Active Learning with Multiple Classifiers for Multimedia Indexing

Bahjat Safadi, Georges Quénot

Active learning with multiple classifiers for multimedia indexing

Bahjat Safadi · Georges Quénot

Published online: 14 September 2010

Abstract We propose and evaluate in this paper a combination of Active Learning and Multiple Classifiers approaches for corpus annotation and concept indexing on highly imbalanced datasets. Experiments were conducted using TRECVID 2008 data and protocol with four different types of video shot descriptors, with two types of classifiers (Logistic Regression and Support Vector Machine with RBF kernel) and with two different active learning strategies (relevance and uncertainty sampling). Results show that the Multiple Classifiers approach significantly increases the effectiveness of the Active Learning. On the considered dataset, the best performance is achieved when 15 to 30% of the corpus is annotated for individual descriptors and when 10 to 15% of the corpus is annotated for their fusion.

Keywords Active learning · Imbalanced datasets · Multimedia indexing

1 Introduction

Image and video databases become more and more common and large. They are found in a variety of places including home, companies and institutions, and also in a variety of applications. In order to keep them manageable, it is essential to create powerful tools for searching and browsing the content of theses databases. These tools need other means for contents indexing. This indexing can be done at the signal level (color, texture, motion ...) or at the semantic level (concepts). For both indexing types, the latter is by far the more useful to the users and it is also the more difficult to extract from the contents. Due to the so-called *semantic gap* between the raw images (or video contents) and the elements that make sense to human beings, indexing concepts in image or video documents is a very hard task. This task is most often

B. Safadi · G. Quénot (✉)
Laboratoire d'Informatique de Grenoble, BP 53, 38041 Grenoble Cedex 9, France
e-mail: Georges.Quenot@imag.fr

carried out using classifiers or networks of classifiers working on "low level feature" vectors extracted automatically from binary multimedia contents [4, 11, 17].

Supervised learning consists in training a system from sets of positive and negative examples. The learning system may be composed of various types of feature extractors, classifiers and fusion modules. The performance of the systems depends very much upon the implementation choices and details but it is also strongly linked to the size and quality of the training examples. While it is quite easy and inexpensive to get large amounts of raw data, it is usually very costly to have them annotated because it involves human intervention for judging the "ground truth".

While the volume of data that can be manually annotated is limited due to the cost of manual intervention, there remains the possibility to select the data samples that will be annotated so that their annotation is as useful as possible [1]. Deciding which samples will be the most useful is not trivial. *Active learning* is an approach in which an existing system is used to predict the usefulness of new samples. This approach is a particular case of *incremental learning* in which a system is trained several times with a growing set of samples. The objective is to select as few sample shots as possible to be manually indexed but still have a good classification performance (in the sense of the Mean Average Precision measure commonly used in information retrieval).

Several strategies or heuristics can be considered to predict samples' usefulness. Most of them operate by *selective sampling* which consists in progressively adding the samples that are expected to be the most informative to the training set. The most popular ones include:

- Choose the most uncertain samples (*uncertainty sampling*, [8]). This strategy tries to increase the sample density in the neighborhood of the frontier between positives and negatives, therefore improves the system's precision.
- Choose the most probable positive samples (*relevance sampling*). This strategy tries to maximize the size of the positive sample set.

Relevance sampling is especially effective for highly imbalanced dataset, which is a very frequent case in video indexing. For instance, in the TRECVID high level features indexing task [15], the average target concept frequency is below 1%. Finding negative samples is easy. Whatever the sampling strategy is, these generally come numerous enough. Active learning with relevance sampling can significantly increases the positive to negative ratio in the set of annotated samples. The imbalance can be reduced by ratio factors of up to 5 in the early iterations [2] but usually the highly imbalanced problem comes while increasing the annotated fraction of the dataset.

The imbalance between positive and negative classes is a serious problem for the classical supervised learning methods. Indeed, these often have a criterion for internal optimization based on the balance assumption. This problem also relates to the fact that the evaluation performance metric, usually external to the learning method, is different from the internal criterion. Hence, the common metric used in information retrieval is the MAP (Mean Average Precision). Simple solution such as giving strong weight to the minority class elements usually leads to good results for small imbalance but produces poor results with large imbalance.

An alternative approach to the imbalanced dataset problem is to randomly select a subset from the set of negative samples with a size comparable to that of the set of all positive samples [5]. It is even possible to compensate the loss of information

related to the sub-sampling of the negative class by making several such selections on this class set and by fusing the outputs of the multiple classifiers built from these subsets. In [9] the authors applied the sub-sampling approach on the majority class (the negative one) by multi-selections and then fused the results of different learners, they took two identical size subsets for each class at each random selection. In [18, 19], the authors went deeper in the sub-sampling by taking subsets from the majority class (the negative class) in size smaller than the minority class (the positive class), thus reversing the balance between classes for each random selection. Due to the inverse proportion, the target class is favored; this is what is required for the evaluation metric that gives more weight to the well-ranked positive samples. This effect is maintained during the fusion of the multiple classifier outputs. The efficiency of the mutli-learner approach for the imbalanced dataset problems have been validated using a number of different descriptors and classifiers [14].

Active learning and multiple classifiers approaches are two different ways of dealing with the imbalanced dataset problem, the former attempts to build more balanced datasets and the latter tries to get the best from an imbalanced dataset. In this paper, we combine both approaches and show that active learning is more effective when combined with the multiple classifiers (or multi-learner) approach.

