Optimizing Boosting with Discriminative Criteria

Rong Zhang and Alexander I. Rudnicky

Language Technologies Institute, School of Computer Science
Carnegie Mellon University, Pittsburgh, PA 15213, USA
{rongz,air}@cs.cmu.edu

Abstract

We describe the use of discriminative criteria to optimize Boosting based ensembles. Boosting algorithms may create hundreds of individual classifiers in order to fit the training data. However, this strategy isn't feasible and necessary for complex classification problems, such as real-time continuous speech recognition, in which only the combination of a few of acoustic models is practical. How to improve the classification accuracy for small size of ensemble is the focus of this paper. Two discriminative criteria that attempt to minimize the true Bayes error rate are investigated. Improvements are observed over a variety of datasets including image and speech recognition, indicating the prospective utility of these two criteria.

1. Introduction

The past ten years have witnessed the success of the boosting algorithm in many fields [1]. In boosting, individual classifiers are iteratively trained in a fashion such that hard-to-classify examples are given increasing emphasis. In generalization, the individual classifiers are composed to form the final ensemble that outputs the hypothesis that receives the weighted majority vote.

Theoretically, the number of individual classifiers generated by boosting could be increased to several hundreds in order to fit the training data. However, two reasons make this strategy neither feasible nor necessary. First, it's impossible for some complex classification problems to use a large size ensemble. For example, a practical real-time continuous speech recognition system consists of large number of parameters and requires the decoding for an utterance to be finished in seconds. Therefore, the number of acoustic models trained from boosting should be restricted to a small number, given the current computer technology [2]. Second, an ensemble with hundreds of individual classifiers will overfit to the misclassified training examples that lead to a biased model whose performance is highly impacted by the training noise.

This raises the question of how to improve the performance of a small size ensemble to make it suitable for complex classification problems. Through an analysis of the boosting algorithm, we find that the boosting training is a one-way and locally-optimized process in which the parameters of a classifier are determined only from the current distribution, and won't be changed any more in the future. Therefore, it would be helpful to post-optimize all the classifiers in the ensemble as a whole using a unitary criterion, especially some discriminative criteria. Namely, using discriminative criteria may be able to further optimize the ensemble obtained from boosting training. Our paper attempts to answer the following three questions: (1) Which parameters need be discriminatively optimized? (2) Which discriminative criteria are beneficial for boosting training result? And (3) how to combine boosting and discriminative training?

Figure 1: Adaboost Algorithm

Input: training set $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$.

Initialize:

- Let $B = \{(i, y) \mid 1 \le i \le N, y \in \mathrm{Y} \text{ and } y \ne y_i\}$.
- Initialize distribution of training data: $D_1(i, y) = 1/|B|$ for all $(i, y) \in B$.

For *k*=1 to *K*:

- Train a new classifier f_k with respect to distribution D_k.
- Compute pseudo loss for f_k $\varepsilon_k = \frac{1}{2} \sum_{(i,y) \in B} D_k(i, y)(1 - f_k(x_i, y_i) + f_k(x_i, y))$.
- Set $\beta_k = \varepsilon_k / (1 - \varepsilon_k)$.
- Set importance factor α_k for f_k that $\alpha_k = -\log \beta_k$.
- Update distribution D_k by $D_{k+1}(i, y) = \frac{D_k(i,y)}{Z_k} \beta_k^{\frac{1}{2}(1 + f_k(x_i, y_i) - f_k(x_i, y))}$ where Z_k is the normalization factor.

Output: ensemble $f(x, y) = \sum_{k=1}^{K} \alpha_k f_k(x, y)$

2. Boosting Algorithm

A brief introduction to Boosting algorithm for multi-class classification is given in this section. Suppose we have a training set $\Psi = \{(\mathbf{x}_i, y_i) \mid 1 \le i \le N\}$, where each instance $\mathbf{x}_i$ is a *D*-dimension feature vector that $\mathbf{x}_i \in \mathbf{R}^D$, and y_i is the corresponding class label that belongs to a finite label space Y, i.e. $\mathrm{Y} = \{c_1, c_2, ..., c_M\}$ for a *M*-class task.

