Discriminative Batch Mode Active Learning

Yuhong Guo **Dale Schuurmans**
Department of Computing Science
University of Alberta
{yuhong,dale}@cs.ualberta.ca.edu

Abstract

Active learning sequentially selects unlabeled instances to label with the goal of reducing the effort needed to learn a good classifier. Most previous studies in active learning have focused on selecting one unlabeled instance at one time while retraining in each iteration. However, single instance selection systems are unable to exploit a parallelized labeler when one is available. Recently a few batch mode active learning approaches have been proposed that select a *set* of most informative unlabeled instances in each iteration under the guidance of some heuristic scores. In this paper, we propose a discriminative batch mode active learning approach that formulates the instance selection task as a continuous optimization problem over auxiliary instance selection variables. The optimization is formulated to maximize the discriminative classification performance of the target classifier, while also taking the unlabeled data into account. Although the objective is not convex, we can manipulate a quasi-Newton method to obtain a good local solution. Our empirical studies on UCI datasets show that the proposed active learning is more effective than current state-of-the art batch mode active learning algorithms.

1 Introduction

Learning a good classifier requires a sufficient number of labeled training instances. In many circumstances, unlabeled instances are easy to obtain, while labeling is expensive or time consuming. For example, it is easy to download a large number of webpages, however, it typically requires manual effort to produce the labels. Randomly selecting unlabeled instances for labeling is inefficient in many situations, since non-informative or redundant instances might be selected. Aiming to reduce labeling effort, active learning methods have been adopted to control the labeling process in many areas of machine learning.

Given a large pool of unlabeled instances, active learning provides a way to iteratively select the most informative unlabeled instances—the queries—from the pool to label. This is the typical setting of pool-based active learning. Most of the active learning approaches, however, have focused on selecting only one unlabeled instance at a time, while retraining the classifier on each iteration (Zhu et al., 2003; Nguyen & Smeulders, 2004; Muslea et al., 2002; Roy & McCallum, 2001; Campbell et al., 2000; Tong & Koller, 2000; McCallum & Nigam, 1998a; Freund et al., 1997; Lewis & Gale, 1994; Cohn et al., 1996; Guo & Greiner, 2007). When the training process is hard or time consuming, this repeated retraining is inefficient. Furthermore, if a parallel labeling system is available, a single instance selection system can make wasteful of the resource. Thus, a batch mode active learning strategy that selects multi-instances each time is more appropriate under these circumstances. Principles for batch mode active learning need to be developed to address the multi-instance selection specifically. Simply using a single instance selection strategy to select more than one unlabeled instance in each iteration does not work well, since it fails to take the information overlap between the multiple instances into account. In fact, a few batch mode active learning approaches have been proposed recently (Schohn & Cohn, 2000; Brinker, 2003; Xu et al., 2003; Hoi et al., 2006b; Hoi

et al., 2006a). However, most extend existing single instance selection strategies into multi-instance selection simply by using a heuristic score or greedy procedure to ensure both the instance diversity and informativeness.

In this paper, we propose a new discriminative batch mode active learning method, which exploits the information contained in the unlabeled set and targets the goal of learning a good classifier directly. We define a good classifier to be one that obtains high likelihood on the labeled training instances and low uncertainty on labels of the unlabeled instances. We therefore formulate the instance selection problem as an optimization problem with respect to auxiliary instance selection variables, taking the measure of good classification as the objective function. This optimization problem is a NP-hard problem, and it is intractable to seek the exact optimal solution. However, we can approximate it locally using the second order Taylor expansion and then find a suboptimal solution using a quasi-Newton local optimization technique.

The instance selection variables we introduce in the optimization in fact indicate the optimistic guesses for the labels of the selected unlabeled instances. A concern about our discriminative instance selection therefore is that some information in the unlabeled data not consistent with the true classification partition might mislead the instance selection process. Fortunately, our active learning method can immediately tell whether it has been misled by comparing the true labels to its prior guesses. Thus we have the opportunity to adjust the active selection strategy it in the next iteration whenever a mismatch between the labeled and unlabeled data is detected. An empirical study on UCI datasets shows that our new batch mode active learning method is more effective than current state-of-the-art batch mode active learning algorithms.