The outline of the paper continues as follows: the combination between the multiple classifier and active learning approaches is presented in Section 2. Section 3 describes the experimental results including the description of the used data collection and the descriptors, while Section 4 presents concluding remarks.

2 Active learning with multiple classifiers

Active learning may be used in different ways in the context of multimedia indexing and retrieval. We shall focus here on the development of a system for automatic multimedia indexing. Alternatively, active learning can also be used for the production of an annotated corpus or as part of an interactive search system (for improved relevance feedback). In our case, the target product is the automatic indexing system while in the two other cases, it would be the annotated corpus or the retrieval system. We describe and evaluate here the proposed method in the context of the first case but it could also be applied to the two others. This would just require some minor changes in the algorithm, especially within the evaluation part. The evaluation part is not actually part of the active learning method but it is included in the algorithm description so that it can be better understood how the evaluations are performed.

For the evaluation, a development set *Dev* and an test set *Test* are used. The test set comes with a fixed set of labels (or judgments) and is used for the evaluation of classification systems while, for the development set, new labels are added at each active learning iteration. The active learning with multiple classifiers is actually performed only within the development set and, at each active learning iteration, a classification system is produced using the current set of labeled data (this classification system is actually made of multiple learners). The same classification system is used to make predictions both on the development set for the selection of the next samples to be annotated and on the test set for the evaluation. There is no feedback of the predictions on the test set in the active learning system; these predictions are only used for the evaluation of performance of the classification

system produced at each iteration. We can then monitor the evolution of this performance with the annotated fraction of the development set. The goal is indeed to reach the highest possible performance as fast as possible i.e. with an annotated fraction as small as possible. Though in an actual application, the active learning would be stopped when an optimal fraction is reached, experiments are conducted here until the whole development set is annotated so that a complete analysis can be made.

The general structure of the active learning algorithm with multiple classifiers is given in Algorithm 1. This algorithm is a classical active learning algorithm in which we have replaced the single classifier by a set of elementary classifiers. For implementation purposes and without loss of generality, the elementary learning algorithm A is split into two parts: Train and Predict. A global parameter, *mono-learner*, can force the classical active learning mode with a single classifier.

Algorithm 1 Multiple Classifiers Active Learning Algorithm

Dev: all development data samples
$Test$: all test data samples
L_i, U_i: labeled and unlabeled subsets of Dev at iteration i
A=(Train, Predict): the elementary learning algorithm
Q: the selection (or querying) function.
Initialize L_i (e.g. 10 positives & 20 negatives)
while $Dev \setminus L_i \neq \emptyset$ **do**
  if mono-learner **then**
    $nbLearners_i = 1$
  else
    $nbLearners_i$ = Calculate the number of Learners
  end if
  for all $j \in [1..nbLearners_i]$ **do**
    Select subset T_{ij} from L_i for training
    $C_{ij} \leftarrow \text{Train}(T_{ij})$
    $P^{ij}_{un} \leftarrow \text{Predict}(U_i, C_{ij})$
    $P^{ij}_{test} \leftarrow \text{Predict}(Test, C_{ij})$
  end for
  $P^{i}_{un} \leftarrow \text{Fuse}(P^{ij}_{un})$
  Apply Q on P^{i}_{un}
  Select $\tilde{x} \in U_i$ samples
  $\tilde{y}$ = Label $\tilde{x}$
  $L_{i+1} \leftarrow L_i \cup (\tilde{x}; \tilde{y})$
  $U_{i+1} \leftarrow U_i \setminus \tilde{x}$
  $P^{i}_{test} \leftarrow \text{Fuse}(P^{ij}_{test})$
  Evaluate P^{i}_{test} and output performance at iteration i
end while

At each iteration *i*, the development set *Dev* is split into two parts: L_i, labeled samples and U_i, unlabeled samples. A global parameter f_{pos} defines the ratio between the negative and positive samples in all learners and for all iterations. This defines the number of negative samples for each learner at iteration *i*. L_i is split into L_{ipos} and L_{ineg} that respectively contain the positive and negative samples of L_i. If $|L_{\text{ineg}}| < f_{\text{pos}} \times |L_{\text{ipos}}|$ or if *mono-learner*is set, there is a single learner with a training set $T_{i1} = L_i = L_{\text{ipos}} \cup L_{\text{ineg}}$. Otherwise a number of subsets L_{ijneg} are

randomly selected out of L_{ineg} so that $|L_{\text{ijneg}}| = f_{\text{pos}} \times |L_{\text{ipos}}|$ for all j. For each of them, there is an associated learner with a training set $T_{ij} = L_{\text{ipos}} \cup L_{\text{ijneg}}$. The number of such sets and of associated learners is computed at each iteration so that each negative sample appears on average a given number of times (usually once) in the different subsets T_{ij}. C_{ij} learners are then trained on the T_{ij} sets and applied for prediction on the U_i set for the selection of the next samples to annotate and on the *Test* set for the evaluation. Predictions from the elementary classifiers are then fuseed in both cases for producing a single prediction score per sample (any late fusion method can be used). The predictions on the U_i set are used by the selection (or querying) function Q to produce a sorted list of the next samples to annotate. From the top of this list, a $\tilde{x}$ set is selected for annotation. The $\tilde{x}$ set is then added with the associated set of labels $\tilde{y}$ obtained from their annotation to the L_i set to produce the L_{i+1} set. The $\tilde{x}$ set is also removed from the U_i set to produce the U_{i+1} set.

