Partition sampling: an active learning selection strategy for large database annotation

Fabrice Souvannavong, Bernard Merialdo and Benoît Huet
Département Communications Multimédias
Institut Eurécom
2229, route des crêtes
06904 Sophia-Antipolis - France
(Fabrice.Souvannavong, Bernard.Merialdo, Benoit.Huet)@eurecom.fr

November 4, 2004

Abstract

Annotating a video database requires an intensive, time consuming and error prone human effort. However, this is a mandatory task to efficiently analyze multimedia contents. We propose an new selection strategy for active learning methods to minimize human effort in labeling a large database of video sequences. Formally, active learning is a process where new unlabeled samples are iteratively selected, presented to users for annotation and added to the training set. The major problem is then to find the best selection function to quickly reach high classification accuracy. We will show that existing active learning approaches using selective sampling do not maintain their performances when the number of selected samples per iteration increases. The presented selection strategy attempt to provide a solution to this problem. In practice, selecting many samples offers many advantages when dealing with a large amount of data; among them the possibility to share the annotation effort between several users. Finally we attempt to tackle the more realistic and challenging task of multiple label annotation. This would reduce to greater extend the human effort for labeling.

Keywords: active learning, selective sampling, nearest neighbors classifier, video database annotation.

1 Introduction

The growth of numerical storage facilities allows for many documents to be archived in huge databases or extensively shared over the Internet. The advantage of such mass storage is undeniable. However the challenging tasks of multimedia content indexing and retrieval remain unsolved without the expensive and necessary human intervention to archive and annotate contents. Indeed annotation is mandatory to either index video content through keywords or to build models for automatic content analysis. Many researchers are now investigating methods to automatically analyze, organize, index and retrieve video information [1, 2, 3, 4, 5]. This effort is further stressed by the emerging MPEG-7 standard that provides a rich and common description tool of multimedia contents [6]. It is also encouraged by Video-TREC [7] which aims at developing and evaluating video content analysis and retrieval methods on large scale databases.

Currently, one of the main challenges in the field is to bridge the gap from low-level video features to the semantic content. Classical approaches build statistical models from training samples. Unfortunately, given the complexity and diversity of semantic contents, a great amount of annotated samples is necessary to build efficient models. In June 2003, Video-TREC has launched a collaborative effort to annotate video sequences in order to build a reference database with its associated ground-truth. It is composed of about 63 hours of news videos that are segmented into shots. These shots were annotated with labels from a list of 133 items which root concepts are the event taking place, the context of the scene and objects involved. Twenty one institutes and laboratories worldwide participated to this huge collaborative annotation effort. We noticed that the database is composed of many redundant shots like news anchor person, weather maps, commercials, . . . In that case, it is very interesting to limit the annotation effort by discarding this redundant information. In an attempt to ease the annotation effort, we propose a new selection strategy for active learning approach to achieve this task.

Active learning aims at training an efficient statistical model with the smallest training data set. To achieve this goal, it iteratively selects new samples to be annotated by users. Samples are selected to optimize the knowledge gain at each iteration. Existing active learning approaches concentrate on the selection or creation of a single element to be annotated by a teacher at each round. We will show that active learning systems based on a selective sampling strategy do not maintain good performances when more than one sample is selected per iteration. We, then, propose a partition sampling approach to select a set of ambiguous samples that contain complementary information to keep system performances at their maximum. This selection strategy also allows to gain time during the annotation effort and to share this effort among several independent users. Furthermore it reduces calls to machine learning algorithms that demand important computational resources. Following the idea of reducing the annotation effort, we also apply partition sampling to the complex task of multiple label annotation that is more relevant in a real world application.

In the following section, we first introduce active learning and related work in the literature. Then, we present a common mathematical approach to uncertainty sampling. Experiments will show the limits of this common approach when the number of selected samples is increased. In section 5, we set up our mathematical framework for partition sampling and detail the algorithm that allows to efficiently annotate many samples in a single round. Then, we extend our approach to the multi-label case to confirm the behavior of partition sampling in this complex context. Finally we conclude with a brief summary including future work.

2 Related Work

Annotating content is time consuming and subject to errors. However it is necessary and compulsory in many applications to build statistical models based on training data. Limiting the effort in constructing a ground truth has raised the interest of the machine learning community. Two approaches were proposed to tackle this problem, semi-supervised and active learning. On one hand, a semi-supervised learner combines a small set of labeled samples with a large set of unlabeled samples [8]. The latter set does not provide any direct information but the distribution of its samples is used to boost the performance of the classifier. On the other hand, an active learner starts from a very small number of labeled samples and then iteratively asks for new samples to be labeled by a user. Thus it optimally updates the statistical model and increases its performance and accuracy with few samples. Using few labeled training samples also allows to better analyze the data and build more accurate models together with better generalization capabilities. Recently a new learning technique was introduced in [9], that combines active learning and unsupervised learning to take advantage of both approaches. In this paper we focus our attention on active learning methods.

The major task in active learning is to determine the optimal sample selection strategy. New samples can either be selected from an unlabeled set or be created by the system. In the latter case, samples might lack of coherence. Typically a digit recognition system could create and ask to be labeled a non existing digit that results from the combination of two digits. The former approach, called selective sampling, is the most common and many researchers proposed selection methods, such as query by committee [10, 11, 12] or uncertainty sampling [13, 14] applied to different classifiers and problems. On one hand, query by committee algorithms aim at selecting samples according to the principle of maximal disagreement between a committee of learning systems. On the other hand, uncertainty sampling algorithms rely on one learning system and its estimations.

