470	4.595	0.304	0.333	0.351	0.347	0.661	0.525	0.191	0.494	0.265	0.907
PCT-AggSingle-CO	0.234	0.557	4.256	0.349	0.376	0.373	0.405	0.579	0.361	0.189	0.516	0.299	0.906
PCT-AggComplete-CO	0.234	0.557	4.256	0.349	0.376	0.373	0.405	0.579	0.361	0.189	0.516	0.299	0.906
PCT-BKmeans-CO	0.229	0.509	4.099	0.355	0.387	0.386	0.421	0.612	0.523	0.186	0.502	0.316	0.904
PCT-ClusterPCTs-CO	0.260	0.658	4.438	0.280	0.309	0.303	0.343	0.636	0.164	0.186	0.517	0.260	0.904
TMC2007													
PCT-FlatMLC	0.028	0.957	2.600	0.807	0.866	0.843	0.942	0.044	0.007	0.907	0.994	0.962	0.955
PCT-AggSingle-FR	0.030	0.948	2.712	0.797	0.859	0.835	0.936	0.052	0.009	0.905	0.993	0.955	0.950
PCT-AggComplete-FR	0.030	0.949	2.705	0.802	0.862	0.836	0.940	0.052	0.009	0.903	0.993	0.955	0.950
PCT-BKmeans-FR	0.029	0.950	2.648	0.807	0.867	0.842	0.943	0.053	0.008	0.925	0.993	0.959	0.955
PCT-ClusterPCTs-FR	0.030	0.950	2.684	0.801	0.862	0.837	0.940	0.048	0.009	0.903	0.993	0.956	0.949
PCT-AggSingle-CO	0.031	0.943	2.739	0.794	0.855	0.837	0.928	0.057	0.010	0.861	0.992	0.953	0.939
PCT-AggComplete-CO	0.030	0.946	2.711	0.797	0.859	0.835	0.937	0.056	0.009	0.870	0.992	0.955	0.942
PCT-BKmeans-CO	0.029	0.954	2.640	0.807	0.866	0.840	0.943	0.049	0.008	0.903	0.993	0.960	0.953
PCT-ClusterPCTs-CO	0.030	0.947	2.719	0.800	0.860	0.841	0.932	0.051	0.009	0.884	0.992	0.955	0.945
YEAST													
PCT-FlatMLC	0.295	0.630	11.124	0.406	0.514	0.516	0.572	0.430	0.299	0.354	0.558	0.483	0.510
PCT-AggSingle-FR	0.290	0.590	11.122	0.429	0.541	0.545	0.600	0.510	0.367	0.365	0.574	0.500	0.528
PCT-AggComplete-FR	0.289	0.608	11.109	0.417	0.526	0.529	0.584	0.507	0.320	0.368	0.578	0.504	0.527
PCT-BKmeans-FR	0.291	0.645	11.372	0.412	0.523	0.533	0.570	0.430	0.261	0.357	0.565	0.488	0.521
PCT-ClusterPCTs-FR	0.292	0.645	11.298	0.415	0.518	0.531	0.561	0.455	0.257	0.358	0.560	0.491	0.525
PCT-AggSingle-CO	0.298	0.648	11.262	0.408	0.516	0.521	0.573	0.353	0.317	0.356	0.556	0.491	0.517
PCT-AggComplete-CO	0.290	0.676	11.144	0.419	0.528	0.530	0.590	0.328	0.263	0.359	0.570	0.502	0.520
PCT-BKmeans-CO	0.286	0.670	11.352	0.412	0.519	0.530	0.566	0.334	0.275	0.363	0.568	0.498	0.523
PCT-ClusterPCTs-CO	0.296	0.668	11.241	0.412	0.520	0.526	0.575	0.400	0.246	0.355	0.558	0.497	0.518
COREL5K													
PCT-FlatMLC	0.015	0.144	352.716	0.091	0.130	0.175	0.125	0.774	0.419	0.027	0.516	0.058	0.114
PCT-AggSingle-FR	0.014	0.187	357.244	0.083	0.124	0.186	0.118	0.752	0.223	0.022	0.514	0.045	0.098
PCT-AggComplete-FR	0.016	0.184	354.216	0.083	0.121	0.142	0.125	0.734	0.409	0.021	0.513	0.055	0.106
PCT-BKmeans-FR	0.016	0.137	360.022	0.092	0.136	0.169	0.142	0.752	0.606	0.031	0.521	0.060	0.115
PCT-ClusterPCTs-FR	0.016	0.217	350.488	0.093	0.134	0.144	0.150	0.716	0.215	0.032	0.523	0.064	0.123
PCT-AggSingle-CO	0.013	0.096	368.088	0.065	0.097	0.169	0.085	0.778	0.712	0.013	0.501	0.037	0.083
PCT-AggComplete-CO	0.013	0.110	367.356	0.073	0.108	0.186	0.095	0.776	0.645	0.020	0.504	0.042	0.092
PCT-BKmeans-CO	0.015	0.181	351.246	0.101	0.147	0.168	0.156	0.700	0.294	0.029	0.518	0.071	0.120
PCT-ClusterPCTs-CO	0.018	0.210	360.764	0.091	0.138	0.145	0.160	0.718	0.149	0.022	0.511	0.051	0.105