The global algorithm is determined by the A = (Train, Predict) elementary learning algorithm (e.g. logistic regression or SVM) and by the Q selection (or querying) function which implements the active learning strategy (e.g. relevance or uncertainty sampling). It is also determined by some global parameters like the f_{pos} ratio between the number of negative and positive samples (in practice, the optimal value for this ratio depends upon the learning algorithm and the descriptor type), by the way of choosing the initial positive and negative samples (cold start), by how the fusion is performed between classifier outputs (the Fuse function) and also by the manner of how we choose the number of new samples to be integrated at each iteration.

3 Experiments

We have evaluated the Multiple Classifiers Active Learning method in a variety of contexts. It has been applied using four types of image descriptors, two types of elementary classifiers that have been each evaluated in their mono- and multi-learner versions, and with two different active learning strategies, relevance and uncertainty sampling, completed by the random and linear scan sampling strategies for comparison. All the elementary classifiers or learners output probability scores.

The other global parameter values or functions like the f_{pos} ratio or the Fuse function were determined by cross-validation using another colection with similar contents (TRECVID 2007) and the same set of descriptors and learning algorithms. These cross-validation experiments were conducted using a multi-learner approach but without active learning [14]. Five fusion function variants were tried: arithmetic mean, geometric mean, harmonic mean, minimum, and maximum, all applied to the probability scores given by the individual learners. The harmonic mean gave the best results and we display here the results obtained with it. We also tried again the other variants and we found again that the harmonic mean performs better (not shown).

The cold start problem was not really explored; a random set of 10 positive and 20 negative samples was used. For the number of samples to be added at each iteration, we chose a variable step size since we observed in previous experiments that having small steps in the beginning of the active learning process is better for improving

performance speed. In practice, we used a logarithmic scale with 40 steps. The evaluations were conducted using the TRECVID 2008 test collection and protocol.

3.1 TRECVID 2008 test collection

The TRECVID 2008 collection contains 43,616 video shots as training set and 35,766 shots as test set. The training set is fully annotated for 20 concepts and nothing remains to be annotated which makes the use of active learning irrelevant but such large fully annotated sets constitute opportunities to simulate, evaluate and compare strategies and methods in active learning without the need of actually involving a teacher [2]. In our experiments, active learning methods are started with very few annotations available from the training set. Then, each time a human annotation is needed, the corresponding subset of the full annotation is made available to the active learner.

3.2 Image representation

Concepts and Images can be represented by their vector descriptors or features. There are many descriptors that could be used to represent a specific concept in an image and it is a wider area of research to discover what features are the best to represent a concept or an image. We evaluated the different methods with descriptors of different types and sizes. These descriptors have been produced by various partners of the IRIM project of the GDR ISIS [13].

- LIG_hg104: early fusion with normalization of an RGB histogram $4 \times 4 \times 4$ and a Gabor transformation (8 orientations and 5 scales), $64 + 40 = 104$ dimensions.
- CEALIST_global_tlep: early fusion of local descriptors of texture and of an RGB color histogram, $512 + 64 = 576$ dimensions.
- ETIS_global_qwm1x3x256: 3 histograms of 3 vertical bands of visual descriptors, standard Quaternion wavelet coefficients at three scales, $3 \times 256 = 768$ dimensions.
- LEAR_bow_sift_1,000: histogram of local visual descriptors, SIFT "classic" [10], 1,000 dimensions.

3.3 Elementary classifiers

Two types of classifiers were used: Logistic Regression (LR) and Support Vector Machines (SVM) with RBF kernel. Logistic LR has proven to be efficient in multi-learner approaches [18, 19]. SVM with RBF kernel is widely used and known to perform very well due to its ability to model non-linear boundaries. For the implementation, we chose the TRIRLS package [7] for LR and the LIBSVM package [6] for SVM. For LR, probabilities are the natural output. For SVM, the values of the decision function are turned into probabilities using the Platt's approach [12] built in libsvm.

3.4 Optimal negative to positive ratios

Table 1 shows the values used for the f_{pos} global parameters on the development set for single- and multiple-learner versions of LR (SLR and MLR) and SVM-

Table 1 Optimal values of the ratio between the numbers of negative to the positive samples for the different methods and on different descriptors

Descriptor	SSVM	MSVM	SLR	MLR
LIG_hg104	4	2	2	0.05
CEALIST_global_tlep	8	4	2	0.2
ETIS_global_qwm1x3x256	4	3	2	0.05
LEAR_bow_sift_1,000	8	4	2	0.2

RBF (SSVM and MSVM) for the four considered descriptors. The optimization of these parameters was done using the TRECVID 2007 test collection. In the multiple-learner versions, the results of the classifiers are fused by the harmonic mean function.

For all cases except MLR, the optimal values for the negative subset size are a few times more than the size of the full positive set. The values are higher for the single learner case than for the multiple learner case. This was expected since the multiple-learner has another way to take into account more negative samples in total. The optimal ratio for the MLR is very low. This is probably because the LR classifier has a linear boundary between the classes and having less negative samples increases the chance that a linear boundary is a good one, as suggested in [18].

