Comparison of Active Learning Strategies and Proposal of a Multiclass Hypothesis Space Search

Davi P. dos Santos and André C.P.L.F. de Carvalho

Computer Science Department
Institute of Mathematics and Computer Sciences, University of São Paulo
São Carlos - SP, Brazil
{davips,andre}@icmc.usp.br

Abstract. Induction of predictive models is one of the most frequent data mining tasks. However, for several domains, the available data is unlabeled and the generation of a class label for each instance may have a high cost. An alternative to reduce this cost is the use of active learning, which selects instances according to a criterion of relevance. Diverse sampling strategies for active learning, following different paradigms, can be found in the literature. However, there is no detailed comparison between these strategies and they are usually evaluated for only one classification technique. In this paper, strategies from different paradigms are experimentally compared using different learning algorithms and datasets. Additionally, a multiclass hypothesis space search called SG-multi is proposed and empirically shown to be feasible. Experimental results show the effectiveness of active learning and which classification techniques are more suitable to which sampling strategies.

Keywords: machine learning, classification, active learning.

1 Introduction

Classification techniques are used in a large number of real problems, like face recognition, news filtering, spam detection and others. However, it is common to find applications which require handling of a huge amount of data. These data are frequently unlabeled and the assignment of a class to each instance may have a high cost. A promising way to selectively use this massive unlabeled data for the induction of classification models is by employing active learning, area dedicated to machines that evolve *by asking questions* [31]. One of the core questions is about the class/label of an instance.

It is precisely the means by which this question can be asked the subject of investigation in this document. The idea of selecting the best instances among others is not new, it has appeared in the literature under different perspectives, e.g *Design of the Experiment* [27]. In the context of classification, it dates back to at least the 1970's [26].

When labeled instances are needed to induce a classifier, it is reasonable to acquire labels for only the most important of them, since each label acquisition

M. Polycarpou et al. (Eds.): HAIS 2014, LNAI 8480, pp. 618–629, 2014.

has a cost. Depending on the application domain, this label acquisition process can be categorized in three major settings: *membership query synthesis*, which allows the learner to synthesize the most informative instance to ask for a label [2]; *stream-based query*, which requires immediate learner's decision about querying or discarding each instance that arrive from a stream; and *pool-based query*, formerly known as *selective sampling*, when the learner is given the freedom to choose the most informative instance among several others in a pool, which is the most common scenario and the focus of this work [18].

There are several successful strategies for the pool-based setting [21]. However, this variety poses the problem of choosing the most appropriate to a given task. The main contribution of this work is to empirically demonstrate the effectiveness of active learning and to confront strategies from different paradigms under the same experimental apparatus. The comparison includes an adaptation of one of the strategies to support multiclass problems.

The remainder of this paper is organized as follows. In Section 2, the most common niches of the pool-based query setting are reviewed; in Section 3 we present the experiments performed in this study and discuss about the results obtained; finally, the main conclusions and future directions are presented in Section 4.

2 Related Work

There are not many comprehensive comparative studies in the active learning literature. Those found are specific to strategies for a particular classification algorithm [29], intended for specific tasks [32] or focused on a single niche of strategies [16].

In the following sections, the most common active learning strategies under the pool-based setting are reviewed and experimentally compared: uncertainty sampling, hypothesis space search, expected error reduction, density-weighted sampling and cluster-based sampling. Additionally, *Expected Model Change* is presented, but not included in the experiments due to its incompatibility with the selected techniques. Similarly, *Query by Committee* is also presented, but excluded from the experiments to avoid unfair comparison of accuracies.

All paradigms will be presented along with their characteristic order of complexity referring to the number of (re)trainings needed to perform each query. In the following sections, the number of classes and the initial number of instances in the pool will be denoted by $|Y|$ and $|\mathcal{U}|$, respectively.