The Boosting algorithm is designed to minimize the following loss function (other variants exist).

$$L_{Boosting} = \frac{1}{N} \sum_{i=1}^{N} \sum_{y \in \mathrm{Y} \text{ and } y \ne y_i} \exp^{f(\mathbf{x}_i, y) - f(\mathbf{x}_i, y_i)} \quad (1)$$

where function $f(\mathbf{x}, y)$ could be interpreted as a classifier or recognition model that maps a feature/label pair to some confidence or probabilistic metrics. For example, in Boosting

style acoustic modeling, $f(\mathbf{x}, y)$ usually takes the form of posterior probability function $P(y|\mathbf{x})$.

Different from traditional training methodology that only outputs a single classifier which is optimized under certain criterion, Boosting algorithm generates a set of classifiers $\{f_1, f_2, ..., f_K\}$, which is called as ensemble in some literatures, by manipulating the distribution of training data. In generalization stage, a weighted majority voting strategy is adopted to combine individual classifiers and make the final hypothesis. For example, for a new instance $\mathbf{x}$, its class label is determined by

$$y^* = \arg\max_{y \in Y} \sum_{k=1}^{K} \alpha_k f_k(\mathbf{x}, y) \quad (2)$$

where α_k is a parameter quantifying the importance of classifier f_k. Figure 1 shows the pseudo code of Adaboost algorithm for multi-class classification [3][4].

As we have mentioned, sometimes it's impractical to exploit a large ensemble due to the complexity of individual classifiers and the level of current computer technology. How to improve the accuracy of small size ensemble is the focus of our research. The following section will describe our solution in detail: post-optimize ensemble with discriminative criterion.

3. Parameters to be Optimized

An ensemble has two types of parameters, the model parameters of each individual classifier f_k and the factor α_k quantifying the classifier's importance in final weighting. Note that the trainings for these two types of parameters in the boosting algorithm are two separate processes with different objective functions. For example, in a neural network based ensemble, the network weights are updated by Back-Propagation algorithm minimizing the mean squared error, while α_k is determined on the basis of the performance of *k*-th classifier (see Figure 1).

It is an easily implemented method to update α_k only. We have investigated several approaches, including optimizing α_k with new loss function or using more powerful learner to combine classifiers. For example, in one of our initial experiments, the output of each individual classifier is treated as new features and Support Vector Machine is adopted and optimized based on these features. However, no obvious improvement was observed in the experiments. We believe that the failure is mainly due to the small number of α_k which makes it relatively less important compared with the large number of classifier's parameters. Therefore, in our research the two types of parameters are regarded as parts of an integrated model and are optimized together under same criterion.

4. Discriminative Criteria

Research into the problem of discriminative criteria for classification could be traced back to the 1980s. Traditional learning approaches such as MLE aim to maximize the observation probability of training examples. However, maximizing this probability doesn't directly lead to minimizing classification error. Discriminative criteria are investigated to correct this deficiency by "discriminating" the correct hypothesis from all the other competing hypotheses. In implementation, the discriminative criteria are expressed as differentiable and heuristic loss functions strongly related to class confusion or classification error. The process minimizing the value of loss function results in the reduction of class confusion or classification errors respectively. Two discriminative criteria for post-optimizing ensemble are discussed as follows.

4.1. MCE and Boosting Criteria

The first criterion we investigate is the *Minimum Classification Error* (MCE) [5][6]. Given a *M*-class training set $\Psi = \{(\mathbf{x}_i, y_i) | 1 \le i \le N\}$, the MCE loss function is defined as follows (other variants exist):

$$L_{MCE} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{1 + f^{\xi}(\mathbf{x}_i, y_i) \Big/ [\sum_{y \neq y_i} f^{\eta}(\mathbf{x}_i, y)]^{\frac{\xi}{\eta}}} \quad (3)$$

where the output $f(\mathbf{x}, y)$ could be probability or confidence function with value within [0,1]; ξ and η are empirically set positive constants to control function's shape. Noting the fact that $\mathbf{x}_i$ will be misclassified when $f(\mathbf{x}_i, y_i) < \frac{1}{M-1} \sum_{y \neq y_i} f(\mathbf{x}_i, y)$, the objective function could be interpreted as an approximation of the error rate on training set. As the most commonly used discriminative method, the characteristics of MCE have been well studied, and we won't repeat them in this paper.

