

# WEIGHTED CO-SVM FOR IMAGE RETRIEVAL WITH MVB STRATEGY

Xiaoyu Zhang, Jian Cheng, Hanqing Lu, Songde Ma

National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China  
{xyzhang, jcheng, luhq}@nlpr.ia.ac.cn, mostma@gmail.com

## ABSTRACT

In relevance feedback, active learning is often used to alleviate the burden of labeling by selecting only the most informative data. Traditional data selection strategies often choose the data closest to the current classification boundary to label, which are in fact not informative enough. In this paper, we propose the Moving Virtual Boundary (MVB) strategy, which is proved to be a more effective way for data selection. The Co-SVM algorithm is another powerful method used in relevance feedback. Unfortunately, its basic assumption that each view of the data be sufficient is often untenable in image retrieval. We present our Weighted Co-SVM as an extension of Co-SVM by attaching weight to each view, and thus relax the view sufficiency assumption. The experimental results show that the Weighted Co-SVM algorithm outperforms Co-SVM obviously, especially with the help of MVB data selection strategy.

**Index Terms**— Image retrieval, Relevance feedback, Active learning, Multi-view learning, Support vector machine

## 1. INTRODUCTION

Relevance feedback is an important method to improve the performance of image retrieval systems. In each round of feedback, only a small number of images in the huge image database are labeled. How to utilize both the limited labeled images and the abundant unlabeled images, and conduct effective relevance feedback has been studied for years.

Active learning is a machine learning method used for effective labeling. Its key idea is choosing the “most informative” data so that knowing the labels of them can greatly boost the performance of the classifier. The selected data will then be labeled and added to the training set to retrain the classifier. This procedure can be repeated, and the goal is to label as little data as possible to achieve a certain performance. Specifically in the field of image retrieval, Tong and Chang [1] proposed an active learning algorithm, named SVM<sub>Active</sub>. They consider the images closest to their current separating hyperplane to be the most informative data. Therefore, in each round of relevance feedback, they select the images closest to the current SVM boundary to label, and retrained a new classifier that better separates the images relevant to user’s query from the rest.

Multi-view learning is another useful learning method using the large amount of unlabeled data to augment classification performance. It is often applied to problems with redundant views. The main idea of multi-view learning is that the same data set can be classified from different views, so classifiers with each single view can be trained separately. Then all predictions on the unlabeled data set can be put together to improve the performance of each other. Most multi-view learning algorithms, such as co-training [2], co-testing [3], etc., are proposed on the assumption that each view is sufficient for learning.

In order to combine the merits of active learning and multi-view learning, Cheng and Wang [4] presented their algorithm, named Co-SVM, for image retrieval. They also assume that different features representing an image are sufficient. In their algorithm, SVM classifiers are firstly learnt with different features separately, and then contention data on which the classifiers disagree are picked out. Finally, the contention data close to boundaries in all views are returned to users for labeling.

In this paper, we indicate that traditional data selection strategies which simply select the data closest to the current classification boundary are not effective enough. We propose a novel informative data selection strategy, the Moving Virtual Boundary (MVB) strategy, which uses the label of current selected data as guidance for the selection of next data. Then, on the basis of Co-SVM, we present our multi-view active learning algorithm, Weighted Co-SVM. We introduce weights into Co-SVM to relax the view sufficiency assumption which often fails in image retrieval, and thus achieve better classification performance.

## 2. MOVING VIRTUAL BOUNDARY STRATEGY

As mentioned above, the key point with active learning is to find out a way to choose the most informative data for labeling. Traditionally, as in SVM<sub>Active</sub> and Co-SVM, the data closest to the current trained separating hyperplane are simply chosen as the most informative ones.

