An Empirical Study of Building Compact Ensembles

Huan Liu, Amit Mandvikar, and Jigar Mody

Computer Science & Engineering
Arizona State University
Tempe, AZ 85281
{huan.liu,amitm,jigar.mody}@asu.edu

Abstract. Ensemble methods can achieve excellent performance relying on member classifiers' accuracy and diversity. We conduct an empirical study of the relationship of ensemble sizes with ensemble accuracy and diversity, respectively. Experiments with benchmark data sets show that it is feasible to keep a small ensemble while maintaining accuracy and diversity similar to those of a full ensemble. We propose a heuristic method that can effectively select member classifiers to form a compact ensemble. The idea of compact ensembles is motivated to use them for effective active learning in tasks of classification of unlabeled data.

Keywords: Image Mining, Ensemble Methods, Compact Ensemble

1 Introduction

Many Applications generate massive unlabeled data in forms of text, image, audio or multimedia. A commonly seen task in real-world applications is *classification*. Using an image classification task as an example, a user may be able to carefully study a few images at a time, but may not have the patience and consistent performance to hand-label a hundred of training images. To make things worse, new images are being collected and waiting to be classified. A real-world problem we face is to classify Egerai Densa in images. Egeria is an exotic submerged aquatic weed causing navigation and reservoir-pumping problems in the west coast of the USA. As a part of a control program to manage Egeria, classification of Egeria regions in aerial images is required. This task can be stated more specifically as one of classifying massive data *without class labels*. Relying on human experts for labeling Egeria regions is not only time-consuming and costly, but also inconsistent in their performance of labeling. Massive manual classification becomes impractical when images are complex with many different objects (e.g., water, land, Egeria) under varying picture-taking conditions (e.g., deep water, sun glint). In order to automate Egeria classification, we need to ask experts to label images, but want to minimize the task.

Many data mining methods for classification are available to help massively process image data. In order for classification methods to work, labeled data is needed for training purpose. We face a dilemma: to classify unlabeled data, we need to rely on a classification algorithm; in order for the classification algorithm to learn from the training data, we need to have data labeled. Since we have to have labeled data for training, we ask if it is possible that the classification algorithm can learn with as few labeled data as possible. By doing so, we can minimize the labeling efforts by experts to turn unlabeled data to labeled one. Active learning [5] is an effective learning framework that can be applied in the above process of working with domain experts and using as few labeled data as possible and learn to perform classification through an iterative (re)learning process. Active learning requires highly accurate classifiers that ideally can generalize well with a small set of labeled data. Therefore, we examine one type of highly accurate classifiers - ensemble methods. We analyze one ensemble method (Bagging [2]) with experiments on benchmark data sets, observe interesting results from the experiments, and evaluate its feasibility and effectiveness to use compact ensembles for active learning.

2 Ensemble Methods

Ensemble methods are learning algorithms that construct a set of classifiers and then classify new instances by combining the individual predictions. An ensemble often has smaller expected loss or error rate than any of the n individual (member) classifiers [7]. A good ensemble is one whose members are both *accurate* and *diverse* [3].

An accurate classifier is one that has an error rate of better than random guessing on new instances; more specifically, each member classifier should have its error rate below 0.5. Two classifiers are diverse if they make different (or uncorrelated) errors on new data points. In reality, the errors made by member classifiers will never be completely independent of each other, unless the predictions themselves are completely random (in which case the error rate will be greater than 0.5) [3]. However, so long as each member's error rate is below 0.5, with a sufficient number of members in an ensemble making somewhat uncorrelated errors, the ensemble's error rate can be very small as a result of voting. Out of the many proposed ensemble methods, we consider Bagging in this work as it is the most straightforward way of manipulating the training data [3]. Bagging relies on bootstrap replicates of the original training data to generate multiple classifiers that form an ensemble. Each bootstrap replicate contains, on the average, 63.2% of the original data, with several instances appearing multiple times.

Ensembles can be assessed by three measures: predictive accuracy, diversity, and size. Accuracy can be estimated using cross validation. Diversity measures how different the predictions member classifiers made in an ensemble. The first two measures are of the goodness of an ensemble. The last one is about ensemble size - the number of member classifiers required for ensemble learning. Ensemble size mainly hinges on the complexity of the training data. For a fixed type of classifier (say, decision trees), the more complex the underlying function of the data is, the more members an ensemble needs. The complexity of the function can always be compensated by increasing the number of members for a given type of classifier until the error rate converges.