2.1 Uncertainty Sampling

Probably, the simplest informativity measure to decide when to select an instance (or a group of instances, in the original proposal) is the maximum posterior probability given by a probabilistic model [18]:

$$P_{max}(\boldsymbol{x}) = \arg\max_{y} P(y|\boldsymbol{x})$$

where $\boldsymbol{x}$ refers to an instance vector sampled from the pool, and $P(y|\boldsymbol{x})$ is the posterior conditional probability of $\boldsymbol{x}$ being of the class y. $P(y|\boldsymbol{x})$ is roughly equivalent, e.g. to the output of a probability-based model. The uncertainty sampling strategy consists of querying the most informative instance, i.e. the instance with the lowest $P_{max}(\boldsymbol{x})$, to explore the decision boundary in the attribute (or parameter) space. The maximum posterior can be substituted by others measures. A similar measure is the *margin* $M(\boldsymbol{x})$ between the two highest posterior probabilities. Given the second most probable class probability P_{2ndmax}, the margin $M(\boldsymbol{x})$ is defined as:

$$M(\boldsymbol{x}) = P_{max}(\boldsymbol{x}) - P_{2ndmax}(\boldsymbol{x})$$

Another measure, which is inversely related to the previous measures is the *Shanon entropy* [34], defined as:

$$E(\boldsymbol{x}) = -\sum_{y} P(y|\boldsymbol{x}) \log P(y|\boldsymbol{x})$$

These three measures depend on a probabilistic model. However, it is possible to roughly approximate such informativity measures or even probability distributions for other families of learning algorithms.

This strategy **requires only a single training** on the labeled instances for all candidates, having $\mathcal{O}(1)$ complexity (a single training per query).

2.2 Hypothesis Space Search

It is possible to perform active learning directly from the hypothesis space perspective. The rationale is to query the most controversial instances when different valid hypotheses are compared with each other, i.e. to query instances that would reduce the *version space* [23] after its inclusion in the training set. One way to search through the hypothesis space is to track the sets S and G of specific and general hypotheses during learning and consider only the most specific $h_S \in S$ and the most general $h_G \in G$ hypotheses.

One important feature of this family of strategies is its **binary decision model**: all instances for which the hypotheses disagree can be queried at once or in any arbitrary order, i.e. there is no precedence among them.

One of the first active learning algorithms is a *query by disagreement*, called *SG-network* [6]. It approximately induces specific/general models θ_S and θ_G by means of generating or sampling random "background" instances and labeling them artificially according to the desired training goal: specificity or generality. Instances are sampled from the region of disagreement between θ_S and θ_G.

The comparison performed in this work is delimited by the pool-based setting, independent on the learning algorithm and the number of classes. Therefore, to fit *SG-network* into the experimental requirements, two sensible adaptations were adopted, *SG-multi* and *SG-multiJS*. The order of complexity of the original work (only binary problems) and the following adaptations is $\mathcal{O}(|Y|)$.

SG-multi. For each class $c \in Y$, there is a pair model/training set $\langle \theta_c, \mathcal{L}_c \rangle$ properly designed to represent the most general hypothesis h_G^c w.r.t. the class c. Initially, all instances $\langle \boldsymbol{x}, y, w \rangle \in \mathcal{L}_c$ are the same instances present in the pool, except for two differences: they are labeled as "positive" to the corresponding class ($y = c$) and weighted to have only a small fraction of the importance of the real labeled instances ($w << 1$), as suggested in the literature [31]. The weight value adopted in this work is $w = \frac{1}{|Y||\mathcal{U}|}$, since it ensures that the summed influence of all background instances is no larger than a single real instance. This measure avoids misleadings due to the scarce initial real training instances.

The prediction function $f(\theta_c, \boldsymbol{x})$ returns the most probable class to a given instance $\boldsymbol{x}$ according to the provided model θ_c. It is possible to determine an instance under disagreement $\boldsymbol{x}^*$ by comparing the outcomes from all different prediction functions. Each prediction function represents the most general concept of each class:

$$\forall a, b \in Y, a \neq b, \exists \boldsymbol{x}^* \mid f(\theta_a, \boldsymbol{x}^*) \neq f(\theta_b, \boldsymbol{x}^*)$$

As soon as the instances from the region of disagreement $\boldsymbol{x}^*$, i.e. those with no consensus, are sampled and queried, they replace their counterparts in all training sets with the real labels and integral weights:

$$\mathcal{L}_c \leftarrow (\mathcal{L}_c - \{\langle \boldsymbol{x}^*, c, w \rangle\}) \cup \{\langle \boldsymbol{x}^*, c, 1 \rangle\} \forall c \in Y$$

In this adapted strategy (*SG-multi*), the decisions based on disagreement were kept binary, i.e. there is no ordering in the sequence of queries, except the precedence of the group of controversial instances over the rest.

