2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Multi-class batch-mode active learning for image classification

Ajay J. Joshi[†] , Fatih Porikli[*], and Nikolaos Papanikolopoulos[†]
[†]University of Minnesota, Twin Cities [*]Mitsubishi Electric Research Labs

***Abstract*— Accurate image classification is crucial in many robotics and surveillance applications – for example, a vision system on a robot needs to accurately recognize the objects seen by its camera. Object recognition systems typically need a large amount of training data for satisfactory performance. The problem is particularly acute when many object categories are present. In this paper we present a batch-mode active learning framework for multi-class image classification systems. In active learning, images are to be chosen for interactive labeling, instead of passively accepting training data. Our framework addresses two important issues: i) it handles redundancy between different images which is crucial when batch-mode selection is performed; and ii) we pose batch-selection as a submodular function optimization problem that makes an inherently intractable problem efficient to solve, while having approximation guarantees. We show results on image classification data in which our approach substantially reduces the amount of training required over the baseline.**

I. Introduction

In this paper, we focus on the problem of object recognition and image classification in real-world problems. For example, consider a robot traversing through an environment in which it continuously encounters new objects. The recognition system has to deal with large variation in illumination, apparent object sizes, different viewpoints, etc. In order to deal with such large variation, a large and diverse training set is typically required for accurate recognition. The diversity in training is essential to handle variation in object appearance. For instance, a car seen from the side and rear appears very different. As such, obtaining enough human training on *diverse* images is one of the most difficult aspects in designing learning systems.

In computer vision applications, large training sets are employed to obtain the required diversity in image views. For example, a robot can be used to collect images of an object from all directions and build a model for recognition. However, this approach might not be feasible in the general setting - where mobility is not available, or where only images can be obtained from a database. In contrast, we explicitly enforce diversity in the training set, by utilizing a batch-mode setup.

In image classification problems, large amounts of human labeled training data are required for satisfactory performance. Therefore, human input often forms the primary bottleneck in achieving good generalization. Active learning attempts to reduce the human effort required in learning good object models thorough intelligent example selection mechanisms. Theoretical results show that in many problems, such active selection methods can substantially reduce the amount of training required to achieve a certain classification rate [2], [7], [10].

ajay@cs.umn.edu, faith@merl.com, npapas@cs.umn.edu

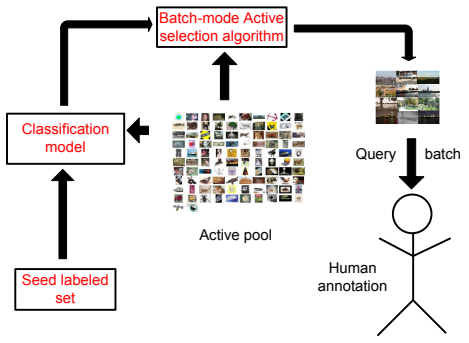


Fig. 1. Batch-mode selection model proposed in this paper. Note the absence of the feedback loop in batch-mode selection – this avoids multiple retraining of the classifier and provides easier user interaction. However, batch selection needs to explicitly handle example redundancy while being computationally tractable – our paper addresses these problems.

Our goal is to focus on active selection in large, multi-class problems that are typical in real-world classification scenarios. Typically, the process of active example selection has been iterative – the classifier queries for labels on certain examples which the human provides, followed by a step of classifier retraining. Such interaction is problematic on two fronts: i) training a classifier at each iteration or round poses computational challenges, especially for classifiers that cannot be trained incrementally; and ii) since the human has to input new labels at each round, the process can be cumbersome, indicating interactive inefficiency.

Most work on active learning in binary problems [4], [17], [25] as well as in multi-class classification [14]–[16], [24] has focused on single return or iterative active learning. In [26], the authors employ user inputs at multiple levels of granularity, also referred to as multi-level annotations. Holub et al. [13] propose entropy-based active learning that can handle batch-mode selection in principle, however, the approach is prohibitively expensive in practice. Recently a few researchers have proposed batch-mode selection algorithms [3], [12], however these are restricted to only binary classification. Efficient batch-mode selection is vitally important in large multi-class classification problems in order to make the methods practically appealing and computationally feasible.

