Asymmetric propagation based batch mode active learning for image retrieval

Biao Niu, Jian Cheng, Xiao Bai, Hanqing Lu

National Laboratory of Pattern Recognition, CASIA, Beijing 100190, China
Beihang University, Beijing 100191, China

Abstract

Relevance feedback is an effective approach to improve the performance of image retrieval by leveraging the labeling of human. In order to alleviate the burden of labeling, active learning method has been introduced to select the most informative samples for labeling. In this paper, we present a novel batch mode active learning scheme for informative sample selection. Inspired by the method of graph propagation, we not only take the correlation between labeled samples and unlabeled samples, but the correlation among unlabeled samples taken into account as well. Especially, considering the unbalanced distribution of samples and the personalized feedback of human we propose an asymmetric propagation scheme to unify the various criteria including uncertainty, diversity and density into batch mode active learning in relevance feedback. Extensive experiments on publicly available datasets show that the proposed method is promising.

1. Introduction

With the development of digital imaging technologies, the huge number of images are produced daily. How to efficiently manage these large scale data has become a challenging task in information retrieval and multimedia analysis communities [1]. In the past decades, content based image retrieval (CBIR) [2–8] has been proved to be one of the most important and effective methods for this task. However, it is well known that the major bottleneck of CBIR is the semantic gap between low-level features and high-level semantic interpretation. Relevance feedback [9–11] is a useful and popular approach to bridge the gap by human–computer interaction. In relevance feedback process, users are taken into the loop of retrieval by iteratively labeling some unlabeled instances to improve the performance of retrieval system. For example, [12] localizes the user’s intention in the visual feature space by using the users’ feedback. However, labeling is tedious and time-consuming, and more labeling does not necessarily lead to better results. Therefore, how to label the limited number of unlabeled images in huge image database to effectively reduce human effort has become one of the key problems in relevance feedback.

Active learning (or more precisely, selective sampling) [13] has been proven to be effective in dealing with classification problems in which labeled data are scare while unlabeled data are abundant and easy to get. A number of active learning methods [14–17], have been developed for data classification. Typically, selective sampling chooses the most informative samples for user to label so that knowing their labels can greatly boost the performance of the classifier. Most active learning methods are selecting only the most informative sample for manually labeling in one iteration and retraining the classifier immediately, named one-by-one mode. Then the updated classifier further decide the next most
informative one to be labeled. The selected data will then be labeled manually and added to the training set to retrain the classifier. However, updating classifier usually occupies numerous time in feedback, which makes one-by-one mode extremely inefficient if not infeasible in practice.

To reduce computational time for training, several active learning methods in batch mode have been introduced. They try to properly assemble their optimal query batch in different ways. Generally apart from considering the criteria uncertainty the selection criteria, diversity and density are also taken into account to entirely reduce the redundancy among the selected instances. These criteria are widely used for active learning in batch mode. In uncertainty sampling framework, the learner choose the instances which is the least certain to label. However these methods usually ignore the distribution of the unlabeled instances and lead to serious sample bias. Density criteria aims to select the instances from the dense unlabeled regions. Yet the methods only requesting measurement of density usually need selecting a relatively large number of instances before the optimal classifier is found. Diversity criterion aims to select the instances among which the overlap in information content is least. But these methods considering the diversity may select outlier. Most active learning in batch mode algorithms only adopt one or two criteria for instance selection, which could partly limit the performance of the active learning in batch mode. Although there are several approaches [18,19] proposed to combine these criteria, they consider criteria separately and then simply adopt linear combination.

In this paper, we present a novel batch mode active learning scheme for informative instance (i.e. sample) selection. Inspired by the method of graph propagation [20], we not only take the correlation between labeled samples and unlabeled samples, but the correlation among unlabeled samples taken into account as well. Especially, by considering the unbalanced distribution of samples and comparing the user’s personalized feedback with the predicted label we adaptively decide to whether increase or reduce the scale parameter of the certainty propagation. Furthermore, we propose an asymmetric propagation scheme to unify the various criteria including uncertainty, diversity and density into batch mode active learning in relevance feedback. In each round feedback, some informative instances are selected to be labeled by user. Then, the certainty gain from labeling will be dynamically propagated to the rest unlabeled data in asymmetric scheme. After several round feedbacks, all labeled data are finally utilized to retrain the classifier model. Extensive experiments show that the proposed method achieve encouraging results.