However, as illustrated in Fig. 1, the data closest to the current classification boundary are not always the most informative data, and sometimes they can be of little help to improve the performance of the classifier, especially when the current classifier is quite different from the optimal one. In fact, the data closest to the optimal classifier, rather than

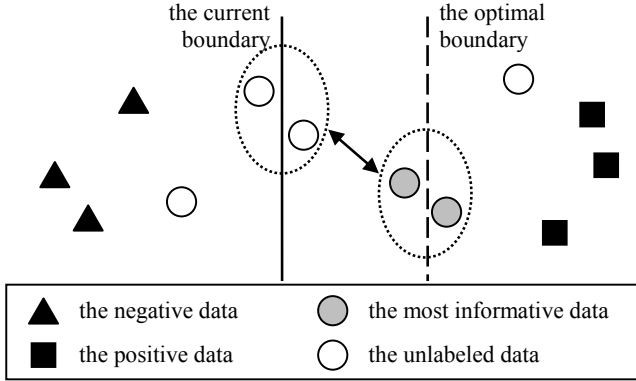


Fig. 1. The data closest to the current classification boundary is not so informative as those in gray since they are quite far from the optimal classification boundary.

the current trained classifier, are the most informative data. Unfortunately, we have totally no idea where the optimal classification boundary lies when selecting data.

In this paper, we propose the Moving Virtual Boundary (MVB) strategy for informative data selection in relevance feedback, which is described in Fig. 2. Different from the traditional strategy which selects a batch of data at a time and labels them all together, our MVB strategy selects data one by one, using the label of the current selected data to guide the selection of the next one (Fig. 3).

The main idea of MVB strategy is to construct a virtual classification boundary moving from the current trained boundary towards the optimal one, and in the meantime select the data closest to the virtual boundary to be labeled. As the virtual boundary moving closer to the optimal boundary, the labeled data are also converging to the most informative data near the optimal boundary.

In every repeat of Step 2 in the MVB strategy, only one single data with the smallest distance to the virtual boundary is labeled, and after each labeling, whether the virtual boundary should move or stay is decided by the labeled data. If the data is correctly classified by the virtual classifier, we call it a “not-very-informative data”, because it offers little information about whether the virtual boundary is correct and where the optimal boundary lies. In this situation, we keep the virtual boundary where it was. If the data’s label does not agree with its classification result according to the virtual classifier, it means that the virtual boundary is not correct and should move to a better position. For example, if a data  $x_l$  is classified positive by the virtual classifier but labeled negative, it means that the optimal classification boundary lies on the positive side of the virtual boundary, so we move the virtual boundary in that direction.

According to the moving step length of the virtual boundary, we can divide the MVB strategy into a naive one and a fast one. In the naive MVB strategy, the virtual boundary simply moves to the middle of the labeled data and its neighbor in the moving direction. In the fast MVB strategy, we define the “unknown space”,  $S_{unknown}$ , as the range of “signed distance” among which the class labels of

**Input:** The current trained classifier  $h$ , the training data set  $D_t$ , the unlabeled data set  $D_u$  and the number of data to be labeled  $N_l$  in a single round of relevance feedback.

**Step1.** Classify  $D_u$  with  $h$ , and initialize the virtual boundary (classifier)  $h_v$  to be the current boundary  $h$ .

**Step2.** Repeat the following procedure for  $N_l$  times:

- (1) Select the data closest to the virtual boundary to be classified by  $h_v$  and labeled, supposing it is  $x_l$ .
- (2) If  $x_l$  is classified positive but labeled negative, move the virtual boundary in the positive direction; If  $x_l$  is classified negative but labeled positive, move the virtual boundary in the negative direction; Otherwise, the virtual boundary stays where it was.
- (3) Add  $x_l$  to  $D_t$  and remove it from  $D_u$ .

**Step3.** Train a new classifier  $h'$  using  $D_t$ .

Fig. 2. The Moving Virtual Boundary (MVB) strategy.