2 Related Work

Many previous researchers have addressed the active learning problem in various different ways. Most have focused on selecting a single most informative unlabeled instance to label each time. Many such approaches make myopic decisions based solely on the current learned classifiers and select the unlabeled instance they are most uncertain about to label. (Lewis & Gale, 1994) chooses the unlabeled instance with conditional probability closest to 0.5 as the most uncertain instance. (Freund et al., 1997) takes the instance on which a committee of classifiers disagree the most. (Tong & Koller, 2000; Campbell et al., 2000) suggest choosing the instance closest to the classification boundary, and (Tong & Koller, 2000) analyzes this active learning as a version space reduction process. Approaches making use of unlabeled data to provide complementary information for active learning have also been proposed. (Cohn et al., 1996; Zhang & Chen, 2002) employ the unlabeled data by using the prior density $p(\mathbf{x})$ as uncertainty weights. (Roy & McCallum, 2001) selects instance that optimizes the expected generalization error over the unlabeled data. (McCallum & Nigam, 1998b) uses an EM approach to integrate the information from unlabeled data. (Muslea et al., 2002; Zhu et al., 2003) consider combining active learning with semi-supervised learning. (Nguyen & Smeulders, 2004) presents a mathematical model that explicitly combines clustering and active learning. (Guo & Greiner, 2007) presents a discriminative active learning approach, which implicitly exploits the clustering information contained in the unlabeled data in an optimistic way.

Considering the problem that single instance selection strategies require tedious retraining with each single instance labeled (and they can not take advantages of parallel labeling systems), many batch mode active learning methods have been proposed recently. (Schohn & Cohn, 2000; Brinker, 2003; Xu et al., 2003) extend single instance selection strategies that use support vector machines. (Brinker, 2003) takes the diversity of the selected instances into account, in addition to individual informativeness (Xu et al., 2003) proposes a representative sampling approach that selects the cluster centers of the instances lying within the margin of the support vector machine. (Hoi et al., 2006b; Hoi et al., 2006a) choose multi-instances that efficiently reduce the Fisher information. Overall, these approaches use various heuristics to guide the instances selection such that the selected batch should be informative about the classification model, while being diverse enough so that their information would not overlap.

Instead of using heuristic measures, in this paper, we formulate batch mode active learning as an optimization problem that aims to learn a good classifier directly. Our optimization selects the best set of unlabeled instances and their labels to produce a classifier that attains maximum likelihood

on labels of the labeled instances while attaining minimum uncertainty on labels of the unlabeled instances. It is apparently intractable to do an exhaustive search for the optimal solution; our optimization problem is a NP-hard problem. Nevertheless we can exploit a second-order Taylor approximation and use a quasi-Newton optimization method to quickly reach a local solution. Optimization techniques have been widely used in machine learning research. (Bennett & Parrado-Hernandez, 2006) examines the interaction of machine learning and mathematical programming. Our proposed approach shows an example of exploiting optimization techniques in the batch model active learning research.

3 Logistic Regression

In this paper, we use binary logistic regression as our base classification algorithm. Logistic regression is a well-known and mature statistical model for probabilistic classification that has been actively studied and applied in machine learning. Given a test instance $\mathbf{x}$, binary logistic regression models the conditional probability of the class label $y \in \{+1, -1\}$:

$$p(y = \pm 1|\mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp(-y\mathbf{w}^\top \mathbf{x})} \tag{1}$$

where $\mathbf{w}$ is the model parameter. Here the bias term is omitted for simplicity of notation. The model parameters can be trained by maximizing the likelihood of the labeled training data, i.e., minimizing the logloss of the training instances:

$$\min_{\mathbf{w}} \sum_{i \in L} \log(1 + \exp(-y_i \mathbf{w}^\top \mathbf{x}_i)) + \frac{\lambda}{2} \mathbf{w}^\top \mathbf{w} \tag{2}$$

where $\frac{\lambda}{2}\mathbf{w}^\top \mathbf{w}$ is a regularization term introduced to avoid overfitting problems. Logistic regression is a robust classifier that can be trained efficiently using various convex optimization techniques (Minka, 2003). It is also a linear classifier, however it is easy to obtain nonlinear classifications by simply introducing kernels (Zhu & Hastie, 2005).