The rest of the paper is structured as follows: the related work is presented in Section 2; Section 3 introduces the proposed batch mode active learning method; we present the experimental results in Section 4; finally, we conclude this paper in Section 5.

2. Related work

As tedious labeling and huge image dataset mentioned above, the critical issue in relevance feedback is how to efficiently and effectively select the helpful unlabeled instances for user to label. With the users’ feedback [21–24] has become a popular dimension-reduction method to narrow the semantic gap.

There are three main different scenarios in which the learner may be able to ask queries. They are (i) membership query synthesis [25], (ii) stream-based selective sampling [26], and (iii) pool-based active learning [27]. In this paper, we focus on the scenario of pool-based active learning, in which large collections of unlabeled data can be gathered and queries are selectively drawn from the pool. Most active learning algorithms are conducted in an iterative manner, which is composed of two parts, that is, a learning part and a sample selection part. In each iteration, the learning part train a model based on the current training set. Then the sample selection part selects the most informative samples for manual labeling according to the current model and these samples are added to the training set for next iteration.

Most active learning methods select only the most informative sample for manually labeling in one iteration and retraining the classifier immediately, named one-by-one mode. Then the updated classifier will decide the next most informative one to be labeled. Exemplar approaches include query-by-committee [28], and uncertainty sampling [14,29]. Tong and Chang [14] regard the task of learning concept of users’ queries as learning SVM binary classifiers. An SVM classifier capture the query concept by separating the relevant images from irrelevant images with a hyperplane in a projected space. The projected points on one side of the hyperplane are considered relevant to the query concept and the rest is irrelevant. They learn an SVM classifier on the current labeled data and choose the next instance to query the pool instance that comes closest to the hyperplane. The main weakness of these approaches is that they are unable to exploit the abundant of unlabeled data and the selection of query instances is only determined by a small number of labeled examples without considering the rest large number of unlabeled data’s distribution. More importantly, classifier training is usually the most computational expensive in relevance feedback, which result in the lower efficiency for one-by-one mode.

To reduce computational time for training, active learning methods in batch mode [18,30–32] have been introduced. In each round of relevance feedback, [18] proposes an approach for SVM that explicitly incorporates a diversity measure that considers the angles between the induced hyperplane. They maximize the angles among instances in the batch. Finally, in order to combine both requirements, viz. minimal distance to the classifier and diversity of these angles, they build the convex combination of both measure. Brinker [18] selects a batch of images taking into account uncertainty and diversity of these unlabeled instances and then manually labeled simultaneously. However, it has been suggested that uncertainty sampling and diversity strategies are prone to querying outliers. Therefore active learning incorporating the information density framework presented by [32] is proposed as a density-weighting technique. The main idea is that informative instances should not only be those
which are uncertain, but also those which are representative of the input distribution. Then the Fisher information \([30]\) avoid such traps implicitly, by utilizing the unlabeled pool \(U\) when estimating ratios. Hoi and Lyu \([33]\) propose a semi-supervised active learning framework for image retrieval. The suggested active learning scheme is based on the fusion of two different types of learning techniques, namely one is supervised and another is semi-supervised. In \([15]\) they propose active learning approach by querying informative and representative examples. They provide a systematic way for measuring and combining the informativeness and the representativeness. Their method is based on the min–max view \([34]\) of active learning. However, this method is not proposed as batch mode active learning and although \([15]\) reduces the redundancy among unlabeled instances, it does not yet consider the correlation between labeled instances and unlabeled instances. Cheng et al. \([35]\) extend this method to batch mode combining with dynamic certainty propagation.

Some other emerging new batch mode active learning techniques have been introduced in recent studies \([30,36–38]\). In terms of whether classifier retraining is required for each labeled instance or not, active learning algorithms can be classified into two modes: one-by-one mode and batch mode. For one-by-one mode active learning methods they query the most informative instance and then retrain the classifier after adding it into the training set. This mode is endowed with higher accuracy but lower efficiency. For batch mode active learning methods they query batch size of informative instances in one round with variant criteria measure and then retrain the classifier.