Fig. 3. Results with the 13 performance measures for *single PCTs* from experiments performed on 8 different datasets. The best results obtained per measure per dataset are highlighted.

Random Forest	Hamming loss	Average precision	Coverage	ML Accuracy	ML F1 measure	ML Precision	ML Recall	One Error	Ranking Loss	AUROC	AUPRC	wAUPRC	pooled AUPRC
ENRON													
RF-FlatMLC	0.047	0.698	13.187	0.402	0.509	0.714	0.435	0.200	0.078	0.241	0.709	0.620	0.577
RF-AggSingle-FR	0.047	0.696	13.028	0.396	0.500	0.706	0.425	0.206	0.077	0.235	0.724	0.615	0.574
RF-AggComplete-FR	0.047	0.695	13.347	0.396	0.499	0.703	0.425	0.206	0.078	0.239	0.724	0.618	0.575
RF-BKmeans-FR	0.046	0.697	12.865	0.404	0.509	0.708	0.434	0.211	0.076	0.242	0.745	0.622	0.582
RF-ClusterPCTs-FR	0.046	0.696	13.180	0.402	0.506	0.704	0.431	0.199	0.076	0.244	0.737	0.620	0.582
RF-AggSingle-CO	0.042	0.686	11.784	0.405	0.507	0.726	0.424	0.193	0.079	0.213	0.728	0.598	0.553
RF-AggComplete-CO	0.042	0.692	11.717	0.410	0.511	0.719	0.430	0.202	0.079	0.215	0.730	0.603	0.559
RF-BKmeans-CO	0.043	0.688	12.223	0.399	0.503	0.728	0.420	0.200	0.078	0.225	0.719	0.600	0.554
RF-ClusterPCTs-CO	0.047	0.692	13.100	0.400	0.504	0.706	0.429	0.199	0.078	0.236	0.742	0.616	0.572
EMOTIONS													
RF-FlatMLC	0.191	0.813	2.812	0.530	0.605	0.674	0.600	0.267	0.152	0.755	0.851	0.754	0.755
RF-AggSingle-FR	0.201	0.815	2.837	0.500	0.569	0.629	0.567	0.282	0.155	0.749	0.852	0.756	0.753
RF-AggComplete-FR	0.196	0.810	2.817	0.502	0.574	0.643	0.564	0.262	0.151	0.766	0.859	0.762	0.769
RF-BKmeans-FR	0.199	0.810	2.817	0.494	0.563	0.626	0.553	0.277	0.153	0.770	0.863	0.767	0.772
RF-ClusterPCTs-FR	0.205	0.814	2.827	0.487	0.559	0.623	0.550	0.282	0.154	0.754	0.856	0.756	0.754
RF-AggSingle-CO	0.199	0.817	2.812	0.504	0.578	0.645	0.572	0.287	0.150	0.755	0.858	0.758	0.757
RF-AggComplete-CO	0.199	0.817	2.812	0.504	0.578	0.645	0.572	0.287	0.150	0.755	0.858	0.758	0.757
RF-BKmeans-CO	0.193	0.815	2.871	0.510	0.580	0.649	0.569	0.297	0.160	0.759	0.854	0.765	0.762
RF-ClusterPCTs-CO	0.191	0.820	2.787	0.512	0.582	0.648	0.575	0.267	0.148	0.764	0.860	0.766	0.766
MEDICAL													
RF-FlatMLC	0.018	0.858	2.571	0.415	0.431	0.462	0.418	0.396	0.023	0.432	0.824	0.787	0.818
RF-AggSingle-FR	0.019	0.856	2.700	0.356	0.371	0.402	0.359	0.459	0.024	0.439	0.812	0.764	0.803
RF-AggComplete-FR	0.018	0.865	2.589	0.417	0.434	0.470	0.418	0.402	0.022	0.458	0.820	0.790	0.828