4 Discriminative Batch Mode Active Learning

During the process of active learning, there will typically be a small number of labeled instances relative to a large number of unlabeled instances. Instance selection strategies based only on the labeled data therefore ignore potentially useful information contained in the unlabeled instances. In this section, we present a new discriminative batch mode active learning algorithm for binary classification, which exploits the information in the unlabeled instances. Our approach is *discriminative*, because: (1) it selects a batch of instances through optimizing a discriminative classification model; (2) it selects the instances by considering the best discriminative configuration of their labels that leads to the best classifier. Unlike other batch mode active learning approaches, which identify the most informative batch of instances by using heuristic measures, our approach aims to identify the batch of instances that directly optimizes classifier accuracy.

4.1 Optimization Problem

The optimal active learning strategy is to select the set of instances to label that lead to learning the best classifier. A good classifier is usually defined as one that has low generalization classification error on testing instances. Supervised classification methods usually maximize the likelihood of training instances to obtain a good classifier. However some semi-supervised classification methods (Grandvalet & Bengio, 2005) exploiting unlabeled instances by maximizing the likelihood of labeled instances, while minimizing the uncertainty of unlabeled instances, to achieve a classifier with low generalization error. We measure generalization error of a classifier in a similar way and use the following measure:

$$\sum_{i \in L} \log P(y_i|\mathbf{x}_i, \mathbf{w}) + \alpha \sum_{j \in U} \sum_{y=\pm 1} P(y|\mathbf{x}_j, \mathbf{w}) \log P(y|\mathbf{x}_j, \mathbf{w}) \tag{3}$$

where α is a tradeoff parameter used to adjust the relative influence of the labeled and unlabeled data for this evaluation L indexes the labeled set, and U indexes the unlabeled set. The idea behind

our active learning strategy is to select a batch of unlabeled instances S from U in iteration t, aiming to maximizing the measure (3) after labeling them. Measure (3) can be rewritten as a score function of selected instances S:

$$f(S) = \sum_{i \in L^t \cup S} \log P(y_i|\mathbf{x}_i, \mathbf{w}^{t+1}) + \alpha \sum_{j \in U^t \setminus S} \sum_{y=\pm 1} P(y|\mathbf{x}_j, \mathbf{w}^{t+1}) \log P(y|\mathbf{x}_j, \mathbf{w}^{t+1}) \tag{4}$$

where the classification model parameter $\mathbf{w}^{t+1}$ is trained on the new labeled set $L^{t+1} = L^t \cup S$.

In practice, the problem with using the $f(S)$ score to guide selection is that we do not know the labels for instances S when we are performing the selection. A typical solution for this problem is to measure the expected $f(S)$ score under the current learned model $\mathbf{w}^t$; that is using $\mathbf{E}[f(S)] = \sum_{\mathbf{y}_S} P(\mathbf{y}_S|\mathbf{x}_S, \mathbf{w}^t) f(S)$. However, using this expectation might aggravate the ambiguity that already exists in the current model $\mathbf{w}^t$, which has been trained on the very small set L^t, using $P(\mathbf{y}_S|\mathbf{x}_S, \mathbf{w}^t)$ as weights. Here, we propose to use an optimistic strategy: select a set of unlabeled instances S which posses one set of label configurations $\mathbf{y}_S^*$ that leads to the best score $f(S)$. This optimistic scoring function can be written as:

$$f(S) = \max_{\mathbf{y}_S} \sum_{i \in L^t \cup S} \log P(y_i|\mathbf{x}_i, \mathbf{w}^{t+1}) + \alpha \sum_{j \in U^t \setminus S} \sum_{y=\pm 1} P(y|\mathbf{x}_j, \mathbf{w}^{t+1}) \log P(y|\mathbf{x}_j, \mathbf{w}^{t+1}). \tag{5}$$

Now the problem becomes how to select the most optimistic set S. Although this problem can be solved using an exhaustive search on all size $m = |S|$ subsets of the unlabeled set U and possible labelings, it is intractable to do so in practice, since the search space is exponentially large. Even explicit heuristic search approaches seeking a local optima do not exist, because it is hard to define an explicit set of systematic operations that can transfer from one position to another one within the search space, while guaranteeing improvements over the score $f(S)$.

