

Active Learning to Recognize Multiple Types of Plankton *

Tong Luo, Kurt Kramer
Dmitry B. Goldgof, Lawrence O. Hall
Dept. of Computer Science & Engineering
University of South Florida
Tampa, FL 33620
{tluo2, kkramer, goldgof, hall}
@csee.usf.edu

Scott Samson, Andrew Remsen
Thomas Hopkins
College of Marine Science
University of South Florida
St. Petersburg, FL 33701
{samson, aremsen, thopkins}
@seas.marine.usf.edu

Abstract

Active learning has been applied with support vector machines to reduce the data labeling effort in pattern recognition domains. However, most of those applications only deal with two class problems. In this paper, we extend the active learning approach to multiple class support vector machines. The experimental results from a plankton recognition system indicate that our approach often requires significantly less labeled images to maintain the same accuracy level as random sampling.

1. Introduction

An automated plankton recognition system [2] is capable of collecting and classifying tens of thousands of plankton images an hour. The system is also expected to evolve from a previous model to a more accurate model by adding some new manually labeled images into the training set. It is impossible to manually label all the new images during run time, because they come too fast, thus making active learning attractive.

Tong [6] and Schohn [5] applied the active learning approach to two class support vector machines (SVMs). They chose to label new examples closest to the decision boundary. Compared to random sampling, this active learning approach reduced the number of labeled images needed in their experiments on text classification.

However, little work in active learning has been done with multiple class SVMs. For instance, SVMs handle multiple class problems by building several two-class SVMs. A new example usually has different distances to

the decision boundaries in those two-class SVMs. It is hard to apply the previous active learning approach because we do not know which distance to choose. In this paper, we propose an active learning approach for multiple class SVMs. After developing a probability model for multiple class SVMs, we label the examples with the least classification confidence. We compared our approach with random sampling for the plankton recognition problem. To obtain the same classification accuracy, our approach required many less labeled examples than random sampling. Also, we studied the conditions under which active learning was significantly better than random sampling. We found that active learning with a bad initial classifier (model) was less effective.

2. Support vector machines and the probability model

Support vector machines (SVMs) [7] have received increasing attention recently and have achieved very good accuracy in pattern recognition, text classification, etc.

In binary classification, SVMs first map the data into a higher dimension feature space with $\phi(x)$, then use a hyperplane in that feature space to separate the data. In the feature mapping stage, the kernel $k(x, y) = \langle \phi(x) \cdot \phi(y) \rangle$ is used to avoid explicit inner product calculation in the high-dimension feature space. C-SVM [7], a typical example of soft margin SVMs, is described as follows.

Given m examples: x_1, x_2, \dots, x_m with class label $y_i \in \{-1, 1\}$.

C-SVM:

$$\text{minimize } \left(\frac{1}{2} \langle w, w \rangle + C \sum_{i=1}^m \xi_i \right) \quad (1)$$

$$\text{subject to: } y_i (\langle w, \phi(x_i) \rangle + b) \geq 1 - \xi_i \quad (2)$$

*The research is partially supported by the United States Navy, Office of Naval Research, under grant number N00014-02-1-0266 and the NSF under grant EIA-0130768.

where w is normal to the class separating hyperplane, C is a scalar value that controls the trade off between the empirical risk and the margin ($\frac{2}{|w|}$), ξ_i is the slack variable to handle non-separable examples, b is a scalar value, and $C, \xi_i > 0$.

The decision function is $f(x) = \sum_i \alpha_i k(x_i, x) + b$, where α_i and b are computed from Eq. (1) and (2).

The Karush-Kuhn-Tucker condition of the optimal solution to Eq. (1) and (2) is:

$$\alpha_i(\langle w, \phi(x_i) \rangle + b - 1 + \xi_i) = 0 \quad (3)$$

An α_i is nonzero only when Eq. (4) is satisfied. In this case the x_i contributes to the decision function and is called a support vector (SV).

$$y_i(\langle w, \phi(x_i) \rangle + b) = 1 - \xi_i \quad (4)$$

We applied the one-vs-one approach to extend SVMs for multiple class problems. All possible groups of 2 classes are used to build binary SVMs. In the N class case, we will build $\frac{N(N-1)}{2}$ binary SVMs.

2.1. Assigning probability values in support vector machines

A probability associated with a classifier is often very useful and it provides confidence about the classification result. Platt [3] introduced the sigmoid function as the probability model to fit $P(y = 1|f)$ directly. The parametric model is shown in Eq. (5).

$$P(y = 1|f) = \frac{1}{1 + \exp(Af + B)} \quad (5)$$

where A and B are scalar values, which are fit with maximum likelihood estimation. f is the decision function of the binary SVM.

We followed the sigmoid model and extended it to the multi-class case. In the one-vs-one multi-class SVM model, since it is time consuming to do the parameter fitting for all $\frac{N(N-1)}{2}$ binary SVMs, we developed a practical approximation method to compute the probability value while avoiding parameter fitting.

