Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

A Comparative Study on the Use of Multi-Label Classification Techniques for Concept-Based Video Indexing and Annotation

Fotini Markatopoulou, Vasileios Mezaris, and Ioannis Kompatsiaris

Information Technologies Institute(ITI), Centre for Research and Technology Hellas (CERTH), Thermi 57001, Greece
(markatopoulou, bmezaris, ikom)@iti.gr

Abstract. Exploiting concept correlations is a promising way for boosting the performance of concept detection systems, aiming at concept-based video indexing or annotation. Stacking approaches, which can model the correlation information, appear to be the most commonly used techniques to this end. This paper performs a comparative study and proposes an improved way of employing stacked models, by using multi-label classification methods in the last level of the stack. The experimental results on the TRECVID 2011 and 2012 semantic indexing task datasets show the effectiveness of the proposed framework compared to existing works. In addition to this, as part of our comparative study, we investigate whether the evaluation of concept detection results at the level of individual concepts, as is typically the case in the literature, is appropriate for assessing the usefulness of concept detection results in both video indexing applications and in the somewhat different problem of video annotation.

Keywords: Concept detection, concept correlation, stacking, multi-label classification.

1 Introduction

Semantic concept detection in videos, often also referred to as semantic indexing or high-level feature extraction, is the task of assigning one or more labels (semantic concepts) to video sequences, based on a predefined concept list [1]. This process is important for several applications such as video search and retrieval, concept-based annotation and video summarization.

The majority of concept detection systems are based on variations of the following process: Ground-truth annotated videos are segmented into shots, visual features are extracted from each shot, and supervised classifiers are trained separately for each concept. Then, a new, non-annotated video shot can be associated with concept labels by applying the trained concept detectors, to get a set of confidence scores. These scores indicate the belief of each detector that the corresponding concept appears in the shot. Assigning concepts to video shots is

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

by definition a multi-label classification problem, since multiple concepts may match a single video shot. The process of training each concept detector independently, as described earlier, is known as Binary Relevance (BR) transformation and is the simplest way of solving multi-label learning problems.

In this baseline BR system, any existing semantic relations among concepts are not taken into account (e.g., the fact that *sun* and *sky* will often appear together in the same video shot). Thus, one way of improving the performance of concept detection is to also consider such concept correlations. A group of methods in this category follow a stacking architecture (e.g. [2], [3]). The predictions of multiple BR-trained concept detectors form model vectors that are used as a meta-learning training set for a second learning round. While there is no strict rule for the selection of the meta-learning algorithm, researchers mainly adopt a second round of BR models. In this work we examine the use of elaborate multi-label classification algorithms instead of BR models for the second-layer learning.

In addition to this, a closer look to the way that concept detection is evaluated shows that researchers focus on evaluating it in a concept-based indexing and retrieval setting, i.e. given a concept, measure how well the top retrieved video shots for this concept truly relate to it. However, besides the retrieval problem, another important problem related to video concept detection is the annotation problem, i.e. the problem of estimating which concepts best describe a given video shot. We argue that the retrieval-based evaluation of concept detection results is not sufficient for assessing the goodness of concept detectors in the context of the annotation problem, and we experimentally underline the importance of reporting evaluation results in both these directions.

2 Related Work

Concept correlation refers to the relations among concepts within a video shot. By using this information we can refine the predictions derived from multiple concept detectors in order to improve their accuracy, a process known as Context Based Concept Fusion (CBCF) [3]. Two main types of methods have been adopted in the literature for this: a) Stacking-based approaches that collect the scores produced by a baseline set of concept detectors and introduce a second learning step in order to refine them, b) Inner-learning approaches that follow a single-step learning process, which jointly considers low-level visual features and concept correlation information [1].

In this work we mainly focus on the first category. Stacking approaches aim to detect dependencies among concepts in the last layer of the stack. One popular group is the BR-based stacking approaches. For example, Discriminative Model Fusion (DMF) [2] obtains concept score predictions from the individual (BR-trained) concept detectors in the first layer, in order to create a *model vector* for each shot. These vectors form a meta-level training set, which is used to train a second layer of BR models. Correlation-Based Pruning of Stacked Binary Relevance Models (BSBRM) [4] extends the previous approach by pruning the

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

predictions of non-correlated concept detectors before the training of each individual classifier of the second-layer BR models. Similarly to DMF, the Baseline CBCF (BCBCF) [3] forms model vectors, in this case using the ground truth annotation, in order to train second-layer BR models. Furthermore, the authors of [3] note that not all concepts can take advantage of CBCF, so their method refines only a subset of them. Another group of stacking approaches are the graph-based ones, which model label correlations explicitly [1]. Multi-Cue Fusion (MCF) method [5] uses the ground truth annotation to build decision trees that describe the relations among concepts, separately for each concept. Initial scores are refined by approximating these graphs.