There are many cases that these criteria are combined explicitly as \([19]\). They have to find the joint parameters empirically, which results in badly adapting the variant data distribution. Instead of the computational retrain process in this paper we use the process of label propagation to estimate the new distribution after new instance labeled. We unify these criteria as well as the user’s feedback to adaptively control the scale parameter of the label propagation and then query the most uncertainty instance. We repeat the process batch size rounds before retraining.

3. Asymmetric propagation based batch mode active learning

In this paper, we propose an asymmetric propagation based active learning algorithm (APAL) to integrate the various sampling criteria including uncertainty, diversity, and density in unified selection scheme. In each iteration, we select and manually label the data, using the last labeled data point and the distribution of the unlabeled data to decide the selection of the next, the architecture model of this part is shown in Fig. 1; after a batch of data has been labeled and added to the training set, we perform training once to get a better performing classifier. In a word, in our scheme, we use one-by-one labeling but still batch training, which can be seen as a compromise of batch labeling and one-by-one training.

Assume \(X = \{x_1,x_2,\ldots,x_n\} = U \cup L\) to denote the dataset, where \(U\) and \(L\) denote the unlabeled and labeled dataset respectively. For each data point \(x_i \in X(1 \leq i \leq n)\), \(y_i \in \{0,1\}\) is its corresponding class label for negative and positive respectively. If \(x_i \in U\), \(y_i\) is unknown and we use the sign of \(f(x_i)\) to estimate \(y_i\), where \(f\) denotes the current trained classifier on \(L\).

3.1. Problem formulation

Our uncertainty sampling method is based on probability estimates of class membership for all the examples in the pool. In order to obtain these estimates we follow the initializing mode of \([33]\). Suppose there is an SVM classifier \(f\) trained on the given labeled data. We use \(f(x_i)\) to represent the distance with sign from an unlabeled data instance \(x_i\) to the current decision boundary of SVM. We employ a Sigmoid function to normalize the distance metric into probability label metric within \([0,1]\) as

\[
p_i = \frac{1}{1 + \exp(-f(x_i))} \quad (1)
\]

Following this proceeding we get the class posterior \(p_i\) for assigning the example to class \(y = 1\). The probability of example on decision boundary of SVM is 0.5. The probability of example that is less than 0.5 is considered as irrelevant, and one that is bigger than 0.5 is classified to relevant class.

According to the most uncertainty strategy we select sample that is closest to the current decision boundary \(f\) that is trained on the current set of labeled examples. We can cast this idea as follows:

\[
x_i = \arg \min_{x \in U} \|f(x_i)\| \quad (2)
\]

We should repeat the operation for selecting batch size number of targets in each iteration. This can be represented as

\[
x_i^k = \arg \min_{x \in U} \|f^+((x_i,y_i))(x_i)\| \quad (3)
\]

here \(+(x_i,y_i)\) means that after we select a new unlabeled example \(x_i\) and label it manually to obtain its label \(y_i\), then we add it to the training set and retrain the classifier. After solving Eq. (3) we find the next unlabeled sample to label. Unfortunately, this is often a retraining problem and so impractical for image retrieval. Hence we have to make some approximation for the optimization in practice.

Inspired by the work \([39]\), we find that harmonic function after adding the labeled data is related to the last labeled data’s changed probability. This concept can be represented as the following equation:

\[
gu_n^{+(x_i,y_i)} = gu_n + (y_k - gu_k)(A_{uu}^{-1})_k
\]

where \((A_{uu}^{-1})_k\) is the \(k\)-th column of the inverse Laplacian on unlabeled data, and \((A_{uu}^{-1})_{kk}\) is the \(k\)-th diagonal element of the matrix \(A_{uu}^{-1}\). Both are available when computing the harmonic function \(g\). We simplify the formulation because we do not include harmonic energy minimization functions here. We simplify it as follows:

\[
p_n^{+(x_i,y_i)} = p_n + (y_k - p_k)w_{ku}
\]