Following [4], let $\hat{Y}(x) = \hat{y}_1(x), ...\hat{y}_n(x)$ the set of the predictions made by member classifiers $c_1, ..., c_n$ of ensemble C on instance $\langle x, y\rangle$ where x is input, and y is prediction. The **ensemble prediction** of a uniform voting ensemble for input x under loss function l is $\hat{y}(x) = arg \min_{y \in Y} E_{c \in C}[l(y, \hat{y}_c(x))]$. The ensemble prediction is the one that minimizes the expected loss between the ensemble prediction and the predictions made by each member classifier c for the instance $\langle x, y\rangle$. The **loss** of an ensemble on instance $\langle x, y\rangle$ under loss function l is given by $L(\langle x, y\rangle) = l(\hat{y}(x), y)$. The error rate of a data set with N instances can be calculated as $e = \frac{1}{N}\sum_1^N L_i$ where L_i is the loss for instance x_i. **Accuracy** of an ensemble is $1 - e$[1]. The **diversity** of an ensemble on input x under loss function l is given by $\overline{D} = E_{c\in C}[l(\hat{y}_c(x), \hat{y}(x))]$.

The diversity is the expected loss incurred by the predictions of the member classifiers relative to the ensemble prediction. Commonly used loss functions include square loss ($l_2(\hat{y}, y) = (\hat{y} - y)^2$), absolute loss ($l_{||}(\hat{y}, y) = |\hat{y} - y|$), and zero-one loss ($l_{01}(\hat{y}, y) = 0$ iff $\hat{y} = y$; $l_{01}(\hat{y}, y) = 1$ otherwise). We use zero-one loss in this work. We conduct experiments below.

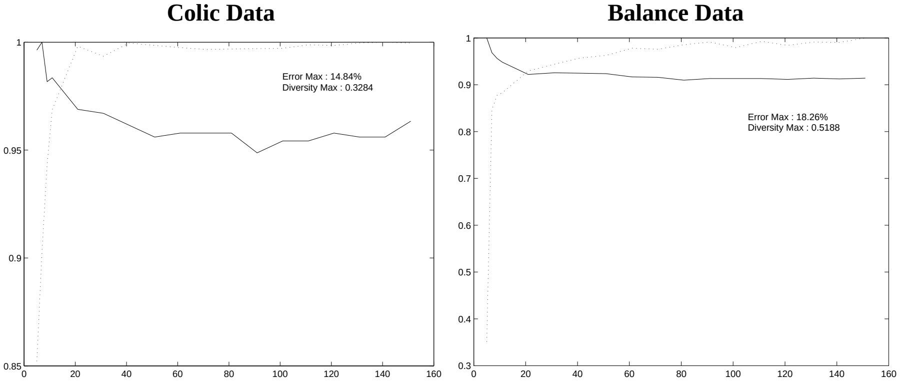

Fig. 1. Plots for normalized diversity and error rates.

Having ensemble loss (or accuracy) and diversity defined, we investigate how ensemble *sizes* influence ensemble *accuracy* and *diversity*. We use benchmark data sets [1] in the experiments. These data sets have different numbers of classes, different types of attributes and are from different application domains.

We use the Weka [6] implementation of Bagging [2] as the ensemble generation method and used J4.8 (the Weka's implementation of C4.5 without pruning as the base learning algorithm. For each data set, we run 10-fold cross validation of Bagging and increase ensemble sizes from 5 to 151 and record each ensemble's error rate e and diversity D. Their average values $\overline{e}$ and $\overline{D}$ are calculated.

We have run experiments with 18 ensemble sizes (5, 7, 9, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101, 111, 121, 131, 141, and 151) with 10-fold cross validation for each data set (29 sets in total). In Figure 1, two illustrative sets of curves are demonstrated. Both diversity values (**dashed lines**) and error rates (**solid lines**) are normalized for plotting purposes. The vertical axis shows percentage (p). The maximum values of diversity and error rate are given in each figure. We can derive absolute values for diversity and error rates following $Max \times p$. The trends of diversity and error rates are of our interest. We

[1] We use *loss* and *error* interchangeably in this paper.

can observe a general trend that diversity values increase and approach to the maximum, and error rates decrease and become stable as ensemble size increases.

The results of this study prompt us to think that if we can find the ensemble with minimum size for every application while maintaining accuracy and diversity, we will be able to make the retraining of the ensemble very fast. One way of searching for such ensembles is what we did in our experiments: increasing ensemble sizes until the curves of error rates and diversity stabilize. However, this is a very costly process when ensemble sizes are large. For example, to estimate error rate and diversity of an ensemble with 100 member classifiers using 10-fold cross validation, we need to build 100×10 classifiers. The total number of classifiers required to build starting an ensemble with 5 classifiers is $(5 + 6 + ... + 100) \times 10 = 50,400$ with each classifier taking $O(N \log N)$ time to train, where N is the number of instances of a training data set. There is a need for an alternative that can determine ensemble size without training so many classifiers.

