MULTIPLE CLASSIFIERS FOR ELECTRONIC NOSE DATA

M. Pardo, G. Sberveglieri
INFM and University of Brescia
Gas Sensor Lab, Dept. of Chemistry and Physics for Materials
Via Valotti 9 - 25133 Brescia –Italy

D. Della Casa, G. Valentini, F. Masulli
INFM and University of Genova
Dept. of Computer and Information Sciences
Via Dodecaneso 35 – 16164 Genova -Italy

In this contribution we apply a method -called boosting- for constructing a classifier out of a set of (base or weak) classifiers for the discrimination of two groups of coffees (blends and monovarieties). The main idea of boosting is to produce a sequence of base classifiers that progressively concentrate on the hard patterns, i.e. those which are near to the classification boundary. Measurement were performed with the Pico-1 Electronic Nose based on thin films semiconductor sensors developed in Brescia. The boosting algorithm was able to halve the classification error for the blends data and to diminish it from 21% to 18% for the more difficult monovarieties data set.

INTRODUCTION

An Electronic Nose (EN) can be briefly schematized as consisting of an odor sampling unit, an array of chemical sensors, electronic circuitry and data analysis software. Data analysis, in turn, can be divided in two parts. The first part, sometimes called (data) preprocessing, deals with signal processing (e.g. removal of spikes, noise filtering), the choice of the features to be considered in the subsequent analysis and data visualization, for example with PCA (Principal Component Analysis) score plots. Drift correction can be also considered as part of this first processing of the data. This part of data analysis is crucial for the quality of the final results and requires a constant exchange with the experimental process, mainly to establish a sufficiently good and reliable measurement protocol.

The second part of the data analysis deals with inferring the relationship between the EN data (patterns) and the corresponding class labels (or the continuous quantities in the case of e.g. concentration determination of gas components in a mixture). This is the subject of supervised learning which comprises a collection of general purpose techniques for determining the relationship from data. The use of a *single* neural network (normally a multilayer perceptron, but radial basis functions also been investigated) as a classifier is a common solution to pattern recognition problems in many application fields, comprising EN odor analysis.

A direction in which research in supervised learning is making great progresses is the study of techniques for combining the predictions of multiple classifiers (briefly called *ensembles*) to produce a single classifier (1,2). The resulting classifier is generally more

accurate than any of the individual classifiers making up the ensemble. Both theoretical and empirical research has demonstrated that a good ensemble is one where the individual classifiers are both accurate and make errors on different parts of the input space (that is to say when they are independent).

Two popular methods for creating accurate ensembles which emerge from the recent machine learning literature are Bagging (1) and Boosting (3,4). These methods rely on resampling techniques to obtain different training sets for each of the classifiers. Boosting were An empirical evaluation of these methods on 23 data sets using both neural networks and decision trees as base classifiers is presented in (5).

In this paper we apply boosting to the classification of data collected with the Pico-1 EN developed at the Gas Sensor Lab in Brescia. Experiments were performed on two groups of coffees, consisting respectively of 7 different blends (containing the Italian Certified Espresso (ICE)) and of 6 single varieties (SV) plus the ICE. The food manufacturing sector is one of the two main application fields for EN together with environmental monitoring. The goal is, in the case of coffee, to use the EN on line for coffee quality control, at least to perform a first stage, gross differentiation of the products.

EXPERIMENTAL

Boosting

Boosting consists in an iterative application of a learning algorithm (MLP in our case) to subsets of the training data. The subset is chosen at every step according to a probability distribution of the data that depends on the actual classification errors. At each iteration the probability distribution is updated in order to improve the weights (probabilities) of misclassified examples. The error on the training set is weighted, depending on the probability distribution of the examples. The final hypothesis is computed by a weighted voting of the generated hypotheses. In our implementation we have used boosting by resampling, i.e. we have chosen a set of examples from the training set at random with replacement, according to the current probability distribution of the data.

A pseudocode for boosting (AdaBoost) can be given as follows:
1. Start with weights $w_i = 1/N$, i=1,…,N; $y_i \in \{1,-1\}$
2. Repeat for m=1,…,M:
    a) Estimate the base (weak) learner $f_m(x)$ from the training data with weights w_i
    b) Compute the weighted misclassification error $e_m=sum_j(w_j)$; j index of misclassified samples
    c) Compute the weight of the m-th classifier $f_m(x)$: $c_m=log((1-e_m)\backslash e_m)$
    d) Update the weights of the misclassified examples: $w_j = w_j \exp(c_m)$ and renormalize so that $sum_i(w_i)=1$
3. Output weighted majority classifier: $C(x)=sign[sum_m(w_m f_m(x))]$, where sign(x)= 1 if x>0 and sign(x)= -1 if x<0

The Pico Nose

The Pico-1 EN makes use of six thin film semiconductor sensors. For this experiment three SnO_2-based (one catalyzed with gold, one with palladium and one with

platinum) and three Ti-Fe sensors were employed. All of them were grown by sputtering with the RGTO technique.