SG-multiJS. A real-valued measure of disagreement can be adopted to soften the binary querying criterion of *SG-multi*. It assumes that the probability distributions $P(\theta_c, \boldsymbol{x})$ can be estimated from the models $\theta_c \forall c \in Y$. Besides the constraint on the classification algorithm being able to output probabilities, *SG-multiJS* differs from *SG-multi* in the querying criterion: the Jensen-Shannon divergence [20]. It is an information theoretic measure that compares probability distributions, commonly used in ensembles to assess the degree of agreement between their members. The non-weighted Jensen-Shannon divergence is defined by the entropy of the distributions:

$$JS(\{\theta_c \forall c \in Y\}) = E(\sum_{c \in Y} P(\theta_c, \boldsymbol{x})) - \sum_c E(P(\theta_c, \boldsymbol{x}))$$

The higher the JS, the further the members are from a consensus. Therefore, the instance with the highest value should be queried first. This criterion disrupts with the binary decision model underlying its theoretical background inspiration and may be more adequate to select instances from the disagreement area.

2.3 Query by Committee

Committees, also called ensemble-based classifiers, are combinations of models whose united predictions are meant to achieve better accuracy than a single

model. Query by Bagging and Query by Boosting are two examples of active learning committees [1]. Depending on the member models output, several measures of disagreement are possible.

In this paper, since the base learning algorithms of all strategies are not ensembles, a comparison that includes *Query by Committee* is deferred to future work. Moreover, a fair comparison between strategies requires the same base learner, otherwise accuracies of classifiers trained on the actively sampled instances could not be compared.

The complexity of Query by Committee is considered here as $\mathcal{O}(1)$, if the ensemble is seen as a single base learner or $\mathcal{O}(M)$, if the number of members M is considered.

2.4 Expected Error Reduction

Probably, the *entropy reduction example* [28] is the first proposal of an *expected error reduction* strategy: the instance that achieves the greatest reduction in the total predicted label entropy is select as the best query.

An important feature of the expected error reduction family of strategies is the possibility to adopt any objective function, like *g-means* or *f-measure* [17] - g-means, e.g. can be employed in the presence of class imbalance, a frequent issue in multi-class problems.

A more recent work [11] presents a method that considers implicitly information about the underlying clustering partitions, instead of relying only on the scarce labeled data. It is the natural choice for the present comparison given its reported superior performance. For each candidate instance $x \in \mathcal{U}$ from the pool, its most probable label y' is calculated optimistically:

$$y' = \arg\min_{y} \sum_{u} H(\boldsymbol{x}_u, \theta_{\mathcal{L} \cup \{\langle \boldsymbol{x}, y \rangle\}})$$

where $H(x, \theta)$ is the objective function. Additionally to the accuracy, and in line with the original work, the entropy on the unlabeled data is also adopted as objective function in this work (amounting two variations of the same strategy: accuracy and entropy).

The high complexity order of the algorithm ($\mathcal{O}(|\mathcal{U}|^2)$) degrades linearly with increases in $|Y|$, which is a major concern in problems with a big number of classes. To alleviate the computational cost, a hundred instances were randomly sampled from $\mathcal{U}$ in each iteration in the experiments of this article.

2.5 Expected Model Change

One can relief the sampling process from the computational complexity of analyzing the expected impact over the pool. This is possible by observing only

the expected impact on the model. One such strategy is the Expected Gradient Length [33]. Since the true label is not known in advance, the expected model change is calculated over all possible labels. The differences between two trainings (the previous and the candidate to be the next training) $\Delta C(\boldsymbol{x}, y, \mathcal{L})$ is weighted by the model's posterior probability estimates $P(\boldsymbol{x})$:

$$EMC(\boldsymbol{x}) = \sum_{c \in Y} P(c|\boldsymbol{x}) \Delta C(\boldsymbol{x}, y, \mathcal{L})$$

$$\Delta C(\boldsymbol{x}, c, \mathcal{L}) = |C(\mathcal{L} \cup \{\langle \boldsymbol{x}, c \rangle\}) - C(\mathcal{L})|$$

Expected Model Change is similar to *uncertainty sampling* because it is based on a localized criterion: it is focused on the relation between the current model and the candidate query instead of the rest of the instances.