3.5 Processing times

Table 2 gives the total processing times (cumulated from several machines or nodes, single-thread programs, 2.66 GHz Intel processor) for the whole active learning process (40 iterations) on all 20 concepts, per method and per descriptor, for one strategy (relevance sampling, processing times are similar for uncertainty sampling).

As expected, the single-learner versions are faster than the multiple-learner ones. The ratio between both is much higher for LR than for SVM. This is due to the much lower f_{pos} ratio for LR that induces a much greater number of learners. This almost compensates the fact that the elementary LR classifier is much more faster than the SVM one. The computation time generally increases with the descriptor dimensionality but not in a simple way and there are some exceptions.

3.6 Active learning effectiveness

Figure 1 compares the effectiveness of the relevance and uncertainty sampling strategies for the four classifier types (SLR, MLR, SSVM and MSVM). The performance of the linear scan and random sampling strategies are also shown as baselines. The results presented here are for the LIG_hg104 descriptor only but a similar behavior is observed with the other descriptors. For the multiple-learner experiments, the fusion

Table 2 Processing times with relevance sampling strategy (hours)

Descriptor	Dims	SSVM	MSVM	SLR	MLR
LIG_hg104	104	3.246	21.118	0.2948	14.152
CEALIST_global_tlep	576	58.283	232.016	0.766	13.960
ETIS_global_qwm1x3x256	768	28.333	377.278	0.775	66.469
LEAR_bow_sift_1,000	1,000	119.8	437.5	0.618	27.499

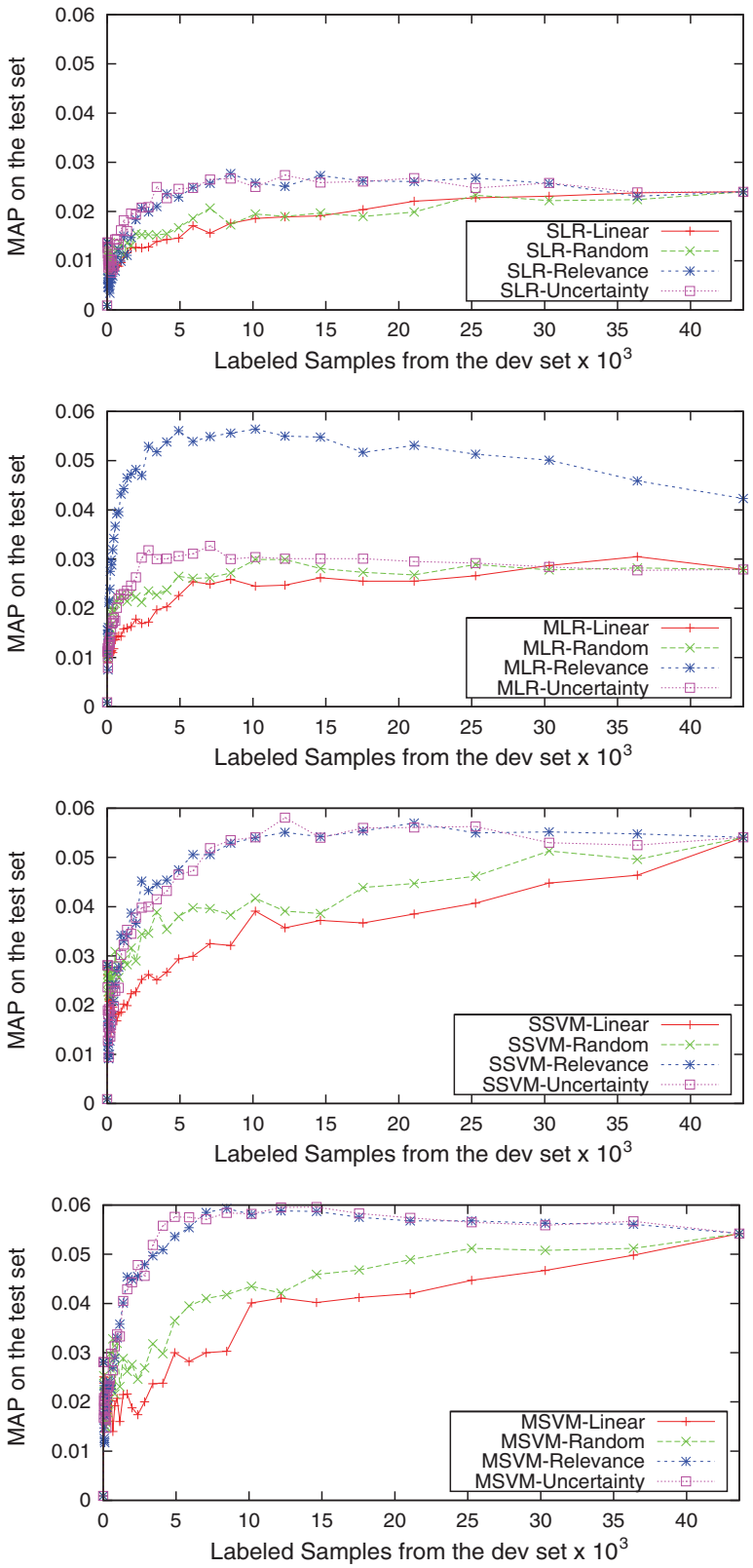

Fig. 1 Linear, random, relevance and uncertainty sampling strategies with the LIG_hg104 descriptor. Classification methods: *top*: LR mono-learner, *top middle*: LR multi-learner, *bottom middle*: SVM-RBF mono-learner and *bottom*: SVM-RBF multi-learner, fusion method for multi-learners: harmonic mean

by harmonic mean has been used. These plots show the evolution of the indexing performance measured by the Mean Average Precision (MAP) measure with the number of annotated samples. The faster it grows, especially in the beginning, the better. The higher it goes, the better also.