The odor to be analyzed can be sampled either in a static way with a programmable autosampler comprising a syringe, or in a dynamic way letting the carrier flush through the headspace, or from stained steel canisters or nalophan bags through a pump. For this application the possibility of easily preparing the sample suggested the adoption of the more reproducible static headspace extraction with the autosampler.

Pico-1 precisely controls the sensor temperature via a feedback loop. Further, there is the possibility to steer the EN remotely via the TCP\IP interface. A simple user interface for the preliminary analysis of data (graphs of sensor responses, time development of extracted features, PCA score and loading plots) has also been implemented in Matlab. A newer version of the Pico Nose is currently in an advanced stage of development: the hardware has been simplified and standardized using commercial components.

RESULTS

We have randomly split the data in a training and in a testing set and we have repeated training of each learning machine six times using different pseudorandom initialization weights. In our experimentation we have used the AdaBoost.M1 algorithm introduced by Freund and Schapire (6) for boosting Multi-Layer Perceptrons (MLP). As base learners we have used MLP with one hidden layer and we have set the maximum number of base learner to 250, i.e. we have fixed the maximum number of rounds of boosting to 250. All the experimentations has been performed using NEURObjects (7), a set of library C++ classes for neural networks development[1].

Results of our experimentation are summarized in Table 1 to Table 4. The tables represent the results on the test sets of blended and monovariety coffee data sets. The first two tables are referred to a single MLP trained with backpropagation algorithm, the last two to boosted MLP ensembles. Each row of the tables show results relative to MLP or boosted MLP with a predefined number of hidden units. The first column of each table refers to the number of hidden units of a single MLP or of a single MLP base learner of the boosted. The next 6 columns correspond to percent error rates obtained by different pseudorandom initialization of the weights of the MLP. The 8th column shows the minimum error achieved (Best), while the next corresponds to the average error (Mean) and the last shows the standard deviation of the percent error rate (Stdev).

Comparing the overall results on the blended coffee data set between MLP and boosted MLP (Table 1 and Table 3), we can remark that the average error (Mean) is halved using boosted MLP ensembles: The percent error rate on the test set drops down from 15.05 to 8.60, using MLP with 7 hidden units as base learners, and similar results are obtained also using MLP with 5 and 9 hidden units. The minimum error, also, is reduced in a similar way, passing from 11.29 to 6.45.

---

[1] NEURObjects software is available on-line for research and education purposes: http://www.disi.unige.it/person/ValentiniG/NEURObjects

Hidden #	Percent error rate on different runs						Best	Mean	Stdev
5	20.97	19.35	16.13	19.35	20.97	16.13	16.13	18.82	2.01
7	11.29	14.52	14.52	11.29	19.35	19.35	11.29	15.05	3.31
9	16.13	17.74	16.13	17.74	17.74	17.74	16.13	17.20	0.76

Table 1 Single MLP results on blended coffees data set.

Hidden #	Percent error rate on different runs						Best	Mean	Stdev
20	21.43	23.21	21.43	21.43	21.43	23.21	21.43	22.02	0.84
30	23.21	23.21	21.43	23.21	21.43	23.21	21.43	22.62	0.84
40	25.00	25.00	23.21	23.21	23.21	23.21	23.21	23.81	0.84

Table 2 Single MLP results on monovariety coffees data set.

Hidden #	Percent error rate on different runs						Best	Mean	Stdev
5	9.68	11.29	9.68	11.29	11.29	11.29	9.68	10.75	0.83
7	6.45	9.68	9.68	9.68	6.45	9.68	6.45	8.60	1.67
9	11.29	9.68	6.45	9.68	6.45	11.29	6.45	9.14	2.20

Table 3 Boosted MLP results on blended coffees data set.

Hidden #	Percent error rate on different runs						Best	Mean	Stdev
20	21.43	21.43	19.64	21.43	19.64	21.43	19.64	20.83	0.92
30	21.43	17.86	19.64	17.86	19.64	21.43	17.86	19.64	1.60
40	23.21	17.86	19.64	19.64	23.21	17.86	17.86	20.24	2.44

Table 4 Boosted MLP results on monovariety coffees data set.