where \(p_k\) and \(p_n\) denote degree of certainty of the last labeled sample and the unlabeled sample respectively.
$y_k$ is the true label of $x_k$, $w_{ku}$ is the similarity between unlabeled data $x_u$ and the last selected sample $x_k$. We define it as follows: We assume a connected graph $G=(V,E)$ to depict the correlation of the data, where the node set $V$ is the dataset $X$. The edges $E$ are represented by an $n \times n$ weight matrix $W$ which is given. For example $W$ can represent the pairwise relationship between data points with the radial basis function (RBF):

$$w_{ij} = \exp\left(-\frac{||x_i-x_j||^2}{\alpha}\right)$$

(6)

where $\alpha$ is a parameter which controls the influential intensity of the correlated data points. It is obvious that nearby points in Euclidean space are assigned large edge weights with the same $\alpha$. Then we update all the rest unlabeled data's degree of certainty according to their correlation to the last selected data point. This procedure can be repeated until a certain number of samples have been labeled. We will find that after simplifying the formulation there is more flexible to implement the sample selection algorithm.

3.2. Sampling criteria

We implement the sample selection algorithm based on our proposed method asymmetric propagation by controlling the magnitude of $\alpha$ incorporating the criteria uncertainty, diversity and density. We introduce this novel method in detail as follows:

3.2.1. Uncertainty

This sampling criterion aims at selecting the unlabeled samples that can add most information to the current model. Generally, the most uncertain sample in the classification process are selected, such as the sample closest to the hyperplane in SVM [14] according to Eq. (2).

Under probability label metric we infer the same result. The degree of certainty closer to 0.5 an unlabeled data point has, the more uncertainty it is. Every time we only select one single data point with the degree of certainty $p_i$ closest to 0.5 to label.

3.2.2. Density and diversity

There is one parameter, $\alpha$, in our scheme according to Eq. (6), which controls the influential radius of the correlated data points and reflects our batch model sample selection criteria. In order to ensure the generalization of our new scheme, we do not fix the parameter $\alpha$ empirically. Instead, we evaluate them adaptively according to the distribution of the dataset itself. According to kernel density estimation (KDE) [40], we estimate the distribution of the dataset and adapt the parameter $\alpha$ on the basis of the density of the last selected sample. The probability density function $\tilde{p}(x)$ can be estimated by

$$\tilde{p}(x) = \frac{1}{n} \sum_{i=1}^{n} K(x_i,x)$$

(7)

where $K(x_i,x) = \exp(-||x_i-x||^2/\beta)$ is a kernel function. $\beta$ is the parameter of RBF kernel which we set as large as in SVM training. Then the density measure of the selected example can be defined by normalizing to $[0,1]$ as follows:

$$\text{density}(x_i) = \frac{\sum_{j=1}^{n} K(x_i,x_j)}{\max_{i} \sum_{j=1}^{n} K(x_i,x_j)}$$

(8)

Here $x_i$ is the last selected sample. Observe that the term on the right-hand side is in proportion to $\sum_{j=1}^{n} K(x_i,x_j)$. The kernel function is equivalent to the expression below:

$$\log(K(x_i,x)) = -\frac{||x_i-x||^2}{\beta} \propto ||x_i-x||^2 = ||x_i||^2 + ||x||^2 - 2\|x_i\|\|x\|\cos(\theta)$$

(9)

where $\theta$ is the angle between $x_i$ and $x$; this follows from the definition of inner product. And Eq. (9) can be used to estimate the diversity measure [18].

Density is a large mutual distance for points in the sample set in which we estimate the density [41]. We only have to compute the density of the instance and then we get the diversity measure of the subset simultaneously.
Therefore, the information of density include the measure of diversity.

3.3. Unified sampling scheme with asymmetric propagation

The distribution of samples is usually in very broad domain. We should model the distribution of examples and learn the classifier with as few samples as possible. Thus the selected samples should be diversity as much as possible. When a sample is classified right by the current classifier, we can infer that the cluster represented by this sample may be well learned and we can increase the parameter \( \alpha \) to increase the propagation radius. On the other hand, when the sample is classified wrong, we can infer that the cluster represented by this sample may be badly learned or this sample may be an outlier. So the area around the last selected samples is informative. Then we can decrease the parameter \( \alpha \) to decrease the propagation radius and choose the sample near the last selected one.