Motivated by the described problems, we propose the first efficient *multi-class batch-mode active selection* framework in this paper. Using the proposed framework, multiple examples can be queried actively in a batch, thereby

minimizing classifier training time while allowing easy user interaction. Figure 1 shows the batch-mode active selection model we propose.

The learning setup and terminology used are as follows. We assume that a very small set of labeled examples (seed set) are used for initial training of the classifiers. Support Vector Machines (SVM) are used as the base binary classifier in this work, since they provide the best results on our datasets. The active pool is a large set of unlabeled examples from which the selection algorithm can select examples to query the user for class labels. The evaluation is performed on a separate test set.

II. Challenges in batch-mode selection

Actively selecting a single example for human labeling (single-return) requires a selection measure for querying *useful* examples at each iteration. By appending the newly labeled example to the training set, a new classifier can be trained for the next iteration. The active selection measure needs to be computed at every iteration since it depends on the current trained classifier. This implicitly minimizes redundancy in the queried examples, since if the classifier is confident on certain examples, they are not queried on future rounds. In batch-mode selection, the redundancy between examples needs to be accounted for explicitly. The primary challenges therefore are the following:

- Along with a measure of 'usefulness' of examples for active selection, we need a criterion to evaluate redundancy of examples - either explicitly through a set redundancy function, or through information theoretic measures. Finding a measure for example redundancy is especially hard in multi-class problems, since redundancy depends heavily on the classifiers employed, the feature space, and class populations among others. It is thus not straightforward to generalize measures of example redundancy from binary to multi-class classification.
- Even if we have redundancy measures, batch-mode selection poses a big computational bottleneck. Consider that we need to select a batch of size p from an unlabeled data pool of size n. The number of possible batches that can be selected is ${}^nC_p$. n and p are typically large – we therefore run into intractable subset selection problems.

In this paper, we first propose an active selection framework that explicitly accounts for example redundancies in multi-class problems. We then devise an active selection function and prove that it is submodular. Exploiting ideas from submodular function optimization [21], we overcome the computational challenge described above; we propose a greedy algorithm for selecting a batch of examples actively – the algorithm is efficient and also guarantees a near-optimal solution. Finally, we give three new example redundancy measures for multi-class problems that can be used within our framework. Our work thus generalizes batch-mode active selection to *multi-class* classification.

III. Active selection framework

In this section we describe our multi-class active learning framework for batch-mode example selection.

A. Overall strategy

Our framework relies on measures of utility and redundancy of examples. Roughly, utility corresponds to the benefit associated with choosing individual examples for active selection, in terms of the potential improvement in classification accuracy. For instance, uncertainty in classification of an unlabeled example could be used as its utility measure. Further, we define a measure of utility of a batch of examples as the sum of their individual utilities, where redundancy is ignored. For a set of examples $\mathcal{S} = \{h_1, \ldots, h_n\}$, denote their utility score by $\mathbf{U}(\mathcal{S})$. If the individual example utilities are denoted by $\mathbf{V}$, then $\mathbf{U}(\mathcal{S}) = \sum_i^n \mathbf{V}(h_i)$. Without loss of generality, we enforce the constraint $\mathbf{V} \in [\delta, 1]$ on the utility score range. 1 indicates maximum utility while $\delta > 0$ indicates a minimum – the rationale for keeping it above zero is that no utility measure can *guarantee* a zero utility for any example without knowing its true class label.

Next, we require a redundancy measure for a set of examples that captures their redundancy from a classification standpoint. For instance, consider unlabeled examples each having a high utility score. Even though each of the examples *individually* are expected to be very useful for active selection, they might contain the same information, i.e., selecting one example for training the classifier might make the other examples redundant. We need to explicitly account for such example redundancies in batch selection. For a set $\mathcal{S}$ of examples, we denote their redundancy score by $\mathbf{R}(\mathcal{S})$.

We employ a *quality* measure of a set of examples, denoted by $\mathbf{Q}(\cdot)$ that takes into account the utility and redundancy of examples

$$\mathbf{Q}(\mathcal{S}) = \mathbf{U}(\mathcal{S}) - \mathbf{R}(\mathcal{S}). \tag{1}$$

We further define $\mathbf{Q}(\phi) = 0$. The quality measure aims to capture the overall quality of a batch of examples for active selection, in terms of potential improvement in classification accuracy. Intuitively, diverse (non-redundant) sets of examples that are maximally informative will have a higher quality score.