The complexity of each query is $\mathcal{O}(|Y|.|\mathcal{U}|)$. Like *Expected Error Reduction*, training time can be reduced when the learning algorithm is incremental. Since none of the learning algorithms adopted in this work have an analogous to the gradient length, *Expected Model Change* was not included in the experiments.

2.6 Density-Weighted Sampling

The general contract of the *Density-weighted* strategies is the *information density* measure [30]:

$$ID(\boldsymbol{x}) = H(\boldsymbol{x}) \frac{1}{|\mathcal{U}|} \sum_{\boldsymbol{u} \in \mathcal{U}} sim(\boldsymbol{x}, \boldsymbol{u})$$

or the *training utility* [10], measure adopted by its improved version and used in this work:

$$TU(\boldsymbol{x}) = ID(\boldsymbol{x}) (\sum_{\boldsymbol{l} \in \mathcal{L}} sim(\boldsymbol{x}, \boldsymbol{l}))^{-1}$$

Any similarity $sim(\boldsymbol{x}, \boldsymbol{u})$ and informativity measures $H(\boldsymbol{x})$ can be adopted. In this work, five distances $d(\boldsymbol{x}, \boldsymbol{u})$ were compared (Euclidian, Minkowsky, Manhattan, Chebyshev and Mahalanobis) and transformed into a similarity measure by the formula:

$$sim(\boldsymbol{x}, \boldsymbol{u}) = \frac{1}{1 + d(\boldsymbol{x}, \boldsymbol{u})}$$

Due to publication restrictions concerning space, only the two best distances were kept in the results (Euclidian and Manhattan). The margin $M(\boldsymbol{x})$ was adopted as the informativity measure.

The complexity order is $\mathcal{O}(1)$, but $|\mathcal{U}|^2$ distance calculations are needed for each query. For this reason, their values should be cached in fast access memory to reduce computational costs by taking advantage of the fact that the pool remains the same along the whole process. The main feature of *density-weighted* methods is their sensitivity to the spatial distribution of the data.

2.7 Cluster-Based Sampling

The learning process can exploit natural clusters in the (unlabeled) pool, instead of performing queries that focus the decision boundaries/version space division. One such approach is the hierarchical sampling [7]. Instances are queried with higher probability from the most impure and representative clusters. The original implementation was adopted in the comparison of this work, with the same clustering algorithm: the Ward's average linkage method[1].

Cluster-based strategies are independent from the classification algorithms. Their hierarchical version is statistically sound, since it draws instances at random from each cluster within estimates for the error induced by each pruning. Therefore, it is guaranteed to not perform worse than random sampling. Because of the independence regarding classification algorithms, they are called *agnostic*. Another example of agnostic strategy is Random Sampling.

3 Experiments

In the evaluation of the active learning strategies, it is important to compare different classification algorithms, because non-agnostic strategies depend heavily on the base learner. Therefore, all the evaluated strategies were assessed using four algorithms commonly used in classification problems: C4.5, Naive Bayes (NB) , Very Fast Decision Trees (VFDT) and 5-NN [24,19,9,15]. Specifically, NB, VFDT and 5-NN are well suited for interactive active learning because they accept incremental training. Redundant results, like the similar performance of entropy $E(\boldsymbol{x})$ and uncertainty $P_{max}(\boldsymbol{x})$ were omitted due to space restrictions.

The active learning process is divided in two phases: sampling and training. For each new query, a new model is built/updated and tested against unknown instances previously set apart. Ten runs of 10-fold cross-validation were used for each dataset [5]. Duplicate instances were removed. Each fold was used as the pool of unlabeled instances - as adopted by [22].

In real applications, at least at the first steps, it is expected from the supervisor to perform some kind of *guided active learning* [3] to reduce the risk of incurring into useless labeling. Therefore, in the experiments, it was assumed that one instance from each class had its label revealed before each active learning strategy took place[2]. One or more than one instance per class have been used in literature [12].

