Correlation-Based Pruning of Stacked Binary Relevance Models for Multi-Label Learning

Grigorios Tsoumakas[1], Anastasios Dimou[2], Eleftherios Spyromitros[1], Vasileios Mezaris[2], Ioannis Kompatsiaris[2], and Ioannis Vlahavas[1]

[1] Dept of Informatics, Aristotle University of Thessaloniki,
Thessaloniki 54124, Greece
{greg,espyromi,vlahavas}@csd.auth.gr
[2] Informatics and Telematics Institute,
6th Km Charilaou-Thermi Rd, Thessaloniki 57001, Greece
{dimou,bmezaris,ikom}@iti.gr

Abstract. Binary relevance (BR) learns a single binary model for each different label of multi-label data. It has linear complexity with respect to the number of labels, but does not take into account label correlations and may fail to accurately predict label combinations and rank labels according to relevance with a new instance. Stacking the models of BR in order to learn a model that associates their output to the true value of each label is a way to alleviate this problem. In this paper we propose the pruning of the models participating in the stacking process, by explicitly measuring the degree of label correlation using the phi coefficient. Exploratory analysis of phi shows that the correlations detected are meaningful and useful. Empirical evaluation of the pruning approach shows that it leads to substantial reduction of the computational cost of stacking and occasional improvements in predictive performance.

1 Introduction

Supervised learning has traditionally focused on the analysis of single-label data, where training examples are associated with a single label λ, from a set of disjoint labels L. However, training examples in several application domains are often associated with a set of labels $Y \subseteq L$. Such data are characterized as multi-label. Though methods for learning from multi-label *textual* data have been proposed since 1999 [14, 19], the years that followed witnessed an increasing number and diversity of applications, such as bioinformatics (e.g. functional genomics) [5, 8, 2, 4, 1, 34], semantic annotation of images [3, 32, 35] and video [17, 20], directed marketing [36], music categorization into genres and emotions [13, 24, 15] and automated tag suggestion in collaborative tagging systems [10, 21].

Binary relevance (BR), one of the most popular multi-label learning methods in the literature, learns a single binary model for each different label of multi-label data independently of the rest of the labels. It has linear complexity with respect to the number of labels and can learn highly optimized (independent parameter optimization process) and potentially specialized (different learning algorithm) binary classifiers for

each label using state-of-the-art learning algorithms. In addition, BR can predict arbitrary combinations of labels, without being restricted to those existing in the training set, as is the case for the label powerset algorithm for example [18]. On the other hand, it does not take into account label correlations and may fail to accurately predict label combinations or rank labels according to relevance with a new instance.

One approach that has been proposed in the past in order to deal with the aforementioned problem of BR, works generally as follows: It learns a second (or meta) level of binary models (one for each label) that consider as input the output of all first (or base) level binary models. It will be called 2BR, as it uses the BR method twice, in two consecutive levels. 2BR follows the paradigm of stacked generalization [31], a method for the fusion of heterogeneous classifiers, widely known as stacking.

Variations of 2BR have been successfully used (i.e. achieved improved accuracy compared to BR) by several communities. To the best of our knowledge, it was firstly used by the machine learning and knowledge discovery community in [9], as part of their SVM-HF method, which was based on a support vector machine (SVM) algorithm for training the binary models of both levels. The abstraction of SVM-HF irrespectively of SVMs and its relation to stacking was pointed out in [26, 25]. A very interesting account of the use of 2BR by the image and video processing community is given in Section 1.2 of [17], where it is called context based conceptual fusion. Some of the references therein, precede [9] in date. Finally, 2BR was very recently applied to the analysis of musical titles [15].

As 2BR is based on binary classification models, it retains the aforementioned advantages of BR, apart from the linear time complexity with respect to the number of labels. The number of labels affects both the number of models trained at the meta-level and the dimensionality of their input vector. Another disadvantage of 2BR raises from the fact that some labels can be completely uncorrelated with others. A label that is irrelevant with the one being modeled is not only lacking additional, valuable information for the classification system, but it also introduces extra inherent noise from the base level.

In this paper we propose pruning the base-level models that are considered as input to the meta-level models based on the correlation of labels. The ϕ coefficient is used to calculate the correlation of each label pair based on an initial single pass over the training set. Labels with correlation below a threshold with the label being learned at the meta-level are pruned and the dimensionality of the meta-level feature space is reduced. This approach improves the system efficiency substantially, without significant loss in predictive performance. In some datasets there are even gains in performance due to the reduced noise being introduced to the system.

The rest of this paper is structured as follows. The next section presents the baseline 2BR algorithm, along with the proposed pruning approach. Section 3 describes an exploratory analysis of the distribution and semantics of the ϕ correlation coefficient based on a variety of multi-label data sets. Section 4 describes the setup and results of the empirical evaluation of the proposed approach. Finally, Section 5 concludes and points to interesting extensions of this work for the future.

2 Pruning the 2BR method

This section first gives a formal description of the baseline 2BR method that we adopted in this work. It then motivates the pruning of base-level predictions at the meta-level and presents our approach to achieving it.