Applications of active learning techniques are now emerging in the field of multimedia database annotation [15, 16, 17]. In the following section we present a common active learning approach using uncertainty sampling. In section 5, we propose a new uncertainty sampling strategy introduced in [18], called partition sampling. This algorithm offers the possibility to select multiple samples as opposed to classical approaches.

3 Active Learning Principle

This section introduces active learning and the uncertainty sampling strategy. Then we present the k-nearest neighbor classifier that is involved in the active learning process.

3.1 Notation and Terminology

We have a database of video sequences, denoted D, whose shots have to be annotated. A shot is represented by a vector x taking values in X. Formally, the learning algorithm takes a set of training examples $L = \{(x_1, y_1), ..., (x_N, y_N)\}$ as input where y_i is the label assigned to x_i. It produces an hypothesis $f_L : X \mapsto \Re$ that minimizes the generalization expected error:

$$E_L = \int_X E_{Y|X}[C(f_L(x), y)]P(x)dx \tag{1}$$

Where P(x) is the marginal distribution of x and $C : \Re, Y \mapsto \Re^+$ a predefined loss function.

Active learning starts from an initial annotated set and lets the learner iteratively update its training set while learning at each step from the new knowledge gain, i.e. knowledge provided by new samples. There are two main components involved in selective sampling: the classifier $f_L(.)$ trained on the labeled samples L; the selection function $s_f(P)$. The goal of $s_f(P)$ is to select the most appropriate samples S of a unlabeled pool P given the knowledge already acquired by the learner. The principle is depicted in figure 1.

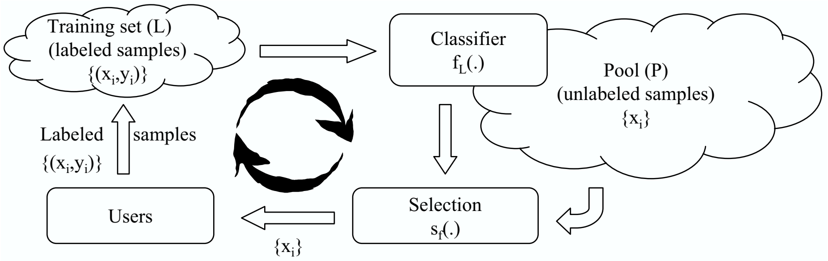


Figure 1: **Active learning principle:**Iterative method to reduce the annotation effort by selecting useful samples from an unlabeled set.

3.2 Active Learning

An active learner has to efficiently select a set of samples S in P to be labeled by users. The optimal set, $L^+ = L \cup S$, is the one that will result in the maximal error reduction, denoted R_S.

$$R_S = \int_X (E_{Y|X}[C(f_L(x),y)] - E_{Y|X}[C(f_{L^+}(x),y)])P(X)dx \tag{2}$$

$$S = s_f(P) = arg \max_S R_S \tag{3}$$

There are two difficulties in the task. First it is intractable to compute all possible combinations for S. Therefore, the common approach is to select one query sample at each round. We call this method common sampling. Secondly, we can not exactly determine the error because the target distributions $P(X)$ and $P(Y|X)$ are not known. Several assumptions have to be made leading to different selection strategies.

A classical approach consists in approximating the integral in equation 1 with a sum over the pool. P is build from a large number of unlabeled samples. Thus we can assume that its size is large enough to approximate the true distribution. Hence, the expected error reduction can be expressed as:

$$\hat{R}_S = \sum_P E_{Y|X}[C(f_L(x),y)] - E_{Y|X}[C(f_{L^+}(x),y)] \tag{4}$$

The major problem is now to learn the hypothesis $f_{L^+}(.)$ of equation 2 for each possible query sample S in order to compute the estimated error reduction. In [17], the authors first assume that all losses for any $x \in P \setminus L$ have an equal influence. Hence, the sum over P is reduced over S. Then, they can neglect $C(f_{L^+}(x),y)$ over $C(f_L(x),y)$ since the new learner is expected to have a very small loss error over S, if not null, compared to the current learner. A worst case model is, then, used to approximate $E_{Y|X}[C(f_L(x),y)]$. Let $\hat{y}$ be the estimated label of x, the best approximated error reduction is finally obtained for:

$$s_f(P) = arg \max_{x \in S} C(f_L(x), \hat{y}) \tag{5}$$

The idea behind this formulation is to select the most ambiguous sample at each iteration.

In order to evaluate the improvement provided by active learning approaches, a comparison with a random selection approach is usually performed. In this case, samples are randomly selected and annotated at each iteration. This is obviously the worst selection strategy. It is also interesting to have an idea of the best selection sequence that can be obtained. An approximation of the optimal selection sequence is given by a greedy maximization of the error reduction R_S, see equation 2, knowing the ground-truth of the database. At each iteration, we compute the improvement provided by the insertion of each sample of the pool knowing all labels. The sample that reduces the most the classification error is then selected. This is an approximation of the optimal solution since the maximization of the error reduction is done iteratively without altering previous decisions. However this optimal solution is already very time consuming and already provides a very good idea of the best performances that can be expected.

The active learning process relies on a classifier to learn models from training samples. The next section presents the classifier used in our system.