The second criterion that we investigate is borrowed directly from the Boosting algorithm (see Formula 1). In Formula 1, $\sum_{y \neq y_i} \exp[f(\mathbf{x}_i, y) - f(\mathbf{x}_i, y_i)]$ is called pseudo loss that measures the classifier's ability to discriminate the desired class y_i from competing class y. Obviously, misclassification will result in a high pseudo loss. The Boosting algorithm we have discussed minimizes this loss function in a sequential-update manner which generates a set of weak classifiers. However, in the research on post-optimization, Formula 1 is treated as a discriminative criterion which goal is to further optimize the entire ensemble. The characteristic of the Boosting criterion is investigated in Section 4.2 with the view of Bayes decision theory.

4.2. Link to Bayes Decision Rule

The probability of classification error for a pattern recognition problem is minimized by Bayes decision rule [7]. Assuming zero-one loss is used, the Bayes rule for classification task is as follows:

Decide y *if* $P(y|\mathbf{x}) > P(y'|\mathbf{x})$ *for all* $y' \neq y$

Where $P(.)$ denotes the *true* distribution of data. Given this decision rule, the expectation of the true Bayes error rate E_{Bayes} is

$$\begin{aligned} E_{Bayes} &= \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} [1 - \max_{y} P(y|\mathbf{x}_i)] \\ &= \int d\mathbf{x} \cdot P(\mathbf{x}) \cdot [1 - \max_{y} P(y|\mathbf{x})] \end{aligned} \quad (4)$$

in which $[1-\max_y P(y|\mathbf{x})]$ is called local Bayes error. Please note that E_{Bayes} is the minimum error rate that we can achieve regardless of what kind of classifier is used.

MCE has been proved to be a perfect upper bound to the true Bayes error rate by theoretic and experimental study. Minimizing the value of MCE's loss function may obtain a model in which performance is close to the true Bayes error. We will show that the Boosting criterion is also a suitable objective function independent of the underlying model. This means that with sufficient training data, minimizing Boosting loss function could also result in a good model with the lowest classification error.

Formula 1 could be rewritten as:

$$\begin{aligned} L_{Boosting} &= \tfrac{1}{N}\sum_{i=1}^{N}\sum_{y\neq y_i}\exp^{f(\mathbf{x}_i,y)-f(\mathbf{x}_i,y_i)} \\ &= \tfrac{1}{N}\sum_{i=1}^{N}[\sum_{y}\exp^{f(\mathbf{x}_i,y)-f(\mathbf{x}_i,y_i)}-1] \\ &= \tfrac{1}{N}\sum_{i=1}^{N}[(\frac{\exp^{f(\mathbf{x}_i,y_i)}}{\sum_{y}\exp^{f(\mathbf{x}_i,y)}})^{-1}-1] \\ &= \tfrac{1}{N}\sum_{i=1}^{N}[Q(y_i|\mathbf{x}_i)^{-1}-1] \end{aligned} \quad (5)$$

where $Q(y_i|\mathbf{x}_i)=\exp^{f(\mathbf{x}_i,y_i)}\Big/\sum_y \exp^{f(\mathbf{x}_i,y)}$. As an often used technique in Softmax Neural Network and exponential model, $Q(y_i|\mathbf{x}_i)$ could be interpreted as the estimated distribution complying with $0<Q(y|\mathbf{x})<1$ and $\sum_y Q(y|\mathbf{x})=1$. By extending the size of training set to infinite, we have:

$$\begin{aligned} L_{Boosting} &= \lim_{N\to\infty}\tfrac{1}{N}\sum_{i=1}^{N}[Q(y_i|\mathbf{x}_i)^{-1}-1] \\ &= \int d\mathbf{x}\cdot P(\mathbf{x})\cdot[\sum_y P(y|\mathbf{x})(Q(y|\mathbf{x})^{-1}-1)] \\ &= \int d\mathbf{x}\cdot P(\mathbf{x})\cdot L_{local}(y|\mathbf{x}) \end{aligned} \quad (6)$$

where $L_{local}(y|\mathbf{x})=\sum_y P(y|\mathbf{x})(Q(y|\mathbf{x})^{-1}-1)$ is the local Boosting loss, whose minimum could be solved by considering the following Lagrangian

