Neurocomputing 134 (2014) 214–221

Contents lists available at ScienceDirect

Neurocomputing

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

A novel classifier ensemble method with sparsity and diversity

Xu-Cheng Yin [a,*], Kaizhu Huang [b], Hong-Wei Hao [c], Khalid Iqbal [a], Zhi-Bin Wang [d]

[a] Department of Computer Science and Technology, School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China
[b] Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China
[c] Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[d] China National Engineering Research Center for Information Technology in Agriculture, Beijing 100097, China

ARTICLE INFO

Article history:
Received 4 March 2013
Received in revised form
20 June 2013
Accepted 12 July 2013
Available online 27 January 2014

Keywords:
Classifier ensemble
Sparsity learning
Diversity learning
Neural network ensembles
Genetic algorithm

ABSTRACT

We consider the classifier ensemble problem in this paper. Due to its superior performance to individual classifiers, class ensemble has been intensively studied in the literature. Generally speaking, there are two prevalent research directions on this, i.e., to diversely generate classifier components, and to sparsely combine multiple classifiers. While most current approaches are emphasized on either sparsity or diversity only, we investigate the classifier ensemble by learning both sparsity and diversity simultaneously. We manage to formulate the classifier ensemble problem with the sparsity or/and diversity learning in a general framework. In particular, the classifier ensemble with sparsity and diversity can be represented as a mathematical optimization problem. We then propose a heuristic algorithm, capable of obtaining ensemble classifiers with consideration of both sparsity and diversity. We exploit the genetic algorithm, and optimize sparsity and diversity for classifier selection and combination heuristically and iteratively. As one major contribution, we introduce the concept of the diversity contribution ability so as to select proper classifier components and evolve classifier weights eventually. Finally, we compare our proposed novel method with other conventional classifier ensemble methods such as Bagging, least squares combination, sparsity learning, and AdaBoost, extensively on UCI benchmark data sets and the Pascal Large Scale Learning Challenge 2008 webspam data. The experimental results confirm that our approach leads to better performance in many aspects.

1. Introduction

We consider the classifier ensemble problem in this paper. Combining multiple classifiers has been intensively studied and widely regarded as an effective technique for overcoming the limitations of individual classifiers [1–5]. In the literature, there have been many famous models [6–15]. Importantly, these classifier ensemble methods, e.g., neural network ensembles, have been applied in many real-world applications and achieved great success [16,17].

Generally speaking, there are two main categories in the classifier ensemble. The first one aims at learning multiple classifiers at the feature level. Namely, they usually train and combine multiple classifiers in the learning process, e.g., Boosting [18] and Bagging [19]. On the other hand, the second school of methods tries to combine classifiers at the output level, where the results of multiple available classifiers are combined to solve the targeted problem, e.g., multiple classifier systems, or mixture of experts [14]. In this paper, we focus on the second one. Namely, given multiple classifiers (available or sequently learned, homogeneous or heterogeneous), the classifier ensemble is learned by combining these component classifiers intelligently. Furthermore, classifiers differing in feature representation, architecture, learning algorithm, or training data exhibit complementary classification behavior and the fusion of their decisions can yield higher performance than the best individual classifier. In addition to the accuracy of classifier components, the performance of a classifier ensemble also relies on the diversity of the classifier components, and the combination strategy. Consequently, the research efforts in the classifier ensemble have focused on two directions: how to generate and select diverse classifiers and how to combine available multiple classifiers.

First, diversity can ensure that all the individual classifiers craft uncorrelated errors. In the classifier ensemble, diversity learning is performed in two approaches, i.e., seeking implicit or explicit diversity [20]. The common way for the prior approach is to train individual classifiers on different training sets, for example Bagging [19] and Boosting [18,21]. Bagging generates several different training sets with bootstrap sampling from the original training

* Corresponding author. Tel.: +86 10 82371191; fax: +86 10 62332873.
E-mail addresses: xuchengyin@ustb.edu.cn (X.-C. Yin), kaizhu.huang@xjtlu.edu.cn (K. Huang), hongwei.hao@ia.ac.cn (H.-W. Hao), kik.ustb@gmail.com (K. Iqbal), wzb1818@yahoo.cn (Z.-B. Wang).

0925-2312/$ - see front matter
http://dx.doi.org/10.1016/j.neucom.2013.07.054

set and then trains a component classifier from each of those training sets. Boosting generates a series of component classifiers whose training sets are different and determined by the performance of former classifiers. Training instances which are wrongly predicted by former classifiers will play a more important role in the training of later classifiers. Similarly, when combining nearest-neighbor classifiers, Zhou and Yu proposed an ensemble with local learners for utilizing multi-modal perturbation to help generate accurate but diverse component learners [22]. Yu et al. proposed the diversity regularized machine, which efficiently generates an ensemble of assorted support vector machines [23]. More recently, Hosseinzadeh and Reza proposed a new classifier combining strategy with virtual voting by random projection, which used the distortion to virtually generate different training sets from the total available samples [24]. Li et al. tried to present a theoretical study on the effect of diversity on the generalization performance of voting in the PAC-learning framework for the classifier ensemble. Following this analysis, they also proposed the diversity regularized ensemble pruning method [25].

