Active Learning Strategies: a case study for detection of emotions in speech

Alexis Bondu, Vincent Lemaire, and Barbara Poulain

R&D France Telecom,
TECH/EASY/TSI
2 avenue Pierre Marzin 22300 Lannion

Abstract. Machine learning indicates methods and algorithms which allow a model to learn a behavior thanks to examples. Active learning gathers methods which select examples used to build a training set for the predictive model. All the strategies aim to use the less examples as possible and to select the most informative examples. After having formalized the active learning problem and after having located it in the literature, this article synthesizes in the first part the main approaches of active learning. Taking into account emotions in Human-machine interactions can be helpful for intelligent systems designing. The main difficulty, for the conception of calls center's automatic shunting system, is the cost of data labeling. The last section of this paper propose to reduce this cost thanks to two active learning strategies. The study is based on real data resulting from the use of a vocal stock exchange server.

1 Introduction

Active learning methods come from a parallel between active educational methods and learning theory. The learner is from now a statistical model and not a student. The interactions between the student and the teacher correspond to the opportunity (possibility) to the model to interact with a human expert. The examples are situations used by the model to generate knowledge on the problem.

Active learning methods allow the model to interact with its environment by selecting the more "informative" situations. The purpose is to train a model which uses as litle as possible examples. The elaboration of the training set is done in interaction with a human expert to maximize progress of the model. The model must be able to detect the more informative examples for its learning and to ask to the expert: "what should be done in these situations".

The purpose of this paper is to present two main active learning approaches found in the state of the art. These approaches are presented in a generic way without considered a kind of model (the one which learns using examples delivered by the expert after every of its requests). Others approaches exist but they are not presented in this paper although references are given for the reader who would be interested in.

The next section of this paper introduce the topic, formalize active learning in a generic way and establish mathematical notations used. The aim of this section is to place active learning among others statistical learning methods (supervised, unsupervised...). The fourth section presents in details two main active learning approaches. These two strategies are then used in the fifth section on a real problem. Finally the last section is a discussion on question open in this paper.

2 Active Learning

2.1 General remarks

The objective of statistical learning (unsupervised, semi-supervised, supervised[1]) is to "inculcate" a behavior to a model using observations (examples) and a learning algorithm. The observations are points of view on the problem to be resolved and constitute the learning data. At the end of the training stage the model has to generalize its learning to unseen situations in a "reasonable" way.

For example let's imagine a model which try to detect "happy" and "unhappy" people from passport photo. If the model realizes good predictions for unseen people during its training stage then the model correctly generalize.

Characteristics of used data change depending on the learning mode. Unsupervised learning is a method of machine learning where a model is fit to observations. It is distinguished from supervised learning by the fact that there are no a priori outputs on data. The learner has to discover itself correlations between examples which are shown to it. In case of the example above ("happy" / "unhappy" people), the model is trained using passport photos deprive of label and has no indication on what we try to make it learn. Among unsupervised learning methods one finds clustering methods [2] and association rules methods [3].

Semi-supervised learning [4] is a class of techniques that makes use of both labeled and unlabeled data for training; typically a small amount of labeled data and a large amount of unlabeled data. Among possible utilization of this learning mode, we could distinguish (i) semi-supervised clustering which tries to group similar instances but using information given by the small amount of labeled data [5] and (ii) semi-supervised classification [6] which is based first on labeled data to elaborate a first model and then unlabeled data to improve the model.

Supervised learning is a machine learning technique for creating a function from training data. The training data consist of pairs of input objects (typically vectors), and desired outputs. The output of the function can be a continuous value (called regression), or can predict a class label of the input object (called classification). The task of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples (i.e. pairs of input and target output). In case of the illustrative example above, examples would be passport photos associated to labels "happy" or "unhappy".

[1] Reinforcement learning is not presented here, reader interested could read [1]

At last, active learning, as the name suggests, is a type of learner which is less passive than the others described above. This strategy allows the model to construct its own training set in interaction with a human expert. The learning starts with few desired outputs (class labels for classification or continuous value for regression). Then, the model selects examples (without desired outputs) that it considers the more informative and asks to the human expert their desired outputs. In case of the illustrative example, the model asks class labels of passport photos presented to the human expert. In this paper, we restrict active learning to classification but it is obvious that our presentation of active learning strategies can be transpose for regression.

