Applied Soft Computing 14 (2014) 554–562

Contents lists available at ScienceDirect

Applied Soft Computing

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

Cost-sensitive decision tree ensembles for effective imbalanced classification

Bartosz Krawczyk[a,*], Michał Woźniak[a], Gerald Schaefer[b]

[a] Department of Systems and Computer Networks, Wroclaw University of Technology, Poland
[b] Department of Computer Science, Loughborough University, Loughborough, UK

ARTICLE INFO

Article history:
Received 26 November 2012
Received in revised form 5 July 2013
Accepted 29 August 2013
Available online 12 September 2013

Keywords:
Machine learning
Multiple classifier system
Ensemble classifier
Imbalanced classification
Cost-sensitive classification
Decision tree
Classifier selection
Evolutionary algorithms
Classifier fusion

ABSTRACT

Real-life datasets are often imbalanced, that is, there are significantly more training samples available for some classes than for others, and consequently the conventional aim of reducing overall classification accuracy is not appropriate when dealing with such problems. Various approaches have been introduced in the literature to deal with imbalanced datasets, and are typically based on oversampling, undersampling or cost-sensitive classification. In this paper, we introduce an effective ensemble of cost-sensitive decision trees for imbalanced classification. Base classifiers are constructed according to a given cost matrix, but are trained on random feature subspaces to ensure sufficient diversity of the ensemble members. We employ an evolutionary algorithm for simultaneous classifier selection and assignment of committee member weights for the fusion process. Our proposed algorithm is evaluated on a variety of benchmark datasets, and is confirmed to lead to improved recognition of the minority class, to be capable of outperforming other state-of-the-art algorithms, and hence to represent a useful and effective approach for dealing with imbalanced datasets.

1. Introduction

Numerous approaches have been introduced in the literature aiming to provide effective and efficient classification systems [19]. However, it is also well known that according to the *no free lunch theory* there is no universal classifier that performs best for all decision problems [41].

Canonical machine learning methods are based on the idea of selecting the single best classifier from a set of available models. However, making a decision based on solely a single classifier also discards the possibility that other models may also offer a valuable contribution. Methods that are trying to exploit the strengths of several models are known as multiple classifier systems (MCSs) or classifier ensembles [29], and are one of the most promising research directions in the current field of machine learning and pattern recognition.

There are typically two main challenges when constructing MCSs: how to select classifiers to form an ensemble, and how to fuse the individual decisions of the base classifiers into a single decision. Poor selection may undermine the whole process of designing MCSs, while a good strategy for building an ensemble should guarantee an improvement in its diversity. This can be achieved by using different partitions of the dataset or by generating a number of datasets through data splitting, a cross-validated committee, bagging, or boosting [29], so that the generated base classifiers, since trained on different inputs, would be complementary. Among the employed approaches, constructing random subspaces [17] is one of the most generic ones, and typically works well with various types of classifiers.

Classifier fusion methods can be categorised into approaches that are based on classifier labels and those that utilise discriminant analysis. The former includes various voting algorithms [4,46]. While (majority) voting schemes are among the most popular fusion methods, often better results are obtained by approaches that consider the importance of decisions coming from particular committee members [39,28].

For methods based on discriminant analysis, the main form of discriminants is *a posterior probability*, although outputs of neural networks or other functions whose values are used to establish the decision of the classifier (so called support functions) can also be considered. While simple aggregation methods (like minimum, maximum, product, mean) can be used, they are typically subject to rather restrictive conditions [12] which limit their practical use. Better results can be achieved by designing fusion models based on a training procedure to arrive at so-called trained fusers [44].

* Corresponding author. Tel.: +48 692979578.
E-mail addresses: bartosz.krawczyk@pwr.wroc.pl (B. Krawczyk), michal.wozniak@pwr.wroc.pl (M. Woźniak), gerald.schaefer@ieee.org (G. Schaefer).

1568-4946/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2013.08.014

The underlying class distribution can play a crucial role in the derivation of effective classifiers. In many cases the distribution is roughly equal among all the classes but this does not hold for every application. When one of the classes (referred to as the majority class) significantly outnumbers the remaining (minority) class(es), we deal with a problem known as imbalanced classification [16] which occurs in a variety of domains including anomaly detection [20], fault diagnosis [47], medical data analysis [23], drug design [22], SPAM detection [48] and face recognition [32]. While the performance of classification algorithms is typically evaluated using predictive accuracy, clearly this is not appropriate when the data is imbalanced as it would favour the correct identification of majority class samples.

In this paper, we propose, based on our earlier work [25,27], a classifier ensemble design algorithm which is built on the basis of a cost matrix for improved minority class prediction. As base classifiers we utilise cost-sensitive decision trees due to their susceptibility to improvement via the ensemble approach, while we employ an evolutionary algorithm to simultaneously perform classifier selection and fusion.

Instead of using a fixed cost matrix we derive its parameters via ROC analysis. To gain a deeper insight into the influence of cost matrices on the minority class recognition, we investigate several imbalanced datasets with different levels of imbalance to identify a useful pattern for setting the cost matrix.