3.3 Classifier

In this paper, we focus our attention on the k-nearest neighbors classifier. This memory based method does not require any assumption about the data distribution which is very convenient for Video-TREC data set that is going to be used.

Let N_s be the neighborhood of a shot s in L, i.e. k-nearest neighbors in the training set, and $y_i \in \{-1,1\}$ the semantic value of the neighbor n_i. The hypothesis f_L is defined as:

$$f_L(s) = \frac{\sum_{N_s} sim(s,n_i) * y_{n_i}}{\sum_{N_s} sim(s,n_i)}$$
$$\text{where } sim(s,n_i) = cosine(s,n_i)$$

The estimated label of s is then:

$$\hat{y}_s = arg \min_y C(f_L(s),y)$$
$$C(u,v) = \|u-v\|$$

Another advantage of this classifier comes from the simple updating scheme that is used in our implementation. Thus time requirements remain satisfying with current data set sizes. The neighborhood of each point is updated only if it changes.

The next section presents preliminary experiments to illustrate the limits of the common approach.

4 Preliminary experiments

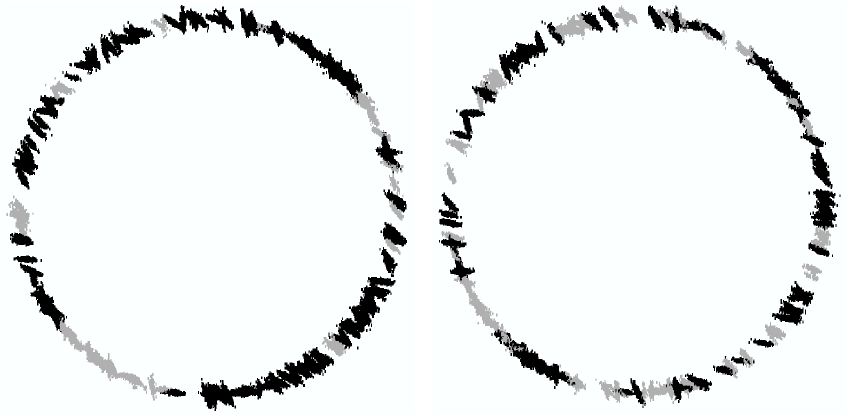
We evaluate active learning performances on both synthetic data and on the Video-TREC 2003 annotated database. First we present both data sets, then the evaluation framework and finally preliminary results.

4.1 Synthetic data sets

The synthetic data sets were generated in a space of dimension 2. The sets are composed of 200 clusters at random positions and with random labels in $\{0,1\}$. Each cluster has a random number of elements between 20 and 400 that are normally distributed around the centroid. Sets are composed of a 10,000 samples. In the first one, clusters are constructed such that the overlapping of classes remains reasonably small (figure 2(a)). The second one have more interlaced classes (figure 2(b)) and it is therefore more difficult to model.

4.2 Video-TREC database

The Video-TREC database is composed of 20,000 annotated shots from ABC and CNN news sequences [19]. We propose a region-based system to efficiently index visual features of video shots. Contrasting to traditional approaches that compute global features, the region-based methods extract features of the segmented frames and perform comparisons at the granularity of the region. The main objective is to keep the local information in a way that reflects the human perception of the content [20, 21]. In order to keep both computational complexity and storage requirements at a reasonable level, region features are usually quantized, thus allowing a compact frame representation as



(a) **First synthetic data set.** Small overlapping of classes. (b) **Second synthetic data set.** Accentuated overlapping of classes.

Figure 2: Illustration of synthetic datasets

a count vector. From previous experiments [22], 2,000 quantification values for each feature, i.e. color and texture, provide the best performances for retrieval tasks. Unfortunately, region-based methods are sensitive to the content, the segmentation and the quantization. We thus introduce latent semantic indexing (LSA), as described in [23] and [24], to reduce the side effects of the segmentation and quantization.

LSA is a method borrowed from the information retrieval community that aims at discovering synonyms and the polysemy of words to identify similar text documents [25]. It describes the semantic content of a context by projecting words (within this context) onto a latent space. In our case, a context is a shot and words composing the context are quantification values that we call visual words. LSA analyzes the occurrence of visual words into shots thanks to the singular value decomposition (SVD) that is used to compute the projection parameters to the latent space. The number of singular values kept for the projection drives the LSA performance. On one hand if too many factors are kept, the noise will remain and the detection of synonyms and the polysemy of visual terms will fail. On the other hand if too few factors are kept, important information will be lost degrading performances. Unfortunately no solution has yet been found and only experiments allow to find the appropriate factor number. From previous experiments [22], reducing the size by 10% allows to achieve good retrieval performances on Video-TREC data.

Since this is not the scope of this paper to deal with fusion methods, color and texture features are first projected in their respective latent spaces and then fused by concatenation. Shots are thus described by a vector of size 400 including color and texture information. We focus our effort on the detection of shots presenting *weather news*, *studio settings*, *vegetation*, *face* or *physical violence* concepts. These features have the particularity to have different a priori properties. *Weather news* and *studio* features are frame level concept. They characterize the complete frame. The former occurs rarely in our data set: 128 positive samples and the latter occurs often: 1287 positive samples. Other features have various a priori probabilities and will be used for the problem of multi-label classification.