Active learning is different from all others learning methods because it interacts with its environment; the examples are not randomly chosen. Active learning strategies allow the model to learn faster (the learner rich the best performances using less data) considering first the more informative examples. This approach is more specifically attractive when data are expensive to obtain or to label.

2.2 Two possible scenarios

The distinction between raw data and data descriptors (which are associated) is important. In the illustrative example above, the raw data are passport photos and the data descriptors are attributes describing the photos (pixel, luminosity, contrasts, etc.). The model makes the prediction of the class "happy" or "unhappy" to every vector of descriptors. The elaboration of descriptors from raw data is not always bijective; sometimes it is impossible to compose raw data using list of descriptors. Adaptive sampling and selective sampling, which are the two main scenarios to set active learning [7], use respectively "data descriptors" and "raw data".

In the case of **adaptive sampling** [8] the model requires of expert labels corresponding to vectors of descriptors. The model is not restricted and can explore all the space of variations of the descriptors, searching area to be sampled more finely. Adaptive sampling can pose problem in its implementation when it is difficult to know if the vectors of descriptors (generated by the model) have a meaning with respect to the initial problem. Let us suppose that the model requires the label associated with the vector $[10, 4, 5..., 12]$. Does this vector correspond to a set of descriptors which represent a passport photo, a human face photo, a flower or something else?

In the case of **selective sampling** [9], the model observes only one restricted part of the universe materialized by training examples stripped of label. Consequently, the input vectors selected by the model always correspond to a raw data. The image of a *"bag"*" of instances for which the model can ask labels associated (to the examples in the bag) is usually used. The model requires the label associated with the vector $[10, 4, 5..., 12]$ which corresponds to a passport photo.

Emotion detection is a problem where it is easy to obtain a great number of unlabeled examples and for which labeling is expensive. Therefore, from now the point of view of selective sampling is only considered. In practice, the choice of selective or adaptive sampling depends primarily on the applicability where the model is authorized, or not, "to generate" new examples.

2.3 Notations

$\mathcal{M} \in \mathbb{M}$ is the predictive model which is trained thanks to an algorithm $\mathcal{L}$. $\mathbb{X} \subseteq \mathbb{R}^n$ represents all the possible input examples of the model and $x \in \mathbb{X}$ is a particular examples. $\mathbb{Y}$ is the set of the possible outputs (answers) of the model; $y \in \mathbb{Y}$ a class label[2] related (associated) to $x \in \mathbb{X}$.

During its training, the model observes (see Figure 1) only one part $\Phi \subseteq \mathbb{X}$ of the universe. The set of examples is limited and the labels associated to these examples are not necessarily known. The set of examples where the labels are known (at a step of the training algorithm) is called L_x and the set of examples where the labels are unknown is called U_x with $\Phi \equiv U_x \cup L_x$ and $U_x \cap L_x \equiv \emptyset$.

The concept which is learned can be seen as a function, $f : \mathbb{X} \to \mathbb{Y}$, with $f(x_1)$ is the desired answer of the model for the example x_1 and $\widehat{f} : \mathbb{X} \to \mathbb{Y}$ the obtained answer of the model; an estimation of the concept. The elements of L_x and the associated labels constitute a training set T. The training examples are pairs of input vectors and desired labels such as $(x, f(x)) : \forall x \in L_x, \exists!(x, f(x)) \in T$.

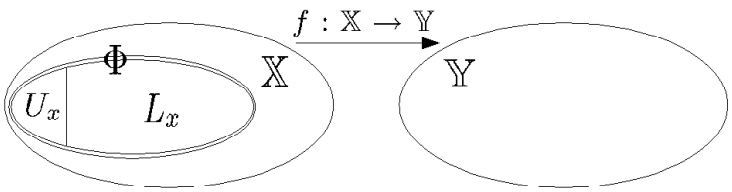

Fig. 1. Notations

[2] The word "label" is used here for a discrete value in classification problems or a continuous value in regression problems.