B. The redundancy matrix

We consider that the redundancy score of a set is composed of pairwise "interference" scores between each pair of examples that the set consists of. Denote the pairwise interference score between examples h_i and h_j as $\mathbf{I}(h_i, h_j)$. Without loss of generality, we assume $\mathbf{I} \in [0, 1]$. In our framework, we employ these pairwise interference scores scaled appropriately, as elements of the redundancy matrix $\mathbf{M}(\mathcal{S}) \in \mathbb{R}^{n \text{x} n}$.

The matrix $\mathbf{M}$ can be interpreted as follows. Each row in the matrix represents the interference caused by one example with each of the other examples in the set. In order to achieve correct scaling for the interference, we scale the i^{th} row of the matrix by the utility value $\mathbf{V}(h_i)$ of the corresponding

example. Therefore, the element in i^{th} row and j^{th} column of the matrix can be written as

$$\mathbf{M}_{ij} = p \cdot \mathbf{V}(h_i) \cdot \mathbf{I}(h_i, h_j) \cdot \mathbb{I}(i \neq j), \tag{2}$$

where p is a constant scaling factor. The last term above is an indicator function, which evaluates to 0 if $i = j$ and is unity otherwise. In essence, it sets the diagonal elements to zero and thus removes the influence of self-interference from the matrix. The scaling by $\mathbf{V}(h_i)$ achieves reasonable values of redundancy for an example having a certain utility – in a sense, it performs normalization for the quality function. As will be seen in Proposition 2, the scaling is crucial for having a non-decreasing function $\mathbf{Q}$ – since $\mathbf{Q}$ measures the overall quality of a set from a classification standpoint, the non-decreasing property is desirable. It captures the fact that more training examples are desirable for classification. We now define the redundancy score of the set as the squared Frobenius norm of the matrix $\mathbf{M}$,

$$\mathbf{R}(\mathcal{S}) = \|\mathbf{M}(\mathcal{S})\|_{\mathrm{F}}^2. \tag{3}$$

C. Selection measure and submodularity

Consider a finite set $\mathcal{U}$ and a function $\mathbf{F} : \mathcal{U} \rightarrow \mathbb{R}$. $\mathbf{F}$ is a submodular set function [18], [21] for all $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{U}, K \notin \mathcal{B}$ if the following holds.

> $\mathbf{F}$ satisfies a diminishing returns property such that $\mathbf{F}(\mathcal{A} \cup \{K\}) - \mathbf{F}(\mathcal{A}) \geq \mathbf{F}(\mathcal{B} \cup \{K\}) - \mathbf{F}(\mathcal{B})$. In other words adding an element to a set increases the function value by at least as much as adding the element to its superset. This property is fairly intuitive and is satisfied by information theoretic measures such as entropy, and information gain (under conditional independence assumptions) [19].

In real problems that typically have high dimensions, real-valued attributes, and in some cases non-vectorial representations, measures such as mutual information (for redundancy) are hard to estimate accurately [9]. In contrast, the quality measure $\mathbf{Q}$ defined previously is very easy to compute and is the fundamental active selection measure we employ in this paper. Here, we prove that the set function $\mathbf{Q}$ is actually submodular, irrespective of the specific functions $\mathbf{V}$ and $\mathbf{I}$ employed.

Proposition 1: $\mathbf{Q}$, as defined in Equation (1) is a submodular set function.

Proof: Consider sets of examples $\mathcal{S}_1$ and $\mathcal{S}_2$ such that $\mathcal{S}_1 \subseteq \mathcal{S}_2$, and an example $x \notin \mathcal{S}_2$. In order for $\mathbf{Q}$ to satisfy the property for submodularity above, we need to show that

$$\mathbf{Q}(\mathcal{S}_2 \cup \{x\}) - \mathbf{Q}(\mathcal{S}_2) \leq \mathbf{Q}(\mathcal{S}_1 \cup \{x\}) - \mathbf{Q}(\mathcal{S}_1).$$

Now using the definition of $\mathbf{Q}$ from Equation (1) and expressing $\mathcal{S}_2 = \mathcal{S}_1 \cup \Delta\mathcal{S}$, we need to show that

$$\mathbf{R}(\mathcal{S}_1 \cup \Delta\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S}_1 \cup \Delta\mathcal{S}) \geq \mathbf{R}(\mathcal{S}_1 \cup \{x\}) - \mathbf{R}(\mathcal{S}_1). \tag{4}$$

The above means that adding an element to a set should not increase the redundancy score more than adding the same element to a superset.