As in the latter approach for diversity learning, the general way is to train multiple classifiers by using different classifier architectures or different feature sets [1,26–28]. Liu investigated and combined classifiers with different structures using a variety of combination rules, such as sum-rule, product-rule, linear discriminants, and weighted combination [26]. Yin et al. proposed a variant of boosting and adaptively integrated classifiers built on different features [27]. Cevikalp and Polikar exploited local classifier accuracy estimates to weight classifier outputs and proposed a dynamic approach with quadratic programming to combine classifiers built on different regions of the input space [29].

Regarding the Random Forests approach [30], it can exploit implicit and explicit diversities together. The method combines the “Bagging” idea for instance sampling and the random selection of variables for feature selection.

Second, in the combination strategy, multiple classifiers with a proper combination of rules or learning methods are grouped. The combination methods can be categorized according to the level of classifier outputs over a given classifier set in terms of abstract level (class level), rank level (rank order), and measurement level (class scores) [26,31]. In such analysis, the class scores provide richer information than the class label and the rank order. Therefore, a number of methods with the measurement level combination such as an average, linear or nonlinear combination rules are employed [2,26,32]. In particular, Bagging combines component classifiers by majority voting and the most-voted class is predicted [19]. AdaBoost combines components by weighted majority voting, where weights are adaptively learned from training rounds [21]. Stacking uses a meta-learner (in the second-level) to learn and combine individual classifiers (from the first-level) [33]. Hao et al. also proposed a stacking-like combination method for neural network classifiers in handwritten Chinese character recognition [34].

Given a number of available component classifiers, most conventional approaches combine all of these classifiers to constitute an ensemble. In the literature, many researchers suggested that ensemble of some parts of the available component classifiers may be better than ensemble as a whole. This leads to sparse ensemble, pruning ensemble or selective ensemble for the combination of multiple classifiers [10,35–40]. Specifically, Martinez-Muoz et al. investigated several pruning strategies with ordered aggregation to select and prune classifiers in the ensemble, and found that the generalization error reached a minimum at intermediate numbers of available classifiers [38]. Zhou et al. analyzed the relationship between the ensemble and its component neural networks, declared that it may be better to ensemble many instead of all of the available neural networks, and proposed an approach named GASEN for their selective ensemble [10]. Zhang and Zhou analyzed sparse ensembles with a weighted combination framework and proposed several linear programming techniques to sparsely combine classifiers [39]. The sparsity learning seeks a sparse weight vector for combining the outputs of all classifiers. Each classifier model has its own weight value, zero or nonzero. In general, a sparse model representation is expected to improve the generalization performance and computational efficiency [41,42].

In conventional approaches, the diversity learning and the sparsity learning for classifier ensemble have different purposes and algorithmic treatments. However, there are few researchers who focus on combining classifiers with both sparsity and diversity learning strategies, though the combination idea is easily understandable. Recently, Chen and Yao analyzed diversity and regularization in neural network ensembles for balancing diversity, regularization and accuracy of multi-objectives [12,43]. In the balance, the optimization algorithms were executed with expectation propagation (for pruning ensemble) or ranking-based fitness assignment (for selective ensemble). Note that their methods were specifically designed for component classifier training and combination in neural network ensembles.

In this paper, considering a general classifier ensemble with numerous available component classifiers, we formulated the sparsity and diversity learning problem in a general mathematical optimization framework. Moreover, we proposed a practical approach based on the genetic algorithm (GA) to heuristically learning sparsity and diversity. In order to conveniently evaluate the diversity of component classifiers, we introduce the diversity contribution ability to select proper classifier components and evolve classifier weights. There are two main steps in our heuristic learning approach. In the first step, a sparsity weight vector with component classifiers is learned with sparsity learning, i.e., the l_1−norm regularization. In the second step, our approach employs GA to evolve these sparse weights according to the ensemble diversity. The optimization rule is to seek a high diversity while retaining a considerable sparsity in the ensemble.

The rest of this paper is organized as follows. Section 2 describes the problem statement and the framework for the classifier ensemble with sparsity or/and diversity. In Section 3, our sparsity and diversity learning for the classifier ensemble is presented in detail. Several comparative experiments with UCI classification data sets and the Pascal Large Scale Learning Challenge 2008 webspam data are demonstrated in Section 4. Finally, conclusion is given in Section 5.

2. Framework for classifier ensemble with sparsity and diversity

2.1. Setting and notation

In ensemble learning with a classification problem, each instance $\mathbf{a}$ is associated with a label y. To classify one instance $\mathbf{a}$ into K classes $\{\omega_1, \dots, \omega_K\}$, assume that we have N different classifiers (classification hypotheses) $\{h_1, \dots, h_N\}$, each using a certain feature vector for $\mathbf{a}$. On an input instance $\mathbf{a}$, each classifier h_n outputs discriminant measures $x^n = h_n(\mathbf{a})$. With all classifiers we get $\mathbf{x} = [x^1 \dots x^N]^T$.

By the classifier ensemble, the decisions of the component classifier are deferred and the final decision is made after the outputs of multiple classifiers are fused to give combined class similarity measures. The combined measures can be computed by a learning function

$$H(\mathbf{a}) = F(h_1(\mathbf{a}), \dots, h_N(\mathbf{a})) = F(\mathbf{x}) = F(x^1, \dots, x^N)$$

In the context of ensemble learning and classifier combination, numerous methods have been proposed. In this paper, we focus on