3 Active Learning Methods

3.1 Introduction

The problem of selective sampling was posed formally by Muslea [10] (see Algorithm 1). It uses an utility function, $\mathcal{U}tility(u, \mathcal{M})$, which estimates the utility of an example u for the training of the model $\mathcal{M}$. Thanks to this function, the model presents to the expert examples for which it hopes the greatest improvement of its performances.

The Algorithm 1 is generic insofar as only the function $\mathcal{U}tility(u, \mathcal{M})$ must be modified to express a particular active learning strategy. How to measure the interest of an example will be discuss now.

Considering:

- $\mathcal{M}$ a predictive model provided with a training algorithm $\mathcal{L}$
- U_x et L_x the sets of examples respectively not labeled and labeled
- n the desired number of training examples
- T the training set with $\|T\| < n$
- $\mathcal{U} : \mathbb{X} \times \mathbb{M} \rightarrow \Re$ the function which estimates the utility of an example for the training of the model

Repeat
(A) Train the model $\mathcal{M}$ thanks to $\mathcal{L}$ and T (and possibly U_x).
(B) Look the example such as $q = argmax_{u \in U_x} \mathcal{U}(u, \mathcal{M})$
(C) Withdraw q of U_x and ask the label $f(q)$ to the expert.
(D) Add q to L_x and add $(q, f(q))$ to T
until $\|T\| < n$

Algorithm 1: Selective sampling, Muslea 2002

3.2 Uncertainty sampling

Uncertainty sampling is an active learning strategy [11, 12] which is based on confidence that the model has in its predictions. The model used must be able to estimate the reliability of its answers, to provide the probabilities, y_j, to observe each class (j) for an examples u. Thus the model can make a prediction choosing the most probable class for u. The choice of new examples to be labeled proceeds in two steps:

- the model available at the iteration t is used to predict the labels of the unlabeled examples;
- examples with the more uncertain prediction are selected.

The uncertainty of a prediction can also be defined using a threshold of decision. For example (see Figure 2) if the model gives answers between 0 and 1

a threshold is defined to take a decision and say which examples will be classified 0 and those which will be classified 1. The closer an answer of the model is to the threshold of decision, the more uncertain is the decision.

This first approach has the advantage to be intuitive, easy to implement and fast. The uncertainty sampling shows its limits however when the problem to be solved is not separable by the model. Indeed, this strategy will tend to select the examples to be labeled in mixture zones, where there is nothing any more to learn.

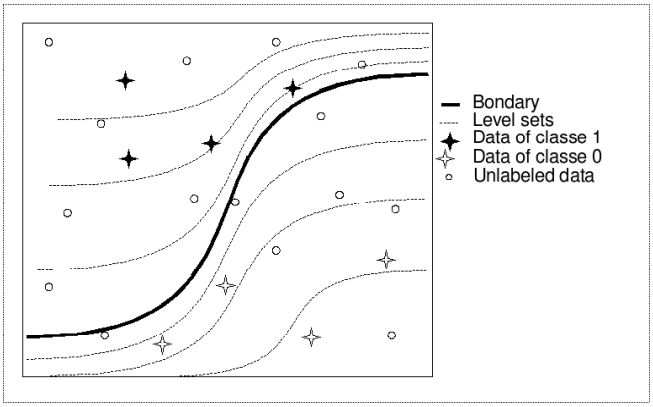

Fig. 2. A binary classification problem: the boundary plotted represents the threshold of decision. Level sets are plotted too and their distances from the boundary lines indicate the uncertainty. Unlabeled data close to the boundary line are the more uncertain and will be selected to be labeled by the expert.

3.3 Risk reduction

The purpose of this approach is to reduce the generalization error, $E(\mathcal{M})$, of the model [13]. It chooses examples to be labeled so as to minimize this error. In practice this error cannot be calculated because the distribution of the examples, $\mathbb{X}$, is unknown. However it can be write, at an iteration t, using a loss function ($\mathcal{L}oss(\mathcal{M}^t, x)$) which evaluates the error of the model for a given example $x \in \mathbb{X}$ such as:

$$E(\mathcal{M}^t) = \int_{\mathbb{X}} \mathcal{L}oss(\mathcal{M}^t, x) P(x) dx$$