If $\mathcal{S}_1$ has n_1 elements and $\mathcal{S}_2$ has $n_2 \geq n_1$ elements,

$$\begin{aligned}
\mathbf{R}(\mathcal{S}_1 \quad \cup \quad & \Delta\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S}_1 \cup \Delta\mathcal{S}) \\
= & \sum_{i,j=1, i\neq j}^{n_2} |p \cdot \mathbf{V}(h_i) \cdot \mathbf{I}(h_i, h_j)|^2 \\
= & \ \mathbf{R}(\mathcal{S}_1 \cup \{x\}) - \mathbf{R}(\mathcal{S}_1) \\
+ & \sum_{i,j=n_1+1, i\neq j}^{n_2} |p \cdot \mathbf{V}(h_i) \cdot \mathbf{I}(h_i, h_j)|^2 \\
\geq & \ \mathbf{R}(\mathcal{S}_1 \cup \{x\}) - \mathbf{R}(\mathcal{S}_1).
\end{aligned}$$

This proves the correctness of Equation (4), and thereby shows that $\mathbf{Q}$ is submodular. ■

Proposition 2: $\mathbf{Q}$ is a monotonically non-decreasing function for $\mathbf{I} \in [0, 1]$, and $p^2 \leq \frac{1}{n(1+1/\delta)}$, where n is the size of the largest set chosen for active selection.

Proof: Now consider a set $\mathcal{S} = \{h_1, \ldots, h_n\}$ which consists of n examples, and an example $x \notin \mathcal{S}$. The difference in the redundancy scores $\mathbf{R}(\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S})$ can be computed using the respective redundancy matrices.

$$\begin{aligned}
\mathbf{R}(\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S}) \quad = \quad & \sum_{i=1}^{n} |p \cdot \mathbf{V}(h_i) \cdot \mathbf{I}(h_i, h_j)|^2 \\
+ \quad & \sum_{j=1}^{n} |p \cdot \mathbf{V}(x) \cdot \mathbf{I}(h_j, x)|^2.
\end{aligned} \tag{5}$$

Since the maximum value of $\mathbf{V}$ and $\mathbf{I}$ is 1 each, the maximum value of the above difference can be obtained as

$$\begin{aligned}
\mathbf{R}(\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S}) \quad \leq \quad & np^2 + np^2\mathbf{V}(x) \\
= \quad & np^2(\mathbf{V}(x) + 1) \leq \mathbf{V}(x).
\end{aligned} \tag{6}$$

The last inequality follows from the upper bound on p^2 and since $\delta < 1$. Using the above, and the fact that $\mathbf{U}(\mathcal{S} \cup \{x\}) = \mathbf{U}(\mathcal{S}) + \mathbf{V}(x)$, we now have

$$\mathbf{U}(\mathcal{S} \cup \{x\}) - \mathbf{R}(\mathcal{S} \cup \{x\}) \geq \mathbf{U}(\mathcal{S}) - \mathbf{R}(\mathcal{S}), \tag{7}$$

which implies that $\mathbf{Q}(\mathcal{S} \cup \{x\}) \geq \mathbf{Q}(\mathcal{S})$ by definition. ■ In other words, the quality function adheres to the intuitively appealing 'information never hurts' principle [6].

D. Performance guarantees

Denote the unlabeled pool by $\mathcal{U}$ and suppose we want to chose a batch of k examples from the pool. In optimal batch-mode selection, our goal is to select a batch of size k, denoted by $\mathcal{A}_{opt}$ that maximizes the quality function $\mathbf{Q}$

$$\mathcal{A}_{opt} = \underset{|\mathcal{A}|=k}{\operatorname{argmax}} \quad \mathbf{Q}(\mathcal{A}). \tag{8}$$

The above problem of maximizing a submodular function is **NP**-hard in general [18]. However, Nemhauser et al. [21] show that for a submodular non-decreasing function $\mathbf{F}$ with $\mathbf{F}(\phi) = 0$, a greedy algorithm gives a solution with a value bounded close to the optimal. The basic idea is to iteratively select an element that maximizes the incremental increase in value of the function. Denote the set obtained by the greedy algorithm as $\mathcal{A}_g$. It is shown in [21] that such a greedy algorithm gives the following bound: $\mathbf{F}(\mathcal{A}_g)/\mathbf{F}(\mathcal{A}_{opt}) \geq (e-1)/e$.