Inner-learning approaches make use of contextual information from the beginning of the concept learning process. For example, the authors of [6] and [7] propose methods that simultaneously learn the relation between visual features and concepts and also the correlations among concepts. In [8] a probabilistic Multi-Label Multi-Instance learning approach is proposed, where the multi-label part models correlations among multiple concepts and the multi-instance part models relations among different image regions. These two parts are combined into a single step in order to develop a complete system that detects multiple concepts in an image. In [9] a combination of a weighted version of kNN and multiple SVM classifiers are used for jointly assessing the semantic similarity between concepts and the visual similarity between images annotated with them. Although inner-learning approaches are out of the scope of this work, they were briefly discussed in this paragraph for the sake of completeness.

In TRECVID 2012 several teams explicitly study label correlations. For example, in [10] the Concept Association Network is used, which is a rule-based system searching for frequent item sets of concepts and extracting association rules. Other systems aim to take advantage of "imply" and "exclude" relations between concepts [11], [12]. However, we did not consider such methods in the present comparative study, because in the TRECVID experiments reported in the aforementioned publications these methods did not exhibit a significant improvement in the goodness of concept detection, compared to the BR baseline.

Label correlation has also been investigated in the broader multi-label learning domain. In [13], multi-label classification methods, including methods that consider contextual relations, are compared on multimedia data. In [9] and [14] such methods are adapted for concept detection. Nevertheless, none of these approaches considers the use of multi-label classification methods as part of a stacking architecture. The latter is the focus of this work, and section 3 describes the way we adapt such methods in order to build models in the second-layer of the stacking architecture that learn the correlations among labels.

3 The Proposed Stacking Architecture

Let $D_1, ..., D_N$ denote a set of N trained concept detectors on N different concepts. Let T denote a validation set of video shots, which will be used for training the second layer of the stacking architecture, and m denote the model vector of a

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

new unlabeled video shot. Figure 1 summarizes the full pipeline from training the second-layer classifiers to using them for classifying an unlabeled sample when using: (1) the BR stacking architecture (Fig. 1(b),(d)), and (2) the proposed stacking architecture (Fig. 1(c),(e)). Both architectures use exactly the same strategy to create the meta-level training set; the trained BR models $(D_1, ..., D_N)$ of the first layer are applied to the validation dataset T and in this way a model vector set M is created, consisting of the scores that each of $D_1, ..., D_N$ has assigned to each video shot of T for every concept (Fig. 1(a)). What distinguishes the two architectures is the way that this meta-learning information is used and therefore the way that the second-layer learning is performed.

During the training phase, the BR stacking architecture builds a new set of BR models $(D'_1, ..., D'_N)$. To train each model, a different subset of M that is ground-truth annotated for the corresponding concept C_n that the meta-concept detector D'_n will be trained for, is used (Fig. 1(b)). In contrast, the proposed architecture uses the whole model vector set and the ground truth annotation at once in order to train a single multi-label classification model D', instead of separate models $D'_1, ..., D'_N$ (Fig. 1(c)).

During the classification phase, a new unlabeled video shot is firstly given to the first layer BR models $(D_1, ..., D_N)$ and a model vector m is returned. On the one hand, the BR stacking architecture will let each of the $D'_1, ..., D'_N$ models to classify m and one score will be returned separately from each model (Fig. 1(d)). On the other hand, the proposed architecture uses the single trained model D' in order to return a final score vector (Fig. 1(e)).

With respect to learning concept correlations, the BR-based stacking methods learn them only by using the meta-level feature space. However, the learning of each concept is still independent of the learning of the rest of the concepts. The rationale behind us proposing the use of multi-label learning algorithms in replacement of the BR models at the second layer of the stacking architecture is based on the assumption that if we choose algorithms that explicitly consider label relationships as part of the second-layer training, improved detection can be achieved. Our stacking architecture learns concept correlations in the last layer of the stack both from the outputs of first-layer concept detectors and by modelling correlations directly from the ground-truth annotation of the meta-level training set. This is achieved by instantiating our architecture in our experiments with different second-layer algorithms that model:

- Correlations between pairs of concepts;
- Correlations among sets of more than two concepts;
- Multiple correlations in the neighbourhood of each testing instance.