The same model at the next iteration $t+1$ is defined as: $\mathcal{M}^{t+1}_{(x^\diamond, y^\diamond)}$. This model takes into account a new training example: $(x^\diamond, y^\diamond)$. For real problems, the output of the model $y^\diamond$ is unknown since $x^\diamond$ is a not labeled data. To estimate the generalization error at $t+1$, all the possibilities of the $\mathbb{Y}$ set have to be considered and to be balanced using their probability to be observed. The generalization

error expected is therefore:

$$E(\mathcal{M}_{x^\diamond}^{t+1}) = \int_{\mathbb{X}} \int_{\mathbb{Y}} P(y|x^\diamond) \mathcal{L}oss(\mathcal{M}_{(x^\diamond,y)}^{t+1}, x) P(x) dx dy$$

This strategies selects the example q which minimizes $E(\mathcal{M}_{x^\diamond}^{t+1})$. Once labeled, this example is incorporated to the training set. Step by step this procedure tries to elaborate an optimal training set.

Nicholas Roy [9] shows how to bring this strategy into play since all the elements of $\mathbb{X}$ are not known. He uses an uniform prior for $P(x)$ which gives :

$$\widehat{E}(\mathcal{M}^t) = \frac{1}{\|L_x\|} \sum_{i=1}^{\|L_x\|} \mathcal{L}oss(\mathcal{M}^t, x_i)$$

This strategy, where different loss functions can be used, is summarized in the algorithm 2. The model is, for all examples i, trained several times ($||\mathbb{Y}||$ times), to estimate $\widehat{E}(\mathcal{M}_{(x_i,y_j)}^{t+1})$. The example i which minimizes the expected loss function ($\widehat{E}(\mathcal{M}_{(x_i)}^{t+1})$) will be incorporated in the training set.

Considering:

- $\mathcal{M}$ a predictive model provided with a training algorithm $\mathcal{L}$
- U_x and L_x the sets of examples respectively not labeled and labeled
- n the desired number of training examples
- T the training set with $\|T\| < n$
- $\mathbb{Y}$ the label set which can be given to the examples of U_x
- $\mathcal{L}oss : \mathbb{M} \rightarrow \Re$ the generalization error
- $\mathcal{E}rr : U_x \times \mathbb{M} \rightarrow \Re$ the expected generalization error for the model $\mathcal{M}$ trained with an additional example, $T \cup (x_i, f(x_i))$

Repeat
  (A) Train the model $\mathcal{M}$ thanks to $\mathcal{L}$ and T
  For all examples $x_i \in U_x$ **do**
    For all label $y_j \in \mathbb{Y}$ **do**
      i) Train the model $\mathcal{M}_{i,j}$ thanks to $\mathcal{L}$ and $(T \cup (x_i, y_j))$
      ii) Compute the generalization error $\widehat{E}(\mathcal{M}_{(x_i,y_j)}^{t+1})$
    end For
    Compute the generalization error
    $\widehat{E}(\mathcal{M}_{x_i}^{t+1}) = \sum_{y_j \in \mathbb{Y}} \widehat{E}(\mathcal{M}_{(x_i,y_j*)}^{t+1}).P(y_j|x_i)$
  end For
  (B) Look for the example $q = argmin_{u \in U_x} \widehat{E}(\mathcal{M}_{x_i}^{t+1})$
  (C) Withdraw q of U_x ans ask the label $f(q)$ to the expert.
  (D) Add q to L_x and add $(q, f(q))$ to T
until $\|T\| < n$

Algorithm 2: Apprentissage actif *"optimal"*, de Nicholas Roy 2000

A example of use of this strategy is presented in [14] where X. Zhu estimates the generalization error ($E(\mathcal{M})$) using the empirical risk:

$$\widehat{E}(\mathcal{M}) = R(\mathcal{M}) = \sum_{n=1}^{N} \sum_{y_j \in \mathbb{Y}} \mathbb{1}_{\{f(l_n) \neq y_j\}} P(y_j|l_n) P(l_n) \text{ with } l_n \in L_x$$

where f is the model which estimates the probability that an example belong to a class (a Parzen window [15] in [14]), $P(y_i|l_n)$ the real probability to observe the class y_i for the example $l_n \in L_x$, $\mathbb{1}$ the indicating function equal to 1 if $f(l_n >) \neq y_i$ and equal to 0 if not. Therefor $R(\mathcal{M})$ is the sum of the probabilities that the model makes a bad decision on the training set (L_x).