RF-BKmeans-FR	0.018	0.865	2.589	0.430	0.447	0.479	0.435	0.393	0.023	0.467	0.823	0.795	0.831
RF-ClusterPCTs-FR	0.019	0.849	2.769	0.388	0.405	0.441	0.391	0.411	0.026	0.422	0.805	0.777	0.818
RF-AggSingle-CO	0.019	0.853	2.841	0.366	0.382	0.416	0.367	0.447	0.027	0.437	0.804	0.771	0.813
RF-AggComplete-CO	0.019	0.852	2.727	0.369	0.386	0.420	0.372	0.438	0.025	0.432	0.817	0.764	0.808
RF-BKmeans-CO	0.018	0.853	2.613	0.421	0.440	0.477	0.424	0.372	0.023	0.455	0.822	0.786	0.821
RF-ClusterPCTs-CO	0.019	0.857	2.586	0.376	0.397	0.438	0.379	0.423	0.023	0.441	0.813	0.778	0.818
MEDIAMILL													
RF-FlatMLC	0.030	0.735	20.676	0.455	0.573	0.798	0.495	0.124	0.047	0.254	0.762	0.671	0.618
RF-AggSingle-FR	0.030	0.733	20.781	0.451	0.570	0.803	0.489	0.124	0.047	0.249	0.765	0.669	0.617
RF-AggComplete-FR	0.030	0.733	20.727	0.451	0.569	0.802	0.487	0.127	0.047	0.254	0.765	0.668	0.616
RF-BKmeans-FR	0.030	0.735	20.546	0.453	0.571	0.800	0.491	0.124	0.046	0.252	0.773	0.671	0.617
RF-ClusterPCTs-FR	0.030	0.734	20.806	0.451	0.569	0.801	0.488	0.126	0.047	0.248	0.765	0.668	0.616
RF-AggSingle-CO	0.031	?	19.722	0.438	0.549	0.777	0.470	0.150	0.047	0.242	0.756	0.657	0.607
RF-AggComplete-CO	0.032	?	19.117	0.440	0.551	0.777	0.471	0.150	0.047	0.249	0.761	0.659	0.610
RF-BKmeans-CO	0.032	?	18.830	0.440	0.551	0.772	0.474	0.153	0.046	0.249	0.768	0.659	0.609
RF-ClusterPCTs-CO	0.030	?	20.681	0.434	0.546	0.775	0.465	0.152	0.045	0.248	0.763	0.656	0.607
SCENE													
RF-FlatMLC	0.169	0.631	2.405	0.202	0.204	0.207	0.202	0.339	0.247	0.193	0.515	0.457	0.906
RF-AggSingle-FR	0.174	0.608	2.496	0.174	0.174	0.174	0.174	0.347	0.272	0.186	0.495	0.440	0.904
RF-AggComplete-FR	0.174	0.624	2.314	0.174	0.174	0.174	0.174	0.331	0.234	0.189	0.502	0.434	0.905
RF-BKmeans-FR	0.172	0.640	2.298	0.198	0.198	0.198	0.198	0.364	0.231	0.189	0.519	0.456	0.905
RF-ClusterPCTs-FR	0.174	0.624	2.314	0.174	0.174	0.174	0.174	0.331	0.234	0.189	0.502	0.434	0.905
RF-AggSingle-CO	0.174	0.590	2.496	0.140	0.140	0.140	0.140	0.298	0.274	0.187	0.507	0.415	0.904
RF-AggComplete-CO	0.174	0.590	2.496	0.140	0.140	0.140	0.140	0.298	0.274	0.187	0.507	0.415	0.904
RF-BKmeans-CO	0.172	0.589	2.595	0.182	0.182	0.182	0.182	0.306	0.292	0.191	0.512	0.434	0.905
RF-ClusterPCTs-CO	0.169	0.614	2.545	0.182	0.182	0.182	0.182	0.339	0.279	0.190	0.513	0.434	0.905
TMC2007													
RF-FlatMLC	0.025	0.976	2.301	0.796	0.848	0.933	0.813	0.039	0.003	0.993	0.999	0.975	0.992
RF-AggSingle-FR	0.025	0.976	2.305	0.796	0.848	0.935	0.812	0.039	0.003	0.993	0.999	0.974	0.992