To model the correlation information described above we exploit methods from the multi-label learning field [15]. Pairwise methods can consider pairwise relations among labels; similar to the multi-class problem, one versus one models are trained and a voting strategy is adopted in order to decide for the final classification. In this category we choose the Calibrated Label Ranking (CLR) algorithm [16] that combines pairwise and BR learning. Label power set (LP) methods search for subsets of labels that appear together in the training set

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

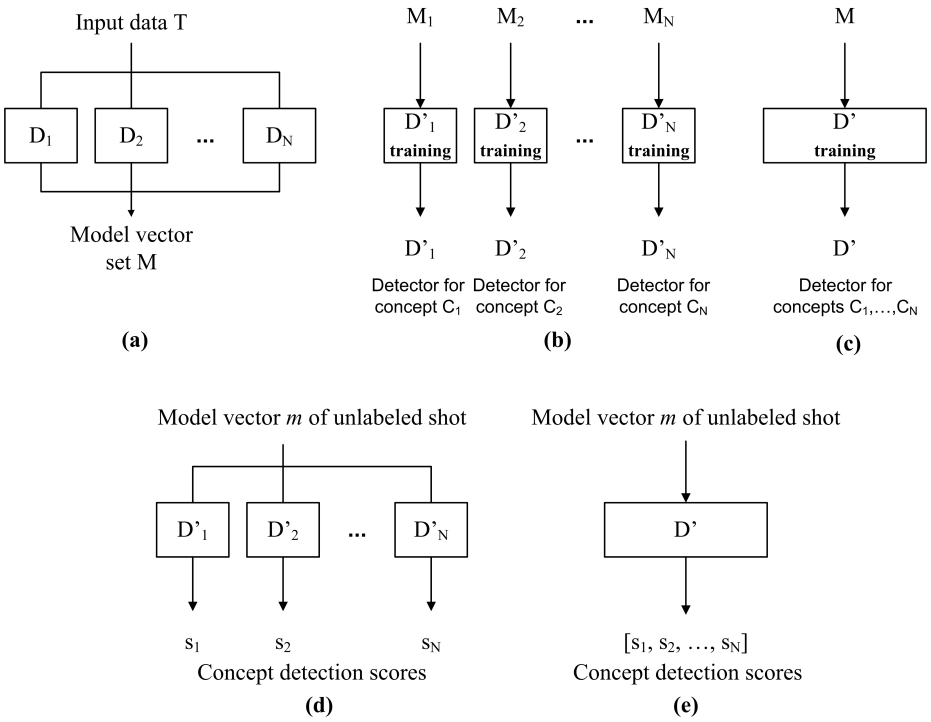

Fig. 1. Comparing BR and the proposed stacking architecture. (a) First layer of a stacking architecture. Video shot set T is given to trained concept detectors $D_1, ..., D_N$, and a model vector set M consisting of the responses of the detectors for each video shot of T is returned. (b) Training of the second layer of a BR-stacking architecture. During the training phase, BR-stacking builds a second set of concept detectors $(D'_1, ..., D'_N)$ separately for each concept, using for training each of $D'_1, ..., D'_N$ a different subset of M according to the availability of ground-truth annotations for each concept. (c) Training of the second layer of the proposed stacking architecture. The proposed architecture uses both the complete model vector set M and the ground truth annotations in order to build a single multi-label model D'. (d)&(e) During the classification phase, a new unlabeled video shot is firstly given to the first layer BR models $(D_1, ..., D_N)$ and a model vector m is returned, to be used as input to the second layer classifiers. (d) The BR stacking architecture will let each of the $D'_1, ..., D'_N$ models to classify m and one score will be returned separately from each model. (e) The proposed architecture uses m as input to the single trained multi-label classification model D'. In both cases, a set of final scores $s_1, ..., s_N$ are produced, corresponding to concepts $C_1, ..., C_N$.

and consider each set as a separate class in order to solve a multi-class problem. We choose the original LP tranformation [15], as well as the Pruned Problem Transformation algorithm (PPT) [17] that reduces the class imbalance problem by pruning label sets that occur less than l times. Finally, lazy style methods most often use label correlations in the neighbourhood of the tested instance,

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

to infer posterior probabilities. In this direction we choose ML-kNN algorithm [18], which models exactly this information. Note that the chosen methods can output both a bipartition of the labels (relevant/irrelevant) and scores in the [0,1] range. In selecting the above methods, we took into account the computational complexity of these and other similar methods and tried to avoid using particularly computationally intensive ones.