Unsurprisingly, the SLR method leads to a much lower performance than the MLR or SVM methods indicating that a single linear boundary is not appropriate for the considered type of data. The MSVM is the best method with the uncertainty strategy being the best one. The SSVM method is almost as good for both strategies: it goes almost as high but it grows more slowly while we can see in Table 2 that it is significantly faster. The MLR method is almost as good also but only with the relevance sampling strategy: it grows as fast as the MSVM but it goes a little bit lower while being also significantly faster (on average, considering all descriptors). For all classifier types and strategies (excluding the baselines), the maximum performance value is achieved when a small fraction (typically between 10% and 25%) of the training set is annotated. The (small) performance drop can be attributed to the fact that the imbalance between the positive and negative sample sets increases significantly: few new positive samples are discovered while no new useful information is found in the next negative samples.

The overall absolute MAP performance is low (about 0.075 for the best descriptor) but it is quite good for individual descriptors considering that a classifier with a significantly higher performance can be built by fusing the outputs of several such classifiers, that this performance can be further improved by using multiple frames in the candidate shots, and that the performance of the best classification system at TRECVID 2008 was of 0.167 (type A run).

Figure 2 compares the effectiveness of the four classifier types for the four considered descriptors. The results are presented here for the relevance sampling strategy. For the multiple-learner experiments, the fusion by harmonic mean has been used.

These plots show a significant variability according to the descriptor type. SLR is always the worst method. MLR is competitive with SVM only for two types of descriptors. MLR increases the imbalance problem between positives to negatives, this can be seen from the figures after annotating 25% of the dataset, which decreases the MAP performance. MSVM is consistently the best method. SSVM is often almost as good as MSVM except for one descriptor. Despite the variability according to the descriptor type, it is a general rule that the slowest method leads to the best result with often a small difference in performance with a large decrease in processing time. This allows to tune the speed versus quality compromise over a wider useful range. Some combination of methods and strategies could also be used like MLR with relevance sampling in the early iterations followed by MSVM with uncertainty sampling. The total processing time of the worst case (437.5 h) is comparable to the total annotation time with a single annotator assigning one label to one video shot on an average of 2 s (485 h). The experiments were conducted here until the whole set is annotated for evaluation purposes. In practice, the annotation would be stopped after only a fraction (e.g. 20%) of the training set is annotated and both the processing time and annotation times would be reduced accordingly.

The actual number of learners involved is very variable and depends upon the imbalance between positive and negative samples, the descriptor type, the learning algorithm and the f_{pos} factor. For MSVM, it ranges from 2 to 471 and for MLR, it ranges from 55 to 18,868. Values are much higher for MLR since the optimal f_{pos} factors are much lower in this case, forcing much less negative samples than positive ones and therfore a very large number of learners if the number of positive samples is quite low.