In this paper, we propose to approach optimistic batch mode active learning by formulating an explicit optimization version of the problem. Given the labeled set L^t and unlabeled set U^t after iteration t, the problem is to select a subset S from U^t that achieves the best score defined in (5) for the next iteration $t+1$. We introduce a set of instance selection variables $\boldsymbol{\mu}$ to determine S, and then formulate instance selection for iteration $t+1$ as the following optimization problem:

$$\max_{\boldsymbol{\mu}} \quad \sum_{i \in L^t} \log P(y_i|\mathbf{x}_i, \mathbf{w}^{t+1}) + \beta \sum_{j \in U^t} \mathbf{v}_j^{t+1} \boldsymbol{\mu}_j^\top + \alpha \sum_{j \in U^t} (1 - \boldsymbol{\mu}_j \mathbf{e}) \mathbf{u}_j^{t+1} \mathbf{e} \tag{6}$$

$$s.t. \quad \boldsymbol{\mu} \in \{0,1\}^{2 \times |U^t|}; \tag{7}$$

$$\sum_{j \in U^t} \boldsymbol{\mu}_j \mathbf{e} = m; \tag{8}$$

$$\boldsymbol{\mu}_j \mathbf{e} \leq 1, \forall j; \tag{9}$$

$$\sum_{j \in U^t} \boldsymbol{\mu}_j \leq (\frac{1}{2} + \epsilon) m \mathbf{e}^\top. \tag{10}$$

where $\mathbf{v}_j^{t+1}$ is a row vector equal to $[\log P(y=1|\mathbf{x}_j, \mathbf{w}^{t+1}), \log P(y=-1|\mathbf{x}_j, \mathbf{w}^{t+1})]$; $\mathbf{u}_j^{t+1}$ is a row vector with length 2, containing $P(y|\mathbf{x}_j, \mathbf{w}^{t+1}) \log P(y|\mathbf{x}_j, \mathbf{w}^{t+1})$ for $y = \{+1, -1\}$ in order; $\boldsymbol{\mu}_j$ is also a row vector with length equal to 2, corresponding to selection indication variables for the two classes $\{+1, -1\}$ of the jth unlabeled instance; $\mathbf{e}$ is a column $\mathbf{1}$ vector with length 2; ϵ is a user-provided parameter controlling the class balance for the instance selection; and β is a parameter we will use later to adjust our belief in the guessed labels. Note that, in fact, the selection variables $\boldsymbol{\mu}$ not only choose instances from U^t, they also select labels for the selected instances. Solving this optimization yields the optimal $\boldsymbol{\mu}$ for instance selection in iteration $t+1$.

The optimization (6) is an integer programming problem that would produce equivalent results to using exhaustive search to optimize (5), except here we have additional class balance constraints (10). Integer programming is an NP-hard problem. Therefore, the first step we make to solve this problem is to relax it into a continuous optimization by replacing the integer constraints $\boldsymbol{\mu} \in \{0,1\}^{2|U^t|}$ with continuous constraints $0 \leq \boldsymbol{\mu} \leq 1$:

$$\max_{\boldsymbol{\mu}} \quad \sum_{i \in L^t} \log P(y_i|\mathbf{x}_i, \mathbf{w}^{t+1}) + \beta \sum_{j \in U^t} \mathbf{v}_j^{t+1} \boldsymbol{\mu}_j^\top + \alpha \sum_{j \in U^t} (1 - \boldsymbol{\mu}_j \mathbf{e}) \mathbf{u}_j^{t+1} \mathbf{e} \tag{11}$$

$$s.t. \quad 0 \leq \boldsymbol{\mu} \leq 1; \tag{12}$$

$$\sum_{j \in U^t} \boldsymbol{\mu}_j \mathbf{e} = m; \tag{13}$$

$$\boldsymbol{\mu}_j \mathbf{e} \leq 1, \forall j; \tag{14}$$

$$\sum_{j \in U^t} \boldsymbol{\mu}_j \leq (\frac{1}{2} + \epsilon) m \mathbf{e}^\top. \tag{15}$$