Using an uniform prior to estimate $P(l_n)$:

$$\hat{R}(\mathcal{M}) = \frac{1}{N} \sum_{n=1}^{N} \sum_{y_j \in \mathbb{Y}} \mathbb{1}_{\{f(l_n) \neq y_j\}} \hat{P}(y_j|l_n)$$

The expected cost for any single example u ($u \in U_x$) added to the training set (for binary classification problem) is then:

$$\hat{R}(\mathcal{M}^{+u}) = \sum_{y_j \in \mathbb{Y}} \hat{P}(y_j|u) \hat{R}(\mathcal{M}^{+(u,y_j)}) \text{ with } u \in U_x$$

3.4 Discussion

Both strategies described above are not the only ones which exist. The reader can see a third main strategy which is based on Query by Committee [16, 17] and a fourth one where authors focus on a model approach to active learning in a version-space of concepts [18–20].

4 Application of active learning to detection of emotion in speech

4.1 Introduction

Thanks to recent techniques of speech processing, many automatic phone call centers appear. These vocal servers are used by customers to carry out various tasks conversing with a machine. Companies aim to improve their customer's satisfaction by redirecting them towards a human operator, in the event of difficulty. The shunting of unsatisfied users amounts detecting the negative emotions in their dialogues with the machine, under the assumption that a problem of dialogue generates a particular emotional state in the subject.

The detection of expressed emotions in speech is generally considered as a supervised learning problem. The detection of emotions is limited to a binary classification since taking into account more classes rises problem of the objectivity of labeling task [21]. The acquisition and the labeling of data are expensive in this framework. Active learning can reduce this cost by labeling only the examples considered to be informative for the model.

4.2 Characterization of data

This study is based on a previous work [22] which characterizes vocal exchanges, in optimal way, for the classification of expressed emotions in speech. The objective is to control the dialogue between users and a vocal server. More precisely, this study deals the relevance of variables describing data, according to the detection of emotions.

The used data result from an experiment involving 32 users who test a stock exchange service implemented on a vocal server. According to the users point of view, the test consists in managing a virtual wallet of stock options, the goal is to realize the strongest profit. The obtained vocal traces constitute the corpus of this study: 5496 "speech turns" exchanged with the machine. Speech turns are characterized by 200 acoustic variables, describing variations of the sound intensity, variations of voice height, frequency of elocution... etc. Data are also characterized by 8 dialogical variables describing the rank of a speech turn in a given dialogue (a dialogue contains several speech turn), the duration of the dialogue... Each speech turn is manually labeled as carrying positive or negative emotions.

The subset of the most informative variables with respect to the detection of expressed emotions in speech is given thanks to a naive Bayesian selector [23]. At the beginning of this process (the selection of the most informative variables), the set of attributes is empty. The attribute which most improves the predictive quality of the model is then added at each iteration. The algorithm stops when the addition of attributes does not improve any more the quality of the model. Finally, 20 variables were selected to characterize vocal exchanges. In this article, used data result from the same corpus and from this previous study. So, every speech turnis characterized by 20 variables.

4.3 The choice of the model

The large range of models able to solve classification problems (and sometimes the great number of parameters useful to use them) may represent difficulties to measure the contribution of a learning strategy. A Parzen window, with a Gaussian kernel [15], is used in experiments below since this predictive model uses a single parameter and is able to work with few examples. The "output" of this model is an estimate of the probability to observe the label y_j conditionally to the instance u:

$$\hat{P}(y_j|u) = \frac{\sum_{n=1}^{N} \mathbb{1}_{\{f(l_n)=y_j\}} K(u,l_n)}{\sum_{n=1}^{N} K(u,l_n)} \qquad avec\, l_n, \in L_x \; et \; u \in U_x \cup L_x \qquad (1)$$

where

$$K(u,l_n) = e^{\frac{||u-l_n||^2}{2\sigma^2}}$$

The optimal value (σ^2=0.24) of the kernel parameter was found thanks to a cross-validation on the average quadratic error, using the whole of available

training data [24]. Thereafter, this value is used to fix the Parzen window parameter. The results obtained by this model (using the whole of training data) are similar with the previous results obtained by a naive Bayesian classifier [22]. Consequently, Parzen windows are considered satisfying and valid for the following active learning procedures. Kernel methods and closer neighbors methods are usually used in classification of expressed emotions in speech [25].