4.3 Evaluation framework

The evaluation consists in comparing system performances, in terms of error rate, when the training set grows. For each system, the error rate with respect to the training size is plotted. In that case, a good system quickly reaches a small error rate after few iterations. Next, depending on the selection strategy the number of iterations differs. For example, if one sample is selected at each iteration, the number of iteration is equal to the size of the training set. But if two hundred samples are selected at each iteration then the number of iteration is two hundred times smaller than the size of the training set. In that case, at a given error rate and training size, a system that involves a small number of iterations is better.

Two reference experiments can be plotted to have a global idea of system performances. The random sampling strategy, i.e. when samples are randomly selected in the pool P, provides the worst performance while the greedy approach provides the best performance that we can expect from evaluated systems.

In order to plot error rate curves, we need the ground-truth that is available with proposed data sets. The user intervention is then not required for the active learning and the evaluation. Systems start with a initial set of training samples and their known labels. Then at each iteration new samples from the pool P are selected with respect to a selection strategy. A virtual user, i.e. the system itself, annotated samples, next they are added with their known labels to the training set *L*. In a real application, a user will have to annotate selected samples in order to teach the system.

4.4 Preliminary Results

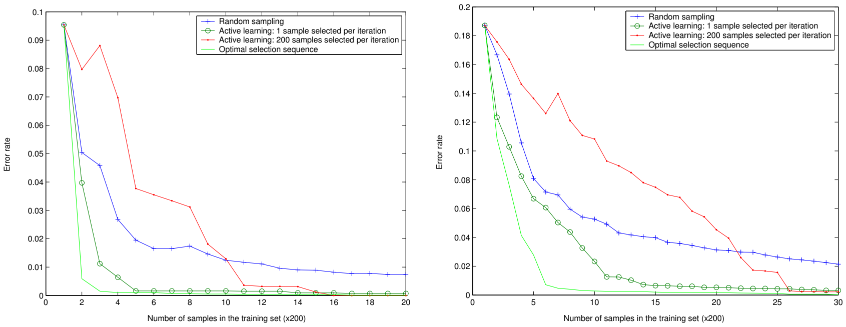
Preliminary experiments presented here have two objectives. The first is to confirm the benefits of the described active learning approach: the hypothesis can be learned thanks to a reduced training set. The second is to analyze its limits when increasing the number of selected samples: the error rate decreases slower.

Figures 3 and 4 show the classification error rate with respect to the size of the training set. They are composed of three plots depending on the selection strategy used. The first plot correspond to the random approach, i.e. the worst case. The second plot correspond to the selective sampling approach when one sample is selected at each iteration. The third plot correspond to the selective sampling approach when many samples are selected at each iteration (The number of selected samples depends on available data). Finally on synthetic datasets, figure 3, the fourth plot is the optimal solution. Note that on real data, it is not possible to compute the optimal selection sequence due to necessary time requirements.

When one sample is selected at each iteration, the advantage of active learning is significant on both synthetic and real data (figures 3 and 4). It allows better performances for a given number of samples or same performances for much less samples.

As expected, the presented approach does not perform as well when we increase the number of samples selected at each round (as shown on figures 3 and 4). This phenomenon was already observed in the article [17]. Intuitively, when more than one sample is selected, it is quite likely that they have are very close one from each other. Their similarity implies redundancy of information which slows down the evolution of the active learning system. This explains why performances can be lower than performances obtained with the random approach on the figure 3. And why we observe a very high error rate at the second and third iterations on the figure 4(a).

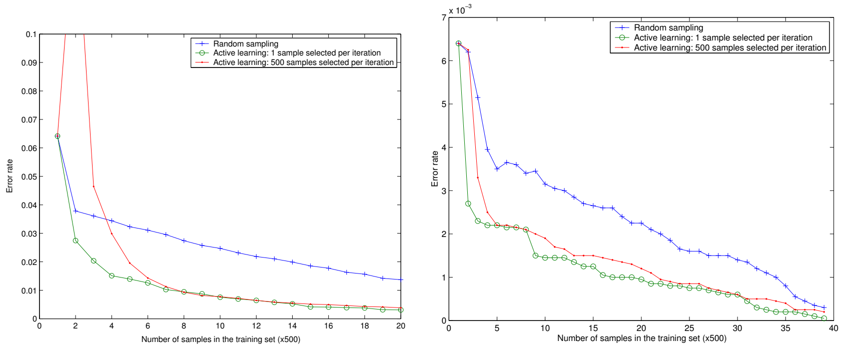
Classical selection strategies are not well suited to select many samples per iteration. In the next section, we propose a novel selection strategy, called partition sampling, to efficiently select multiple samples without altering system performances.



(a) First synthetic data set

(b) Second synthetic data set

Figure 3: **Active learning on synthetic data.** Evaluation of active learning. The error rate decreases slower when the number of samples selected per iteration increases.



(a) Studio Video-TREC feature

(b) Weather Video-TREC feature

Figure 4: **Active learning on Video-TREC data.** Evaluation of active learning. The error rate decreases slower when the number of samples selected per iteration increases.

5 Partition Sampling

This section presents our approach to efficiently select a set of samples at each round. First we detail the mathematical model then its implementation will be exposed.