2.1 Baseline 2BR

We first provide some notation for the formal description of the algorithm. Let $L = \{\lambda_j : j = 1 \dots M\}$ denote the finite set of labels in a multi-label learning task and $D = \{(\boldsymbol{x_i}, Y_i), i = 1 \dots N\}$ denote a set of multi-label training examples, where $\boldsymbol{x_i}$ is the feature vector and $Y_i \subseteq L$ the set of labels of the i-th example.

2BR accepts as parameters a number of folds F, a base-level binary classification algorithm and a meta-level binary classification algorithm.

The first step of 2BR concerns training the base-level models and gathering their predictions on a set of training examples in order to construct the meta-level training data. One approach here is to use the full training set for both base-level training and prediction gathering [9]. This however can lead to biased meta-level training data. An alternative approach is to hold part of the training set for gathering the predictions. This can lead to a reduced meta-level training set if the original training set is small. This fact was actually one of the two arguments posed against 2BR in [17]. An approach in-between these two was used in [15]: for each label the training was based on a sample of the majority class (typically corresponding to the absence of a label) in order to balance the class distribution.

We follow a different approach here, which makes use of the complete training set for both training and prediction gathering but avoids the biasing problem. Initially, the algorithm splits the training data randomly into F disjoint parts, D_k, $k = 1 \dots F$, of approximately equal size. This process is done separately for each label and independently of the rest of the labels, so that the distribution of each label in each part is similar to its distribution in the complete training set, as in stratified cross-validation. Subsequently, the base-level algorithm is employed $k = 1 \dots F$ times for each label using the set $D \setminus D_k$ for training and the set D_k for evaluation. This process leads to a meta-level training set $D' = \{(\boldsymbol{y_i}, Y_i), i = 1 \dots N\}$, where $\boldsymbol{y_i}$ is a vector containing the predictions of the base-level algorithm for each λ_j given $\boldsymbol{x_i}$. Value y_{ij} denotes the confidence of the algorithm in the annotation of $\boldsymbol{x_i}$ with label λ_j.

The last two steps of 2BR involve: a) training one base-level binary classification model B_j for each label using the base-level algorithm on the complete training set, and b) training one meta-level classification model M_j using the meta-level algorithm on the meta-level training set.

For the classification and/or ranking of a new instance $\boldsymbol{x'}$, first the decision y'_j of each model B_j is obtained. Then the vector of all decisions $\boldsymbol{y'}$ forms a meta-instance, which is given as input to each of the meta-models M_j. Based on the binary output of these models we can obtain a bipartition of the labels (multi-label classification), while if the output is numeric (confidence, probability estimate, etc), then a ranking can also be obtained.

The complexity of 2BR depends on the complexity of the base-level and meta-level algorithms used. If these are given by $f(A, N)$ and $g(A, N)$ respectively for a training set with N examples and A attributes, then the time complexity of 2BR is $O(M\,[Ff(A, N) + g(M, N)])$. The complexity of 2BR with respect to M depends on that of the meta-level learning algorithm with respect to the number of features. For example, if $g(A, N)$ is linear with respect to A, then the complexity of 2BR is quadratic with respect to M.

2.2 Correlation-Based Pruning

In this paper we argue that each meta-level model should not be trained using the predictions of all base-level models. Base-level models corresponding to labels that are not correlated with the label of the given meta-level model should instead be pruned. The motivation is that the higher dimensionality of the input space will only lead to higher cost of training the meta-models, while it might also hurt the generalization of the meta-models.

To achieve our goal, we utilize the ϕ correlation coefficient, a specialized version of the Pearson product moment correlation coefficient for categorical variables with two values, also called dichotomous variables [6]. Given two labels, λ_i and λ_j, and the frequency counts of the combinations of their values given in Table 1, the coefficient is defined as follows:

$$\phi = \frac{AD - BC}{\sqrt{(A + B)(C + D)(A + C)(B + D)}} \tag{1}$$

	λ_j	$\neg\lambda_j$
λ_i	A	B
$\neg\lambda_i$	C	D

Table 1. Contingency table for labels λ_i and λ_j.

The pruned version of 2BR accepts a threshold of ϕ correlation as a parameter, denoted t, where $0 \leq t \leq 1$. When constructing the meta-level training examples for a label λ it only takes into account the predictions of the base-level models for those labels $\lambda' \in L$ whose absolute value of the ϕ correlation with λ is greater or equal to t: $|\phi(\lambda', \lambda)| \geq t$. Obviously, the predictions of the base-level model for λ will always be taken into account, as $\phi(\lambda, \lambda) = 1$.

If M' is the largest number of base-level models whose predictions are taken into account at the meta-level, then the complexity of the pruned version of 2BR becomes $O(M\,[Ff(A, N) + g(M', N)])$. With appropriate selection of the threshold ϕ, M' can be a much smaller number compared to M, reducing the complexity of 2BR to linear with respect to the number of labels. Alternatively, by explicitly selecting a small number M' of most correlated labels, the linear complexity can be guaranteed.

An approach based on a similar idea, but applied to multiple numerical target variables using decision tree learners, is the Empirical Asymmetric Selective Transfer (EAST) [16]. For each label, EAST performs a greedy forward selection hill climbing search in the space of label subsets, guided by the accuracy of the model. This search process has quadratic complexity with respect to the number of labels. If we also consider the need to train the model at each step, then in the best case this raises the complexity to cubic with respect to the number of labels. Finally, since this process has to be done for all labels, the overall complexity of EAST is quartic with respect to the number of labels. Therefore, EAST is highly inefficient and unsuitable for domains with large numbers of labels. EAST has a clearly different focus (accuracy) compared to our approach (efficiency).