The model must be able to assign a label $\hat{f}(u)$ to an input data u, so a decision threshold noted $\mathcal{T}h(L_x)$ is calculated at each iteration. This threshold minimizes the error of the model[3] on the available training set. The label attributed is $\hat{f}(u_n) = 1$ if $\{\hat{P}(y_1|u_n) > \mathcal{T}h(L_x)\}$, else $\hat{f}(u_n) = 0$. Since the single parameter of the Parzen window is fixed, the training stage is reduced to count instances (within the meaning of the Gaussian kernel). The strategies of examples selection, without being influenced by the training of the model, are thus comparable.

4.4 Used Active Learning strategies

Two Active learning strategies are considered in this paper; the active learning strategy which tries to reduce the generalization error of the model and the strategy consists in selecting the instance for which the prediction of the model is most uncertain have been tested.

For the first strategy the Parzen window estimates $P(y_i|l_n)$. The empirical risk is approximated adopting a uniform a priori on the $P(l_n)$. The purpose is to select the unlabeled instance $u_i \in U_x$ which will minimize the risk of the next iteration. $R(\mathcal{M}^{+u_n})$ the *"expected"* risk resulting from the labeling of the instance u_n (iteration $t+1$) is estimated. Available labeled data are used to do this estimation when the assumption $f(u_n) = y_1$ ***[resp*** $f(u_n) = y_0$***]*** to estimate $\hat{R}(\mathcal{M}^{+(u_n,y_1)})$ ***[resp*** $\hat{R}(\mathcal{M}^{+(u_n,y_0)})$ ***]*** is done.

For the second strategy the *uncertainty* of a prediction is maximum when the output probability of the model approaches the decision threshold.

Apart from these two active strategies, a "stochastic" approach which uniformly selects the examples according to their probability distribution is considered. This last approach play a role of reference used to measure the contribution of the active strategies.

4.5 Results

The presented results come from several experiments on previous learning strategies. Each experiment has been done five times[4]. At the beginning of the experiments, the training set is only constituted by two examples (one positive and one negative) selected randomly. At each iteration, ten examples are selected to

[3] The used error measurement is the *"Balanced Error Rate"*, for more details see section 4.5

[4] the natches on the curves of the figure 3 correspond to 4 times the variance of the results ($\pm 2\sigma$).

be labeled and added to the training set. The considered classification problem, here, is unbalanced: there are 92% of "positive or neutral" emotions and 8% of "negative" emotions. To observe correctly the classification profits (when adding labeled examples), the model evaluation is done using the area under ROC curve (AUC) on the test set[5]. A ROC curve is calculated from the detection rate of a single class. Consecently, we use the sum of the AUCs weighted by reference class's prevalence in the data.