RF-AggComplete-FR	0.025	0.976	2.305	0.797	0.849	0.933	0.815	0.038	0.003	0.993	0.999	0.974	0.992
RF-BKmeans-FR	0.025	0.977	2.303	0.797	0.849	0.933	0.815	0.039	0.003	0.993	0.999	0.975	0.991
RF-ClusterPCTs-FR	0.026	0.976	2.309	0.789	0.842	0.931	0.805	0.042	0.003	0.992	0.999	0.973	0.992
RF-AggSingle-CO	0.027	0.976	2.309	0.776	0.831	0.928	0.790	0.044	0.004	0.993	0.999	0.973	0.992
RF-AggComplete-CO	0.031	0.947	2.749	0.795	0.857	0.834	0.933	0.052	0.009	0.872	0.992	0.954	0.941
RF-BKmeans-CO	0.025	0.976	2.305	0.791	0.844	0.931	0.808	0.040	0.003	0.993	0.999	0.975	0.992
RF-ClusterPCTs-CO	0.026	0.976	2.308	0.788	0.842	0.933	0.805	0.041	0.003	0.993	0.999	0.974	0.992
YEAST													
RF-FlatMLC	0.197	0.759	7.176	0.482	0.587	0.741	0.530	0.241	0.166	0.508	0.710	0.722	0.675
RF-AggSingle-FR	0.199	0.755	7.308	0.471	0.578	0.743	0.514	0.241	0.170	0.501	0.699	0.717	0.669
RF-AggComplete-FR	0.200	0.753	7.269	0.469	0.576	0.740	0.513	0.246	0.172	0.500	0.682	0.713	0.665
RF-BKmeans-FR	0.199	0.755	7.215	0.473	0.580	0.737	0.521	0.248	0.167	0.505	0.704	0.716	0.669
RF-ClusterPCTs-FR	0.198	0.755	7.252	0.477	0.583	0.739	0.524	0.244	0.169	0.504	0.692	0.714	0.669
RF-AggSingle-CO	0.198	0.757	7.201	0.479	0.586	0.742	0.530	0.242	0.168	0.506	0.699	0.719	0.673
RF-AggComplete-CO	0.196	0.759	7.218	0.484	0.591	0.742	0.535	0.240	0.167	0.511	0.707	0.717	0.674
RF-BKmeans-CO	0.196	0.759	7.215	0.483	0.588	0.740	0.529	0.246	0.166	0.508	0.698	0.719	0.674
RF-ClusterPCTs-CO	0.199	0.758	7.217	0.474	0.581	0.738	0.522	0.241	0.168	0.503	0.695	0.716	0.671
CORELSK													
RF-FlatMLC	0.009	0.317	103.856	0.016	0.025	0.056	0.016	0.298	0.107	0.068	0.656	0.200	0.230
RF-AggSingle-FR	0.009	0.298	105.210	0.020	0.030	0.069	0.020	0.236	0.109	0.066	0.658	0.185	0.229
RF-AggComplete-FR	0.009	0.319	101.606	0.015	0.023	0.052	0.015	0.306	0.107	0.068	0.660	0.208	0.236
RF-BKmeans-FR	0.009	0.327	102.092	0.012	0.018	0.042	0.012	0.320	0.106	0.067	0.665	0.219	0.236
RF-ClusterPCTs-FR	0.009	0.313	107.224	0.017	0.026	0.058	0.017	0.286	0.110	0.070	0.654	0.201	0.234
RF-AggSingle-CO	0.009	0.266	121.804	0.020	0.031	0.072	0.020	0.206	0.127	0.061	0.636	0.155	0.215
RF-AggComplete-CO	0.009	0.269	120.950	0.021	0.032	0.074	0.021	0.228	0.126	0.064	0.636	0.155	0.218
RF-BKmeans-CO	0.009	0.343	97.858	0.014	0.022	0.047	0.014	0.364	0.101	0.075	0.674	0.227	0.245
RF-ClusterPCTs-CO	0.009	0.301	106.638	0.017	0.027	0.062	0.017	0.264	0.109	0.066	0.654	0.186	0.224