3.1 Stopping Criterion

Learning stops after Q queries. Q is dataset-dependent and defined as follows. In the literature, arbitrary values (50, 100, 200, $|\mathcal{U}|$ etc.) have been used [25,12]. However, arbitrary values do not take into account dataset's peculiarities. In this

[1] Clusterer and classifiers implementation, including their default parameters, are from Weka library [14].

[2] Except for the Cluster-based strategy.

work, Q is the average number of queries the best strategy needed to achieve the average *passive accuracy*. The *passive accuracy* was calculated after training the classifier with all available instances in the pool and testing it in the test folds.

To assess the quality of the learned model, its accuracy was averaged along all possible budgets until Q, resulting in the *Area under the Learning Curve* [13].

3.2 Datasets

Twenty-eight labeled data sets from the UCI repository [4] were used in the experiments. They are detailed in Table 1. Datasets with imbalance level larger or equal any of the average passive accuracies were discarded.

Table 1. Dataset details. Last column indicates the proportion of examples from the majoritary class.

Dataset	#Instances	#Numeric	#Nominal	#Classes	%Majoritary class
colon32	62	32	0	2	0.65
bodies	62	3721	0	2	0.55
subject	63	229	0	2	0.56
hayes-roth	84	4	0	3	0.37
accute-i	99	1	6	2	0.56
leukemia-h	100	50	0	2	0.51
breast-t	105	9	0	6	0.21
tae	106	3	2	3	0.36
molecular-p	106	0	57	2	0.50
iris	147	4	0	3	0.34
wine	178	13	0	3	0.40
connection	208	60	0	2	0.53
newthyroid	215	5	0	3	0.70
statlog-h	270	13	0	2	0.56
flare	287	0	11	6	0.30
ionosphere	350	34	0	2	0.64
monk1	432	0	6	2	0.50
breast-c	569	30	0	2	0.63
balance	625	4	0	3	0.46
australian	690	8	6	2	0.56
pima	768	8	0	2	0.65
vehicle	846	18	0	4	0.26
tic-tac-toe	958	0	9	2	0.65
vowel	990	10	0	11	0.09
yeast	1269	8	0	4	0.35
cmc	1358	2	7	3	0.44
wineq-r	1359	11	0	6	0.42
car	1728	0	6	4	0.70

3.3 Experimental Results

In Table 2, all pairs of strategies are compared by the rankings shown in Table 3. Each symbol $s_{r,c}$ in a cell at row r and column c indicates that the strategy r is better than strategy c within the confidence interval 0.05 according to the Friedman test with the Nemenyi post-hoc test [8].

Table 2. Each placed symbol indicates when the strategy at the row is better than the strategy at the column: C4.5 (◯), NB (□), 5-NN (△) and VFDT (·)

Active Learning strategy	1	2	3	4	5	6	7	8	9	10
1 - Random Sampling	-									
2 - Uncertainty	△	-					△			
3 - Cluster-based			-							
4 - Margin	△			-			△			
5 - SGmulti	⊡	⊡	□	⊡	-	◯	⊡	⊡		
6 - SGmultiJS	□	⊡	□			-	⊡	·		
7 - Exp. Error Reduction (entropy)						◯	-			
8 - Exp. Error Reduction (accuracy)							△	-		
9 - Density Weighted Training Utility (euclidian)	⧋	◯̲	⧋	◯	△	⧋	⧋	⧋	-	
10 - Density Weighted Training Utility (manhattan)	⧋	⊙	⧋	⊙	△	⧋	⧋	⧋		-

Table 2 shows that the performances of the strategies are strongly related to the classification algorithm used. The proposed SG-network adaptations were better than almost all other strategies when NB (□) was used. VFDT (·) also presented a positive response under these strategies. The density-weighted strategies achieved similar performance with C4.5 (◯) and 5-NN (△); again VFDT was partially well suited, but mostly for the Manhattan variation of the density-based approaches. Uncertainty and Margin sampling using 5-NN were better than random sampling, the baseline of most studies. They were also better than expected error reduction (entropy) when using 5-NN. The worst strategies were based on expected error reduction and random sampling because of the significant losses. Cluster-based was outperformed only by SGmulti and density-based variations, but did not outperformed any strategy with statistical significance.