Input: Unlabeled data pool $\mathcal{U}$, $\mathbf{R}$, $\mathbf{U}$, and k

1. $\mathcal{A} := \{\phi\}$, the current batch of examples;
2. **for** $i := 1$ **to** k, **do**
3. **foreach element** $h_j \in \mathcal{U} \backslash \mathcal{A}$, **do**
4. $\mathcal{A}_j := \mathcal{A} \cup \{h_j\}$;
5. Compute $\mathbf{Q}_j = \mathbf{U}(\mathcal{A}_j) - \mathbf{R}(\mathcal{A}_j)$;
6. **end**
7. select the example giving the highest improvement: $b = \text{argmax}_j \, \mathbf{Q}_j$;
8. $\mathcal{A} := \mathcal{A} \cup \{h_b\}$;
9. **end**
10. return $\mathcal{A}$.

Output: The actively selected batch $\mathcal{A}$, $|\mathcal{A}| = k$.

Fig. 2. A greedy batch-mode active selection algorithm.

E. A greedy algorithm

Based on the above, we propose a greedy batch-mode active selection algorithm in Figure 2.

Theorem 1: The selected batch of examples $\mathcal{A}$ output by the algorithm described in Figure 2 is near-optimal for the given set quality function $\mathbf{Q}$. Precisely, it satisfies the bound $\mathbf{Q}(\mathcal{A})/\mathbf{Q}(\mathcal{A}_{opt}) \geq (e-1)/e$, where $\mathcal{A}_{opt}$ is as defined in Equation (8).

Proof: The proof follows from Propositions 1 and 2 above, the fact that $\mathbf{Q}(\phi) = 0$, and the approximation result given by Proposition 4.3 in Nemhauser et al. [21]. ■

Therefore, the proposed algorithm is computationally inexpensive and at the same time gives a solution that is guaranteed close to optimal for a given quality function $\mathbf{Q}$.

If at any iteration in the above algorithm, no quality improvement is obtained (line 7), Proposition 4.2 in [21] implies that the chosen set $\mathcal{A}$ is actually optimal. In such a case when none of the examples in $\mathcal{U} \backslash \mathcal{A}$ give a positive improvement in the quality function, the algorithm can be terminated, while guaranteeing optimality.

IV. Multi-class redundancy measure

In this section, we propose three probabilistic multi-class redundancy measures. Specifically, our goal is to define an 'interference' function $\mathbf{I}$ employed previously that is easy to compute, and captures possible redundancies for the classifier. For instance, if two unlabeled examples carry the same information, they should have a high interference score. Apart from the batch selection framework above, we note that the aspect of multi-class redundancy is also new in our work – we are not aware of any multi-class batch-mode selection methods, and thus any measures of example redundancy for multi-class problems. We show here that such interference measures can be obtained with simple computations, and they give good results in practice.

Jain and Kapoor [14] note the important problem that it is difficult to compare uncertainty of binary subproblems in multi-class classification. In this paper, we use the one-vs-one method for multi-class classification, wherein a classifier is trained for each pair of classes. In order to compare uncertainty across binary subproblems, all the measures we propose are probabilistic, i.e., they rely on estimated class membership probabilities of unlabeled examples in the active pool. Probabilistic measures aid in coming up with *comparable* measures of uncertainty and redundancy across binary subproblems. The following section describes a way in which probabilities for unlabeled data can be estimated.

Probability estimates

For margin-based classifiers such as SVM, we require a way of estimating class membership probabilities from the margin of the unlabeled examples. We employ the 2-step approach proposed by Joshi et al. [15]. In the first step, binary probability values of class membership are estimated for all the binary subproblems. This estimation is performed using sigmoid function fitting on the margins of unlabeled examples [20], [23]. Once binary probabilities are estimated, multi-class estimation is performed by combining the binary estimates. The method of pairwise coupling [11], [27] is used to couple the binary estimates and infer a multi-class membership distribution. Such an estimation approach is computationally efficient and generalizes well to problems having many categories [15]. For implementation, the LIBSVM [5] toolbox is employed.