3 Exploratory Analysis of the ϕ Coefficient

This section explores the distribution of the ϕ coefficient in several multi-label data sets in order to derive conclusions on a meaningful range of values for setting the threshold parameter t of the pruned 2BR during the experiments that follow. We also attempt to gain some insight on the semantics underlying the numerical values of the ϕ coefficient by examining the names of the labels in two of these data sets.

3.1 Data sets

We explore the ϕ coefficient on 7 multi-label data sets[3]. Table 2 includes several statistics for each of these data sets [25], including the average number of labels per example (*label cardinality*) and the number of distinct label combinations *distinct labelsets*. Short descriptions of these data sets are given in the following paragraphs.

Table 2. Multi-label data sets and their statistics.

		attributes			label	label	distinct
name	**examples**	**nominal**	**numeric**	**labels**	**cardinality**	**density**	**labelsets**
bibtex	7395	1836	0	159	2.402	0.015	2856
enron	1702	1001	0	53	3.378	0.064	753
mediamill	43907	0	120	101	4.376	0.043	6555
medical	978	1449	0	45	1.245	0.028	94
reuters	6000	0	47236	101	2.880	0.029	1028
tmc2007	28596	49060	0	22	2.158	0.098	1341
yeast	2417	0	103	14	4.237	0.303	198

The *yeast* data set [8] contains micro-array expressions and phylogenetic profiles for 2417 yeast genes. Each gene is annotated with a subset of 14 functional categories (e.g. *metabolism*, *energy*, etc) from the top level of the functional catalogue (FunCat).

[3] Available at `http://mlkd.csd.auth.gr/multilabel.html`

The *tmc2007* data set is based on the data of the competition organized by the text mining workshop of the 7th SIAM international conference on data mining[4]. The original data contained 28596 aviation safety reports in free text form, annotated with one or more out of 22 problem types that appear during flights [22]. Text representation follows the boolean bag-of-words model.

The *medical* data set is based on the data made available during the computational medicine center's 2007 medical natural language processing challenge[5]. It consists of 978 clinical free text reports labeled with one or more out of 45 disease codes.

The *enron* data set is based on a collection of email messages exchanged between the Enron Corporation employees, which was made available during a legal investigation. It contains 1702 email messages that were categorized into 53 topic categories, such as *company strategy*, *humor* and *legal advice*, by the UC Berkeley Enron Email Analysis Project[6].

The *mediamill* data set was part of the Mediamill challenge for automated detection of semantic concepts in 2006 [20]. It contains 43907 video frames annotated with 101 concepts (e.g. *military*, *desert*, *basketball*, etc). The specific dataset we used corresponds to experiment 1 (visual feature extraction) as described in [20]. Each video frame is characterized by a set of 120 visual features.

The *bibtex* data set [10] is based on the data of the ECML/PKDD 2008 discovery challenge. It contains 7395 bibtex entries from the BibSonomy social bookmark and publication sharing system, annotated with a subset of the tags assigned by BibSonomy users (e.g. *statistics*, *quantum*, *datamining*). The title and abstract of bibtex entries were used to construct features using the boolean bag-of-words model.

The *reuters* (rcv1) data set is a well known benchmark for text classification methods. We have used a subset (rcv1subset1) that contains 6000 news articles assigned into one or more out of 101 topics. An extensive description of the rcv1 dataset can be found in [12].

3.2 Mean Label Correlation

Figure 1 depicts a plot of the number of label pairs that exhibit ϕ correlation greater or equal to the corresponding value of the x axis, divided by the total number of labels. We call this number *mean label correlation* as it corresponds to the mean number of correlations of each label that surpass a given threshold of positive or negative ϕ correlation. The plot shows the mean label correlation for the 7 multi-label datasets with respect to a threshold ranging from 0 to 0.3 with a step of 0.01.

Constructing such a plot prior to the execution of 2BR is fast, as it requires a single pass over the data. Based on the calculated correlations of all label pairs, we can choose the threshold parameter t of 2BR, in a way such that a small number of base-level classifiers remains on average for each label, depending on the boost in efficiency that we would like to achieve. The plot shows that after a threshold of 0.3, each label is on average correlated with less than one label (apart from itself) for all datasets.