5.1 Theory

We propose a new selection strategy to efficiently select a set of samples at each round. Learning algorithms make the assumption that similar elements belong to the same class. Thus the knowledge of one sample should induce the knowledge of its similar neighbors. This is implicitly used in common active learning approaches and it is emphasized in [16], where they proposed to weight the selection function value of a sample with an estimation of its probability density function to increase learning speed. However, most ambiguous points are likely to be neighbors. Thus a strategy that would select the n most ambiguous samples would mostly ask the teacher to annotate similar contents; resulting in sub-optimal selections.

It is therefore important to select ambiguous points spread over the distribution of X. We have to ensure that most of selected points are far from each other and also as ambiguous as possible. Let assume that we constructed a partition of P, i.e. $P = \cup U_i$ and $U_i \cap U_j = \emptyset$ for $i \neq j$, such that U_i are connex and that given $\varepsilon \in \Re$ then:

$$\forall (x_1, x_2) \in U_i \times U_i$$
$$\|x_1 - x_2\| < \varepsilon$$

Consider a representative element of each set selected with a selection function $m_i = \dot{s}_f(U_i)$, for example mean element, maximum ambiguity, maximum density. Let $M = \{m_i\}$, then we approximate equation 4 with:

$$\hat{R}_S = \sum_M (E_{Y|X}[C(f_L(x_i), y_i)] - E_{Y|X}[C(f_{L^+}(x_i), y_i)])N_i \tag{6}$$

Where N_i is the cardinal of U_i. This approximation relies on the assumption that neighbors have the same behavior with respect to learners, i.e. similar loss value for a given learner. Let

$$\Delta_{L,L^+}(x_i, y_i) = E_{Y|X}[C(f_L(x_i), y_i)] - E_{Y|X}[C(f_{L^+}(x_i), y_i)]$$

We are looking for S such that:

$$s'_f(P) = arg\max_S[\sum_S \Delta_{L,L^+}(x_i,y_i))N_i + \sum_{M\setminus S} \Delta_{L,L^+}(x_i,y_i)N_i] \tag{7}$$

We now further assume that for $x \in M \setminus S$, $\Delta_{L,L^+}(x_i,y_i)$ is small. Indeed, given the partition we do not expect L^+ to improve classification of elements of $M \setminus S$. Hence,

$$s'_f(P) = arg\max_S \sum_S \Delta_{L,L^+}(x_i,y_i)N_i \tag{8}$$

Moreover the new learner is expected to have a very small loss error on S,

$$\forall S,\ \sum_S \Delta_{L,L^+}(x_i,y_i))N_i \approx \sum_S E_{Y|X}[C(f_L(x_i),y_i)]N_i \tag{9}$$

Finally,

$$s'_f(P) = arg\max_{S\subset M} \sum_S E_{Y|X}[C(f_L(x_i),y_i)]N_i \tag{10}$$

The idea behind this formulation is to select the most ambiguous samples spread over the distribution of x.

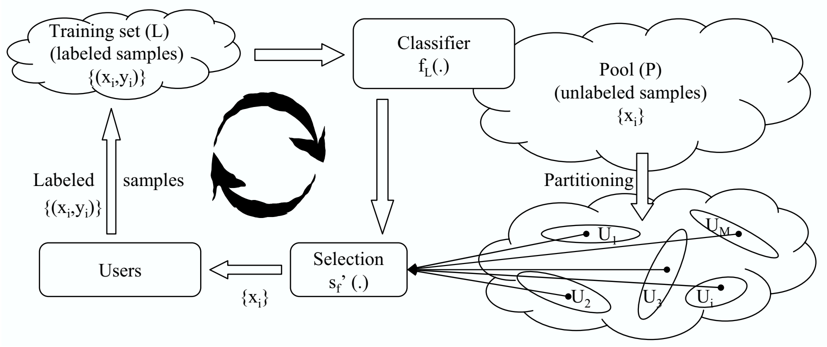


Figure 5: **Partition sampling strategy principle.** Before the selection, the pool is partitioned to select useful samples spread over the data set.

5.2 How to create partitions?

In practice, we propose to create a partition of the pool thanks to clustering techniques. The partition can either be created once at the beginning or at each iteration. In the experimental section, we will have a look at these two possibilities as well as the partition size that is required. We propose to use the well-known k-means algorithm for its simplicity and efficiency to create necessary partitions.

Once a partition is computed on the pool, equation 10 is used to select representative elements of each set of the partition, i.e we select the most ambiguous element per cluster. Finally the set S of samples to be labeled is composed of the n most relevant representatives.

6 Experimental results

Here, experimental results are presented. The first section deals with the partition sampling selection strategy methods and their parameters. Both approaches presented in section 5.2 to create the partition will be investigated. The second section compares the partition sampling selection strategy to the basic approach

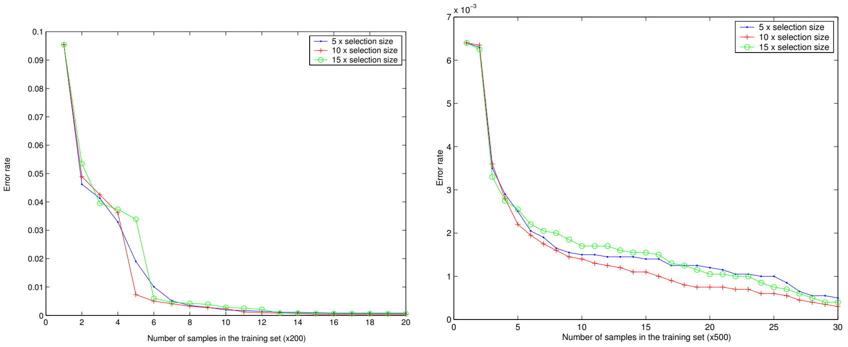
6.1 Evaluation of partitions

The proposed approach to create partitions is based on the k-means algorithm. As most algorithms for partitioning, it requires the number of desired clusters on input parameter. In following experiments we consider that partition sizes are relative to the number of samples to be selected. Thus, we define the partitioning factor f such that:

$$(\text{the size of the partition}) = f \times (\text{the number of selected samples})$$