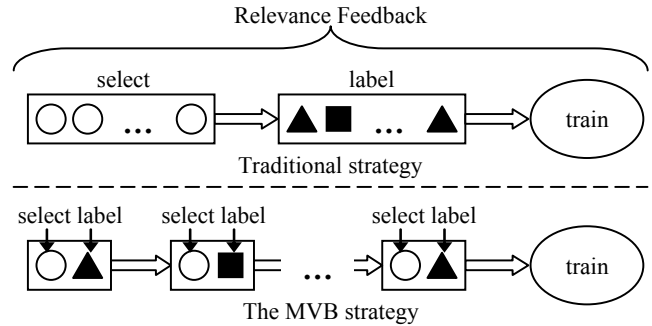


Fig. 3. Traditional strategy vs. MVB strategy.

the data are totally unknown. Here, the signed distance means the distance between a data and the current boundary with a sign (“+” or “-”) to indicate the data’s class. Initially, as we know nothing about the unlabeled data set before the first labeling, we set the unknown space to be  $S_{unknown} = (-Max_{negative}, Max_{positive})$ , where the  $Max_{negative}$  and  $Max_{positive}$  represent the largest distance of the data on the negative and positive side respectively. Every time a wrongly classified data is detected, the range of the  $S_{unknown}$  is changed, and the virtual boundary moves to the position where the new  $S_{unknown}$  is halved. To illustrate, if a data  $x_l$  is classified negative but labeled positive, we claim that most data on the positive side of  $x_l$  are also positive, so we change the initial unknown space into  $S'_{unknown} = (-Max_{negative}, h(x_l))$ , where  $h(x_l)$  is the signed distance of  $x_l$ . Then we move the virtual boundary to the middle of  $S'_{unknown}$ , which divides it in half. With larger step length, the fast MVB moves much faster than the naive one, so that the selected data converge to the most informative data in fewer steps.

Our MVB strategy is a generic data selection strategy, and can be used in most machine learning algorithms with selective sampling.

### 3. WEIGHTED CO-SVM

Most multi-view learning algorithms are based on the

assumption that different views of data are sufficient for learning. In other words, all views are equally important. For example, in Co-SVM, they claim that features, such as color, texture, etc., are all sufficient views of an image, and defined the confidence degree as follows to reflect the consistency of all classifiers with different views:

$$D(x) = \left| \sum_{i=1}^k \text{sign}[h_i(x)] \right|,$$

where  $k$  is the number of features ( $k \geq 2$ ) and  $h_i$  corresponds to the classifier trained solely with the  $i$ th feature. As can be seen from the formula above, each classifier's contribution to  $D(x)$  is the same.

However, Yan and Naphade [5] pointed out that the view sufficiency assumption often fails in semantic concept classification, and different views are not necessarily of the same importance. So it is not appropriate to simply put together different single-view classifiers.

In this paper, we introduce weights into Co-SVM to represent different importance of different features, and propose our Weighted Co-SVM algorithm.

As illustrated in Fig. 4, the Weighted Co-SVM starts with building a training set by labeling the  $l$  most similar images with all features. Then, in the first step of the repeat in Step 2,  $m$  most relevant images classified by a SVM classifier with all features are also labeled and added to the training set. The purpose is to provide the following SVM classifiers with enough positive data (i.e., relevant images), because in a large image database the number of irrelevant images is always much larger than that of relevant images. If we choose images randomly instead, we will most probably get few positive data, which makes it very hard to train robust classifiers in the following steps.

After we have trained  $k$  SVM classifiers with  $k$  corresponding features separately, we combine them into a much stronger classifier:

$$H = \sum_{i=1}^k w_i h_i.$$

With the aid of the validation data set  $D_v$ , the Weighted Co-SVM algorithm decides the importance of each feature by examining how much benefit can be achieved from each single  $h_i$ , and calculates the weight  $w_i$  accordingly. There are many ways to get  $w_i$  for each  $h_i$ . A simple choice is using each single classifier  $h_i$  to classify  $D_v$ , and regarding the classification accuracy as its combination weight  $w_i$ .

At the end of the repeat in Step 2 of Weighted Co-SVM, we select  $n$  images for labeling, and we can use both the traditional strategy and the novel MVB strategy proposed above. If we use the MVB strategy here, we set the input as: classifier  $h = H$ , training set  $D_t$ , unlabeled set  $D_u$  and the number of data to be labeled  $N_l = n$ .

After several rounds of feedback, the classifier  $H$  is well trained and strong enough. At last, the most relevant images are returned.