1. We assume $P(y = 1|f = 0) = P(y = -1|f = 0) = 0.5$. It means that a point right on the decision boundary will have 0.5 probability of belonging to each class. We get rid of parameter B in this way.
2. Since each binary SVM has a different margin, a crucial criterion in assigning the probability, it is not fair to assign a probability without considering the margin. Therefore, the decision function $f(x)$ is normalized by its margin in each binary SVM. The probability model of SVMs is shown in (6) and (7). P_{pq}

represents the probability output for the binary SVM on class p vs. class q , class p is +1 and class q is -1.

$$P_{pq}(y = 1|f) = \frac{1}{1 + \exp(\frac{-Af}{\|w\|})} \quad (6)$$

$$P_{pq}(y = -1|f) = 1 - P_{pq}(y = 1|f) = P_{qp}(y = 1|f) \quad (7)$$

3. Assuming $P_{pq}, q = 1, 2, \dots$ are independent, the final probability for class p is computed as follows:

$$P(p) = \prod_{q \neq p} P_{pq}(y = 1|f) \quad (8)$$

Normalize $P(p)$ to make $\sum_p P(p) = 1$.

4. Output $k = \arg \max_p P(p)$ as the prediction.

A is determined through numeric search based on the cost function L from (9), where t_i is the true class label of x_i .

$$L = \sum_i \log P(t_i) \quad (9)$$

[2] provides detailed description and experiments for this probability model.

3. Approach

We make use of an estimated probability that we calculate for each classification from a modified support vector machine. The algorithm is as follows:

1. Start with an initial training set and an unclassified set of images.
2. A multi-class support vector machine is built using the current training set.
3. Compute the probabilistic outputs of the classification results for each image on the unclassified set. Assign each image to the class with highest probability and associate that probability as the classification confidence.
4. Remove the image from the unclassified set that has the smallest classification confidence, obtain the label from human experts and add it to the current training set.
5. Go to 2.

4. Experiments

The data used for the experiments consists of plankton images that have been classified by a marine biologist. The source of these images is an underwater scanning device referred to as SIPPER [4] that takes a continuous scan of particles in the water. A separate application extracts discrete images from these scans and creates image files. The images consist of three bit grayscale values giving 8 levels. Based on [2], 17 features were extracted from images of the five most abundant types of plankton. There are 7440 images in total: 1488 images from each type of plankton. 1000 images (200 each type of plankton) were randomly selected as the validation set.

The Libsvm [1] support vector machine software was modified to produce probabilistic outputs. A one-vs-one approach was applied to multi-class classification. In all experiments the gaussian radial basis function (RBF) was used as the kernel.

The gaussian RBF kernel: $k(x, y) = \exp(-g\|x - y\|^2)$ where g is a scalar value.

The three parameters (g, C, A) were optimized by performing a grid-search across the initial data set consisting of 200 images per class. A five-fold cross validation was used to evaluate each combination of parameters based on the loss function L from (9). The parameters (g, C, A) were varied with a certain interval in the grid space. Since the parameters are independent, the grid-search ran very fast in parallel. The values of $g = 0.04096$, $C = 16$, and $A = 100$ were found to produce the best results.

We did a series of retraining for active learning and random sampling on the training data with N randomly selected images per class as the initial training set. The experiment was performed 30 times and the average statistics were recorded.

Several variations of the procedure described above were performed. We varied both the number of images per class in the initial training sets (IPC) and the number of images selected for addition to training sets at each retraining step (IPR). Due to space limits, we only describe experiments with $IPR=1$.

4.1. Experiments with $IPC=200, IPR=1$

As shown in Figure 1, with starting accuracy of 83.29% and 200 images from each class, active learning is very effective. Random selection required approximately 4 times the number of images to reach the same level of accuracy as selection by lowest probability.

Active learning tries to label the most "informative" new images, thus helping improve the classifier. In SVM, the decision boundary is represented by support vectors (SVs). Therefore, a good active learning approach should find

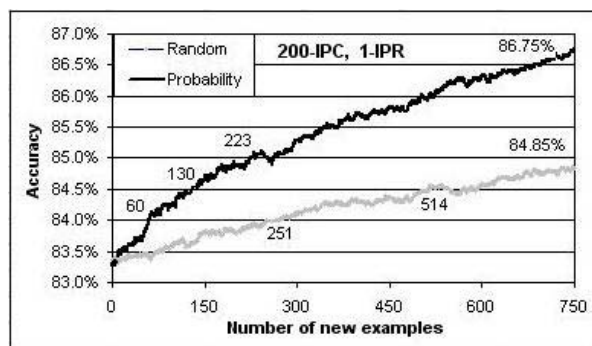


Figure 1. Comparison of active learning and random sampling in terms of accuracy: initial training set is 200, one new labeled image added at a time.

more SVs than random sampling. Figure 2 shows the average number of SVs versus the number of images added into initial training set from the 30 runs. Active learning labeled many more SVs than random sampling. Also, the slope of active learning curve is 0.88, which means that 88 percent of labeled images turn out to be SVs. Our active learning approach efficiently captured support vectors.