Figure 6 shows the effect of this factor on the first synthetic data set and the Video-TREC set on the *weather* feature. The value of the partitioning factor has an impact on performances. However, we note that performance variations are small and thus the partitioning factor can be empirically selected. For next experiments, the partitioning factor is set to 10.

Figure 7 shows the performance differences when the partition is create once at the beginning or at each iteration. Partitioning at each iteration provides slightly better

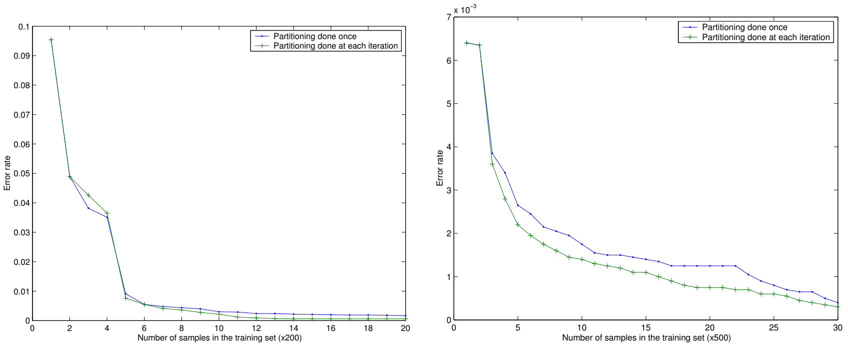


(a) First synthetic data set

(b) Video-TREC data set (weather feature)

Figure 6: **How to select the number of partitions?** Experiments to illustrate the impact of the partition size on performances.

error rates. However, the gain is not worth the computation requirements. In the current system, the partition can thus be created once at the beginning without altering the error rate evolution. In systems that need very accurate classification, a compromise is to compute the partition every N iterations of the active learning.



(a) First synthetic data set

(b) Video-TREC data set (weather feature)

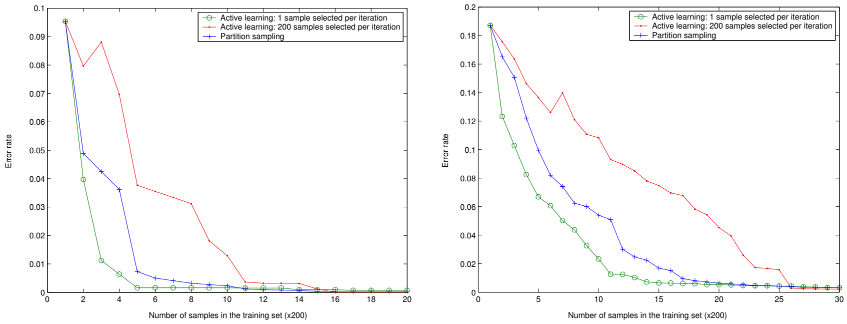
Figure 7: **When are partitions created?** Once at the beginning or at each iteration?

6.2 Evaluation

Figure 8 compares the different approaches when increasing the number of selected samples per iteration on both synthetic data sets introduced in section 4.1. As expected, uncertainty sampling has its performances drastically decreased, as explained in section 4.4, if many samples are selected at each iteration. The partition sampling strategy is then a good method to greatly reduce this side-effect of traditional active learning algorithm. Furthermore its performance are close to the objective, i.e. the performance of active learning with a selection of one sample per iteration. The mathematical framework presented in section 5.1 is then well adapted to the problem and provides a good solution.

Partition sampling provides more advantages than traditional active learning with similar performances. First of all, users are involved in less iterations and annotations. The annotation can also be shared among many users. Finally, we can reserve more computational power between rounds to find optimal elements since we do expect users to have a rest between labeling rounds.

Figure 9 presents results on the Video-TREC database introduced in section 4.2. We draw the same conclusion: the partition sampling strategy allows to select many samples at each iteration without a major impact on the error rate evolution.



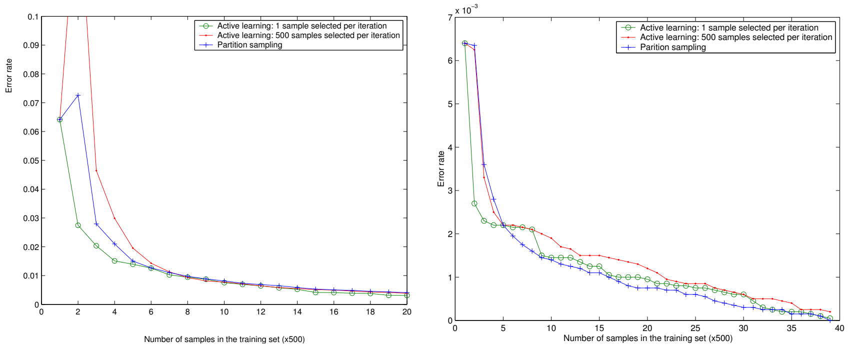
(a) First synthetic data set

(b) Second synthetic data set

Figure 8: **Active learning and partition sampling on synthetic data.** Comparison of the partition sampling and uncertainty sampling strategies.