The expected error reduction strategy was not impacted by the 100-instance subsampling. This is evidenced by noting that its performance was not better even in datasets with less than 100 instances in the pool. The first nine rows of tables 1 and 3 represent the small datasets, which required no subsampling.

Table 3. ALC ranking for the first Q queries (Section 3.1). Lower is better. Strategy numbers are the same given in the Table 2. The last row is the median for all datasets.

	C4.5										VFDT										5-NN										NB									
Strat.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
col.	6	8	5	7	2	0	3	9	1	4	8	5	7	3	2	4	6	9	1	0	9	4	2	3	6	7	8	5	1	0	8	3	9	2	6	4	5	7	1	0
bod.	3	6	9	5	2	4	7	8	1	0	8	7	0	6	4	3	5	9	1	2	8	6	3	5	9	2	7	4	1	0	8	6	9	5	2	0	1	7	4	3
sub.	5	8	3	7	6	9	0	4	2	1	8	7	1	6	5	4	0	9	2	3	9	3	7	4	5	8	6	2	0	1	9	6	7	4	5	0	1	8	2	3
hay.	6	3	8	5	7	9	0	4	1	2	0	8	2	9	1	3	6	4	7	5	4	5	7	6	0	3	9	8	1	2	2	8	3	7	1	0	9	4	6	5
acc.	7	4	5	3	6	9	2	8	0	1	5	7	3	6	1	0	9	8	2	4	8	3	5	2	6	4	9	7	1	0	8	5	6	4	1	2	9	7	3	0
leu.	3	6	9	5	1	8	4	7	2	0	3	7	1	8	0	2	4	9	5	6	8	4	2	3	7	6	9	5	0	1	8	5	1	4	0	2	7	9	3	6
bre.	6	7	5	8	4	2	3	9	0	1	5	7	4	6	1	0	8	9	3	2	8	7	3	5	4	2	9	6	1	0	2	9	3	8	1	0	4	5	7	6
tae	4	0	6	3	1	5	7	9	2	8	5	2	9	7	8	0	4	3	6	1	4	1	8	2	6	7	9	3	0	5	3	9	4	8	2	1	6	0	5	7
mol.	8	3	7	2	5	4	6	9	0	1	3	8	2	9	5	1	6	0	7	4	6	2	1	3	5	8	9	7	4	0	7	9	1	6	4	0	8	5	3	2
iris	8	5	7	4	2	6	3	9	1	0	4	6	3	7	1	0	9	8	5	2	8	2	6	5	4	3	9	7	1	0	6	7	5	8	1	0	9	4	3	2
wine	8	5	3	4	2	7	6	9	1	0	7	5	4	6	2	0	9	8	3	1	8	2	7	3	5	6	9	4	1	0	9	5	8	4	1	0	6	7	3	2