Fig. 4. Results with the 13 performance measures for *Random Forest* from experiments performed on 8 different datasets. The best results obtained per measure per dataset are highlighted.

independent measures (*AUPRC, AUROC, wAUPRC* and *pooledAUPRC*) for the mediamill and tmc2007 datsets are improved for almost twice, which is consistent to the notion from the literature that ensembles of PCTs improve the performance over single predictive models. Hierarchies created with clustering of the space consisting of feature rankings outperform both hierarchies obtained using label co-occurrences and flat MLC for the threshold independent measures on the medical, enron and emotions datasets. *RF-BKmeans-FR* performs the best for medical dataset in seven evaluation measures. Considering the hierarchies obtained with clustering the space of label co-occurrences, we can note that they outperform the other methods for the corel5k dataset. Using hierarchies (i.e., label dependences) rather than flat multi-label task improves the predictive performance generally for most of the evaluation measures, but not for (*ML F1 measure, ML accuracy, ML precision* and *ML recall*) in the emotions and scene datasets.

Finally, in our study we also considered training errors i.e., the errors made in the learning phase. There, in a large majority of the cases, the original *FlatMLC* method performed the best. This means that other methods we use for constructing the hierarchies do not overfit as the original one. This is another advantage of methods for construction the hierarchies identified from the obtained results.

6 Conclusions and Further Work

In this work, we have presented an approach for hierarchy construction and structuring the output (label) space by using feature ranking. More specifically, we cluster the feature rankings to obtain a hierarchical representation of the potential relations existing among the different labels. We then address the task of MLC as a task of HMLC. Moreover, we compare our approach with the approach of clustering the space consisting of label co-occurrences [6].

We investigated four clustering methods for hierarchy creation, agglomerative clustering with single and complete linkage, balanced k-means and clustering using predictive clustering trees (PCTs). The resulting problem was then approached as a HMLC problem using PCTs and random forests of PCTs for HMLC. We used eight benchmark datasets to evaluate the performance.