7 Multi-labeling Case

Focusing the effort on reducing the number of samples to annotate is a first step to reduce the annotation effort. Another important issue is the annotation of samples with multiple labels. In most situations, users are required to attribute many labels to a given multimedia content. For example, 133 items were selected to annotate video shots of



(a) Studio Video-TREC feature

(b) Weather Video-TREC feature

Figure 9: **Active learning and partition sampling on Video-TREC data.** Comparison of the partition sampling and uncertainty sampling strategies.

Video-TREC database. Therefore it is becoming important for active learning systems to deal with such requirements.

In the literature, few systems propose to learn multiple labels simultaneously [17, 16]. The basic solution is to compute the mean utility over binary classifiers. Unfortunately, these systems also loose their efficiency when many samples are selected per iteration. We, then, propose to study the effect of the novel partition sampling strategy on performances. We expect that this selection strategy will help to maintain performances at their maximum.

The k-nearest neighbors classifier presented in section 3.3 is easily extended to multi-class problems. Furthermore the complexity of the classification is not changed since the neighborhood does not depend on sample classes. Let K be the number of classes and $y = \{y^k\}, k = 1,..,K \in Y^K$.

For the label k, the hypothesis f_L^k is defined as:

$$f_L^k(s) = \frac{\sum_{N_s} sim(s,n_i) * y_{n_i}^k}{\sum_{N_s} sim(s,n_i)}$$

$$\text{where } sim(s,n_i) = cosine(s,n_i)$$

The estimated label of s is then:

$$\hat{y}_s^k = arg\min_y C(f_L^k(s), y^k)$$

$$C(u,v) = \|u - v\|$$

From equation 5, we define a new selection function when many labels are involved in active learning:

$$s_f(P) = arg \max_{x \in S} \sum_{y^k} C(f_L^k(x), \hat{y}^k) \tag{11}$$

And for partition sampling equation 10 becomes:

$$s_f(P) = arg \max_{S \subset M} \sum_{S} \sum_{y^k} C(f_L^k(x_i), \hat{y}^k) N_i \tag{12}$$

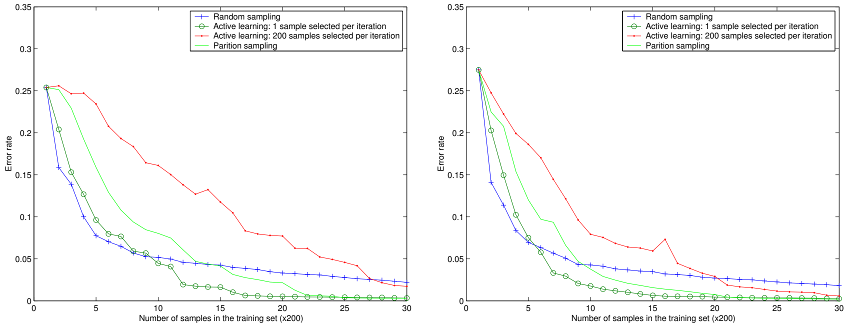
Finally, virtual users are asked to label selected samples at each iterations.

Figure 10 shows the behavior of described systems when dealing with multiple labels in a synthetic environment. Active learning allows to efficiently annotate samples with multiple labels. The problem which arises when selecting more samples per single round still remains: performances drastically decrease. Partition sampling strategy is a new solution to keep good performances even when selecting hundreds of samples per round.

Figure 11 which show system performances on real data, illustrates the problem that arises when labels are uncorrelated. In that case different samples are required to train all classes resulting in a selection sequence close to random. With *weather news* and *studio settings* concepts, active learning still allows to save a lot annotation effort (see figure 11(a)). While training five concepts simultaneously (*vegetation*, *studio settings*, *weather news*, *face* and *physical violence*) reduces consequently the benefits of active leaning (see figure 11(b)). This behavior can be explained by the fact that the correlation between labels is less strong when more labels are introduced.

8 Conclusion

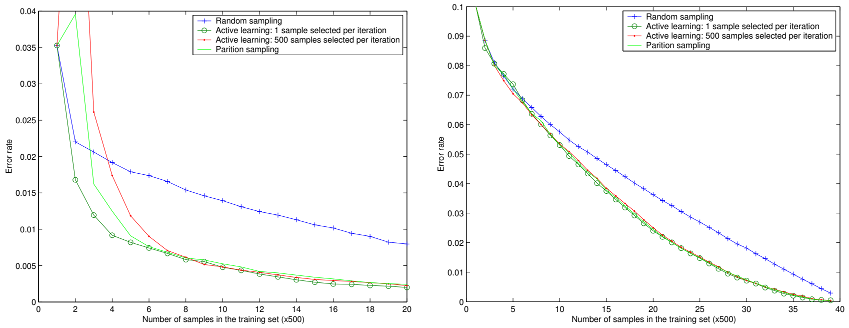
We proposed a new selection strategy, named partition sampling, that allows to build a set of optimal query samples to be annotated. The set may then be shared among users in a collaborative work to efficiently annotate complex and numerous contents



(a) Two labels

(b) Five labels

Figure 10: **Active learning for the annotation with multiple labels: synthetic data.**