Finally according to this idea we can define the parameter \( \alpha \) in Eq. (6) incorporated with measurement of density as

\[
\alpha = \frac{b}{\text{density}(x_i)}
\]  

(10)

Here \( b \) is a constant positive coefficient which is less than one. \( \ell(x_i) \in \{0,1\} \) is the true label of the last selected sample which is queried by the user. \( \hat{\ell}(x_i) \in \{0,1\} \) is the predicted label of the last selected sample. When the label is predicted wrong by the classifier in other words \( \hat{\ell}(x_i) \neq \ell(x_i) \), then \( |\ell(x_i) - \hat{\ell}(x_i)| = 1 \). Then Eq. (10) becomes \( \alpha = b/\text{density}(x_i) \). This means when the predicted label is different to the true label we can infer that the cluster represented by this sample may be badly learned. So we should increase the parameter \( \alpha \) to enhance the influence from the just labeled instance. We want to reduce the redundant among the queries as much as possible.

As illustrated in Fig. 2, when the red point is classified wrong by the current classifier, we are more likely to select point C near the labeled point which is predicted wrong to query. And when the red point is classified right, it seems that A and B becomes more informative than C. The method that adaptively adjust the scale parameter of the graph propagation is defined as asymmetric propagation in this paper.

To summarize, the batch mode active learning based on asymmetric propagation algorithm we proposed here is shown in Fig. 3. Here \( h \) is the number of unlabeled examples which have been selected for labeling in one iteration.

4. Experiment

In this section, in order to validate the effectiveness of the proposed APAL algorithm for relevance feedback in image retrieval, extensive experiments are conducted on three publicly available datasets by comparing with some state-of-the-art algorithms.

- SVM active learning: The baseline method for the original SVM active learning algorithm that simply choose the batch size of instances closest to the current decision boundary \([14]\), denoted by SVM_{al}.
- SVM active learning with diversity: The baseline method for batch mode SVM active learning by incorporating diversity among selected samples \([18]\), denoted by SVM^{div}_{al}.
- Batch mode active learning for kernel logistic regression: The state-of-the-art kernel version of batch mode active learning using the kernel logistic regression \([30]\), denoted by KLR_{bmal}.
- Representative sampling with certainty propagation: Based on \([35,15]\), selecting the samples both representative and informative, denoted by QUIRE.
Algorithm 1. Asymmetric Propagation based batch mode Active Learning algorithm (APAL)

Input: a query image

Parameter: $\alpha$ defined in Eq. 10

Output: the relevant images

Procedure:

1. Rank the database in descending order of similarity to the query.
   Select the top-$k$ images for manual labeling as the initial labeled images. Move them to the set $L$;
2. Train $f$ with the labeled training data set $L$ using SVM;
3. Get the class posteriors of the unlabeled data in the set $U$ referring to Eq. 1;
4. $h = 0$;
5. while ($h < k$) do
6. select $x_s$ whose degree of certainty is closest to 0.5;
7. query the label of $x_s$ and move it to the set $L$ and $h = h + 1$;
8. accord the label of $x_s$ to obtain $\alpha$ referring to Eq. 10;
9. update the degree of certainty of rest samples according to Eq. 5;
10. end
11. go to step 2, until user is satisfied.

4.1. Dataset

In this section, we conduct experiments on the USPS dataset, the widely used benchmark Corel dataset and a light version of NUS-WIDE dataset, named NUS-WIDE-OBJECT [42] to evaluate the proposed approach. Some examples from the three datasets are shown in Fig. 4.

We classify handwritten digits from the USPS dataset which contains grayscale handwritten digit images scanned from envelopes by the U.S. Postal Service. There are 10 categories of “0” through “9”. Each category contains 1100 images, 11,000 images in all. These images are of size $16 \times 16$, with pixel values of 8-bit grayscale. We directly employed pixel values to represent these images. In total, a 256-dimensional vector is used to represent each image.