[4] http://www.cs.utk.edu/tmw07/

[5] http://www.computationalmedicine.org/challenge/

[6] http://bailando.sims.berkeley.edu/enron_email.html

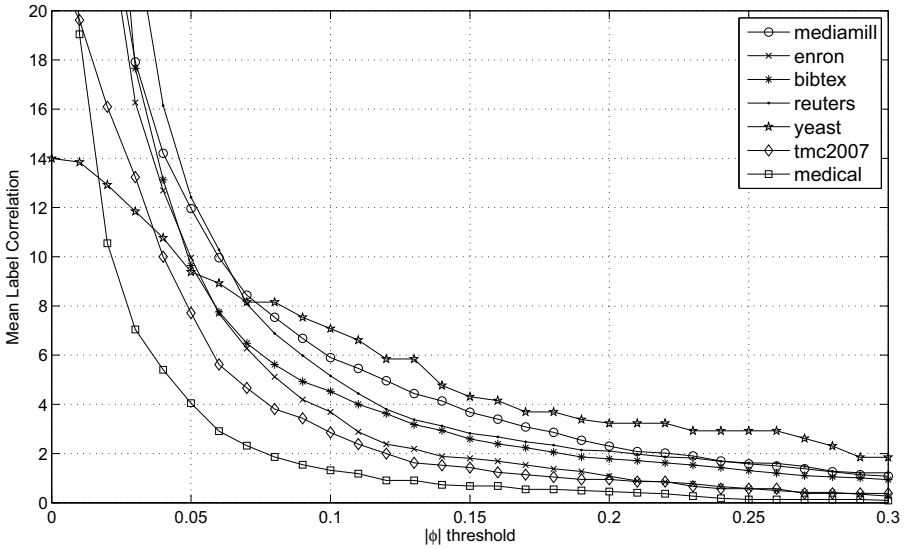


Fig. 1. Mean label correlation with respect to increasing absolute ϕ thresholds in seven multi-label data sets.

Note also that there seems to be a slight correlation between the cardinality of the dataset and the mean number of label correlations per given ϕ threshold. For example, the curves of *yeast* and *mediamill* with a cardinality of around 4 seem to be on top of the rest, despite the difference in number of labels, while on the other side, the curves of *tmc2007* and *medical*, with the smallest cardinality, seem to be lower than the rest of the curves.

3.3 Semantic Analysis of the ϕ Coefficient

The ϕ coefficient of correlation between pairs of labels is estimated from data sets that are typically annotated manually. Manual annotations can be incomplete, erroneous and even biased depending on target and time constraints of the labeling procedure, the accuracy of the labeling effort, and the content of the dataset being labeled. Taking into account all these possible complications, it is useful to examine whether a direct analogy between the ϕ-based and the semantic-based correlation of two labels exists in practice.

In the radar/spider diagram of Figure 2, the ϕ coefficient between two indicative labels, namely *building* and *government leader*, and the rest of the labels from the *mediamill* dataset is depicted. Due to the high number of labels (101), some of them were removed from the diagram to make it more legible. The labels that were removed had zero or close to zero ϕ values with both of the illustrated labels. Moreover, the zero correlation level is highlighted to help the reader distinguish between positive and negative values. It must also be noted that the autocorrelation for both examined labels, equal to 1, has been removed to enhance legibility.

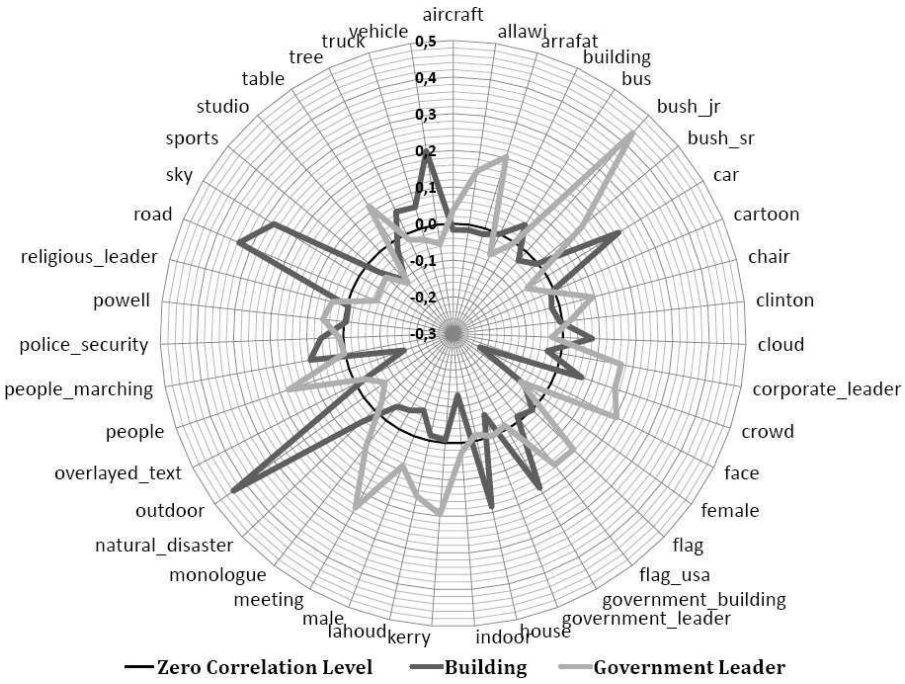


Fig. 2. Radar/spider diagram of the ϕ correlation coefficient of *Building* and *Government Leader* with the rest of the labels.

The label *building* is depicted in the graph with the thick black line. It is positively correlated with the labels *outdoor*, *sky*, *road*, *car*, *truck*, *vehicle*, *tree*, *crowd*, *house*, and *government building*. The labels *house* and *government building* have a direct semantic relation with the label *building*. *outdoor* and *sky* are not part of the concept *building* but they often accompany it in images. This can be explained by the tendency of human annotators to label an image as *building* when the entire building, or at least a good part of it, is depicted. To have such a perspective, the image is taken outdoors and most probably the sky is part of the image. Moreover the concept *building* is strongly related with an urban or suburban environment. In such an environment, the labels *road*, *car*, *truck*, *vehicle*, *tree* and *crowd* are very common. On the other hand, *studio*, *face*, *indoor*, *male*, *government leader*, *meeting* and *people* are labels that do not co-occur with *building* in this dataset. With the exception of label *people* that will be further discussed, all labels are not conceptually compatible with *building*. Label *people* is very similar to *Crowd* and there seems to be an incompatibility. The difference here is in the area of interest. An image depicting a building is not focused on people, making them a crowd without faces and gender. There are also a number of labels like *religious leader*, *overlayed text*, *cartoon*, *Clinton*, *Arrafat*, *sports*, *natural disaster* etc that have zero correlation. These labels show a random relation with the examined one, signaling that the added value that they may offer is very limited if existent.