(a) Two labels

(b) Five labels

Figure 11: **Active learning for the annotation with multiple labels: Video-TREC data.**

that require many examples. In the context of a single user, it simply reduces the time spend by the annotator. An initial mathematical framework was set up to introduce active learning. Then, we proposed a new mathematical framework for the partition sampling strategy. We presented experimental results on both synthetic data and the real problem of video database annotation. First results illustrated the problem of classical approaches when selecting many samples per iteration: the error rate decreases slowly. Secondly, we presented the performances of the partition sampling strategy that allows to efficiently select more samples per iteration. These experiments allowed to validate our mathematical framework. The partition sampling approach outperformed random sampling and almost reached its optimal learning sequence. Finally we tackled the more realistic and challenging task of multiple label annotation and raised an issue concerning the correlation between classes on active learning performances.

Future work will involve the improvement of selection strategies and the partitioning to achieve better performances, i.e. closer to the optimal selection strategy. Then, we will concentrate on the annotation task of multiple labels.

References

[1] S.-F. Chang, W. Chen, H. Meng, H. Sundaram, and D. Zhong, “A fully automated content-based video search engine supporting spatiotemporal queries,” in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, 1998, pp. 602– 615.

[2] M. Naphade, T. Kristjansson, B. Frey, and T. Huang, “Probabilistic multimedia objects (multijects): a novel approach to video indexing and retrieval,” in *IEEE International Conference on Image Processing*, vol. 3, 1998, pp. 536–540.

[3] A. Ekin, A. M. Tekalp, and R. Mehrotra, “Automatic soccer video analysis and summarizatio,” *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 796–807, 2003.

[4] V. Mezaris, I. Kompatsiaris, N. V. Boulgouris, and M. G. Strintzis, “Real-time compressed-domain spatiotemporal segmentation and ontologies for video indexing and retrieval,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 606–621, 2004.

[5] M. R. Naphade, I. V. Kozintsev, and T. S. Huang, “Factor graph framework for semantic video indexing,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 1, pp. 40–52, 2002.

[6] “Information technology - multimedia content description interface - part 5: Multimedia description schemes,” ISO/IEC 15938-5, 2003.

[7] “TRECVID: Digital video retrieval at NIST,” http://www-nlpir.nist.gov/projects/trecvid/.

[8] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell, “Text classification from labeled and unlabeled documents using EM,” *Machine Learning*, vol. 39, no. 2/3, pp. 103–134, 2000.

[9] I. Muslea, S. Minton, and C. Knoblock, “Active + semi-supervised learning = robust multi-view learning,” in *Proceedings of the 19th International Conference on Machine Learning*, 2002, pp. 435–442.

[10] A. McCallum and K. Nigam, “Employing em and pool-based active learning for text classification,” in *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.

[11] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, “Selective sampling using the query by committee algorithm,” *Machine Learning*, vol. 28, pp. 133–168, 1997.

[12] H. Seung, M. Opper, and H. Sompolinsky, “Query by committee,” in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 1992.

[13] M. I. J. David A. Cohn, Zoubin Ghahramani, “Active learning with statistical models,” *Journal of Artificial Intelligence Research*, vol. 4, pp. 129–145, 1996.

[14] M. Lindenbaum, S. Markovitch, and D. Rusakov, “Selective sampling for nearest neighbor classifiers,” *Machine Learning*, vol. 54, no. 125-152, February 2004.

[15] S. Tong and E. Chang, “Support vector machine active learning for image retrieval,” in *ACM Internation Conference on Multimedia*, 2001, pp. 107–118.

[16] T. C. Cha Zhang, “An active learning framework for content-based information retrieval,” in *IEEE Transactions on Multimedia*, vol. 4, 2002, pp. 260–268.

[17] A. H. Rong Yan, Jie Yang, “Automatically labeling video data using multi-class active learning,” in *IEEE International Conference on Computer Vision*, 2003.

[18] F. Souvannavong, B. Merialdo, and B. Huet, “Partition sampling for active video database annotation,” in *Proceedings of the 5th International Workshop on Image Analysis for Multimedia Interactive Services*, 2004.

[19] C.-Y. Lin, B. L. Tseng, and J. R. Smith, “Video collaborative annotation forum: Establishing ground-truth labels on large multimedia datasets,” in *Proceedings of the TRECVID 2003 Workshop*, 2003.

[20] C. Carson, M. Thomas, and S. Belongie, “Blobworld: A system for region-based image indexing and retrieval,” in *Third internation conference on visual information systems*, 1999.

[21] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, “An effective region-based image retrieval framework,” in *ACM Multimedia*, 2002.

[22] F. Souvannavong, B. Merialdo, and B. Huet, “Latent semantic analysis for an effective region-based video shot retrieval system,” in *International Workshop on Multimedia Information Retrieval*, 2004.

[23] F. Souvannavong, B. Merialdo, and B. Huet, “Video content modeling with latent semantic analysis,” in *Third International Workshop on Content-Based Multimedia Indexing*, 2003.

[24] F. Souvannavong, B. Merialdo, and B. Huet, “Latent semantic analysis for semantic content detection of video shots,” in *International Conference on Multimedia and Expo*, 2004.

[25] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, “Indexing by latent semantic analysis,” *Journal of the American Society of Information Science*, vol. 41, no. 6, pp. 391–407, 1990.