A reduction of the percent error rate, both for the average and the minimum error can be observed also on the monovariety coffee data set (Table 2 and Table 4), but with a remarkably lower decrement. In this case the average error decreases only from 22.02 to 19.64 and the minimum error from 21.43 to 17.86.

Figure 1 and Figure 2 show the error rate of the boosted MLP. The error rate on the training set drops to 0 after about 10 rounds of boosting on the blended coffee data set (Figure 1), and after about 150 rounds on the monovariety coffee data set (Figure 2). In both cases an exponential decrement of the error can be observed, according to Freund and Schapire's theorem stating that the training error exponentially falls to zero incrementing the number of base learners, given that the weighted error of each base learner is less than 0.5. Note that the spikes in the error curves are due to the relative small number of examples in the testing set.

The test error on the blended data set continues to decrease, even after the training error reaches zero. A similar trend can also be noted in the monovariety data set, even if the test error lowers more slowly. This fact has been observed also in other cases (3,4) and has
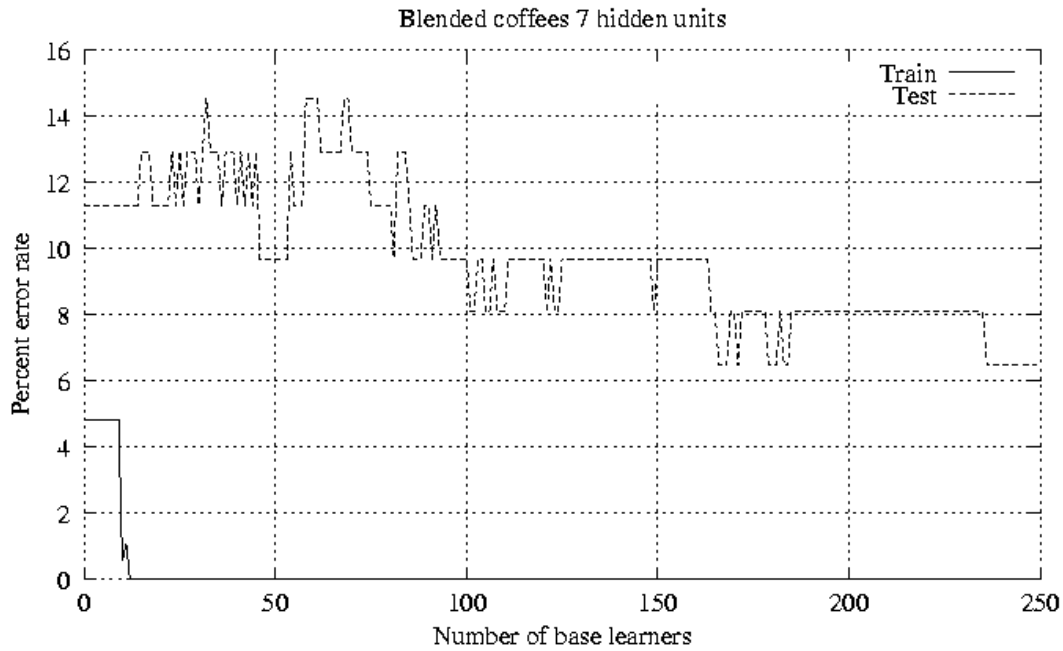
Figure 1 Error curves for boosting MLP on the blended coffees data set. The training and test error curves of the combined classifiers are represented as a function of the number of rounds of boosting. The base classifiers are MLP with 7 hidden units.
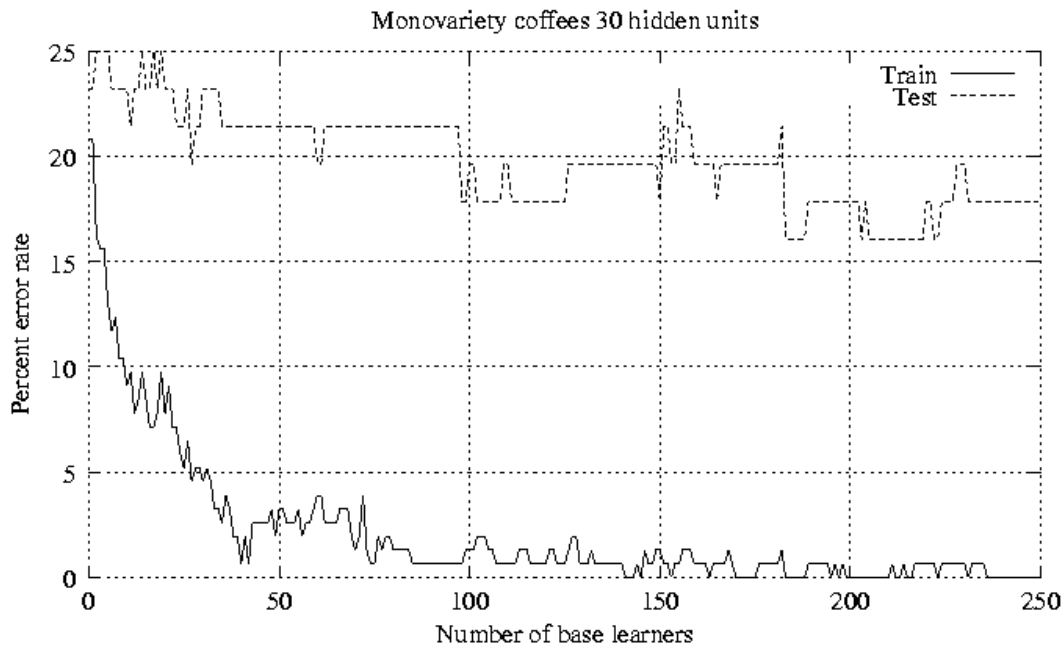

