Reduced Reward-punishment editing for building ensembles of classifiers

Loris Nanni *, Annalisa Franco

DEIS, IEIIT, Università di Bologna, viale Risorgimento 2, 40136 Bologna, Italy

ARTICLE INFO

ABSTRACT

In this work a novel technique for building ensemble of classifiers is presented. The proposed approaches are based on a Reduced Reward-punishment editing approach for selecting several subsets of patterns, which are subsequently used to train different classifiers. The basic idea of the Reduced Reward-punishment editing algorithm is to reward patterns that contribute to a correct classification and to punish those that provide a wrong one.

We propose ensembles based on the perturbation of patterns; in particular we propose a bagging-based algorithm and two variants of recent feature transform based ensemble methods (Rotation Forest and Input Decimated Ensemble). In our variants the different subsets of patterns find by the Reward-punishment editing are used to create a different subspace projection (the Principal Component Analysis and the Independent Component Analysis are tested in this work). These feature transformations are applied to the whole dataset and a classifier D_i is trained using these transformed patterns. To combine the set of classifiers obtained the sum rule is used.

Experiments carried out on several classification problems show the superiority of this method with respect to other well known state-of-the-art approaches for building ensembles of classifiers.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

A feasible solution for improving the performance of the classifiers is the combination of multiple classifiers (Kuncheva & Whitaker, 2003). Due to the complex relationships among the data, in difficult classification problems, the performance of the stand-alone methods is too low for practical applications (Lumini & Nanni, 2006), to solve this problem the idea is to average the different hypotheses of different classifiers to produce a good approximation of the true hypothesis (Kittler, Hatef, Duin, & Matas, 1998). In the last years several methods have been proposed in the literature for building a multi-classifier system; the most used idea is to train each different classifier, that belong to the multi-classifier system, with a perturbate version of the training set. The basic steps (see Fig. 1) for building a multi-classifier are:

- to generate K new training sets starting from the original one;
- to train a different classifier for each of the K new training set;
- to combine these K classifiers with a decision rule.

The good performance of the multi-classifier system has been demonstrated both theoretically and empirically (Kittler et al., 1998). In the following the "classic" approaches for constructing a multi-classifier are briefly explained:

- *pattern perturbation*: each new training set is built changing the patterns that belong to the training set. Some examples of this class are: Bagging (Breiman, 1996), the new training sets $S_1,\ldots,S_K$ are subsets of the original one; Arcing (Bologna & Appel, 2002), the patterns contained in each new training set are selected according to the probabilities calculated considering the number of times that a given training pattern is misclassified by the previous $K-1$ classifiers of the multi-classifier system; Class Switching (Martínez-Muñoz & Suárez, 2005), the K training sets are created randomly changing the classes of a subset of the training examples; Decorate (Melville & Mooney, 2005), the K training sets are created by adding artificial patterns whose labels disagree with the current decision of the multi-classifier; Boosting (Freund & Schapire, 1997), a given weight is assigned to each training pattern, the weights are increased at each iteration for the patterns that are difficult to classify (the ith classifier is built considering the patterns that have been difficult to classify for the previous $(i-1)$ classifier of the ensemble); in Nanni and Lumini (2006) the new training sets $S_1,\ldots,S_K$ are obtained considering different clusterization of the training patterns.
- *Perturbation of the features*[1]: each new training set is built changing the feature set. Some examples are: random subspace (Ho, 1998), the new training sets $S_1,\ldots,S_K$ contain only a subset

* Corresponding author.
   E-mail addresses: loris.nanni@unibo.it, lnanni@deis.unibo.it (L. Nanni), annalisa.franco@unibo.it (A. Franco).

---

[1] For this kind of approach few general purpose systems are proposed, for this reason some papers that proposed a system for trying to solve a specific problem are briefly explained
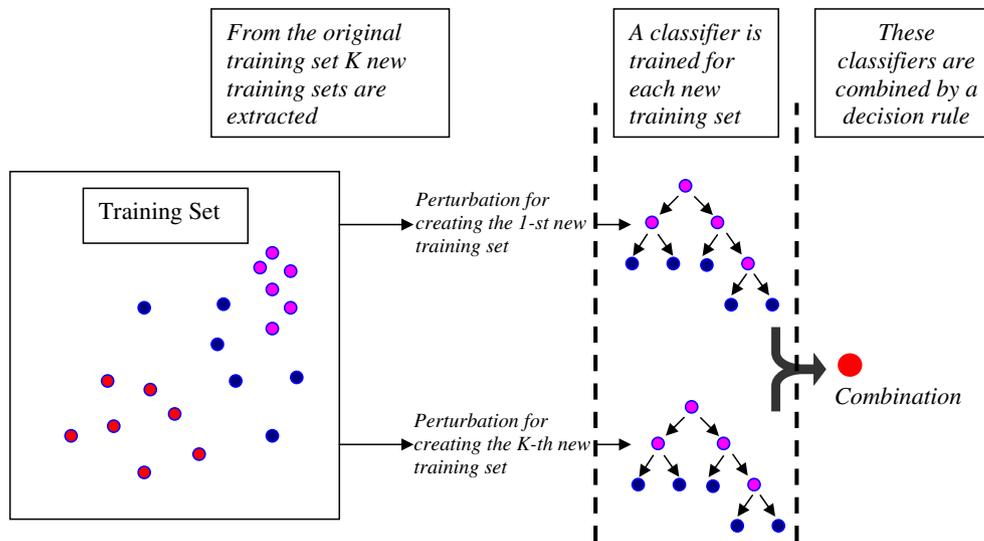
Fig. 1. Multiclassifier system.