The main contributions of this paper are as follows:

- A new ensemble pruning method based on the combination of decision trees trained on different sets of features.
- Use of an evolutionary algorithm for simultaneous classifier selection and fusion to promote the best base classifiers and boost the recognition rate of the minority class.
- In-depth analysis of the influence of the cost matrix parameters and data imbalance ratio on the performance of the proposed ensemble based on ROC analysis.

The remainder of the paper is organised as follows. In Section 2 we present the pattern recognition background that our approach is based on, while Section 3 discusses the problem of imbalanced classification. Our new algorithm is introduced in detail in Section 4. Experimental results are reported and discussed in Section 5, while Section 6 concludes the paper.

2. Model of pattern recognition task

The aim of pattern recognition is to assign a given sample to one of a number of pre-defined categories. A pattern recognition algorithm Ψ thus maps the feature space X to the set of class labels $\boldsymbol{M}$

$$\Psi : X \rightarrow \boldsymbol{M}. \tag{1}$$

This mapping is typically established on the basis of examples from a training set which contains learning examples, i.e. observations of features together with their correct classifications. Although it is important for the performance of a classifier, we do not focus on feature selection in this paper, but assume that the set of features is given by an expert or chosen by an appropriate feature selection method [11].

Let's assume that we have n classifiers $\Psi^{(1)}, \ldots, \Psi^{(2)}, \ldots, \Psi^{(n)}$. For a given object x, each of them makes a decision regarding class $i \in \boldsymbol{M} = \{1, \ldots, M\}$. The combined classifier $\bar{\Psi}$ then makes a decision according to a weighted voting rule

$$\bar{\Psi}(\Psi^{(1)}(x), \Psi^{(2)}(x), \ldots, \Psi^{(n)}(x)) = \underset{j \in M}{\arg\max} \sum_{l=1}^{n} \delta(j, \Psi^{(l)}(x)) w^{(l)}, \tag{2}$$

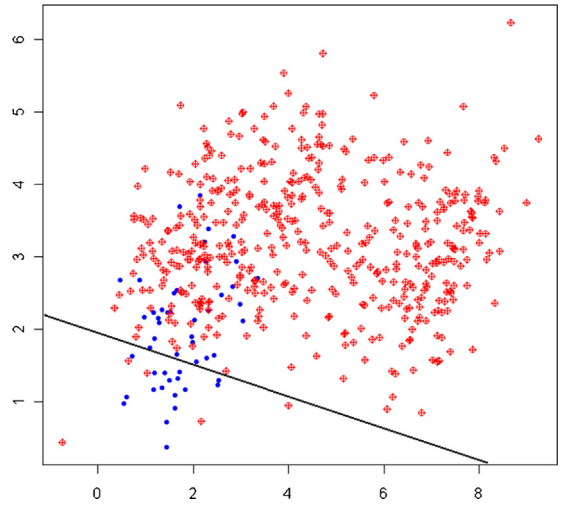

Fig. 1. Example of bias towards the majority class in linear classification of an imbalanced problem. The established decision boundary (line) would give poor prediction for minority class samples.

where

$$\delta(j, i) = \begin{cases} 0 & \text{if} \quad i \neq j \\ 1 & \text{if} \quad i = j \end{cases}, \tag{3}$$

and $w^{(l)}$ is the weight assigned to the lth classifier. The weights used in Eq. (2) play a key-role in establishing the quality of $\bar{\Psi}$ [42]. In this paper, we construct an ensemble with decision tree classifiers as base classifiers. Therefore, it is not possible to use support functions [45] and we consequently revert to a weighted voting approach which has been shown to behave better than canonical voting methods [43].

3. Imbalanced classification

The performance and quality of machine learning algorithms is conventionally evaluated using predictive accuracy. However, this is not appropriate when the data under consideration is strongly imbalanced, since the decision boundary may be strongly biased towards the majority class, leading to poor recognition of the minority class as illustrated in Fig. 1.

Class imbalance not only makes the learning task more complex [38], it is usually accompanied also by other difficulties such as:

- *Small sample size*: In many cases the number of minority class samples is insufficient to properly train a classifier, hence resulting in poor generalisation and possibly leading to overfitting. Even though it has been shown, that when the number of minority samples is sufficient the uneven class distribution itself does not cause a significant drop in recognition rate [10], often this is not possible for real-life classification problems.
- *Small disjuncts*: This problem is connected to the previous one, as it may happen that the minority class is represented by a number of subconcepts, meaning that its objects form several spread "chunks" of data [34]. This leads to difficulties due to the lack of uniform structure in the minority class and low sample count in each of the subconcepts.
- *Class overlapping*: When discriminative rules are constructed in such a way as to minimise the number of misclassified instances, this may lead to poor performance for objects in the overlap area to the minority class [14].

Techniques that address the problems associated with imbalanced datasets can in general be divided into three groups [33]: