Information Sciences 258 (2014) 200–216

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Using Bayesian networks for selecting classifiers in GP ensembles

C. De Stefano [a], G. Folino [b], F. Fontanella [a,*], A. Scotto di Freca [a]

[a] *Dipartimento di Ingegneria Elettrica e dell'Informazione, Università di Cassino e del Lazio meridionale, Via G. Di Biasio 43, 03043 Cassino (FR), Italy*
[b] *ICAR-CNR, Istituto di Calcolo e Reti ad Alte Prestazioni, Via P. Bucci, 87036 Rende (CS), Italy*

ARTICLE INFO

Article history:
Received 2 July 2012
Received in revised form 24 July 2013
Accepted 24 September 2013
Available online 17 October 2013

Keywords:
Genetic programming
Classifier ensemble
Bayesian Networks
Evolutionary computation

ABSTRACT

Ensemble techniques have been widely used to improve classification performance also in the case of GP-based systems. These techniques should improve classification accuracy by using voting strategies to combine the responses of different classifiers. However, even reducing the number of classifiers composing the ensemble, by selecting only those appropriately "diverse" according to a given measure, gives no guarantee of obtaining significant improvements in both classification accuracy and generalization capacity. This paper presents a novel approach for combining GP-based ensembles by means of a Bayesian Network. The proposed system is able to learn and combine decision tree ensembles effectively by using two different strategies: in the first, decision tree ensembles are learned by means of a boosted GP algorithm; in the second, the responses of the ensemble are combined using a Bayesian network, which also implements a selection strategy to reduce the number of classifiers. Experiments on several data sets show that the approach obtains comparable or better accuracy with respect to other methods proposed in the literature, considerably reducing the number of classifiers used. In addition, a comparison with similar approaches, confirmed the goodness of our method and its superiority with respect to other selection techniques based on diversity.

1. Introduction

Ensemble techniques have been taken into account [2,8,25] in the last few years in order to further improve the performance of classification algorithms. They try to combine the responses provided by several experts effectively, in order to improve the overall classification accuracy [28]. Such techniques rely on: (i) a diversification heuristic, used to extract sufficiently diverse classifiers; (ii) a voting mechanism, to combine the responses provided by the learned classifiers. If the classifiers are sufficiently diverse, i.e. they make uncorrelated errors, then the majority vote rule tends to the Bayesian error as the number of classifiers increases [28].

Ensemble techniques have also been used to enhance the performance of classification systems in which decision trees are learned by means of Genetic Programming (GP). Examples of GP–based approaches using ensemble techniques can be found in [6,14,18,24,31]. Moreover, in [6,14,24], ensembles of decision trees are evolved, and the diversity among the ensemble members is obtained by using bagging or boosting techniques. According to these approaches, an ensemble can be obtained by evolving each decision tree with reference to a different subset of the original data. Instead, in the bagging approach [5], different subsets (called *bags*), the same size as the original training set, are obtained by applying a random

* Corresponding author. Tel.: +39 0776 2993382.
E-mail addresses: destefano@unicas.it (C. De Stefano), folino@icar.cnr.it (G. Folino), fontanella@unicas.it (F. Fontanella), a.scotto@unicas.it (A. Scotto di Freca).

0020-0255/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.09.049

sampling with replacement. Then the ensemble is built by training each classifier on a different *bag*. Finally, the responses provided by these classifiers are combined by means of the majority vote rule: an unknown sample is assigned the class label that has the highest occurrence among those provided by the whole set of classifiers. While, in the boosting approach [16], the classifier ensemble is trained by means of a stepwise procedure. At each step, a new classifier is trained by choosing the training set samples from the original training set, according to a suitable probability distribution. This distribution is adaptively changed at each step in such a way that samples misclassified in the previous steps have a better chance of being chosen. Eventually, classifier responses are combined by the weighted majority vote rule, where the weight associated with a classifier takes into account its overall accuracy on the training data.