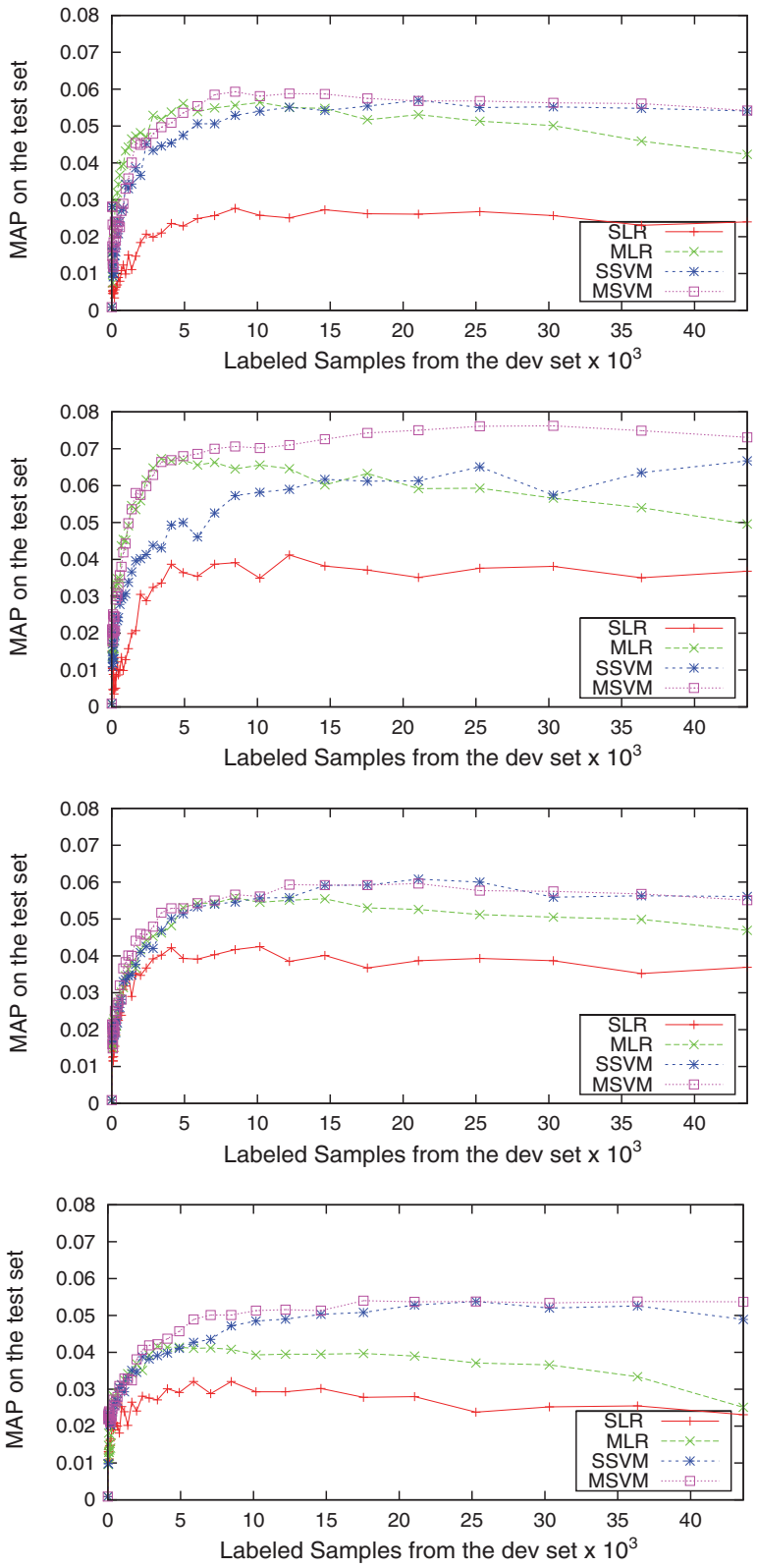

Fig. 2 The four classifiers using relevance sampling strategy. Descriptors: *top*: LIG_hg104, top middle: CEALIST_global_tlep, *bottom middle*: ETIS_global_qwm1x3x256, and *bottom*: LEAR_bow_sift_1,000

3.7 Descriptor fusion

Until now, we have studied the combination of multiple learners and active learning approaches only using individual descriptors. The most efficient methods for concept classification actually use a number of descriptors. This can be done with a number

of fusion strategies, among early and late fusion [16] or kernel fusion [3]. The performance of a system combining several individual descriptors is generally significantly higher than the performance of a system using a single descriptor used for the fusion. The gain is more important when individual descriptors are of different origin, for instance color, texture and SIFT and, in this case, a gain can be obtained relatively to