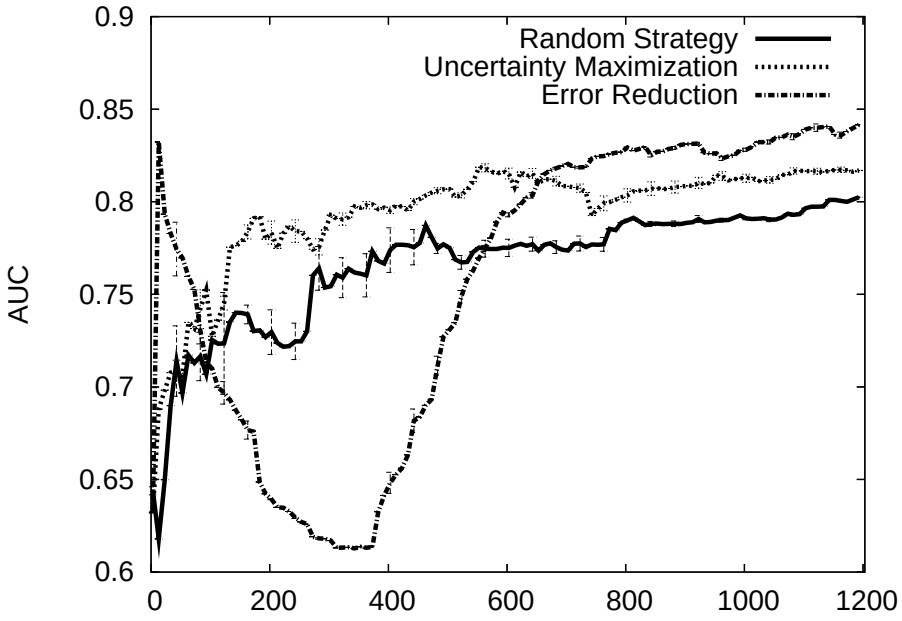

Fig. 3. Focus of the results on the test using [0:1200] training examples

The "risk reduction" is the strategy which maximizes the quality of the model for a number of training examples in the range [2:100]. Between 100 and 700 the strategy based on uncertainty wins. After 600 training examples the three strategies converge to the optimal AUC (Area Under Roc Curve).

The two active strategies allow obtaining faster than the random strategy the optimal result (the optimal AUC is 0.84 using the whole training set). The use of active learning is positive in this real problem. However the results obtained raise questions which will detailed in the next section.

[5] The test set include 1613 examples and the training set 3783 examples

5 Discussion and conclusion

This paper shows the interest of active learning for a field where acquisition and labeling of data are particularly expensive. Obtained results show that active learning is relevant for the detection of expressed emotions in speech. But whatever the strategy considered (even the two strategies evoked in section 3.4 but not derailed in this paper) several question exist and can be raised:

- **evaluation -** The quality of an active strategy is usually represented by a curve assessing the performance of the model versus the number of training examples labeled (see Figure 3). The performance criterion used can take several different ways according to the problem. This type of curve allows only comparisons between strategies in a punctual way, i.e. for a point on the curve (a given number of training examples). If two curves pass each other (as in the Figure 3, it is impossible to determine if a strategy is better than another (on the total set of training examples). The elaboration of a criterion which measures the contribution of a strategy compared to the random strategy on the whole data set should be interesting. This point will be discussed in a future paper.
- **test set -** Active learning strategies are, often, used when data acquisition is expensive. Therefore, in practice, a test set is not available (otherwise it can be used to the training) and the evaluation of the model during a strategy is difficult to implement.
- **stopping criterion -** The maximal number of examples to labeled, or an estimation of the progress of the model, can be used to stop the algorithm. This is very link to the use of a test set or the model employed. For example in the Figure 3 the strategy based on the risk gives the same results when 15 examples have been labeled than results using all the available data. In this case the cost using 15 examples and 600 will be not the same... A good criterion should be independent of the model and of a test set. Actual experiments (not yet published) will allow us to propose a criterion of this type at the end of 2007.
- **number of examples to be labeled** : the state of the art seems to incorporate an only one example at each step of the strategy. But in real case the expert is a human and when the model needs time to learn at each iteration of the strategy this could be not efficient. Sometimes more than one example must be incorporated. This aspect has been a litle bit studied in [26] but it has to be more analyse in the future.
- **uncertain environment -** If an answer could be given to the points above then active strategies could be used for on-line learning in an uncertain environment. For example to tag part of graph (graph here is social network). When writing this paper we hope that this point will accepted and incorporated in a proposition sent to a French Project (and financed by the French National Agency of Research (ANR)) grouping industrial and universities.

Generally, active learning strategies estimate the utility of training examples. These criteria could be used for on-line training. The training set would be

consisted of the N examples the more "useful" seen until now (with N fixed). This approach would be able to consider non stationary problems and it is able to train a model which adapts itself to the variations of the observed system.

For the detection of expressed emotions in speech could be treated by a double strategy reducing the cost linked to the data : (i) a variables selection allowing to preserve only the necessary and sufficient characteristics for classification; (ii) an examples selection allowing to preserve only useful instances for training. This will be explored in future work.