3 Compact Ensembles via Classifier Selection

In general, 50-100 member classifiers have been used to build ensembles. In the context of active leaning, since the initial training of ensembles is off-line, it is feasible to have an ensemble with 100 member classifiers. To generate a training set for the task of selecting member classifiers, we first perform Bagging with 100 member classifiers. We then use the learned classifiers (c_k) to generate predictions for instance $\langle x_i, y_i \rangle$: $\hat{y}_i^k = c_k(x_i)$. The resulting data set consists of instances of the form $((\hat{y}_i^1, ..., \hat{y}_i^M), y_i)$. After this data set is constructed, the problem of selecting member classifiers becomes one of feature selection.

As discussed before, when member classifiers are equally good, the key issue to find a subset of classifiers that maintain diversity. When we have the newly formed training data, selecting a subset of diverse classifiers is equivalent to selecting a subset of features using diversity of the subset as the goodness criteria. In order to build dual ensembles, we divide data D_{new} into two data sets D_{new}^1 and D_{new}^0 according to the class labels. For each data set, we can calculate the full ensemble's diversity, then look for the smallest subset of classifiers that can maintain this diversity. This will result in two ensembles E_1 for D_{new}^1 and E_0 for D_{new}^0.

Selecting Diverse Classifiers

```
1. Divide data $D_{new}$ according to its last column $y_j$
      form data $D_{new}^1$ and $D_{new}^0$ for classes 1 and 0;
2. For data set $D_{new}^1$
      Calculate diversity $D_{full}$ for ensemble $E_{full}$;
      $S_{best} = S_{full}$;
      $D_{best} = D_{full}$;
      Apply $LVFd$ with $D_{full}$ as a goodness criterion:
         $E_{temp}$ is generated by $LVFd$;
         Calculate $D_{temp}$ and $S_{temp}$ for $E_{temp}$
         If $D_{temp} \approx D_{full} \wedge S_{temp} < S_{best}$
            $D_{best} = D_{temp}$;
            $S_{best} = S_{temp}$;
3. Repeat step 2 for data $D_{new}^0$
```

Fig. 2. Searching for small ensembles.

We implement a modified version of LVF - a filter model of feature selection algorithm (its implementation can be found in Weka [6]) and we call it $LVFd$ as it uses diversity instead of consistency as the goodness measure of feature subsets. The basic idea is as follows: randomly generate a subset of classifiers as E_{temp}; calculate the diversity D_{temp} and size S_{temp} of E_{temp}; if D_{temp} is similar to the diversity D_{full} of the full ensemble C, E_{temp}'s diversity and size are remembered as D_{best} and S_{best}; in the subsequent steps, only if a new E_{temp}'s diversity D_{temp} is similar to D_{full} and size S_{temp} is smaller than S_{best}, D_{best} and S_{best} with E_{best} are updated with those of E_{temp}. $LVFd$ stops when S_{best} does not change after a given number of iterations. The algorithm of $LVFd$ is given in Figure 2 in which $\approx$ means "is similar to". Similarity can be defined by the measures for two comparing ensembles. In our implementation, we define $p \geq 1$ as a threshold for similarity definition: if the difference between the two measures is less than $p\%$, we consider they are similar. p is set as 1 in our experiments.

4 Experiments

Table 1. Selected Compact vs. Full Ensembles for training data

Dataset	E_{full}			E_s					Acc Diff
	Accuracy	$DivE1$	$DivE0$	Accuracy	$DivE1$	$DivE0$	$SizeE1$	$SizeE0$	
breast	98.43 ± 0.12	0.04	0.18	97.00 ± 0.51	0.04	0.02	8	13	-1.43
breast-c	83.21 ± 1.72	0.21	0.06	79.72 ± 1.53	0.22	0.07	17.33	7	-3.50
colic	87.64 ± 1.05	0.08	0.04	87.77 ± 1.15	0.09	0.05	26.33	11	0.13
credit-a	93.62 ± 0.24	0.07	0.08	91.74 ± 0.51	0.07	0.09	18.33	5	-1.88
credit-g	92.90 ± 0.64	0.27	0.10	88.30 ± 0.26	0.26	0.10	19	10	-4.60
diabetes	95.31 ± 0.16	0.21	0.10	90.36 ± 0.56	0.21	0.11	18.33	11	-4.95
heart-st	96.67 ± 0	0.13	0.10	92.78 ± 2.00	0.94	0.11	18.33	10	-3.89
hepatits	94.84 ± 0.52	0.06	0.26	89.36 ± 1.64	0.08	0.28	8.33	19	-5.48
ionosphr	99.72 ± 0.12	0.03	0.09	95.58 ± 1.37	0.04	0.10	3.67	13	-4.13
kr	99.56 ± 0.05	0.01	0.01	99.23 ± 0.16	0.01	0.01	6.33	4	-0.33
labor	73.68 ± 8.13	0.09	0.12	92.11 ± 0	0.11	0.14	7.67	22	18.42
vote	97.36 ± 0.25	0.02	0.02	96.78 ± 0.47	0.03	0.02	8.33	11	-0.58