We are now ready to describe the proposed intersection functions. Note that $\mathbf{I}$ needs to be in the range $[0, 1]$ as defined in Section III-A above.

A. Jensen-Shannon divergence

Jensen-Shannon divergence (JSD) is a popular measure of similarity between two probability distributions. JSD is based on the Kullback-Leibler (KL) divergence, with the notable exceptions that JSD is finite and symmetric. If P and Q are two probability distributions,

$$JSD(P\|Q) = \frac{1}{2}KL(P\|M) + \frac{1}{2}KL(Q\|M), \tag{9}$$

where $M = (P+Q)/2$. We employ JSD as one measure of interference between two examples $\mathbf{I}$.

B. Histogram intersection

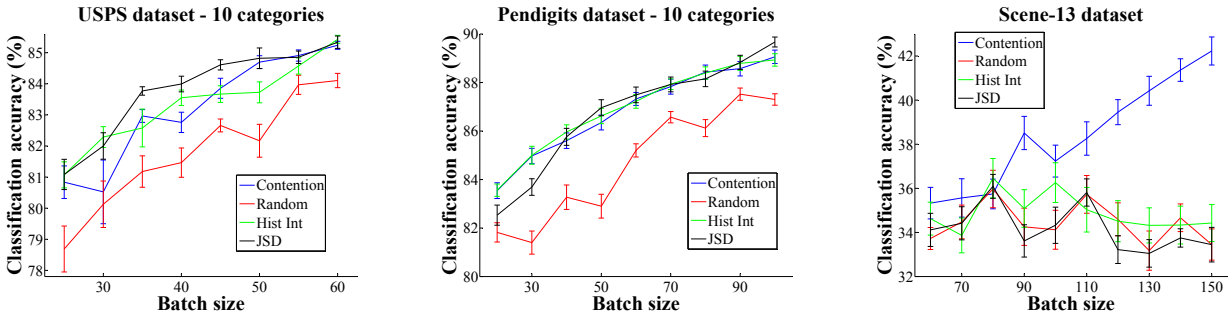
Histogram intersection is a measure of similarity between two histograms or discrete probability densities. For distributions P and Q, it is defined as

$$\mathbf{I}(P, Q) = \sum_i \min(P_i, Q_i). \tag{10}$$

C. Classifiers in contention

For multi-class problems, a concept referred to as "classifiers in contention" (the classifiers most likely to be affected by choosing an example for active learning) is introduced in [15]. This concept can also be employed for forming an interference measure – if two examples are likely to affect two different classifiers, they likely carry different information and are not redundant. Through this idea, we can capture the potential redundancies in *multi-class* problems, which is much more challenging that redundancy estimation in binary classification.

We now define the interference function for two distributions P and Q in the following. Denote the top two



(a) (b) (c)

Fig. 3. Batch-mode active selection v/s random batch selection on different datasets: (a) USPS, (b) Pendigits, (c) Scene-13. Redundancy measures used – Contention: Classifiers in contention, Hist Int: Histogram intersection, JSD: Jensen-Shannon divergence. Similar results observed on other datasets also.

classes with highest probability values for the distributions P and Q to be $c_1(P), c_2(P)$ and $c_1(Q), c_2(Q)$ respectively.

$$\begin{aligned}\mathbf{I}(P,Q) &= 1, \text{ if } c_1(P) = c_1(Q), \text{ and } c_2(P) = c_2(Q).\\ &= 0.1, \text{ if } c_1(P) = c_1(Q), \text{ and } c_2(P) \neq c_2(Q).\\ &= 0, \text{ if } c_1(P) \neq c_1(Q).\end{aligned}$$

The above implies the highest interference score for examples that have the exact same classifier in contention. On the other hand, for examples that are likely to belong to different categories, the interference score is zero.