After solving this continuous optimization, we then can use a greedy strategy to recover the integer solution by taking the biggest $\boldsymbol{\mu}$ values with respect to the constraints. However, this relaxed optimization problem maximizes a non-concave objective function (11) with respect to a set of linear constraints (12), (13), (14) and (15). Therefore it is hard to find a globally optimal solution. To see this more clearly, note that $\mathbf{w}^{t+1}$ is the parameter for classification model trained on $L^{t+1} = L^t \cup S$, where S indexes the unlabeled instances selected by $\boldsymbol{\mu}$. Therefore $\mathbf{w}^{t+1}$ is in fact a function of $\boldsymbol{\mu}$: $\mathbf{w}^{t+1} = g(\boldsymbol{\mu})$. The relaxed optimization problem is still very complex, and its objective function can be viewed as an arbitrary function of $\boldsymbol{\mu}$. Nevertheless, standard continuous optimization techniques can be used to solve for a local minima.

4.2 Quasi-Newton Method

The objective function (11) in the formulated maximization problem is only a function of the instance selection variables $\boldsymbol{\mu}$:

$$f(\boldsymbol{\mu}) = \sum_{i \in L^t} \log P(y_i | \mathbf{x}_i, \mathbf{w}^{t+1}) + \beta \sum_{j \in U^t} \mathbf{v}_j^{t+1} \boldsymbol{\mu}_j^\top + \alpha \sum_{j \in U^t} (1 - \boldsymbol{\mu}_j \mathbf{e}) \mathbf{u}_j^{t+1} \mathbf{e}. \tag{16}$$

However, this function is non-concave. Therefore convenient convex optimization techniques that achieve globally optimal solutions cannot be applied. Nevertheless, we develop a local optimization method that uses a variant of a quasi-Newton method to quickly determine a locally optimal solution. At each iteration, the optimization starts from a fixed point reached in the last iteration, and then makes a local move that allows it to make the biggest improvement in the objective function, along the direction decided by cumulative information obtained from the sequence of local gradients.

At each iteration k, we start from point $\bar{\boldsymbol{\mu}}$ which is the initial point or the point we reached in the last iteration. We first derive a second-order Taylor approximation of the objective function $f(\boldsymbol{\mu})$ to replace the original function within the small neighborhood of the current point $\bar{\boldsymbol{\mu}}$:

$$\tilde{f}(\boldsymbol{\mu}) = f(\bar{\boldsymbol{\mu}}) + \nabla f_k^\top (\boldsymbol{\mu} - \bar{\boldsymbol{\mu}}) + \frac{1}{2} (\boldsymbol{\mu} - \bar{\boldsymbol{\mu}})^\top H_k (\boldsymbol{\mu} - \bar{\boldsymbol{\mu}}), \tag{17}$$

where we assume $\boldsymbol{\mu}$ and $\bar{\boldsymbol{\mu}}$ take forms of column vectors; $\nabla f_k = \nabla f(\bar{\boldsymbol{\mu}})$. Since our original optimization function is smooth, the quadratic (17) can reasonably approximate it in a small neighborhood of $\bar{\boldsymbol{\mu}}$. Therefore we can determine the best search step by solving a quadratic programming with the objective (17) and linear constraints (12), (13), (14) and (15). Suppose the optimal solution for this quadratic program is $\tilde{\boldsymbol{\mu}^*}$, we then obtain a reasonable update direction $\mathbf{d}_k = \tilde{\boldsymbol{\mu}^*} - \bar{\boldsymbol{\mu}}$ for the kth iteration. Given this direction, a back-track line search is used to guarantee improvement over the original objective (16). Note that for each update of $\boldsymbol{\mu}$, $\mathbf{w}^{t+1}$ has to be retrained on $L^t \cup S$ to evaluate the new objective value where S indexes the instances selected by $\boldsymbol{\mu}$. In order to reduce the computational cost, we approximate the training of $\mathbf{w}^{t+1}$ in our empirical study, by limiting it to a few Newton-steps with a starting point given by $\mathbf{w}$ trained only on L^t.