Corel dataset is one of the most used datasets by many groups in the area of image retrieval. We select 102 categories from the Corel image CDs with different semantic meanings, such as tiger, antelope, butterfly, car, cat, dog, horse, lizard, etc. Each category contains 100 images, so there are altogether 10,200 images. In our experiments, the main purpose is to verify if the learning mechanisms of our proposed method are useful, so we only employed simple color and texture features to represent images. The color features include 125-dimensional color histogram vector and six-dimensional color moment vector in RGB. The texture features are extracted using 3-level discrete wavelet transformation (DWT). The mean and variance averaging on each of the 10 subbands are arranged to a 20-dimensional texture feature vector. In total, a 151-dimensional feature vector was extracted from each image.

NUS-WIDE-OBJECT is a real world object image dataset. As a subset of NUS-WIDE, it consists of 31 object categories.
and 30,000 images in total. The low features we used here include the 64-dimensional color histogram, 144-dimensional color correlogram, 73-dimensional edge direction histogram, 128-dimensional wavelet texture, and 225-dimensional block-wise color moments. We combine these features directly by concatenating the five feature vectors of each sample. We normalize the feature vectors of all samples to ensure the square sum of the elements in every feature vector to be one.

4.2. Experimental setup

We simulate Content Based Image Retrieval procedure by querying an image and returning the top images based on the Euclidean distances. In experiments, for USPS and Corel dataset, we select the first 10 images of each category as query. For NUS-WIDE-OBJECT dataset, we randomly select 100 images as query. All the SVM classifiers in our experiments use the same RBF kernel with fixed kernel width. The kernel width is learnt by cross-validation approach. Regarding the parameter setting, the penalty parameter $C$ of SVM is set to 100 (or $\lambda = 0.01$) in all experiments, and the number of initial labeled images and the batch size $k$ are set to the same constant (i.e. 10, 15, 20). The parameter $b$ in Eq. (10) is set to 0.9 empirically.

4.3. Compared schemes

In order to evaluate the performance, we depict the top-N accuracy vs. scope curves of the five algorithms after several rounds and adopt the evaluation metric based on mean average precision follow as Eq. (13) considering 100 queries on handwritten digits dataset and 1020 queries on Corel dataset:

$$\text{AveP} = \frac{\sum_{r=1}^{N} (P(r) \times r)}{\text{number of relevant images}}$$

where $r$ is the rank, $N$ is the number retrieved, $\text{rel}()$ is a binary function on the relevance of a given rank, and $P(r)$ is the precision at a given cut-off rank:

$$P(r) = \frac{|\text{relevant retrieved images of rank } r \text{ or less}|}{r}$$

Table 1

<table>
<thead>
<tr>
<th>Model</th>
<th>Batch</th>
<th>Uncertainty</th>
<th>Diversity</th>
<th>Density</th>
<th>Propagation</th>
<th>Average time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM$_{al}$</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>–</td>
<td>0.1096</td>
</tr>
<tr>
<td>SVM$_{al}^\text{div}$</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>–</td>
<td>0.2742</td>
</tr>
<tr>
<td>KLR$_{mal}$</td>
<td>Yes</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>13.2824</td>
</tr>
<tr>
<td>QUIRE</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Symmetric</td>
<td>2.0181</td>
</tr>
<tr>
<td>one-by-one</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>–</td>
<td>0.5990</td>
</tr>
<tr>
<td>ours</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Asymmetric</td>
<td>0.3277</td>
</tr>
</tbody>
</table>
Mean average precision for a set of queries is the mean of the average precision scores for each query:

\[ \text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP}(q)}{Q} \]

where \( Q \) is the number of queries.

### 4.3.1. Complexity of the algorithm

At each iteration \( t \) the scheduling overhead includes training SVM classifier, predicting the labels and selecting \( k \) instances to query. First, training SVM classifier takes \( O(N_t^3 + (N_t^2)n_{t,\text{labeled}} + N_t d n_{t,\text{labeled}}) \) floating point operations, where \( n_{t,\text{labeled}} \) is the labeled sample size and \( N_t \) is the number of support vectors during iteration, \( d \) is the dimension size of the feature. Second, predicting the label requires \( O(n_{t,\text{unlabeled}} N_s d) \) floating point operations, where \( n_{t,\text{unlabeled}} \) is the unlabeled sample size and RBF kernel is used. Finally, selecting \( k \) instances requires repeating the procedures 6–9 in Fig. 3 \( k \) times which takes \( O(k n_{t,\text{unlabeled}} d) \) floating point operations as we can obtain kernel density estimation and similarity measure in Eq. (5) simultaneously. Putting the both things together, the overall cost at iteration \( t \) is \( O(N_t^3 + (N_t^2)n_{t,\text{labeled}} + N_t d n_{t,\text{labeled}} + n_{t,\text{unlabeled}} N_s d + k n_{t,\text{unlabeled}} d) \).