Utility measure

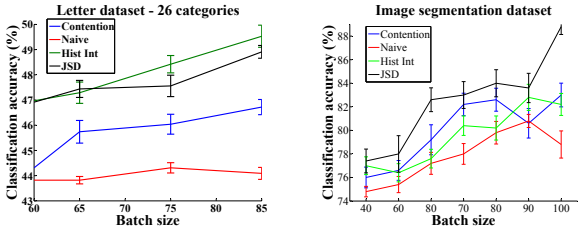
Apart from the redundancy measure described above, we also require a multi-class utility measure for selecting useful examples. We choose the probabilistic measure proposed in [15], and shown to be effective for problems having a large number of categories. The selection measure is the difference between the probability values of the top two classes obtained from the estimated distribution, or $(c_1(P) - c_2(P))$ in the above notation. The smaller the difference, the larger is the uncertainty, indicating a good candidate for active selection. We now define $\mathbf{V}(P) = \max(\delta, 1 - (c_1(P) - c_2(P)))$. Therefore, larger values of $\mathbf{V}$ imply greater uncertainty – also a lower limit δ is imposed for highly peaked distributions with little uncertainty, so that $\mathbf{V} \in [\delta, 1]$. We choose $\delta = 0.1$ for all the experiments.

V. Experiments

This section demonstrates the strength of our batch-mode selection algorithm on real-world data. We show results using the 3 different redundancy measures proposed, and compare it with random selection. Further, we also compare the results with "naive batch-mode selection": or selecting a batch by choosing the most informative examples based on their utility score only (the examples are sorted according to their utility score and then the top examples are chosen, without considering redundancy). In all experiments, we fix the scaling parameter $p = 0.02$, so that the bound on p^2 in Proposition 2 is satisfied. We use many multi-class image datasets for our experiments – Abalone, USPS, Pendigits, Image segmentation, and Letter datasets from the UCI repository [1], and a dataset of images from 13 natural scene categories [8] (scene-13). For scene-13, we use GIST features of Oliva and Torralba [22], since GIST shows good discriminatory power. The datasets we use are typically used for evaluating image classification systems since the they are collected from real sources having noise, have many categories, and also have many examples for testing efficiency of the algorithms.

A. Classification rate

In this section, we compare the classification rate of different active learning schemes. Figure 3[1] shows classification accuracy results (with 1-standard deviation bars) on USPS, Pendigits, and Scene-13 dataset. We can see that on the USPS and Pendigits datasets, all the redundancy measures outperform random selection significantly, showing that even in multi-class batch-mode selection, a lot of annotation effort can be reduced by the proposed algorithm. On the Scene-13 dataset, JSD and Histogram intersection perform poorly, giving accuracy values similar to random example selection. However, the method using classifiers in contention beats all other methods by a large margin. The result indicates that capturing redundancy is crucial to good performance. In this case, the 'contention' method looks greedily for distributions that peak at the same categories, while 'JSD' and 'histogram intersection' look at the entire distributions, and therefore fail to capture the corresponding example redundancies.



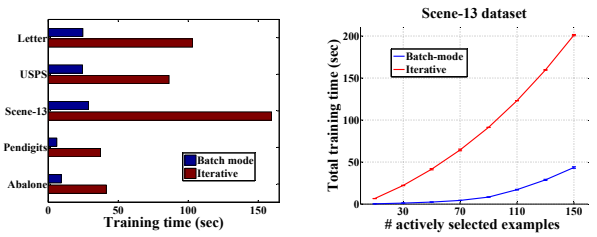
(a) (b)

Fig. 4. Classification accuracy comparisons with naive batch selection on (a) 'Letter', (b) 'Image segmentation' datasets.

B. Comparisons with naive batch-mode selection

Here, we evaluate the methods on another important aspect – comparisons with naive batch selection. Note that naive selection also uses an active selection measure – thus the baseline is much stronger, and for outperforming naive

[1]Figures best viewed in color.



(a) (b)

Fig. 5. Our method scales much better than the baseline in training time.

selection it is essential to capture example redundancies appropriately. Figure 4 shows results on the 'Letter' and 'Image segmentation' datasets that have 26 and 7 categories respectively. Plots in 4(a) and 4(b) show that all the interference functions outperform naive batch selection on both datasets. We observed similar behavior with other datasets also, and representative cases are shown due to space constraints. The experiment demonstrates that accounting for example redundancies is critical in real problems, and naively selecting the most informative individual examples gives poor results.