The label *government leader* is depicted in the graph with the thick grey line. It is strongly and positively correlated with the labels *Allawi*, *Bush Jr*, *Bush Sr*, *Kerry*, *Lahoud*, *Powel*, *chair*, *corporate leader*, *crowd*, *face*, *flag*, *flag USA*, *meeting*, *table*, *people* and *male*. All of them can be mapped to international meetings between leaders and public appearances. Negative ϕ correlation exists with *female*, *studio*, *sports*, *road,* outdoor, *car* and *building*. Judging from the correlations, we can argue that label *government Leader* seems to be biased due to the dataset that is available for the training. It includes mostly American and Arab leaders meeting indoors in an office. Outdoor appearances, female leaders, talks in front of buildings that could also be relevant to the *government leader* label are under-represented in this dataset.

The radar/spider diagram of Figure 3, depicts the ϕ coefficient between two indicative labels, namely *alliances/partnerships* and *company image - changing/influencing*, and the rest of the labels from the *enron* data set. For legibility reasons some labels were removed, and the names of others were shortened or abbreviated in the diagram. All labels that were removed have again zero correlation with the examined labels or a uniform correlation with all the labels.

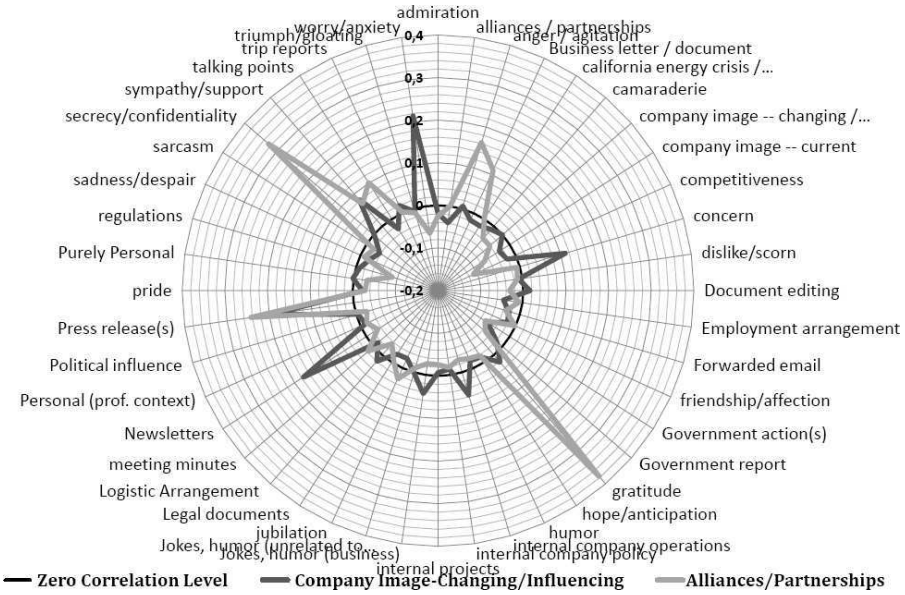


Fig. 3. Radar/spider diagram of the ϕ correlation coefficient of *alliances/partnerships* and *company image - changing influencing* with the rest of the labels.

At this point it is useful to provide some additional background information about the *enron* dataset in order to support the task of the semantic analysis that follows. As discussed in section 3.1, *enron* is based on a collection of e-mail messages. These messages were exchanged between employees of the Enron energy corporation and became

available during the legal investigation of a financial scandal, involving Enron and its accounting firm. The data set consists of 53 labels, belonging to 4 main categories, namely *Coarse genre*, *Included/forwarded information*, *Primary topic* and *Emotional tone*. Taking this information into consideration we can explain the inconsistency between some label groups, since they refer to 4 different aspects of an e-mail message. Both the illustrated labels of the radar diagram fall into the *Primary topics* main category.

The label *alliances/partnerships* is depicted in the graph with the thick grey line. It has a strong positive correlation with the labels *gratitude*, *secrecy/confidentiality* and *press release(s)* while a weaker positive correlation emerges with the labels *anger/agitation* and *business letter(s)/document(s)*. An e-mail message with *alliances/partnerships* as the primary topic, is likely to have an emotional tone of *gratitude*, following a successful partnership, or a tone of *secrecy/confidentiality* regarding a prospective alliance, product etc. Furthermore, the reference to *press release(s)* or *business letter(s)/document(s)* in such a message is normal in business practice. On the other hand, a failed partnership can provoke negative emotions, justifying the correlation with the label *anger/agitation*. The most negatively correlated label is *competitiveness/ag-gressiveness*. Again this seems reasonable, since a message labeled as *alliances/ partnerships* is usually lacking aggressive and competitive tones.