Our Weighted Co-SVM algorithm can be regarded as the extension of Co-SVM in situation where the view

---

**Input:** The query image  $q$ , the validation data set  $D_v$ , the training data set  $D_t$  ( $= \emptyset$ ), the unlabeled data set  $D_u$  and the entire feature space  $V = V_1 \times V_2 \times \dots \times V_k$  ( $k \geq 2$ ).

**Step 1.** Rank the data in  $D_u$  in descending order of similarity to  $q$  with entire feature  $V$ . Label the top  $l$  images. Add the labeled images to  $D_t$  and remove them from  $D_u$ .

**Step 2.** Repeat the following procedure until the user is satisfied with the result:

- (1) Train a SVM classifier  $h$  with entire feature  $V$  using  $D_t$  and classify  $D_u$ . Label the top  $m$  images with the highest  $h(x)$ . Add the labeled images to  $D_t$  and remove them from  $D_u$ .
- (2) Train SVM classifiers  $h_1, h_2, \dots, h_k$  separately with each single feature  $V_1, V_2, \dots, V_k$  using  $D_t$  and classify  $D_u$ .
- (3) Combine  $h_1, h_2, \dots, h_k$  into  $H = \sum_{i=1}^k w_i h_i$ , using  $D_v$  to determine the combination weights  $w_i$ .
- (4) Select  $n$  informative images to label. Add the labeled images to  $D_t$  and remove them from  $D_u$ .

**Step 3.** Return the images most relevant to  $q$ .

---

**Fig. 4. The Weighted Co-SVM algorithm.**

sufficiency assumption fails.

## 4. EXPERIMENTS

Our experiments are performed on a subset selected from the Corel image CDs. In our subset, there are 50 categories with different semantic meanings, such as tiger, car, flag, etc. Each category contains 100 images, so there are altogether 5000 images. In the experiments, the first 10 images of each category, 500 in all, are picked out as query images to test the retrieval performance of different algorithms. We employ color and texture features to represent images. The color features consist of 125-dimensional color histogram and 6-dimensional color moment in RGB. The texture features are extracted using 3-level discrete wavelet transformation, and the mean and variance averaging on each of 10 sub-bands form a 20-dimensional vector. All the SVM classifiers in our experiments use the same RBF kernel. In each round of relevance feedback, we label the same amount of images selected by different algorithms. All the accuracy is the average accuracy over all the test images.

Our experiments consist of two parts as follows:

### 4.1. Comparison of Data Selection Strategies

Firstly, we compare our MVB strategy (both naive and fast) with the traditional data selection strategy which simply chooses the data closest to the current classification boundary. To provide a fair comparison, we apply the three data selection strategies to Tong and Chang's SVM<sub>Active</sub>.

Fig. 5 shows the accuracy vs. scope curves of SVM<sub>Active</sub> algorithm with the three different data selection strategies, where scope =  $x$  means the accuracy is calculated within the top- $x$  returned images. Fig. 6 depicts the curves of top-30 accuracy vs. feedback rounds. The figures indicate that our MVB strategies consistently outperform the traditional

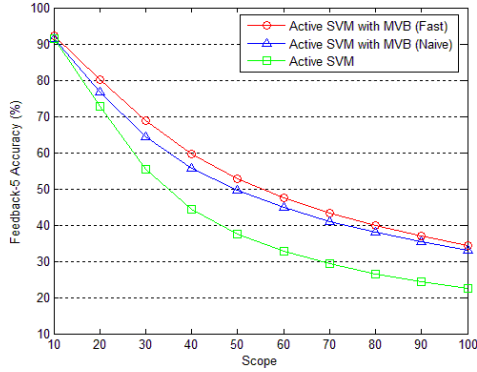


Fig. 5. The accuracy of image retrieval with 5 feedbacks

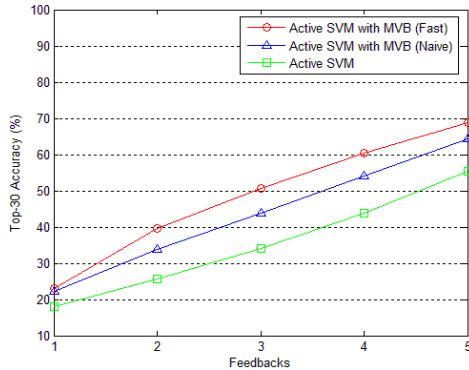


Fig. 6. The top 30 accuracy of image retrieval

strategy, and within the MVB strategies, the fast one performs even better than the naive one.