Note here that the interference measures we employ are at best simplistic – they are not problem specific and rely only on estimated class membership distributions. In a sense, this shows the potential of the proposed framework. The framework is generic and can accommodate other measures of utility and redundancy that capture the problem structure better. We believe that more advanced interference measures can make a big difference in the classification rate – it is our hope that further research is pursued in this direction based on the promising results shown.

C. Computational advantages

One of the primary motivations of batch-mode selection is to reduce the amount of computation involved – iterative active learning involves classifier retraining and probability estimation at each round of learning. Figure 5 demonstrates the time required for active example selection in both iterative and batch-mode settings on various datasets. Batch-mode selection is an order of magnitude faster, and also scales better as seen in Figure 5(b). Further note that labeling examples in batches is also much easier from a user interaction standpoint, making it appealing in practice. In summary, batch-mode selection is both computationally and interactively efficient, and thus extremely useful for large real-world applications.

VI. Conclusion

In this paper, we address the problem of batch-mode selection in multi-class problems. We demonstrate an efficient generic framework and give associated performance guarantees. We also propose ways to measure redundancy in batches of examples. The experiments show the importance of accounting for example redundancies, the improvement in classification rate achieved by the proposed framework, and significant computational savings. Measures that better capture redundancy and utility in example batches promise to improve the results even further. This will be one important direction for future work.

VII. Acknowledgments

This material is based upon work supported in part by, the U.S. Army Research Laboratory and the U.S. Army Research Office under contract #911NF-08-1-0463 (Proposal 55111-CI) and the National Science Foundation through grants #CNS-0324864, #CNS-0420836, #IIP-0443945, #IIP-0726109, #CNS-0708344, #CNS-0821474, and #IIP-0934327.

References

[1] A. Asuncion and D. J. Newman. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences, 2007. Available at `http://archive.ics.uci.edu/ml/datasets.html`.

[2] M.-F. Balcan, S. Hanneke, and J. Wortman. The true sample complexity of active learning. In *Conf. Computational Learning Theory*, 2008.

[3] K. Brinker. Incorporating diversity in active learning with support vector machines. In *ICML*, 2003.

[4] C. Campbell, N. Cristianini, and A. J. Smola. Query learning with large margin classifiers. In *ICML*, 2000.

[5] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[6] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley Interscience, 1991.

[7] S. Dasgupta. Analysis of a greedy active learning strategy. In *NIPS* . MIT Press, 2005.

[8] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005.

[9] K. Fukumizu, A. Gretton, X. Sun, and B. Scholkopf. Kernel measures of conditional dependence. In *Advances in Neural Information Processing Systems 20*. MIT Press, 2008.

[10] S. Hanneke. A bound on the label complexity of agnostic active learning. In *ICML*, 2007.

[11] T. Hastie and R. Tibshirani. Classification by pairwise coupling. *Annals of Statistics*, 26(2):451–471, 1998.

[12] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *ICML*, 2006.

[13] A. Holub, P. Perona, and M. Burl. Entropy-based active learning for object recognition. In *CVPR, Workshop on Online Learning for Classification*, 2008.

[14] P. Jain and A. Kapoor. Active learning for large multi-class problems. In *CVPR*, 2009.

[15] A. J. Joshi, F. Porikli, and N. Papanikolopoulos. Multi-class active learning for image classification. In *CVPR*, 2009.

[16] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with Gaussian Processes for object categorization. In *ICCV*, 2007.

[17] A. Kapoor, E. Horvitz, and S. Basu. Selective supervision: Guiding supervised learning with decision-theoretic active learning. In *IJCAI*, 2007.

[18] A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In *UAI*, 2005.

[19] A. Krause, H. B. McMahan, C. Guestrin, and A. Gupta. Robust submodular observation selection. Technical Report CMU-ML-08-100, Carnegie Mellon University, 2008.

[20] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68:267–276, 2007.

[21] G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming*, 14:265–294, 1978.

[22] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[23] J. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In *Advances in Large Margin Classifiers*. MIT Press, 2000.

[24] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang. Two-dimensional active learning for image classification. In *CVPR*, 2008.

[25] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *JMLR*, 2:45–66, 2001.

[26] S. Vijayanarasimhan and K. Grauman. Multi-level active prediction of useful image annotations for recognition. In *NIPS* . MIT Press, 2008.

[27] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *JMLR*, 5:975–1005, 2004.