The remaining issue is computing the local gradient $\nabla f(\bar{\boldsymbol{\mu}})$ and the Hessian matrix H_k. For the local optimization we can assume $\mathbf{w}^{t+1}$ remains constant $g(\bar{\boldsymbol{\mu}})$ with small local updates on $\bar{\boldsymbol{\mu}}$. Therefore, the local gradient can be approximated as:

$$\nabla f(\bar{\boldsymbol{\mu}_j}) = \beta \, \mathbf{v}_j^{t+1} - \alpha \, [\mathbf{u}_j \mathbf{e}, \mathbf{u}_j \mathbf{e}], \tag{18}$$

where all the $\nabla f(\bar{\boldsymbol{\mu}_j})$ can be put together into a column vector $\nabla f(\bar{\boldsymbol{\mu}})$. We then use BFGS (Broyden-Fletcher-Goldfarb-Shanno) to compute the Hessian matrix, which starts as an identity matrix for the first iteration, and then is updated in each iteration as follows (Nocedal & Wright, 1999):

$$H_{k+1} = H_k - \frac{H_k s_k s_k^\top H_k}{s_k^\top H_k s_k} + \frac{y_k y_k^\top}{y_k^\top s_k}, \tag{19}$$

where $y_k = \nabla f_{k+1} - \nabla f_k$, and $s_k = \bar{\boldsymbol{\mu}}_{(k+1)} - \bar{\boldsymbol{\mu}}_{(k)}$. This Hessian matrix accumulates information from the sequences of local gradients to help determine better update directions.

The local convergence optimization algorithm will stop iterating when no progress can be made in the objective (16).

4.3 Adjustment Strategy

In the discriminative optimization problem we formulated in Section 4.1, the $\boldsymbol{\mu}$ variables are used to optimistically select both instances and their labels, aiming to achieve the best classification model according to the objective (11). When the labeled set is small, and the discriminative partition (clustering) information contained in the large unlabeled set is not consistent with the true classification, the labels optimistically "guessed" for the selected instances through $\boldsymbol{\mu}$ might not be consistent with the true classification either. Thus, the instance selected won't be very useful to identify our true classification model. Furthermore, the unlabeled data might continue to mislead the next instance selection iteration.

Fortunately, we can immediately identify this problem of being misled after obtaining the true labels for the selected instances. If the true labels are different from the guessed labels that are returned by the optimization, we need to make some adjustment for the next instance selection iteration.

We have tried a few adjustment strategies in our study. We report the most effective one in this paper. Note that the *being-misled* problem is in fact caused by the unlabeled data that affects the target classification model through the term $\beta \sum_{j \in U^t} \mathbf{v}_j^{t+1} \boldsymbol{\mu}_j^\top$. Therefore, a simple way to fix the problem is to adjust the parameter β. At the end of each iteration t, we obtain the true labels $\mathbf{y}_S$ for the selected instances S, then we compare them with our guessed labels $\hat{\mathbf{y}}_S$ indicated by $\boldsymbol{\mu}^*$. If they are consistent, then we will set $\beta = 1$, which means we trust the partition information from the unlabeled data as same as the label information in the labeled data for building the classification model $\mathbf{w}$. If $\mathbf{y}_S \neq \hat{\mathbf{y}}_S$, then apparently we should reduce the β value, that is, reducing the influence of the unlabeled data for the next selection iteration $t+1$. We use a simple heuristic to determine the β value. Starting from $\beta = 1$, we iteratively reduce its value by a small factor 0.5, until we get a better objective value for (16) when replacing the guessed indication variables $\boldsymbol{\mu}^*$ with the true labeling indications. Note that, if we reduce β to zero, our optimization will be exactly equivalent to picking the most uncertain instance when $m = 1$.

5 Experiments

To investigate the empirical performance of our discriminative batch mode active learning algorithm, we conducted a set of experiments on nine two-class UCI datasets, comparing with a baseline random instance selection algorithm and two batch mode active learning methods proposed in the literature: *svmD*, an approach that incorporates diversity in active learning with SVM (Brinker, 2003); *Fisher*, an approach that uses Fisher information matrix for instance selection (Hoi et al., 2006b). The UCI datasets we used include (we show the name, followed by the number of instances and the number of attributes): Australian(690;14), Cleve(303;13), Corral(128;6), Crx(690;15), Flare(1066;10), Glass2(163;9), Heart(270;13), Hepatitis(155;20) and Vote(435;15).

We investigate a hard case of active learning, where we start active learning from only a few labeled instances. In each experiment, we start with four randomly selected labeled instances, two in each class. We then randomly select 2/3 of the remaining instances as the unlabeled set, using the remaining instances for testing. All the algorithms start with the same initial labeled set, unlabeled set and testing set. For a fixed batch size n, each algorithm repeatedly select n instances to label each time, with maximum 120 to select in total when it is possible. We repeat each experiment 20 times and report the average performance.