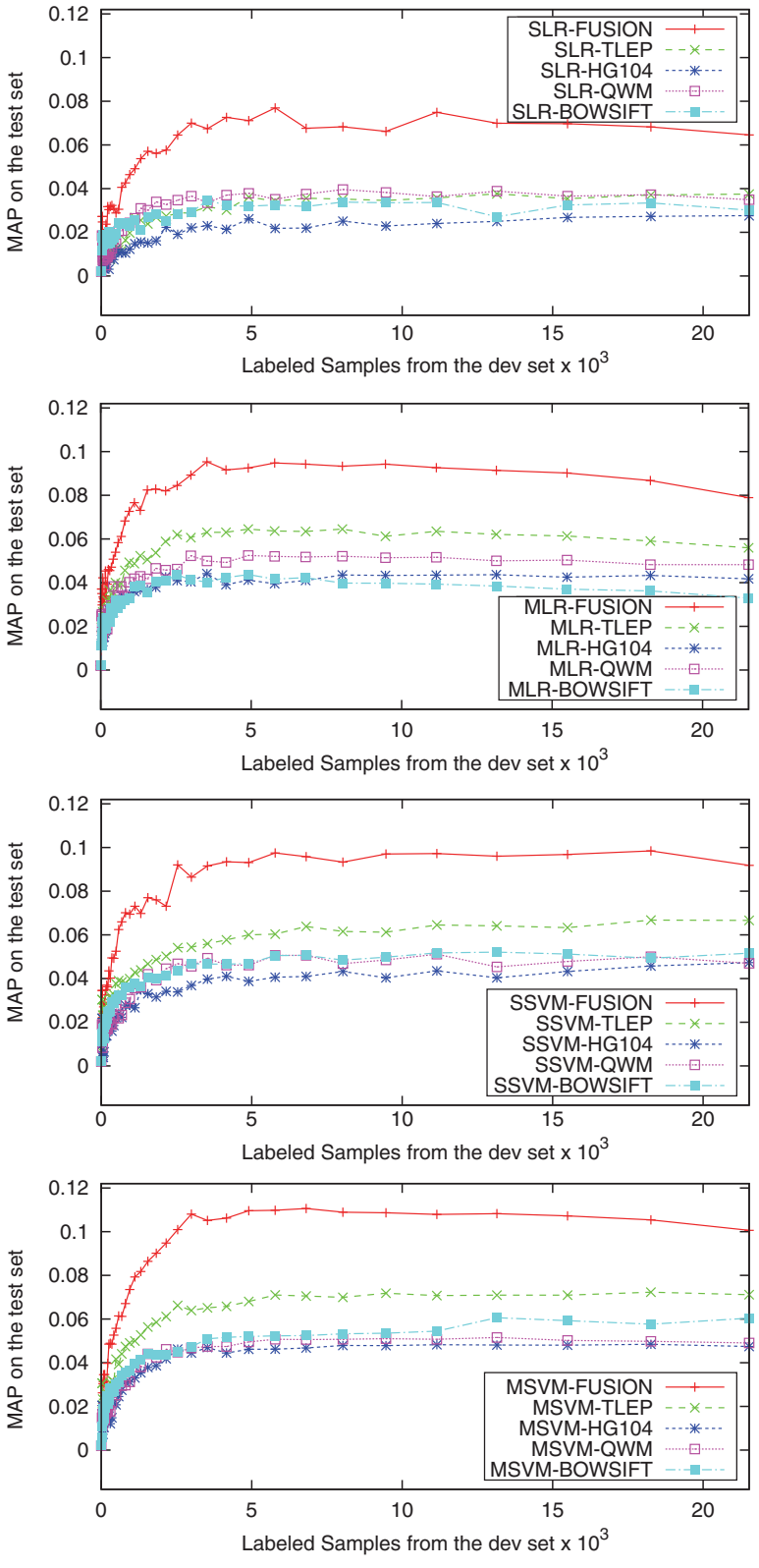

Fig. 3 Combination of fusion and active learning with mono- and multi-learner approaches: *top*: LR mono-learner, *top middle*: LR multi-learner, *bottom middle*: SVM-RBF mono-learner and *bottom*: SVM-RBF multi-learner

the best individual descriptor even if there is a large disparity among the performance of the individual descriptors.

Late fusion has been successfully combined with the multi-learner approach. We used again the harmonic mean function applied to the scores given by the classifiers assosiated to the individual descriptors [14]. We have evaluated it here in combination with the multi-learner and the active learning approaches simultaneously. Figure 3 shows the behavior of active learning using the late fusion of the four studied descriptors with mono- and multi-learner approaches and with the two considered classifiers. We observe that:

- in all cases, the fusion significantly improves the performance of the active learning as in classical learning;
- as for individual descriptors, the SVM-RBF classifier is better that the LR one and the multi-learner version is better than the mono-learner one;
- the maximum performance is obtained when 10 to 15% of the dataset is annotated which is less than individual descriptors; this absolute value is probably dependent upon the size of the dataset as observed in [2].

Figure 4 shows the behavior of active learning when fusion is done within the active learning or separately. In the second case, active learning is performed separately for each descriptor and the fusion is done one the resulting classifiers. We observe that the inclusion of the fusion within the active learning improves both the speed at which the maximum performance is reached and the level of this maximum performance. The effect is more significant in the case of the SVM-RBF classifier.

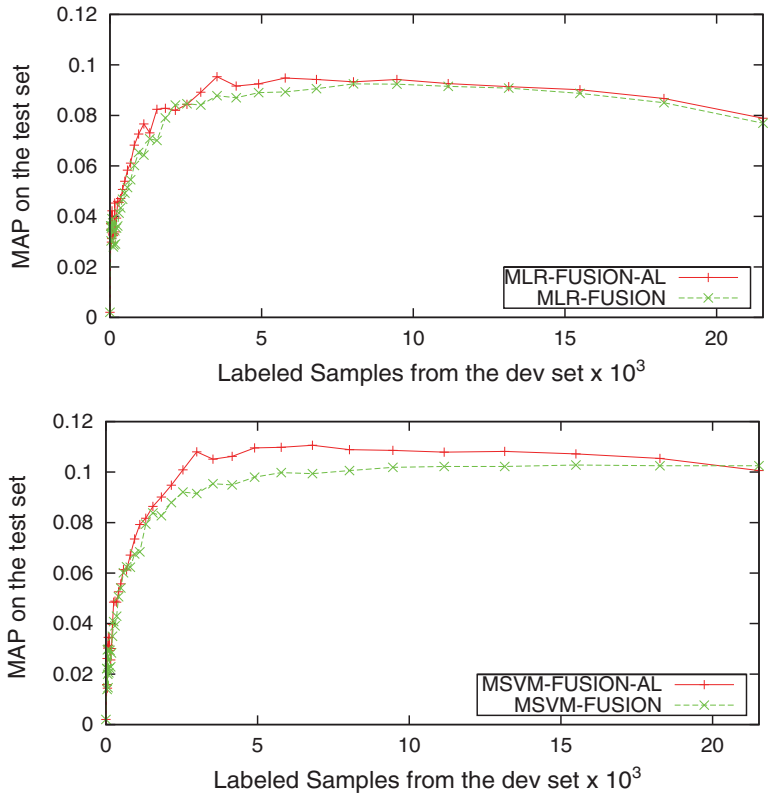

Fig. 4 Performnce of active learning when the fusion is done within the active learning process or separately: *top*: LR multi-learner and *bottom*: SVM-RBF multi-learner

4 Conclusion

We have proposed and evaluated in this paper a combination of Active Learning and Multiple Classifiers approaches for corpus annotation and concept indexing on highly imbalanced datasets. Experiments were conducted using TRECVID 2008 data and protocol with four different types of video shot descriptors, with two types of classifiers (Logistic Regression and Support Vector Machine with RBF kernel) and with two different active learning strategies (relevance and uncertainty sampling). Results show that the Multiple Classifiers approach significantly increases the effectiveness of the Active Learning. On the considered dataset, the best performance is reached when 15 to 30% of the corpus is annotated for individual descriptors and when 10 to 15% of the corpus is annotated for their fusion.