In [13] a novel GP-based classification system, named Cellular GP for data Classification (CGPC), is presented. In the CGPC approach, individuals interact according to a cellular automata inspired model, whose goal is to enable a fine-grained parallel implementation of GP. In this model, each individual has a spatial location on a low-dimensional grid and interacts only with other individuals within a small neighborhood. In [14], an extension of CGPC, called *BoostCGPC*, is presented. It is based on two different ensemble techniques: the Breiman's bagging algorithm [5] and the AdaBoost.M2 boosting algorithm [16]. Despite the significant performance improvements classifier ensembles can provide, their major drawback is that, usually, it is necessary to combine a large number of classifiers in order to obtain a marked error reduction. This implies large memory requirements and slow classification speeds. In fact, these aspects can be critical in some applications [32,36], but this problem can be solved by selecting a fraction of the classifiers from the original ensemble. This reduction, often called "ensemble pruning" in the literature, has other potential benefits. In particular, an appropriate subset of complementary classifiers can perform better than the whole ensemble [32,43,44,33,4]. When the cardinality L of the whole ensemble is high, the problem of finding the optimal subset of classifiers becomes computationally intractable because of the resulting exponential growth of the search space, made of all the 2^L possible subsets. It is worth mentioning that several heuristic algorithms can be found in the literature for finding near optimal solutions. Examples of such heuristics are: Genetic algorithms (GAs) [43,44] and semidefinite programming [42].

In order to be successful, any ensemble learning strategy should ensure that the classifiers making up the ensemble are suitably diverse, so as to avoid correlated errors [28]. In fact, as the ensemble size increases, it could happen that a correct classification provided by some classifiers is overturned by the convergence of other classifiers on the same wrong decision. If the ensemble classifiers are not sufficiently diverse, this event is much more likely and can reduce the obtainable performance, regardless of any combination strategy. Classifier diversity for bagging and boosting are experimentally investigated in [26,27]. The results have shown that these techniques do not ensure obtaining sufficiently diverse classifiers. As regards boosting, in [26] it is observed that whereas highly diverse classifiers are obtained at the first steps, as the boosting process proceeds, classifier diversity strongly decreases.

More recently, AdaBoost has also been applied for generating more training subsets, used to learn an ensemble of Extreme Learning Machine (ELM) classifiers [40]. This approach tries to overcome some of the drawbacks in traditional gradient-based learning algorithm, and can also alleviate instability and over-fitting problems of ELM. The diversity issue has been also considered in the unsupervised learning of ensembles of clusters [41].

In a previous work [10], the classifier combination problem was reformulated as a pattern recognition one, in which the pattern is represented by the set of class labels provided by the classifiers when classifying a sample. Following this approach, a Bayesian network (BN) [35] was learned in order to estimate the conditional probability of each class, given the set of labels provided by the classifiers for each sample of a training set. Here, we have used Bayesian Networks because they provide a natural and compact way to encode joint probability distributions through graphical models, and allow probabilistic relationships among variables to be derived by using effective learning algorithms for both the graphical structure and its parameters. In particular, the joint probability among variables is modeled through the structure of a Direct Acyclic Graph (DAG), whose nodes are the variables while the arcs are their statistical dependencies. In this way, the conditional probability of each class, given the set of responses provided by the classifiers, can be directly derived by the DAG structure applying the Bayes Rule. Thus, the combining rule is automatically provided by the learned BN. Moreover, this approach makes it possible to identify redundant classifiers, i.e. classifiers whose outputs do not influence the output of the combiner: the behavior of these classifiers is very similar to that of other classifiers in the ensemble. For this reason, they may be discarded without affecting the overall performance of the combiner, thus overcoming the main drawback of the combining methods discussed above. In [11] the learning of the BN is performed by means of an evolutionary algorithm using a direct encoding scheme of the BN structure (DAG). This encoding scheme is based on a specifically devised data structure, called *Multilist*, which allows an easy and effective implementation of the genetic operators. Indeed, the rationale behind the choice of an evolutionary approach is that of trying to solve one of the main drawbacks of standard learning algorithms, such as the k2 one, which adopt a greedy search strategy and thus are prone to be trapped in local optima. On the contrary, evolutionary algorithms allow us to effectively explore complex high dimensional search space.

This paper presents a new classification system that merges the two aforementioned approaches. The goal is to build a high performance classification system, able to deal with large data sets but selecting only a reduced number of classifiers. For this purpose, we built a two-module system that combines the BoostCGPC algorithm [14], which produces a high performing ensemble of decision tree classifiers, with the BN-based approach to classifier combination [11]. In particular, the BN module evaluates classifiers diversity by estimating the statistical dependencies of the responses they provide. This ability is used to select the minimum number of classifiers, among those provided by the BoostCGPC module, required to effectively classify the data at hand. Moreover, the responses provided by the selected classifiers are effectively combined