One could argue that graph-based methods also search for meta-models that model label correlations for all concepts at once; however, we differ from these methods as we choose multi-label learning approaches rather than probabilistic models. The use of multi-label classification algorithms as the second layer of a stacking architecture has the significant advantage of allowing the representation of the videos using state-of-the-art high dimensional low-level features (for describing the video at the first layer of the stack), as opposed to simpler features used in e.g. [13], [9], while at the same time keeping relatively low the dimensionality of the input to the multi-label classifier of the second layer, thus making the overall concept detection architecture applicable even to large-scale problems.

4 Experimental Setup

4.1 Dataset and Evaluation Methodology

We tested our framework on the TRECVID 2011 and 2012 Semantic Indexing (SIN) datasets [19], [20]. Each of them consists of a development set and a test set (approximately 400 and 200 hours of internet archive videos for training and testing, respectively, for TRECVID 2011, and another 600 and 200 hours for TRECVID 2012). We further partitioned the original test set into 2 sets (validation and test set, 50% each) by using the Iterative Stratification algorithm [21], suitable for multi-label data, and evaluated all techniques on the latter set using the 50 and 46 concepts that were evaluated as part of the TRECVID 2011 and 2012 SIN Task, respectively.

Regarding the annotations for these datasets, we augmented those used by TRECVID in 2011 with the results of collaborative annotation [22], [23] that was carried out for the same dataset (among other datasets) as part of the 2012 edition of the SIN Task. We solve disagreements between the two annotations by using the max operator (where in each collection of annotations, numbers 1, 0, -1 for a given shot-concept pair denote the following: 1=concept appears, -1=does not appear, 0=ambiguous). We further augmented the ground truth by using the concept "imply" relations provided by TRECVID. Finally, ambiguous and missing annotations are ignored during evaluation. A similar process was performed in order to augment the original annotations for the TRECVID 2012 dataset exploiting the results of the 2013 collaborative annotation [22], [23].

As discussed in the introduction, we also want to investigate if the typical way of evaluating concept detection results [19] is suitable for assessing their goodness for different applications. Based on this, we adopt two evaluation strategies: i)

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

Considering the video indexing problem, given a concept, we measure how well the top retrieved video shots for this concept truly relate to it. ii) Considering the video annotation problem, given a video shot, we measure how well the top retrieved concepts describe it. For each such strategy we calculate Mean Average Precision (MAP) and Mean Precision at depth k (MP@k).

4.2 Baseline Detectors and Comparisons

For the first layer of the stacking architecture (which also serves as the baseline for comparisons) we use one concept detection score per concept, extracted by combing the output of 25 linear SVM classifiers trained for the same concept, following the methodology of [24].

We instantiate the second layer of the proposed architecture with four different multi-label learning algorithms as described in section 3, and will refer to our framework as P-CLR , P-LP, P-PPT and P-MLkNN when instantiated with CLR [16], LP [15], PPT [17] and ML-kNN [18] respectively. The value of l for P-PPT was set to 30.

We compare the proposed framework against BCBCF [3], DMF [2], BSBRM [4] and MCF [5], which were reviewed in section 2. For BCBCF we use the concept predictions instead of the ground truth in order to form the meta-learning dataset, as this was shown to improve its performance in our experiments; we refer to this method as CBCFpred in the sequel. Regarding the concept selection step we use these parameters: $\lambda = 0.5$, $\theta = 0.6$, $\eta = 0.2$, $\gamma =$ the mean of Mutual Information values. For MCF we only use the spatial cue, so temporal weights have been set to zero. Finally, the ϕ coefficient threshold, used by BSBRM, was set to 0.09.

For the purpose of implementing the above techniques the Logistic Regression learning algorithm [25] is used for the classification tasks considered by some of the methods. The WEKA [26] and MULAN [27] machine learning libraries were used as the source of single-class and multi-label learning algorithms, respectively.

5 Results and Discussion

We performed two sets of experiments for each of the two TRECVID datasets [1]. For the TRECVID 2011 dataset, in the first set, the meta-level training set is composed of predictions from 50 concept detectors (the 50 concepts for which ground-truth annotations exist not only in the training set but also in the test set). In the second set of experiments, we include information from 296 more first-layer concept detectors (346 in total). For the TRECVID 2012 dataset the two experiment sets were similarly instantiated with 46 and 346 concepts respectively. Table 1 summarizes the results for the two datasets.