con.	3	9	2	8	6	4	5	7	1	0	5	8	1	7	3	2	6	9	4	0	8	2	5	1	6	7	9	4	3	0	6	9	1	8	0	4	3	2	7	5
new.	5	8	9	7	2	6	4	3	1	0	8	2	4	5	1	0	9	7	6	3	9	2	6	3	7	8	5	4	0	1	9	6	7	5	1	0	8	4	3	2
stat.	5	8	2	7	3	4	6	9	1	0	0	9	1	8	4	3	7	2	5	6	8	3	5	2	6	7	9	4	0	1	5	9	6	7	0	1	8	2	4	3
flare	7	5	2	6	9	8	4	3	0	1	8	5	9	6	1	2	7	4	3	0	8	2	7	3	6	4	9	5	0	1	8	1	9	3	6	0	7	5	2	4
ion.	7	4	6	3	2	9	5	8	1	0	5	7	8	6	0	1	9	4	3	2	7	1	8	0	6	9	3	2	4	5	5	7	4	8	0	1	3	2	9	6
mon.	8	3	6	2	7	9	1	5	0	4	6	4	8	1	3	5	9	7	2	0	6	2	7	3	5	9	8	4	1	0	7	2	8	0	3	5	9	6	4	1
br.c	3	7	4	6	2	8	5	9	1	0	7	6	4	5	1	0	8	9	3	2	7	3	5	2	8	6	9	4	1	0	8	4	6	5	1	0	7	9	3	2
bal.	5	8	6	9	3	7	1	4	0	2	2	9	4	6	0	7	8	5	3	1	6	7	8	4	5	2	9	3	0	1	4	7	6	3	1	8	9	5	2	0
aus.	8	4	1	3	5	9	6	7	2	0	1	9	4	8	0	2	5	3	6	7	6	3	4	2	8	5	9	7	1	0	5	9	3	8	2	0	7	1	6	4
pim.	8	3	9	2	4	7	6	5	1	0	2	8	4	9	0	1	7	3	5	6	7	0	5	2	9	4	8	6	3	1	3	8	4	9	2	1	7	0	6	5
veh.	6	7	8	9	4	3	2	5	1	0	6	5	3	7	2	1	8	9	4	0	8	2	6	3	9	4	5	7	1	0	4	7	1	6	0	3	8	9	2	5
tic.	4	7	2	6	3	9	1	0	8	5	5	4	9	3	0	7	6	8	2	1	8	3	7	2	4	6	9	5	0	1	8	4	9	2	3	5	6	7	1	0
vow.	8	4	7	3	6	9	2	5	1	0	8	7	9	3	2	6	4	5	1	0	8	4	6	2	9	5	7	3	0	1	8	9	7	3	2	6	4	5	1	0
yea.	9	3	4	6	1	2	7	5	8	0	2	8	1	5	0	7	6	9	4	3	9	2	3	4	5	8	7	6	1	0	6	9	3	4	2	8	7	5	0	1
cmc	0	8	2	5	3	6	4	1	9	7	8	6	4	1	3	9	5	7	0	2	1	9	5	6	0	4	3	2	7	8	6	9	5	3	2	4	7	8	1	0
win.r	8	1	9	0	2	6	4	3	5	7	5	4	6	2	1	9	7	8	0	3	2	9	7	3	0	1	4	5	8	6	7	1	8	0	4	9	6	5	2	3
car	8	2	6	3	7	9	4	5	1	0	8	0	7	1	3	9	6	2	4	5	8	5	7	0	4	9	6	3	1	2	8	5	7	4	0	9	6	2	3	1
Med.	6	5	6	5	3	7	4	6	1	0	5	7	4	6	1	2	6	7	3	2	8	3	6	3	6	6	9	4	1	1	7	7	6	4	1	1	7	5	3	2