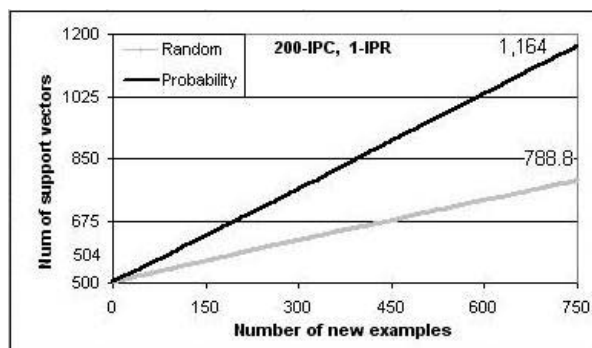


Figure 2. Comparison of active learning and random sampling on chosen examples which became SVs: number of SVs vs. number of additional images

4.2. Varying the initial training set (IPC)

By starting with only 10 images per class in the initial training sets we started off with rather poor accuracy (64.63%). With poor initial accuracy, active learning did not give us a significant advantage over random selection until the accuracy reached a high enough level. Figure

3 shows that until accuracy reached a high enough level (81%), active learning gave no advantage over random selection.

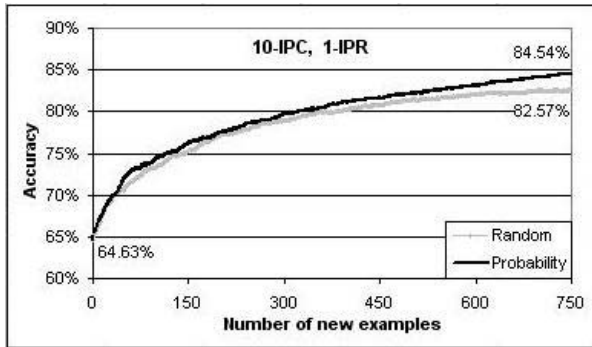


Figure 3. Comparison of active learning and random sampling: initial training set is 10, one new labeled image added a time.

In Figure 4, with 50 IPC in the initial training set we started with 77% accuracy. As compared with 10 IPC, the accuracy for active learning improved faster than random selection. Random selection required almost twice as many new images to achieve the same level of accuracy as selection by lowest probability.

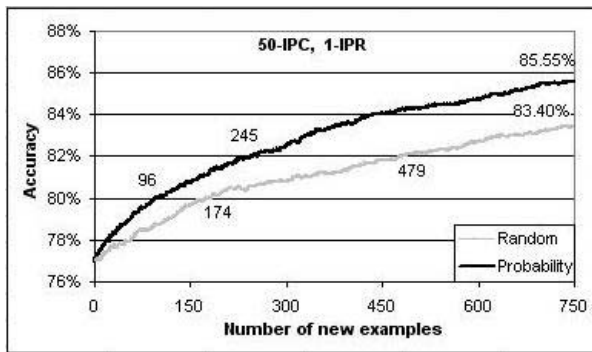


Figure 4. Comparison of active learning and random sampling: initial training set is 50, one new labeled image added a time.

It makes sense that the initial classifier affects the performance of active learning and random sampling. Active learning greedily chooses the most "informative" examples based on the previous model. So a bad model may mislead the active learning approach to choose non-informative examples, which do not help to improve the classifier. While random sampling provides the classifier

with average "informative" examples whatever the initial classifier is. Therefore, if the initial classifier helps active learning to choose examples more informative than average (random sampling), active learning wins. The better the initial classifier, the more labeling effort is saved. Otherwise, active learning may potentially perform worse than random sampling. Due to space limits, we skip the experimental detail and only describe the results. In two-class problem experiments which only dealt with two types of plankton, random sampling outperformed active learning in some cases when the initial classifiers were not very accurate and unbalanced. An unbalanced classifier means that classification accuracy is high for one type of plankton but very low for the other type of plankton.

5. Conclusions

This paper presents an active learning approach for multiple class SVMs. It was applied to reduce the labeling effort on the plankton recognition problem. The experimental results showed that our active learning approach successfully reduced the number of labeled images while maintaining the same accuracy level as random sampling. We found that the goodness of the initial classifier affects the performance of active learning. A somewhat accurate initial classifier that has approximately the same error on all classes allows active learning to be more effective.

References

- [1] C. Chang and C. Lin. LIBSVM: a library for support vector machines (version 2.3). 2001.
- [2] T. Luo, K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, and T. Hopkins. Recognizing plankton images from the shadow image particle profiling evaluation recorder. In *to appear in IEEE Transactions in System, Man and Cybernetics, Part B*, 2004.
- [3] J. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74, 2000.
- [4] S. Samson, T. Hopkins, A. Remsen, L. Langebrake, T. Sutton, and J. Patten. A system for high resolution zooplankton imaging. *IEEE Journal of Ocean Engineering*, pages 671–676, 2001.
- [5] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proc. 17th International Conf. on Machine Learning*, pages 839–846. Morgan Kaufmann, San Francisco, CA, 2000.
- [6] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In P. Langley, editor, *Proceedings of ICML-00, 17th International Conference on Machine Learning*, pages 999–1006, Stanford, US, 2000. Morgan Kaufmann Publishers, San Francisco, US.
- [7] V. N. Vapnik. *The nature of statistical learning theory*. Springer, 2000.