We start the analysis, based on the MAP and MP@k results, separately for each evaluation strategy. Results regarding the indexing problem (Table

[1] The experiments were conducted on a PC with 3.5 GHz CPU and 16GB of RAM.

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

Table 1. Performance, in terms of MAP, MP@k and CPU time, for the different methods that are compared on the TRECVID 2011 and 2012 datasets. The number of concepts that are evaluated on these datasets are 50 and 46, respectively. The meta-learning feature space for the second layer of the stacking architecture is constructed using detection scores for (I) the same 50 and 46 concepts and (II) an extended set of 346 concepts. The $\sim$ symbol indicates that the difference in MP@k between the denoted method and the best-performing method in the same column of the table is not statistically significant (thus, the absence of $\sim$ suggests statistical significance). CPU times refer to mean training (in minutes) for all 50 or 46 concepts on datasets of 67874 and 72818 shots, respectively, and application of the trained second-layer detectors on one shot of the test set (in milliseconds). Evaluation was performed only on shots that are annotated with at least one concept.

	(I) Using the output of 50 (TRECVID 2011) and 46 (TRECVID 2012) detectors for meta-learning								
	TRECVID 2011				TRECVID 2012				
Method	(a) MAP (Indexing)	(b) MP@100 (Indexing)	(c) MAP (Annotation)	(d) MP@3 (Annotation)	(e) MAP (Indexing)	(f) MP@100 (Indexing)	(g) MAP (Annotation)	(h) MP@3 (Annotation)	(i) mean Exec. Time Training /Testing
Baseline	0.3391	0.6608	0.6150	0.3697	0.2052	0.3711	0.6006	0.3251	N/A
DMF[2]	0.4068	0.7448	**0.6878**	**0.4226**	0.2614	0.4350	**0.7610**	**0.4102**	1.33/0.30
BSBRM[4]	0.3744	0.7038	0.6785	0.4174	0.2260	0.3848	0.7499	0.4046	0.39/0.09
CBCFpred[3]	0.3321	0.6146	0.6586	0.4081	0.1675	0.2700	0.6529	0.3582	0.96/0.35
MCF[5]	0.3388	0.6630	0.6122	0.3762	0.2039	0.3654	0.6661	0.3581	24.86/0.32
P-CLR	0.3876	0.7238	0.6876	0.4183	0.1997	0.3335	0.7530	0.4041	3.63/3.85
P-LP	0.3925	0.7404	0.6852	0.4174	0.2667	0.4430	0.7603	0.4074$\sim$	74.27/63.08
P-PPT	0.3614	0.6334	0.6797	0.4162	0.2443	0.4213	0.7536	0.4048	29.88/0.20
P-MLkNN	**0.4727**	**0.7998**	0.6667	0.4073	**0.2760**	**0.4893**	0.7487	0.4021	21.41/17.72

	(II) Using the output of 346 detectors for meta-learning								
	TRECVID 2011				TRECVID 2012				
Method	(j) MAP (Indexing)	(k) MP@100 (Indexing)	(l) MAP (Annotation)	(m) MP@3 (Annotation)	(n) MAP (Indexing)	(o) MP@100 (Indexing)	(p) MAP (Annotation)	(q) MP@3 (Annotation)	(r) mean Exec. Time Training /Testing
Baseline	0.3391	0.6608	0.6150	0.3697	0.2052	0.3711	0.6006	0.3251	N/A
DMF[2]	0.4095	0.7480	0.6833	0.4177	0.2611	0.4383	0.7538	0.4075	10.03/0.48
BSBRM[4]	0.4114	0.7472	0.6905	0.4231$\sim$	0.2778	0.4517	0.7645	0.4111$\sim$	1.66/0.08
CBCFpred[3]	0.3643	0.6782	0.6713	0.4109	0.2294	0.3854	0.7218	0.3824	12.68/0.28
MCF[5]	0.3440	0.6702	0.5979	0.3667	0.2030	0.3628	0.6717	0.3656	131.68/0.81
P-CLR	0.3310	0.6320	0.6731	0.4111	0.2071	0.3578	0.7508	0.4030	28.76/7.47
P-LP	0.4281	0.7684	**0.7001**	**0.4242**	0.2940	0.4761	**0.7733**	**0.4125**	390.99/68.26
P-PPT	0.3710	0.6268	0.6879	0.4176	0.2848	0.4663	0.7622	0.4100$\sim$	144.34/0.23
P-MLkNN	**0.4959**	**0.8078**	0.6810	0.4145	**0.3182**	**0.5278**	0.7704	0.4111$\sim$	135.30/115.82