The label *company image - changing influencing* is depicted in the graph with the thick black line. It is positively associated with the labels *worry/anxiety*, *press release(s)*, *newsletters*, *concern*, *sympathy/support* and *internal company operations*. For these labels, the conceptual correlation with the changing image of the company is quite straightforward. For example the strong correlation of *worry/anxiety* and *concern* is totally substantiated, considering the darksome future of the company which came in the verge of bankruptcy. On the other hand a negative ϕ correlation is revealed with the labels *government action(s)*, *employment arrangements*, *sarcasm* and *alliance/partnerships*. All these labels are referring to actions and emotions regarding the internal affairs of a company, thus having a negative correlation with *company image*.

Concluding this section we can state that the ϕ coefficient is able to capture both real-life and semantic-based correlations between labels. Furthermore, it is able to point out relations that are not straightforward, e.g. the differences between *people* and *crowd* or the differences between internal and public affairs in a company. Our experiments have shown that in two publicly available datasets with diverse data the relationships mapped are valid and they can prove valuable. Despite its effectiveness, the ϕ coefficient is still dependent on the dataset from which it is extracted and the quality of the annotation.

4 Empirical Evaluation

We empirically evaluate the utility of the proposed approach by measuring the performance of the pruned BR^2 on the *yeast* and *enron* data sets, using threshold values ranging from 0 (no pruning) to 0.3 with a step of 0.03. As we saw in the previous section, threshold values greater than 0.3 are not expected to lead to great changes in

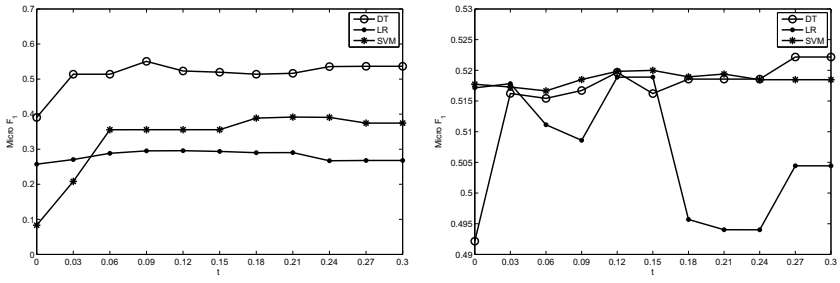
predictive performance or computational efficiency, as the mean number of correlated labels is already less than two.

In order to be able to draw general conclusions, this comparison should be made with a variety of base and meta-level algorithms. To restrict the large space of experiments, in this paper we focus on decision trees (DT) and linear kernel support vector machines (SVM) for both base-level and meta-level learning. We also use linear regression (LR) at the meta-level, based on the good results for stacking heterogeneous classifiers (i.e. based on different learning algorithms) reported in [23]. In the plots that follow, meta-level DTs, SVMs and LRs are marked with circles, stars and dots respectively.

We used the implementations of the above algorithms from the Weka library [30]. We did not perform any parameter optimization for the above algorithms and used them with their default settings, apart from DTs, where Laplace smoothing of the predicted probabilities was enabled. We implemented 2BR by extending the Mulan open source Java library for multi-label learning [29], which works on top of the Weka API.

The performance of 2BR is evaluated in terms of two criteria: a) efficiency, which is measured by the average dimensionality of the meta-level feature vector, and b) accuracy, which is measured using: i) micro F_1, which evaluates bipartitions, and ii) average precision, which evaluates rankings. A description of these and other evaluation measures for multi-label data can be found in [28].

Figure 4 shows the micro F_1 in *enron*. DTs and SVMs are employed as base-level algorithms in sub-figures (a) and (b) respectively. For base-level DTs, pruning seems to increase performance. All meta-level algorithms, namely DT, SVM, and LR, achieve their peak values while employing pruning. Especially for DTs and SVMs, which are the best performing algorithms, the improvement of the F1 measure is substantial. For base-level SVMs, the performance of the meta-level algorithms is either improved or stable with minor variations in terms of the F1 measure.

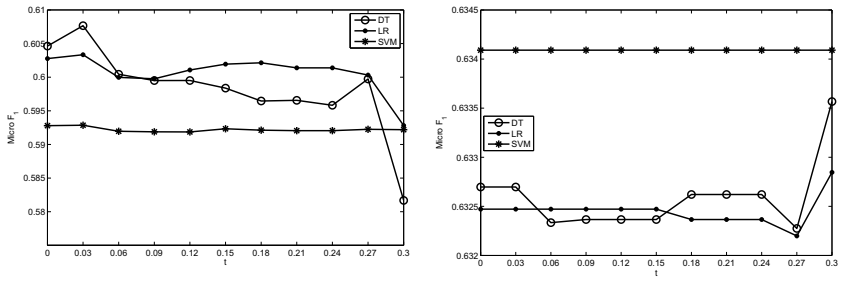


(a) Decision trees. (b) Support vector machines.

Fig. 4. Micro F_1 in *enron*.