References

1. Harmon, M.: Reinforcement learning: a tutorial. http://eureka1.aa.wpafb.af.mil/rltutorial/ (1996)
2. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Computing Surveys **31**(3) (1999) 264–323
3. Jamy, I., Jen, T.Y., Laurent, D., Loizou, G., Sy, O.: Extraction de règles d'association pour la prédiction de valeurs manquantes. Revue Africaine de la Recherche en Informatique et Mathématique Appliquée ARIMA **Spécial CARI04** (2005) 103–124
4. Chapelle, O., Schölkopf, B., Zien, A.: Semi-Supervised Learning. MIT Press, Cambridge, MA (in press) (2006) `http://www.kyb.tuebingen.mpg.de/ssl-book/ssl_toc.pdf`.
5. Cohn, D., Caruana, R., McCallum, A.: Semi-supervised clustering with user feedback. Technical Report 1892, Cornell University (2003)
6. Chapelle, O., Zien, A.: Semi-supervised classification by low density separation. In Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (2005)
7. Castro, R., Willett, R., Nowak, R.: Faster rate in regression via active learning. In: NIPS (Neural Information Processing Systems), Vancouver (2005)
8. Singh, A., Nowak, R., Ramanathan, P.: Active learning for adaptive mobile sensing networks. In: IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks, New York, NY, USA, ACM Press (2006) 60–68
9. Roy, N., McCallum, A.: Toward optimal active learning through sampling estimation of error reduction. In: Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA (2001) 441–448
10. Muslea, I.: Active Learning With Multiple View. Phd thesis, University of southern california (2002)
11. Lewis, D., Gale, A.: A sequential algorithm for training text classifiers. In Croft, W.B., van Rijsbergen, C.J., eds.: Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval, Dublin, Springer Verlag, Heidelberg (1994) 3–12
12. Thrun, S.B., Möller, K.: Active exploration in dynamic environments. In Moody, J.E., Hanson, S.J., Lippmann, R.P., eds.: Advances in Neural Information Processing Systems. Volume 4., Morgan Kaufmann Publishers, Inc. (1992) 531–538
13. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. In Tesauro, G., Touretzky, D., Leen, T., eds.: Advances in Neural Information Processing Systems. Volume 7., The MIT Press (1995) 705–712

14. Zhu, X., Lafferty, J., Ghahramani, Z.: Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In: ICML (International Conference on Machine Learning), Washington (2003)
15. Parzen, E.: On estimation of a probability density function and mode. Annals of Mathematical Statistics **33** (1962) 1065–1076
16. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning **28**(2-3) (1997) 133–168
17. Seung, H.S., Opper, M., Sompolinsky, H.: Query by committee. In: Computational Learning Theory. (1992) 287–294
18. Dasgupta, S.: Analysis of greedy active learning strategy. In: NIPS (Neural Information Processing Systems), San Diego (2005)
19. Cohn, D.A., Atlas, L., Ladner, R.E.: Improving generalization with active learning. Machine Learning **15**(2) (1994) 201–221
20. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. In Langley, P., ed.: Proceedings of ICML-00, 17th International Conference on Machine Learning, Stanford, US, Morgan Kaufmann Publishers, San Francisco, US (2000) 999–1006
21. Liscombe, J., Riccardi, G., Hakkani-Tür, D.: Using context to improve emotion detection in spoken dialog systems. In: InterSpeech, Lisbon (2005)
22. Poulain, B.: Sélection de variables et modélisation d'expressions d'émotions dans des dialogues hommes-machine. In: EGC (Extraction et Gestion de Connaissance), Lille. + Technical Report avalaible here: http://perso.rd.francetelecom.fr/lemaire (in french) (2006)
23. Boullé, M.: An enhanced selective naive bayes method with optimal discretization. In Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L., eds.: Feature extraction, foundations and Application. Springer (August 2006) 499–507
24. Chappelle, O.: Active learning for parzen windows classifier. In: AI & Statistics, Barbados (2005) 49–56
25. Guide, V., Rakotomamonjy, Canu, S.: Méthode à noyaux pour l'identification d'émotion. In: RFIA (Reconnaissance des Formes et Intelligence Artificielle). (2003)
26. Bondu, A., Lemaire, V.: Etude de l'influence du nombre d'exemples à étiquetter dans une procédure d'apprentissage actif. submitted to CAP 2006 (Conference francophone sur l'apprentissage automatique)