The algorithms are all written in MATLAB and executed on 2.8 GHz Core2 processor. Table 1 shows the average running time of our method compared with other methods at the first iteration which takes on the USPS dataset and the batch size is set to 5. The method one-by-one selects just one instance after training the classifier once and in this experiment we repeat this procedure 5 times to select five queries. In SVMal we just select the top five queries closest to the hyperplane at the same time. In KLRbmal they chose a batch of examples that effectively maximizes the Fisher information of a classification model, which consider these criteria implicitly. So we fill in the grids with “–” about KLRbmal. We can see that our proposed batch mode method is efficiency effectiveness compared to other methods.

### 4.3.2. Experimental results on USPS dataset

Table 2 shows the accuracy vs. scope results after one round of relevance feedback and the number of initial labeled images and the batch size \( k \) are set to 10, where scope = \( x \) means the accuracy is calculated within top \( x \) returned images. We can see that most of the batch mode active learning algorithms, KLRbmal, QUIRE and our proposed method have better performance than the baseline method SVMal. However, SVMal still outperforms the method SVMal. It shows that only considering diversity has negative improvement under such circumstances. It seems that the method of active learning with diversity tends to select outliers when consistency of inner class is high in the dataset.

We also show the performance of different algorithms under the different batch size in Table 3. It shows the average

![Fig. 5. The top-20 accuracy vs. varying iteration on USPS dataset when the batch size is 10.](image-url)
precision of top-20 returned images at the first iteration with the batch size set to 10, 15, 20 respectively. From Table 3, we can see that the proposed algorithm has almost the same performance as KLRlmal here.

In USPS dataset the number of outlier is few and the number of class is small. So our density criterion can have a good performance to explicitly estimate the data distribution. We can see that our method obtain the best performance which is comparable to the KLRlmal immediately in Fig. 5. Our approach achieves about 5% accuracy improvement at the first round as compared to SVMlal. The methods considering the criterion density or representative like KLRlmal, QUlRE, and our scheme all have good performance on this dataset.

### 4.3.3. Experimental results on Corel dataset

Figs. 6 and 7 show the accuracy vs. scope curves after three and four rounds of relevance feedback and the number

<table>
<thead>
<tr>
<th>Labels</th>
<th>SVMlal</th>
<th>SVMlal*</th>
<th>KLRlmal</th>
<th>QUlRE</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.389</td>
<td>0.382</td>
<td>0.334</td>
<td>0.432</td>
<td>0.446</td>
</tr>
<tr>
<td>15</td>
<td>0.479</td>
<td>0.470</td>
<td>0.500</td>
<td>0.476</td>
<td>0.536</td>
</tr>
<tr>
<td>20</td>
<td>0.560</td>
<td>0.543</td>
<td>0.555</td>
<td>0.509</td>
<td>0.604</td>
</tr>
<tr>
<td>MAP</td>
<td>0.476</td>
<td>0.465</td>
<td>0.463</td>
<td>0.472</td>
<td>0.528</td>
</tr>
</tbody>
</table>

Table 4

MAP of top-20 images for different batch size labeled images at the fourth iteration.

Fig. 6. The accuracy of image retrieval with three feedbacks.

Fig. 7. The accuracy of image retrieval with four feedbacks.
Fig. 8. The top-20 accuracy vs. varying iteration on Corel dataset when the batch size is 20.

Fig. 9. The accuracy of image retrieval with one feedback.