We now conduct some further study on the benchmark data sets and evaluate if we can apply the algorithm in Figure 2 to find compact ensembles with comparable performance of full ensembles.

Among the benchmark data sets used in Section 2, we use those with binary classes for further experiments using dual ensembles - building one ensemble for each class. As we discussed earlier, an ensemble with 100% accurate member classifiers has diversity value 0. Hence, this data set is excluded. 3-fold cross validation is conducted in this experiment. We record the results in the training and testing phases. For the training data, we record diversity values for E^1_{full} and E^0_{full} and for E^1_s and E^0_s as well as their accuracy rates in Table in Figure 3. We also record the average ensemble sizes for E^1_s and E^0_s.

Table 2. Selected Compact vs. Full Ensemble for testing data

Dataset	E_{full}	E_s	Acc Diff
	Accuracy	Accuracy	
breast	96.57 ± 0.73	95.85 ± 1.04	-0.72
breast-c	68.51 ± 2.70	65.71 ± 2.62	-2.80
colic	85.59 ± 1.61	85.05 ± 1.47	-0.54
credit-a	87.39 ± 0.36	83.04 ± 0.71	-4.35
credit-g	75.60 ± 0.85	69.80 ± 0.75	-5.80
diabetes	75.26 ± 1.29	69.01 ± 2.30	-6.25
heart-st	80.74 ± 1.32	74.81 ± 1.68	-5.93
hepatits	80.67 ± 1.74	76.80 ± 2.33	-3.87
ionosphr	92.02 ± 1.86	86.32 ± 1.85	-5.70
kr	99.28 ± 0.09	98.78 ± 0.16	-0.50
labor	77.19 ± 3.79	84.21 ± 0	7.02
vote	95.40 ± 0.82	94.94 ± 0.50	-0.46

We observe that (1) **Diversity** - Selected ensembles can maintain similar or higher diversity than those of full ensembles (Table 1) on the training data. (2) **Ensemble**

size - Average sizes for E_1 and E_0 are 13.33 and 11.33, the size difference between selected ensembles and full ensembles is about 75 (the reduction is significant). (3) **Accuracy** - Selected ensembles have lower accuracy than that of the full ensemble on both training and testing data (on average, 1.01% and 2.49% less, respectively). Along with the reduction in ensemble size, it is reasonable.

5 Conclusions

Classification of unlabeled data is a common task in real-world applications. We discuss an application of classifying unlabeled images for the purpose of detecting Egeria. Motivated by this application and aiming to alleviate the burden of experts to manually label numerous images, we propose to employ active learning. We analyze what is required to apply active learning and conclude that highly accurate ensemble methods can be used as base classifiers for active learning with class-specific ensembles (e.g., dual ensembles for a binary class problem). As active learning is an iterative process, it requires that each base classifier is efficient to train and test. This directly translates to the necessity of using compact ensembles. We suggest that ensemble size can be reduced as long as its diversity is maintained. A classifier selection algorithm is proposed to find the smallest ensemble that maintains similar accuracy. Various experiments are conducted using benchmark data sets for demonstration and validation purposes.

Acknowledgments This work is in part sponsored by CEINT ASU 2004.

References

1. C. Blake and C. Merz. UCI repository of machine learning databases, 1998.
2. L. Breiman. Bagging predictors.
3. T. Dietterich. Ensemble methods in machine learning. In *First International Workshop on Multiple Classifier Systems*, pages 1–15. Springer-Verlag, 2000.
4. M. Goebel, P. Riddle, and M. Barley. A unified decomposition of ensemble loss for predicting ensemble performance. In *Proceedings of the 19th ICML*, pages 211–218. 2002.
5. G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of the 17th ICML*, pages 839–846, 2000.
6. I. Witten and E. Frank. *Data Mining - Practical Machine Learning Tools and Techniques with JAVA Implementations*. Morgan Kaufmann Publishers, 2000.
7. Z.-H. Zhou, Y. Jiang, and S.-F. Chen. Extracting symbolic rules from trained neural network ensembles. *AI Commun.*, 16(1):3–15, 2003.