1:(a),(b),(e),(f),(j),(k),(n),(o)) clearly show the effectiveness of the proposed stacking architecture when combined with ML-kNN. ML-kNN assumes that similarity among predictions means semantic similarity and also that the same errors that are observed in the first layer will be performed to images with similar concepts. While ML-kNN can model any possible correlation in the neighbourhood of a testing instance, LP and PPT can model only those that have appeared in the training set. Modelling pairwise correlations can not been considered as robust, because CLR exhibits moderate to low performance.

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

When assessing the performance of detectors in relation to the annotation problem, P-LP appears more suitable: In the first round of experiments (Table 1:(c),(d),(g),(h)), it performs slightly worse than DMF; in the second round (Table 1:(l),(m),(p),(q)) it exhibits the best performance. In general, though, with respect to the annotation problem, there is no clear winner: the performance differences between methods that model label correlations and BR methods are often limited (although in most cases statistically significant).

In order to investigate the statistical significance of the differences in MP@k observed in Table 1, the chi-square test [28] is used together with the following null hypothesis: "there is no significant difference in the total number of correct shots/concepts that appear in the first k positions between the results obtained after the application of the best performing method and the results obtained after the application of another competing approach". This test is performed separately for each of the MP@k columns of Table 1 (columns (b),(d),(f),(h),(k),(m),(o),(q)). Methods that do not have statistically significant difference ($p \geq 0.05$) from the best performing method are indicated with the $\sim$ symbol.

Regarding the second fold of this work, we observe in Table 1 that good results in the indexing-based evaluation do not guarantee the same when the system is assessed with respect to the annotation problem, and vice versa. There is not any method that reaches top performance for both of these problems. The differences in the ordering of the tested methods according to their goodness in the different experiments are striking, thus highlighting the importance of following both evaluation strategies and reporting results in both these directions when evaluating general-purpose concept detection methods. In addition to this, researchers should bear in mind that every top-performing method is shown in our experiments to be most appropriate for addressing only one of these two problems. The results presented in this work could be used as a guide in order for researchers to choose the appropriate method based on the specific task that they are interested in.

Finally, we take a look at the execution times that each method requires (Table 1:(i),(r)). One could argue that the proposed architecture that uses multi-label learning methods requires considerably more time than the typical BR-stacking one. However, we should note here that extracting one model vector from one video shot, using the first-layer detectors for 346 concepts requires approximately 1.33 minutes in our experiments, which is about three orders of magnitude slower than the slowest of the second-layer methods. As a result of the high computational complexity of the first layer of the stack, the execution time differences between all the second-layer methods that are reported in Table 1 can be considered negligible. At this point it would be reasonable to compare the stacking-based multi-label architecture to the one-layer alternative, i.e., building a multi-label classifier directly from the low-level visual features of video shots. However, the high requirements for memory space and computation time that the latter methods exhibit do not make this comparison practically feasible for our datasets on typical PCs, as we explain in the following.

Proc. 20th Int. Conf. on MultiMedia Modeling (MMM'14), Dublin, Ireland, LNCS vol 8325, pp 1-12, January 2014.
The final publication is available at link.springer.com