Fig. 10. The accuracy of image retrieval with two feedbacks.
of initial labeled images and the batch size \( k \) are set to 20. First, we can see that most of the batch mode active learning algorithms, \( SVM_{al} \), \( KLR_{bmal} \), \texttt{QUIRE} and our proposed method have better performance than the baseline method \( SVM_{al} \) before the fourth iteration. When the number of iteration increases the difference between the baseline method SVM active learning and other active learning methods decreases. In the fourth iterations the performance of \( KLR_{bmal} \) begins to decent. The fixed number of top eigens and the limited number of labeled samples restrict the performance of the method. In every iteration our proposed method have stable performances which are at least 5% better than \( SVM_{al} \).

Moreover, we also show the result of five algorithms under the different batch size in Table 4. It shows at the fourth iteration the average precision of top-20 returned images when the batch size are set to 10, 15, 20 respectively. From Table 4, we can see that the proposed algorithm has the best performance here.

In Corel dataset there are many images with ambiguous semantic and some images are outliers for the relevant clusters of classes. So at first the methods considering the criterion density or representative like \texttt{QUIRE}, and our scheme only achieve about 2% compared to \( SVM_{al} \) as shown in Fig. 8. Our method obtains better performance than \texttt{QUIRE} as our proposed method asymmetrically chooses the queries.

4.3.4. Experimental results on NUS-WIDE-OBJECT dataset

Figs. 9 and 10 show the accuracy vs. scope curves after one and two rounds of relevance feedback and the number of initial labeled images and the batch size \( k \) are set to 5 for different active learning methods. Several observations can be drawn from the experimental result in the figures. First, we can see that most of the batch mode active learning algorithms, \( SVM_{al}^{div} \), \( KLR_{bmal} \), \texttt{QUIRE} and our proposed method have better performance than the baseline method \( SVM_{al} \), especially when the labeled samples are scarce. When the number of iteration increases the difference between the baseline method \( SVM_{al} \) and the other batch mode active learning methods decreases. Second, in each iteration our proposed method has stable performances which are at least 5% better than \( SVM_{al} \). At the second iteration, our method is able to achieve 2–3% improvement on the runner-up. The MAP result are shown in Table 5. It shows at the second iteration the average precision of top-20 returned images when the batch size are set to 5, 10, 15, 20 respectively. We can see that the proposed algorithm has the best performance even on the complex real world dataset.

In the NUS-WIDE-OBJECT dataset the number of outlier is fewer than the Corel dataset. So the density criterion can be well used. At first iteration our proposed method almost has the best performance as shown in Fig. 11.

5. Conclusion

In this paper, we have proposed a novel active learning algorithm for selective sampling in relevance feedback, namely asymmetric propagation based batch mode active learning (APAL). According to the unbalanced distribution of the samples and the personalized feedback of human we provide a explicit way, asymmetric propagation scheme, to measure and incorporating the criteria: uncertainty, diversity and density. Experimental results show that after adopting our novel scheme in the selective sampling, the performance of retrieval can be obviously improved on real-world dataset.

<table>
<thead>
<tr>
<th>Labels</th>
<th>( SVM_{al} )</th>
<th>( SVM_{al}^{div} )</th>
<th>( KLR_{bmal} )</th>
<th>\texttt{QUIRE}</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.472</td>
<td>0.477</td>
<td>0.481</td>
<td>0.494</td>
<td>0.518</td>
</tr>
<tr>
<td>10</td>
<td>0.623</td>
<td>0.61</td>
<td>0.58</td>
<td>0.655</td>
<td>0.668</td>
</tr>
<tr>
<td>15</td>
<td>0.673</td>
<td>0.682</td>
<td>0.65</td>
<td>0.763</td>
<td>0.758</td>
</tr>
<tr>
<td>20</td>
<td>0.767</td>
<td>0.77</td>
<td>0.671</td>
<td>0.796</td>
<td>0.802</td>
</tr>
<tr>
<td>MAP</td>
<td>0.634</td>
<td>0.635</td>
<td>0.596</td>
<td>0.677</td>
<td>0.687</td>
</tr>
</tbody>
</table>

Table 5
MAP of top-20 images for different batch size labeled images at the second iteration.

Fig. 11. The top-20 accuracy vs. varying iteration on NUS-WIDE-OBJECT dataset when the batch size is 5.
Acknowledgments

This work was supported in part by the 973 Program under Project 2010CB327905, by the National Natural Science Foundation of China under Grant nos. 61170127, 60975010, 60833006, and 61070104.

References