In our study, we tested the algorithms using varying batch sizes: 5, 10 and 15. The comparison results are quite similar for these batch sizes. However, differences between algorithms are easier to be detected with more evaluation points. Thus we only report the results for batch size 5, due to the limited space.

Figure 1 shows the comparison results on the nine UCI datasets. Apparently, comparing with the other two batch mode algorithms, overall, our proposed discriminative batch model algorithm wins

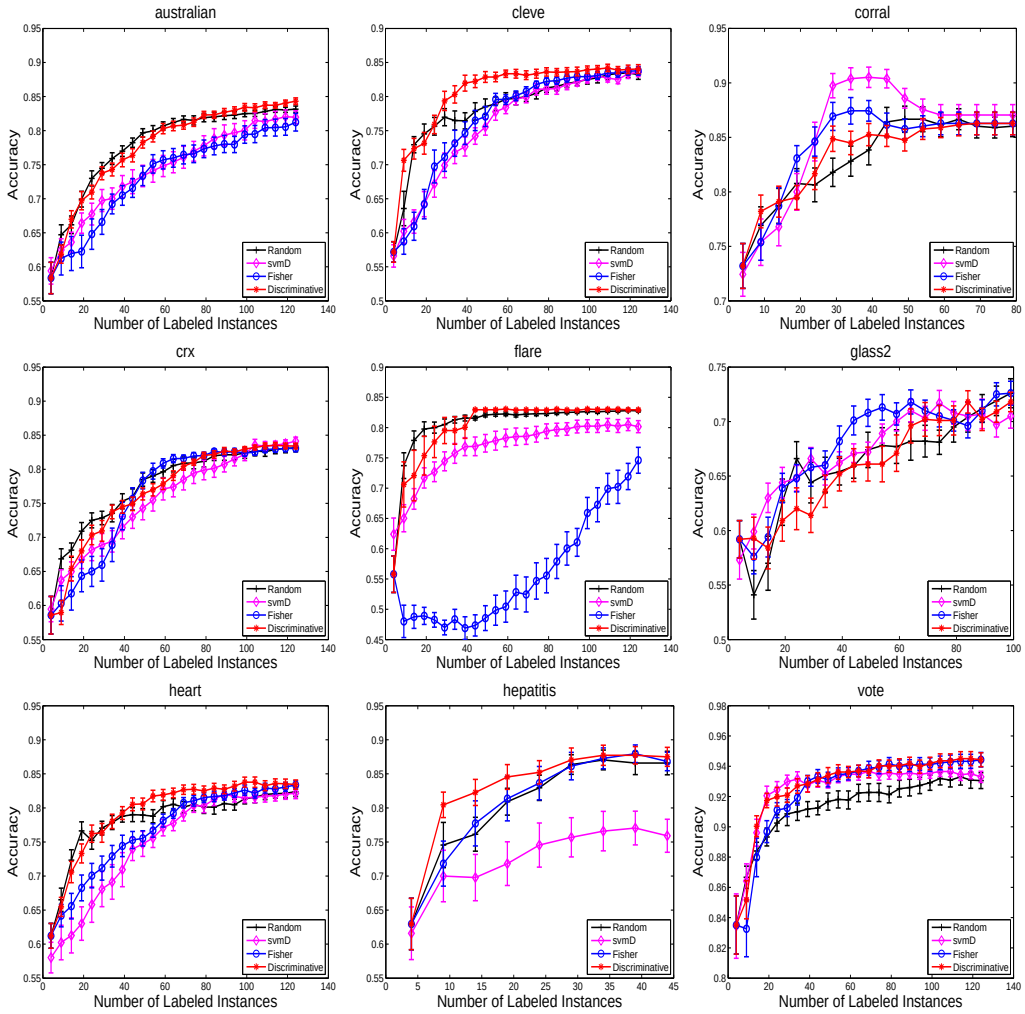

Figure 1: Results on UCI Datasets.

on 5 datasets: Australian, Cleve, Flare, Heart and Hepatitis; reaches a tie on 2 datasets: Crx and Vote; and loses on the remaining 2 datasets: Corral and Glass2. Though the baseline random sampling method works surprisingly well, our algorithm always performs better than it or at least achieves a comparable performance.