$$L_Q = L_{local}(y|\mathbf{x})+\lambda[\sum_y Q(y|\mathbf{x})-1] \quad (7)$$

This derivation result tells us that the minimum of $L_{local}(y|\mathbf{x})$ is reached:

$$L_{\min}=[\sum_y P(y|\mathbf{x})^{\frac{1}{2}}]^2-1 \quad (8)$$

when

$$Q(y|\mathbf{x})=\frac{P(y|\mathbf{x})^{\frac{1}{2}}}{\sum_{y'}P(y'|\mathbf{x})^{\frac{1}{2}}} \quad (9)$$

Formula 9 shows that when use $Q(y|\mathbf{x})$ as the decision rule for classification task, we could obtain same hypothesis as use Bayes decision rule. Namely, the decision rule that

Decide y *if* $Q(y|\mathbf{x})>Q(y'|\mathbf{x})$ *for all* $y'\neq y$

is equivalent to

Decide y *if* $P(y|\mathbf{x})>P(y'|\mathbf{x})$ *for all* $y'\neq y$

since

$$\arg\max_y Q(y|\mathbf{x})=\arg\max_y \frac{P(y|\mathbf{x})^{\frac{1}{2}}}{\sum_{y'}P(y'|\mathbf{x})^{\frac{1}{2}}}=\arg\max_y P(y|\mathbf{x}) \quad (10)$$

This means that in the ideal case the size of training set is infinite and the optimum of $Q(y|\mathbf{x})$ is reached, the expectation of classification error using model $Q(y|\mathbf{x})$ is equal to E_{Bayes}, the minimum error rate that we can achieve. On the other hand, (9) could be understood as a smoothing function that the class with small probability gets more emphasis. This characteristic indicates that the Boosting criterion may be able to avoid overfitting when sufficient training data are not available. Therefore, we believe that the Boosting criterion is also a suitable candidate for discriminative optimization.

5. Combining Boosting Training with Discriminative Criteria

The methods for combining boosting training with discriminative criteria can be divided into two types, sequential-update and post-update. Sequential-update has been investigated in [8]; it works as follows: at each iteration *k* of boosting training, discriminative criteria are used to optimize the *k*-th individual classifier based on the re-weighted distribution; finally, all the discriminatively trained classifiers are linearly combined to form the ensemble. The post-update method regards the ensemble as an integral model and each individual classifier as a part of the model. When boosting training is performed, discriminative criteria are applied to the ensemble, optimizing each classifier's parameters and α_k together. There also exists a third combination method that uses discriminative criteria for both sequential-update and post-update. The experimental results for these three methods will be reported in the next section.

6. Experiments

6.1. Dataset and Configuration

We use five artificial and real world datasets to evaluate the performance of combining boosting training with discriminative criteria. The first four are selected from the UCI machine learning benchmark repositories. These are *crx*, *image-segmentation*, *waveform* and *vowel*. The fields they span include decision-making, image classification and speech recognition. Moreover, we also added a dataset that was collected in the course of our earlier research on confidence annotation, an important topic in speech recognition. Ten fold cross-validations are used except where the dataset comes with a pre-defined training-test division. Table 1 gives the details of these 5 datasets.

Neural Network and Gaussian Mixtures are selected as the base classifiers for boosting and discriminative training. Because these are two commonly used methods in the speech field, we believe their experimental results could give us some initial proof of the effectiveness of combining boosting and discriminative training for continuous speech recognition.

Name	Examples	Classes	Attributes	Classifier
Crx	690	2	15	Neural
Segment	2310	7	19	Neural
Waveform	5300	3	21	Neural
Vowel	990	11	10	Gaussian
Confidence	4783	2	11	Gaussian

Table 1 Datasets

Both the MCE and Boosting criteria, along with all the three combination methods, sequential-update, post-update and both, are investigated in our experiments. Gradient descent is used for discriminative optimization.

6.2. Experimental Results

Our first experiment is to build the baseline for each dataset. Table 2 shows the classification accuracies of boosting training varying with the number of classifiers in the ensemble. Just as we discussed before, increasing the size of an ensemble doesn't always yield the improvement of performance because of the overfitting caused by the noise in training set. To our surprise, Neural Network doesn't work well on the dataset *image-segment*. We believe this is partly due to the simple architecture we selected: one hidden layer with small number of nodes.