The computational complexity of BR, CLR, LP and PPT when used in a single-layer architecture depends on the complexity of the base classifier, in our case the Logistic Regression, and on the parameters of the learning problem. Let us assume that N concepts need to been detected and m training examples are available for learning to detect them. In this learning problem the BR algorithm, which builds N models (one for each concept), is the simplest one. CLR is the next least complex algorithm, requiring the building of N BR-models and additionally $N*(N-1)/2$ one-against-one models. LP is the most complex algorithm, since it trains a multi-class model, with the number of classes being equal to the number of distinct label sets in the training set. PPT works in the same fashion as the LP with the difference that only a pruned set of distinct label sets will be used to train the multi-class model. Finally, the training of ML-kNN is linear with respect to the size of the training set and the length of the training vectors, but the algorithm needs to make many calculations that involve the consideration of all k-neighbours of all m training examples. Given that the training datasets used in this work consist of more than 200.000 training examples, and each training example (video shot) is represented by a 4000-element low-level feature vector and is associated with a few tens of concepts (e.g. 46 for TRECVID 2012), according to the above, for the TRECVID 2012 dataset the BR algorithm would build 46 models, CLR would build 46 BR-models and 1035 one-against-one models, LP and PPT would build a multi-class classifier of 1544 and 152 (for pruning threshold equal to 30 as reported in section 4.2) classes, respectively, and finally ML-kNN would compare each training example with all other (200.000) available examples; in all these cases, the 4000-element low-level feature vectors would be employed. Taking into consideration the dimensionality of these feature vectors all above actions require considerably more time compared to the BR alternative that we employ as the first layer in our proposed stacking architecture. In addition to this, the software (MULAN [27]) used in our experiments requires the full training set to be loaded on memory at once, which again is practically unfeasible without extending the MULAN code, which is out of the scope of this work. We conclude that the two major obstacles of using multi-label classification algorithms in a one-layer architecture are the high memory space and computation time requirements, and this finding further stresses the merit of our proposed multi-label stacking architecture.

6 Conclusion and Future Work

This paper proposed an alternative way of employing the stacking architecture, used for concept detection score refinement. Multi-label classification algorithms that consider label correlations appear to be more suitable for a meta-learning training, instead of the commonly used Binary Relevance models. This conclusion is supported by a comparative study on two challenging datasets involving a multitude of diverse concepts. Furthermore, this paper compared concept detection approaches on two different problems, video indexing and annotation. In relation to this comparison, the message that this work aims to pass is that

there is not a method able to deal with both these problems in the best possible way; good performance of video indexing according to each concept separately is not a good indicator of the suitability of the method for addressing different problems such as concept-based video annotation. Future directions of work include improving the speed of some of the second-layer learning methods and also experimenting with modifications of methods that gave promising results, such as MLkNN and LP.

Acknowledgements This work was supported by the EC under contracts FP7-287911 LinkedTV and FP7-318101 MediaMixer.

References

1. Snoek, C.G.M., Worring, M.: Concept-Based Video Retrieval. Foundations and Trends in Information Retrieval **2**(4) (2009) 215–322
2. Smith, J., Naphade, M., Natsev, A.: Multimedia semantic indexing using model vectors. In: 2003 Int. Conf. on Multimedia and Expo. ICME '03., New York, IEEE press (2003) 445–448
3. Jiang, W., Chang, S.F., Loui, A.C.: Active context-based concept fusion with partial user labels. In: IEEE Int. Conf. on Image Processing, New York, IEEE press (2006)
4. Tsoumakas, G., Dimou, A., Spyromitros-xioufis, E., Mezaris, V., Kompatsiaris, I., Vlahavas, I.: Correlation-Based Pruning of Stacked Binary Relevance Models for Multi-Label learning. In: ECML/PKDD 2009 Workshop on Learning from Multi-Label Data (MLD'09), Berlin, Heidelberg, Springer-Verlag (2009) 101–116
5. Weng, M.F., Chuang, Y.Y.: Cross-Domain Multicue Fusion for Concept-Based Video Indexing. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(10) (2012) 1927–1941
6. Qi, G.J., Hua, X.S., Rui, Y., Tang, J., Mei, T., Zhang, H.J.: Correlative multi-label video annotation. In: 15th international conference on Multimedia. MULTIMEDIA '07, New York, ACM (2007) 17–26
7. Zha, Z.J., Mei, T., Wang, J., Wang, Z., Hua, X.S.: Graph-based semi-supervised learning with multiple labels. Journal of Visual Communication and Image Representation **20**(2) (2009) 97–103
8. Zha, Z.J., Hua, X.S., Mei, T., Wang, J., Qi, G.J., Wang, Z.: Joint multi-label multi-instance learning for image classification. In: Computer Vision and Pattern Recognition (CVRP 2008), New York, IEEE (2008) 1–8
9. Wang, M., Zhou, X., Chua, T.S.: Automatic image annotation via local multi-label classification. In: Int. Conf. on Content-based image and video retrieval - CIVR '08, New York, ACM Press (2008) 17–26
10. Zhu, Q., Liu, D., Meng, T., Chen, C., Shyu, M., Yang, Y., Ha, H.Y., Fleites, F., Chen, S.C.: Florida International University and University of Miami TRECVID 2012. In: TRECVID 2012 Workshop, Gaithersburg, MD, USA (2012)
11. Yu, S.I., Xu, Z., Ding, D., Sze, W., Vicente, F., Lan, Z., Cai, Y., Rawat, S., Schulam, P., Markandaiah, N., Bahmani, S., Juarez, A., Tong, W., Yang, Y., Burger, S., Metze, F., Singh, R., Raj, B., Stern, R., Mitamura, T., Nyberg, E., Jiang, L., Chen, Q., Brown, L., Datta, A., Fan, Q., Feris, R., Yan, S., Pankanti, S., Hauptmann, A.: Informedia @TRECVID 2012. In: TRECVID 2012 Workshop, Gaithersburg, MD, USA (2012)