Figure 2 Error curves for boosting MLP on the monovariety coffees data set. The training and test error curves of the combined classifiers are represented as a function of the number of rounds of boosting. The base classifiers are MLP with 30 hidden units.

been explained in the framework of large margin classifiers, interpreting boosting as an algorithm that enlarges the margins of the training examples (3): even if the training error

reaches zero the boosting algorithm continues to enhance the margins, focusing on the hardest examples. As a consequence, the generalization capabilities of the boosted ensemble are improved (3).

The test error error on the monovariety data set decreases slowly compared with the blended data set and using a less complex MLP as base learner the error remains unchanged at about 20 % . Moreover, the training error drops to zero only after more than 100 rounds of boosting. These results on the monovariety coffee data set can be explained by the presence of outliers. The high values of the weights registered in subsets of the data suggest that some data are hard learnable, i.e. they are candidates for being outliers (6). In fact the PCA plot shows that, for one class, a subset of the data is distinctly separated from the others. This could be related to the fact that, for each class, three carousels of vials were analyzed. It is possible that for one of these carousels the autosampler's settings have been changed.

Boosting enhances classification performances, but it requires training of ensembles of learning machines, with increasing computational costs. However, we need to perform an accurate model selection to achieve good results with a single MLP, and this requires an accurate and time consuming planning of the experimentation, while with boosting also a weak learner not accurately tuned for a particular problem can achieve good generalization results. For instance, in the presented experimentation, the worst boosted MLP achieves better result than the best single MLP, both for blended and monovariety data sets. Moreover, we can also note that sometimes a remarkable reduction in the test error is reached even after few iterations of the boosting algorithm (Figure 1), reducing in such a way the computational cost.

CONCLUSIONS

Boosting improve classification performances of electronic noses, reducing in a significant way both the minimum and the average testing error on multiple runs of the boosted ensemble of MLP. Moreover a remarkable reduction of the error is reached even after few iterations of boosting. Even if boosting achieves its best performances with complex algorithms such as C4.5 or backpropagation when there is a reasonably large amount of data available, we have halved the testing error on the blended coffee data set, with only 187 training examples. On the other hand the moderate reduction of the test error achieved on the monovariety coffee data set can be explained considering that boosting is especially susceptible to noise and outliers.

REFERENCES

1. L. Breiman, Machine Learning, **24,** 1 (1996) 49-64
2. T. Dietterich, in Ensemble Methods in Machine Learning, J. Kittler and F. Roli, Springer-Verlag 2000
3. R. Schapire et al., T*he Annals of Statistics*, **26,** 5 (1998):1651-1686
4. R. Schapire, 16th International Joint Conference on Artificial Intelligence, T. Dean Editor, Morgan Kauffman 1997

5. D. Opitz and R. Maclin, J. of Artificial Intelligence Research, 11 (1999) 169-198
6. Y.Freund and R.Schapire, In *Machine learning: Proceedings of the Thirtheenth International Conference*, Morgan Kauffman 1996
7. G. Valentini and F. Masulli, in Proceedings of the third International ICSC Symposia on Intelligent Industrial Automation (IIA'99) and Soft Computing (SOCO'99), ICSC Academic Press 1999