of all the features; Cluster-based Pattern Discrimination (Nanni, 2006), the classes are independently partitioned into clusters and a different feature selection is defined for each cluster, these sets of features are used to build the new training sets $S_1, \ldots, S_K$; Input Decimated Ensemble (Tumer & Oza, 2003), the new training set S_i is obtained using the data transformed by the Principal Component Analysis (PCA) transform calculated using the training patterns that belong to class i; to avoid the drawback of Input Decimated Ensemble (the size of the ensemble is bounded by the number of classes), in Nanni and Lumini (2008, 2009) the training patterns are partitioned into clusters and the PCA transform is calculated using the training patterns that belong to each cluster; in Ranawana and Palade (2005) four neural networks trained on different encoding models are combined for identifying *Escherichia Coli* promoter sequences in strings of DNA, the weights of each classifier are determined by a genetic algorithm; in Nanni and Lumini (2006) (for peptide classification) and in Nanni and Lumini (2006) (for protein–protein interaction) a multi-classifier where each training set is built using a different physicochemical property of amino-acids, selected by Sequential Forward Floating Selection (Pudil, Novovicova, & Kittler, 1994), is proposed; in Guo and Lin (2006) the authors show that a system where each classifier is trained using a different feature extraction method outperforms a stand-alone classifier for a protein classification problem.

- *Perturbation of the classifiers*[1]: each classifier has different values for its parameters or different classifiers are combined (the training set does not change). Some examples are: (Lan, Carson, Provart, & Bonner, 2007) where four different classifiers (Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Naive Bayes and *k*-nearest neighbors) are combined, by a weighted-vote decision rule, to predict which genes respond to stress; in Nanni and Lumini (2007) a linear classifier (linear Support Vector Machine), a non-linear classifier (radial-basis Support Vector Machine) and a subspace classifier (Karhunen–Loeve Subspace) are combined and evaluated in different bioinformatics problems: sub-cellular localization, peptide classification and micro-array classification.
- *Hybrid methods*: different perturbations are used. Some examples are here described. Random Forest (Breinman, 2001) consists of a bagging ensemble of decision tree where a random selection of features are used to split a given node. In Rotation

Forest (Rodriguez, Kuncheva, & Alonso, 2006) an ensemble of decision tree is used, where the new training sets $S_1, \ldots, S_K$ are built by several Principal Component Analysis (PCA) projections applied on a subset of the training patterns. Several authors (Nanni & Lumini, 2008, 2009; Liu & Huang, 2008) have tested other feature transform for building a Rotation Forest ensemble, concluding that Independent Component Analysis (ICA) can improve the performance of this method for building multi-classifiers. RotBoost (Zhang & Zhang, 2008) proposes an ensemble of decision tree constructed by combining Rotation Forest and AdaBoost; in Zhang and Zhang (2008) it is shown that RotBoost outperforms Bagging, MultiBoost, Rotation Forest and AdaBoost.

In this paper we propose some variants of feature transform based ensemble methods (Rotation Forest and Input Decimated Ensemble). The different training subsets, used for calculating the feature transform projections, are obtained varying the parameters of the editing technique. The editing technique used is a reduced version of our Reward-punishment editing approach (Franco, Maltoni, & Nanni, in press). Experiments carried out on several classification problems show the superiority of the proposed methods with respect to other state-of-the-art multi-classifier systems as Rotation Forest and RotBoost. Moreover, a bagging-based algorithm is proposed where each different training set of the ensemble contains a different subset of the training patterns, selected by applying the editing technique with different parameter combinations.

Editing techniques (Sanchez, Barandela, Marquez, Alejo, & Badenas, 2003) are an attempt to overcome the drawbacks of nearest neighbours (NN) classifier as CPU and memory expensive, when the number of samples in the training set is high, and the sensitivity to the presence of noise. For a recent survey on the editing techniques please read (Franco et al., in press). Among the several editing techniques recently proposed in the literature two interesting methods are the WPE technique (Paredes & Wagner, 2000) and the Reward-punishment editing (RP-editing) (Franco et al., in press). The first method is based on weighing the training patterns according to the ratio between the dissimilarity of the pattern from others of the same class and the dissimilarity of the pattern from patterns belonging to different classes; the patterns whose weight is higher than a given threshold are discarded. The latter cited editing technique is based on a global criterion, that rewards the train-