12. Wang, F., Sun, Z., Zhang, D., Ngo, C.: Semantic Indexing and Multimedia Event Detection: ECNU at TRECVID 2012. In: TRECVID 2012 Workshop, Gaithersburg, MD, USA (2012)
13. Nasierding, G., Kouzani, A.Z.: Empirical Study of Multi-label Classification Methods for Image Annotation and Retrieval. In: 2010 Int. Conf. on Digital Image Computing: Techniques and Applications, China, IEEE (2010) 617–622
14. Kang, F., Jin, R., Sukthankar, R.: Correlated Label Propagation with Application to Multi-label Learning. In: IEEE Computer Society Conf. on Computer Vision and Pattern Recognition - CVPR'06, New York, IEEE press (2006) 1719–1726
15. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining multi-label data. In: Data Mining and Knowledge Discovery Handbook. Springer, Berlin (2010) 667–686
16. Fürnkranz, J., Hüllermeier, E., Loza Mencía, E., Brinker, K.: Multilabel classification via calibrated label ranking. Machine Learning **73**(2) (2008) 133–153
17. Read, J.: A pruned problem transformation method for multi-label classification. In: 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008), New Zealand (2008)
18. Zhang, M.L., Zhou, Z.H.: ML-KNN: A lazy learning approach to multi-label learning. Pattern Recognition **40**(7) (2007) 2038–2048
19. Over, P., Awad, G., Fiscus, J., Antonishek, B., Michel, M., Smeaton, A.F., Kraaij, W., Queenot, G.: Trecvid 2011 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In: TRECVID 2011, NIST, USA (2011)
20. Over, P., Fiscus, J., Sanders, G., Shaw, B., Awad, G., Qu, G.: Trecvid 2012 an overview of the goals , tasks , data , evaluation mechanisms , and metrics. In: TRECVID 2012, NIST, USA (2012)
21. Sechidis, K., Tsoumakas, G., Vlahavas, I.: On the Stratification of Multi-Label Data. In: 2011 Europ. Conf. on Machine Learning and Knowledge Discovery in Databases - ECML PKDD'11, Berlin, Heidelberg, Springer-Verlag (2011) 145–158
22. Ayache, S., Qunot, G.: Video corpus annotation using active learning. In Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R., eds.: Advances in Information Retrieval. Volume 4956 of LNCS. Springer Berlin Heidelberg (2008) 187–198
23. Hradi, M., Kol, M., Lnk, A., Krl, J., Zemk, P., Smr, P.: Annotating images with suggestions user study of a tagging system. In Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P., Zemk, P., eds.: Advanced Concepts for Intelligent Vision Systems. Volume 7517 of LNCS. Springer Heidelberg (2012) 155–166
24. Moumtzidou, A., Gkalelis, N., Sidiropoulos, P., Dimopoulos, M., Nikolopoulos, S., Vrochidis, S., Mezaris, V., Kompatsiaris, I.: ITI-CERTH participation to TRECVID 2012. In: TRECVID 2012 Workshop, Gaithersburg, MD, USA (2012)
25. Le Cessie, S., Van Houwelingen, J.: Ridge estimators in logistic regression. Journal of the Royal Statistical Society. Series C (Applied Statistics) **41**(1) (1992) 191–201
26. Witten, I., Frank, E.: Data Mining Practical Machine Learning Tools and Techniques. Second edn. Morgan Kaufmann, San Francisco (2005)
27. Tsoumakas, G., Spyromitros-xioufis, E., Vilcek, J., Vlahavas, I.: MULAN : A Java Library for Multi-Label Learning. Journal of Machine Learning Research **12** (2011) 2411–2414
28. Greenwood, P., Nikulin, M.: A guide to chi-squared testing. Wiley-Interscience, Canada (1996)