4 Conclusions

Despite its statistical soundness, sophisticated methods, like the cluster-based, did not perform better than ad hoc approaches, like SGmulti and density-based training utility. Therefore, possibly the *sampling bias* plays an important role in active learning, analogous to the *learning bias* (representation/search bias) of a classifier (learning algorithm): whilst generalization of learning is only possible with a bias, a good choice of queries for a given pair dataset/classifier implies the adoption of a strategy with the correct types of exploration and exploitation, and also the adequate balance between both.

The results obtained in this study suggests that active learning can be effective, but dependent on the classification algorithm. It is worth to mention the good overall results for the first proposed multiclass adaptation of SG-network (SGmulti) and density-based training utility. An investigation of the relationship between dataset features and strategy performance, and the use of other classifiers, with different learning biases, are intended as future works.

Acknowledgments. The authors would like to thank CAPES, CNPq and FAPESP for the financial support.

References

1. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging. In: Shavlik, J.W. (ed.) ICML, pp. 1–9. Morgan Kaufmann (1998)
2. Angluin, D.: Queries and concept learning. Machine Learning 2(4), 319–342 (1987)
3. Attenberg, J., Provost, F.J.: Why label when you can search?: alternatives to active learning for applying human resources to build classification models under extreme class imbalance. In: KDD, pp. 423–432. ACM (2010)
4. Bache, K., Lichman, M.: UCI repository of machine learning databases. Machine-readable data repository, University of California, Department of Information and Computer Science, Irvine, CA (2013)
5. Bouckaert, R.R., Frank, E.: Evaluating the replicability of significance tests for comparing learning algorithms. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 3–12. Springer, Heidelberg (2004)
6. Cohn, D.A., Atlas, L.E., Ladner, R.E.: Improving generalization with active learning. Machine Learning 15(2), 201–221 (1994)
7. Dasgupta, S.: Two faces of active learning. Theoretical Computer Science 412(19), 1767–1781 (2011)
8. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
9. Domingos, P., Hulten, G.: Mining high-speed data streams. In: Ramakrishnan, R., Stolfo, S.J., Bayardo, R.J., Parsa, I. (eds.) KDD, pp. 71–80. ACM (2000)
10. Fujii, A., Inui, K., Tokunaga, T., Tanaka, H.: Selective sampling for example-based word sense disambiguation. Computational Linguistics 24(4), 573–597 (1998)
11. Guo, Y., Greiner, R.: Optimistic active-learning using mutual information. In: Veloso, M.M. (ed.) IJCAI, pp. 823–829 (2007)
12. Guo, Y., Schuurmans, D.: Discriminative batch mode active learning. In: Platt, J.C., Koller, D., Singer, Y., Roweis, S.T. (eds.) NIPS. Curran Associates, Inc. (2007)
13. Guyon, I., Cawley, G.C., Dror, G., Lemaire, V.: Results of the active learning challenge. In: Active Learning and Experimental Design @ AISTATS, vol. 16, pp. 19–45. JMLR.org (2011)
14. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations 11(1), 10–18 (2009)
15. Hart, P.E.: The condensed nearest neighbor rule (corresp.). IEEE Transactions on Information Theory 14(3), 515–516 (1968)
16. Körner, C., Wrobel, S.: Multi-class ensemble-based active learning. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) ECML 2006. LNCS (LNAI), vol. 4212, pp. 687–694. Springer, Heidelberg (2006)

17. Kubat, M., Holte, R.C., Matwin, S.: Learning when negative examples abound. In: van Someren, M., Widmer, G. (eds.) ECML 1997. LNCS, vol. 1224, pp. 146–153. Springer, Heidelberg (1997)
18. Lewis, D.D.: A sequential algorithm for training text classifiers: Corrigendum and additional data. SIGIR Forum 29(2), 13–19 (1995)
19. Lewis, D.D.: Naive (bayes) at forty: The independence assumption in information retrieval. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)
20. Lin, J.: Divergence measures based on the shannon entropy. IEEE Transactions on Information Theory 37(1), 145–151 (1991)
21. McCallum, A., Nigam, K.: Employing EM and pool-based active learning for text classification. In: Shavlik, J.W. (ed.) ICML, pp. 350–358. Morgan Kaufmann (1998)
22. Melville, P., Mooney, R.J.: Diverse ensembles for active learning. In: Proceedings of the Twenty-First International Conference on Machine Learning, p. 74. ACM, New York (2004)
23. Mitchell, T.M.: Machine learning. McGraw Hill Series in Computer Science. McGraw-Hill (1997)
24. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann (1993)
25. Raghavan, H., Madani, O., Jones, R.: Active learning with feedback on features and instances. Journal of Machine Learning Research 7, 1655–1686 (2006)
26. Ritter, G.L., Woodruff, H.B., Lowry, S.R., Isenhour, T.L.: An algorithm for a selective nearest neighbor decision rule (corresp.). IEEE Transactions on Information Theory 21(6), 665–669 (1975)
27. Robertson, A.: The sampling variance of the genetic correlation coefficient. Biometrics 15(3), 469–485 (1959)
28. Roy, N., McCallum, A.: Toward optimal active learning through sampling estimation of error reduction. In: Brodley, C.E., Danyluk, A.P. (eds.) ICML, pp. 441–448. Morgan Kaufmann (2001)
29. Schein, A.I., Ungar, L.H.: Active learning for logistic regression: an evaluation. Machine Learning 68(3), 235–265 (2007)
30. Settles, B.: Curious machines: active learning with structured instances. Ph.D. thesis, University of Madison Wisconsin (2008)
31. Settles, B.: Active Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool (2012)
32. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: EMNLP, pp. 1070–1079. ACL (2008)
33. Settles, B., Craven, M., Ray, S.: Multiple-instance active learning. In: Platt, J.C., Koller, D., Singer, Y., Roweis, S.T. (eds.) NIPS. Curran Associates, Inc. (2007)
34. Shannon, C.E.: Communication theory of secrecy systems. Bell System Technical Journal 28(4), 656–715 (1949)