The results reveal that the best methods for hierarchy construction are agglomerative clustering methods and balanced k-means. Compared to the original MLC method where there is no hierarchy this improves the performance in most of the datasets. In four datasets, the hierarchies obtained by clustering the label space consisting of feature rankings improve the predictive performance compared to the hierarchies obtained by clustering the space consisting of label co-occurrences. Similar conclusions, but to a lesser extent, can be made for the random forests of PCTs for HMLC - in many of the cases (datasets and evaluation measures) the predictive models exploiting the hierarchy of labels yield better predictive performance. Finally, by considering the training error performance, we find that original MLC models overfit more than the HMLC models.

For further work, we plan to make more extensive evaluation on more datasets with diverse properties and to try more different feature ranking methods. Furthermore, we assume that potential improvement of the performance can be achieved with cutting the hierarchies based on some conditions such as density, distribution or distance between nodes. Moreover, we plan to include a comparison to network approaches given by Szymanski et al. [11]. Finally, we plan to extend this approach to other tasks, such as multi-target regression.

Acknowledgment. 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), the project LANDMARK - Land management, assessment, research, knowledge base (H2020 Grant number 635201) and Teagasc Walsh Fellowship Programme.

References

1. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
2. Dimitrovski, I., Kocev, D., Loskovska, S., Džeroski, S.: Fast and scalable image retrieval using predictive clustering trees. In: International Conference on Discovery Science, pp. 33–48 (2013)
3. Huynh-Thu, V.A., Irrthum, Wehenkel, L., Geurts, P.: Inferring regulatory networks from expression data using tree-based methods. PLos One **5**(9) (2010)
4. Kocev, D.: Ensembles for predicting structured outputs. Ph.D. thesis, IPS Jožef Stefan, Ljubljana, Slovenia (2011)
5. Kocev, D., Vens, C., Struyf, J., Džeroski, S.: Tree ensembles for predicting structured outputs. Pattern Recogn. **46**(3), 817–833 (2013)
6. Madjarov, G., Dimitrovski, I., Gjorgjevikj, D., Džeroski, S.: Evaluation of different data-derived label hierarchies in multi-label classification. In: International Workshop on New Frontiers in Mining Complex Patterns, pp. 19–37 (2014)
7. Madjarov, G., Kocev, D., Gjorgjevikj, D., Džeroski, S.: An extensive experimental comparison of methods for multi-label learning. Pattern Recogn. **45**(9), 3084–3104 (2012)
8. Malinen, M.I., Fränti, P.: Balanced K-means for clustering. In: Fränti, P., Brown, G., Loog, M., Escolano, F., Pelillo, M. (eds.) S+SSPR 2014. LNCS, vol. 8621, pp. 32–41. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44415-3_4
9. Silla, C.N., Freitas, A.: A survey of hierarchical classification across different application domains. Data Min. Knowl. Disc. **22**, 31–72 (2011)
10. Struyf, J., Džeroski, S.: Constraint based induction of multi-objective regression trees. In: Bonchi, F., Boulicaut, J.-F. (eds.) KDID 2005. LNCS, vol. 3933, pp. 222–233. Springer, Heidelberg (2006). https://doi.org/10.1007/11733492_13
11. Szymanski, P., Kajdanowicz, T., Kersting, K.: How is a data-driven approach better than random choice in label space division for multi-label classification? Entropy **18**, 282 (2016)
12. Tsoumakas, G., Katakis, I.: Multi label classification: an overview. Int. J. Data Warehouse Min. **3**(3), 1–13 (2007)
13. 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)

14. 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, Boston (2010). https://doi.org/10.1007/978-0-387-09823-4_34
15. Vens, C., Struyf, J., Schietgat, L., Džeroski, S., Blockeel, H.: Decision trees for hierarchical multi-label classification. Mach. Learn. **73**(2), 185–214 (2008)
16. Verikas, A., Gelzinis, A., Bacauskiene, M.: Mining data with random forests: a survey and results of new tests. Pattern Recogn. **44**(2), 330–349 (2011)