Figure 5 depicts the micro F_1 in *yeast*. The performance trend is the same as in the previous data set. For all combinations of algorithms, the peak value of micro F_1 is obtained with pruning for different thresholds of ϕ. Overall, pruning is enhancing

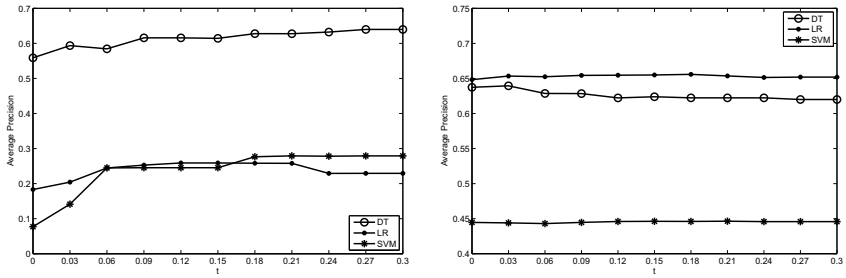
the accuracy of the examined algorithms. Their accuracy is either kept constant, or improved, sometimes substantially.



(a) Decision trees (b) Support vector machines

Fig. 5. Micro F_1 in *yeast*.

In analogy to micro F_1 the average precision is depicted in Figures 6 and 7. In *enron*, pruning is improving the average precision in all cases. Using a DT at both the base and the meta-level gives the best performance in Figure 6(a) showing an almost linear improvement as the ϕ threshold increases. When SVMs are used at the base-level, average precision is almost constant with the peak values obtained always with the aid of pruning.

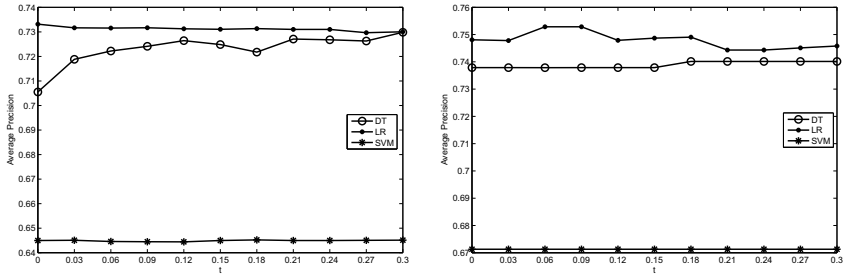


(a) Decision trees (b) Support vector machines

Fig. 6. Average precision in *enron*.

For the *yeast* dataset the conclusions are similar. With the exception of base-level DTs with meta-level LRs, where the performance shows an insignificant decrease, in all other cases pruning seems to improve average precision, including the peak performance. Especially the combination of base and meta-level DT seems to significantly benefit from pruning.

In order to validate the usefulness of pruning we have employed two data sets, three learning algorithms in eight different configurations and two different metrics.



(a) Decision trees (b) Support vector machines

Fig. 7. Average precision in *yeast*.

According to our experiments, the learning procedure seems to benefit from pruning. In several cases there are significant performance improvements, while in all other cases there is no substantial deviation from the accuracy of the baseline 2BR results.

Moreover, the label elimination that takes place, significantly reduces the system complexity, thus increasing its time-efficiency, in all cases where $|\phi| > 0$. Table 3 shows the reduction of the average number of classifiers employed in the meta-level learning for different ϕ thresholds. It suggests that if we bounded the number of labels from the base-level that contribute to the prediction of a label at the meta-level, to a small number (e.g. 5), we could achieve the same or better results at a linear time complexity with respect to M.

	0.03	0.06	0.09	0.12	0.15	0.18	0.21	0.24	0.27	0.30
enron	19.34	10.25	7.08	5.42	4.70	4.21	3.98	3.64	3.34	3.34
yeast	11.57	8.85	8.00	6.29	4.86	4.57	4.00	3.86	3.57	2.86

Table 3. Average number of classifiers being stacked for different ϕ thresholds and data sets.

5 Conclusions and Future Work

In our view, one of the important contributions of this paper is the use of the ϕ coefficient to explicitly quantify the correlation between labels. We believe that this can help improve the scalability and predictive performance of other multi-label methods beyond BR[2] as well. For example, in RAkEL [29], it could be used to construct subsets of correlated labels, with potentially improved performance. It could play the role of the similarity measure in the clustering phase of HOMER [27], perhaps leading to more appropriate clusters. Finally, in DML-kNN [33], it could be utilized in the construction of the margin vectors that are used to characterize the dependency level between labels.

One of the advantages of 2BR, as a classifier fusion method, is that it can encompass additional base-level models at the meta-level. This can be very helpful when the

objects to be classified are characterized by different representations (e.g. textual, microarray and clinical descriptors of gene functions) [11, 7]. It can also help with the fusion of heterogeneous base-level classifiers, leading to an improvement of the overall performance. Exploring this direction is among our near future plans, especially since the results of this paper showed that different algorithms work well in different datasets. In both of these scenarios (heterogeneous descriptors and learning algorithms) the pruning is expected to play a more prominent role, as the dimensionality of the meta-level feature vector will grow linearly by a factor equal to the number of different representations/algorithms, unless a hierarchical stacking approach is employed.

We also plan to investigate the relation of pruning to a number of variations of the baseline 2BR algorithm, such as extending the meta-level feature vector with the original base-level features [9, 17] and replacing the numeric meta-level features, which represent the confidence in the binary decision of base-level classifiers, with binary ones, representing the boolean decisions themselves [9, 15].