#### 4.2. Comparison of Learning Algorithms

In order to validate the effectiveness of our Weighted Co-SVM algorithm (with and without MVB strategy), we compare it with Co-SVM and  $SVM_{Active}$  as well as the traditional relevance feedback algorithm using SVM. Here we use the 131-dimensional color feature and the 20-dimensional texture feature as two views in both Weighted Co-SVM and Co-SVM.

As illustrated in Fig. 7 and Fig. 8, our Weighted Co-SVM achieves the best performance. Especially when we apply the MVB strategy (fast) in data selection, the performance can be further improved remarkably, which demonstrates that data selection plays a critical role in learning algorithms.

#### 5. CONCLUSIONS

In this paper, we have proposed a novel generic data selection strategy, the MVB strategy, for learning algorithms with selective sampling. With the help of a virtual boundary moving towards the optimal classification boundary, we make the selected data gradually come near to the most informative data. Experimental results show that, the data selected by our strategy are much more informative than those by the traditional strategy. Then we improved the Co-SVM algorithm by giving weight to each view according

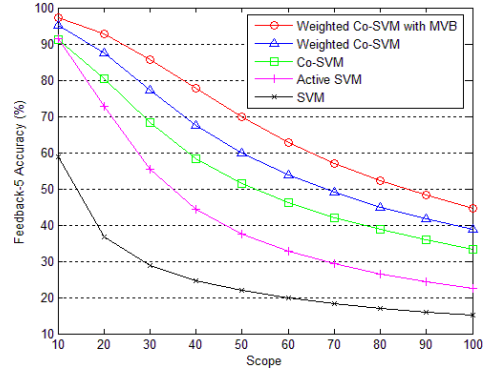


Fig. 7. The accuracy of image retrieval with 5 feedbacks

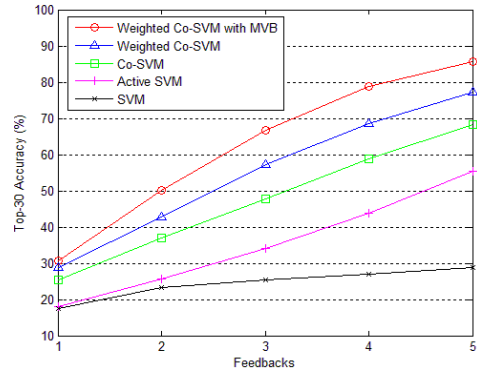


Fig. 8. The top 30 accuracy of image retrieval

to its importance. As the experiments reveal, our Weighted Co-SVM, especially when using MVB strategy, achieves obvious improvement compared with Co-SVM,  $SVM_{Active}$  and traditional algorithms without active learning.

#### 6. ACKNOWLEDGEMENTS

This work is supported by National Hi-Tech R&D Program of China (Grant No. 2006AA01Z315), NSF of China (Grant No. 60121302), NSF of Beijing (Grant No. 4072025), and CASIA Innovation Fund for Young Scientists.

#### 7. REFERENCES

- [1] S. Tong and E. Chang, "Support vector machine active learning for image retrieval", *Proceedings of the 9th ACM International Conference on Multimedia*, pp. 107-118, 2001.
- [2] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training", *Proceedings of the 11th Annual Conference on Computational Learning Theory*, pp. 92-100, 1998.
- [3] I. Muslea, S. Minton, and C.A. Knoblock, "Selective sampling with redundant views", *Proceedings of the 17th National Conference on Artificial Intelligence*, pp. 621-626, 2000.
- [4] J. Cheng and K. Wang, "Active learning for image retrieval with Co-SVM", *Pattern Recognition*, pp. 330-334, 2007.
- [5] R. Yan and M. Naphade, "Semi-supervised cross feature learning for semantic concept detection in videos", *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2005.