These results suggest that selecting unlabeled instances through optimizing the classification model directly would get the most relevant and informative instances, comparing with using heuristic scores to guide the selection. Though the original optimization problem is NP-hard, a relaxed local optimization method that leads to a local optima still works effectively.

6 Conclusion

In this paper, we proposed a discriminative batch mode active learning, which exploits the information contained in the unlabeled set and selects a batch of instances through optimizing the target classification model. Though it could be overly optimistic about the information contained in the unlabeled set and cause the problem of being misled. The problem can be identified immediately after obtaining the true labels. A simple adjustment strategy can then be used to avoid the problem in the following iteration. The experimental results on the UCI datasets show that this approach is more effective comparing with other batch mode active learning methods. Though our current work

is focused on 2-class classification problems, it is easy to be extended to multiclass classification problems.

References

Bennett, K., & Parrado-Hernandez, E. (2006). The interplay of optimization and machine learning research. *Journal of Machine Learning Research, 7*.

Brinker, K. (2003). Incorporating diversity in active learning with support vector machines. *Proceedings of the 20th international conference on Machine learning*.

Campbell, C., Cristianini, N., & Smola, A. (2000). Query learning with large margin classifiers. *Proceedings of the 17th International Conference on Machine Learning*.

Cohn, D., Ghahramani, Z., & Jordan, M. (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research, 4*.

Freund, Y., Seung, H. S., Shamir, E., & Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning, 28*.

Grandvalet, Y., & Bengio, Y. (2005). Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems (NIPS)*.

Guo, Y., & Greiner, R. (2007). Optimistic active learning using mutual information. *Proceedings of the International Joint Conference on Artificial Intelligence*.

Hoi, S., Jin, R., & Lyu, M. (2006a). Large-scale text categorization by batch mode active learning. *Proceedings of the international World Wide Web conference*.

Hoi, S., Jin, R., Zhu, J., & Lyu, M. (2006b). Batch mode active learning and its application to medical image classification. *Proceedings of the 23rd international conference on Machine Learning*.

Lewis, D., & Gale, W. (1994). A sequential algorithm for training text classifiers. *Proceedings of the International ACM-SIGIR Conference on Research and Development in Information Retrieval*.

McCallum, A., & Nigam, K. (1998a). Employing EM in pool-based active learning for text classification. *Proceedings of the 15th International Conference on Machine Learning*.

McCallum, A., & Nigam, K. (1998b). Employing em in pool-based active learning for text classification. *ICML*.

Minka, T. P. (2003). *A comparison of numerical optimizers for logistic regression* (Technical Report). http://research.microsoft.com/ minka/papers/logreg/.

Muslea, I., Minton, S., & Knoblock, C. (2002). Active + semi-supervised learning = robust multi-view learning. *Proceedings of the 19th International Conference on Machine Learning*.

Nguyen, H. T., & Smeulders, A. (2004). Active learning using pre-clustering. *Proceedings of the 21st International Conference on Machine Learning*.

Nocedal, J., & Wright, S. (1999). *Numerical optimization*. Springer, New York.

Roy, N., & McCallum, A. (2001). Toward optimal active learning through sampling estimation of error reduction. *Proceedings of the 18th International Conference on Machine Learning*.

Schohn, G., & Cohn, D. (2000). Less is more: Active learning with support vector machines. *Proceedings of the 17th International Conference on Machine Learning*.

Tong, S., & Koller, D. (2000). Support vector machine active learning with applications to text classification. *Proceedings of the 17th International Conference on Machine Learning*.

Xu, Z., Yu, K., Tresp, V., Xu, X., & Wang, J. (2003). Representative sampling for text classification using support vector machines. *Proceedings of the 25th European Conference on Information Retrieval Research*.

Zhang, C., & Chen, T. (2002). An active learning framework for content-based information retrieval. *IEEE Trans on Multimedia, 4*, 260–258.

Zhu, J., & Hastie, T. (2005). Kernel logistic regression and the import vector machine. *14*.

Zhu, X., Lafferty, J., & Ghahramani, Z. (2003). Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. *Proceedings of the ICML 2003 Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*.