References

1. Zafer Barutcuoglu, Robert E. Schapire, and Olga G. Troyanskaya. Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836, 2006.
2. H. Blockeel, L. Schietgat, J. Struyf, S. Dzeroski, and A. Clare. Decision trees for hierarchical multilabel classification: A case study in functional genomics. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4213 LNAI:18–29, 2006.
3. M.R. Boutell, J. Luo, X. Shen, and C.M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.
4. Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Hierarchical classification: combining bayes with svm. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 177–184, 2006.
5. A. Clare and R.D. King. Knowledge discovery in multi-label phenotype data. In *Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD 2001)*, pages 42–53, Freiburg, Germany, 2001.
6. Jacob Cohen, Patricia Cohen, Stephen G. West, and Leona S. Aiken. *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. Psychology Press, 2002.
7. A. Dimou, G. Tsoumakas, V. Mezaris, I. Kompatsiaris, and I. Vlahavas. An empirical study of multi-label learning methods for video annotation. In *Proc. 7th International Workshop on Content-Based Multimedia Indexing, CBMI '09*, Chania, Greece, 2009.
8. A. Elisseeff and J. Weston. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems 14*, 2002.
9. S. Godbole and S. Sarawagi. Discriminative methods for multi-labeled classification. In *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2004)*, pages 22–30, 2004.
10. Ioannis Katakis, Grigorios Tsoumakas, and Ioannis Vlahavas. Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD 2008 Discovery Challenge*, Antwerp, Belgium, 2008.
11. Hans-Peter Kriegel, Peer Kröger, Alexey Pryakhin, and Matthias Schubert. Using support vector machines for classifying large sets of multi-represented objects. In *Proc. 4th SIAM Int. Conf. on Data Mining*, pages 102–114, 2004.

12. David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.
13. T. Li and M. Ogihara. Toward intelligent music information retrieval. *IEEE Transactions on Multimedia*, 8(3):564–574, 2006.
14. A. McCallum. Multi-label text classification with a mixture model trained by em. In *Proceedings of the AAAI' 99 Workshop on Text Learning*, 1999.
15. F. Pachet and P. Roy. Improving multilabel analysis of music titles: A large-scale validation of the correction approach. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(2):335–343, 2009.
16. Beau Piccart, Jan Struyf, and Hendrik Blockeel. Empirical asymmetric selective transfer in multi-objective decision trees. In *Proceedings of the 11th International Conference on Discovery Science*, Budapest, Hungary, 2008.
17. Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, Tao Mei, and Hong-Jiang Zhang. Correlative multi-label video annotation. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 17–26, New York, NY, USA, 2007. ACM.
18. J. Read. A pruned problem transformation method for multi-label classification. In *Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008)*, pages 143–150, 2008.
19. Y. Schapire, R.E. Singer. Boostexter: a boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.
20. Cees G. M. Snoek, Marcel Worring, Jan C. van Gemert, Jan-Mark Geusebroek, and Arnold W. M. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 421–430, New York, NY, USA, 2006. ACM.
21. Yang Song, Lu Zhang, and Lee C. Giles. A sparse gaussian processes classification framework for fast tag suggestions. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 93–102. ACM, 2008.
22. A. Srivastava and B. Zane-Ulman. Discovering recurring anomalies in text reports regarding complex space systems. In *IEEE Aerospace Conference*, 2005.
23. K.M. Ting and I.H. Witten. Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289, 1999.
24. K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. Multilabel classification of music into emotions. In *Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008), Philadelphia, PA, USA, 2008*, 2008.
25. G. Tsoumakas and I. Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2007.
26. G. Tsoumakas, I. Katakis, and I. Vlahavas. A review of multi-label classification methods. In *Proceedings of the 2nd ADBIS Workshop on Data Mining and Knowledge Discovery (ADMKD 2006)*, pages 99–109, 2006.
27. G. Tsoumakas, I. Katakis, and I. Vlahavas. Effective and efficient multilabel classification in domains with large number of labels. In *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*, pages 30–44, 2008.
28. G. Tsoumakas, I. Katakis, and I. Vlahavas. Mining multi-label data (accepted). In O. Maimon and L. Rokach, editors, *Data Mining and Knowledge Discovery Handbook*. Springer, 2nd edition, 2009.
29. G. Tsoumakas and I. Vlahavas. Random k-labelsets: An ensemble method for multilabel classification. In *Proceedings of the 18th European Conference on Machine Learning (ECML 2007)*, pages 406–417, Warsaw, Poland, September 17-21 2007.
30. Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

31. David H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.
32. Ying Yang, G. I. Webb, J. Cerquides, K. B. Korb, J. Boughton, and Kai M. Ting. To select or to weigh: A comparative study of linear combination schemes for superparent-one-dependence estimators. *IEEE Transactions on Knowledge and Data Engineering*, 19(12):1652–1665, 2007.
33. Z. Younes, F. Aballah, and T. Denoeux. Multi-label classification algorithm derived from k-nearest neighbor rule with label dependencies. In *roceedings of the 16th European Signal Processing Conference*, August 2008.
34. M-L Zhang and Z-H Zhou. Multi-label neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338–1351, 2006.
35. M-L Zhang and Z-H Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038–2048, 2007.
36. Yi Zhang, Samuel Burer, and W. Nick Street. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 7:1315–1338, 2006.