Name	# = 1	# = 5	# = 10	# = 20
Crx	60.87%	61.16%	62.03%	62.03%
Segment	29.57%	30.29%	31.62%	29.24%
Waveform	60.30%	78.55%	82.60%	83.05%
Vowel	56.49%	56.28%	58.22%	58.95%
Confidence	77.54%	78.10%	77.92%	78.21%

Table 2 Baseline of Accuracy

The next two experiments verify the effectiveness of combining boosting and discriminative training. The size of ensemble is set to 5, to allow room for improvement. Table 3 and Table 4 give the results for the MCE and Boosting criteria with each of the three combination methods: sequential-update, post-update, and sequential plus post-update.

Discriminative criteria demonstrate encouraging improvement in performance compared to standard boosting training. We observed improvements on all of the five datasets, and obtained substantial increase of classification accuracy on two of them: *image-segment* and *waveform*. The accuracy that we achieved on *vowel* outperforms all other approaches according to the dataset's documentation.

As we expected, the Seq.+Post provides the best performance of the three combination methods. Furthermore, comparison shows that post-update outperforms sequential-update in most of these five datasets. This could be interpreted as indicating that optimizing a single classifier isn't as useful as optimizing the entire ensemble together.

From Table 3 and 4, we observe that the Boosting criterion performs as well as MCE, even though it approaches to a slacker bound than MCE. This demonstrates that the Boosting criterion is also a possible choice for the classification problem. Moreover, we believe that the comparative study of the Boosting with MCE criterion could give us a more thorough understanding of the characteristics of Boosting and discriminative training.

Because we are still in the early stage of this research, the interpretation and conclusion made here are tentative and could be modified by subsequent findings.

Name	Baseline	Seq.	Post	Seq. + Post
Crx	61.16%	63.19%	62.47%	64.64%
Segment	30.29%	64.24%	70.76%	83.52%
Waveform	78.55%	85.90%	86.80%	86.85%
Vowel	56.28%	58.44%	59.74%	60.82%
Confidence	78.10%	79.07%	79.22%	79.68%

Table 3 Discriminative Optimization with MCE Criterion

Name	Baseline	Seq.	Post	Seq. + Post
Crx	61.16%	62.90%	62.75%	62.90%
Segment	30.29%	59.81%	75.76%	82.38%
Waveform	78.55%	86.20%	86.75%	87.05%
Vowel	56.28%	58.44%	58.84%	60.67%
Confidence	78.10%	78.95%	79.10%	79.36%

Table 4 Discriminative Optimization with Boosting Criterion

Acknowledgement

This work was supported under DARPA grant NBCH-D-03-0010. The content of the information in this publication does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred.

7. References

[1] R. E. Schapire, "A Brief Introduction to Boosting", Proc. of 6th IJCAI, 1999.

[2] Rong Zhang and Alex I. Rudnicky, "Comparative Study of Boosting and Non-Boosting Training for Constructing Ensembles of Acoustic Models", Proc. of Eurospeech 2003.

[3] Yoav Freund and Robert E. Schapire, "A Decision Theoretic Generalization of On-line Learning and an Application to Boosting", Journal of Computer and System Science, 55(1): 119-139, 1997.

[4] Holger Schwenk, "Using Boosting to Improve a Hybrid HMM Neural Network Speech Recognizer", Proc. of ICASSP 1999.

[5] B. H. Juang and S. Katagiri, "Discriminative Learning for Minimum Error Classification", IEEE trans. on Signal Processing, Vol. 40, No. 12, 1992.

[6] R. Schlüter, W. Macherey, B. Müller, H. Ney, "Comparison of Discriminative Training Criteria and Optimization Methods for Speech Recognition". Speech Communication, Vol. 34, pp. 287-310, 2001.

[7] R.O. Duda and P. E. Hart, "Pattern Classification and Science Analysis", John Wiley & Sons, New York, 1973.

[8] Imed Zitouni, Hong-Kwang Jeff Kuo, Chin-Hui Lee, "Combination of Boosting and Discriminative Training for Natural Language Call Steering Systems", Proc. of ICASSP 2002.