This work was mostly experimental and it would have been very interesting to complete it with theoretical studies about the convergence of the method. Unfortunately, its algorithmic complexity is very high and it seems very unlikely to us that any theoretical result can be produced, even just for guaranteeing a convergence, not mentionning an upper bound about its speed. Our results shows the effectiveness of the proposed method in a number of contexts and some empirical rules have also been deduced from them about its actual convergence and its speed of convergence in relation with the problem parameters (e.g. the collection size). Further work needs be done to confirm and refines these results using other types of data, of descriptors and of learning algorithms. Especially, general and accurate rules for the selection of the best active learning system configuration and the determination of the optimal annotation fraction should be obtained.

Acknowledgement This work was partly realized as part of the Quaero Program funded by OSEO, French State agency for innovation.

References

1. Angluin D (1988) Queries and concept learning. Mach Learn 2:319–342
2. Ayache S, Quénot G (2007) Evaluation of active learning strategies for video indexing. Image Commun 22(7–8):692–704. doi:10.1016/j.image.2007.05.010
3. Ayache S, Quénot G, Gensel J (2007) Classifier fusion for svm-based multimedia semantic indexing. In: ECIR'07: 29th European conference on information retrieval
4. Ayache S, Quénot G, Gensel J (2007) Image and video indexing using networks of operators. Image and Video Proc 2007(4):1–13. doi:10.1155/2007/56928
5. Bishop CM (2007) Pattern recognition and machine learning (Information science and statistics), 1st edn. Springer
6. Chang CC, Lin CJ (2001) LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm. Accessed 2009
7. Komarek P (2005) LR-TRIRLS: logistic regression for binary classification. Software available at http://komarix.org/ac/lr. Accessed 2009
8. Lewis DD, Gale WA (1994) A sequential algorithm for training text classifiers. In: Croft WB, van Rijsbergen CJ (eds) Proceedings of SIGIR-94, 17th ACM international conference on research and development in information retrieval. Springer, Heidelberg, pp 3–12
9. Liu XY, Wu J, Zhou ZH (2009) Exploratory undersampling for class-imbalance learning. Trans Sys Man Cyber Part B 39(2):539–550. doi:10.1109/TSMCB.2008.2007853
10. Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110

11. Naphade MR, Smith JR (2004) On the detection of semantic concepts at trecvid. In: MULTIMEDIA'04: Proceedings of the 12th annual ACM international conference on multimedia. ACM Press, New York, pp 660–667. doi:10.1145/1027527.1027680
12. Platt JC (1999) Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Advances in large margin classifiers, pp 61–74
13. Quénot G, Delezoide B, le Borgne H, Moëllic PA, Gorisse D, Precioso F, Wang F, Merialdo B, Gosselin P, Granjon L, Pellerin D, Rombaut M, Bredin H, Koenig L, Lachambre H, Khoury EE, Mansencal B, Benois-Pineau J, Jégou H, Ayache S, Safadi B, Fabrizio J, Cord M, Glotin H, Zhao Z, Dumont E, Augereau B (2009) Irim at trecvid 2009: high level feature extraction. In: TREC2009 notebook
14. Safadi B, Quénot G (2010) Evaluations of multi-learners approaches for concepts indexing in video documents. In: RIAO. Paris, France
15. Smeaton AF, Over P, Kraaij W (2006) Evaluation campaigns and trecvid. In: MIR'06: Proceedings of the 8th ACM international workshop on multimedia information retrieval. ACM Press, New York, pp 321–330. doi:10.1145/1178677.1178722
16. Snoek CG, Worring M, Smeulders AW (2005) Early versus late fusion in semantic video analysis. In: Proceedings of ACM multimedia
17. Snoek CGM, Worring M, Hauptmann AG (2006) Learning rich semantics from news video archives by style analysis. ACM Trans Multimedia Comput Commun Appl 2(2):91–108. doi:10.1145/1142020.1142021
18. Tahir MA, Kittler J, Mikolajczyk K, Yan F (2009) A multiple expert approach to the class imbalance problem using inverse random under sampling. In: MCS '09: Proceedings of the 8th international workshop on multiple classifier systems. Springer, Berlin, pp 82–91. doi:10.1007/978-3-642-02326-2_9
19. Tahir MA, Kittler J, Yan F, Mikolajczyk K (2009) Concept learning for image and video retrieval: the inverse random under sampling approach. In: Eusipco 2009, 17th European signal processing conference

Bahjat Safadi received a Master in Computer Science in 6h from the University of Caen Basse-Normandie, France. He is currently a PhD student at the Laboratoire d'Informatique de Grenoble, University of Grenoble, France. His research interests include machine learning and multimedia intexing and retrieval.

Georges Queénot is Researcher at CNRS (French National Centre for Scientific Research). He has an engineer diploma of the French Polytechnic School (1983) and a PhD in computer science (1988) from the University of Orsay. He is currently with the Multimedia Information Indexing and Retrieval group (MRIM) of the Laboratoire d'informatique de Grenoble (LIG) where he is responsible for their activities on video indexing and retrieval. His current research activity is about semantic indexing of image and video documents using supervised learning, networks of classifiers and multimodal fusion.