Pattern Recognition 83 (2018) 34–51

Contents lists available at ScienceDirect

Pattern Recognition

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

Dynamic ensemble selection for multi-class classification with one-class classifiers

Bartosz Krawczyk [a,*], Mikel Galar [b], Michał Woźniak [c], Humberto Bustince [b], Francisco Herrera [d,e]

[a] Department of Computer Science, Virginia Commonwealth University, USA
[b] Departament of Automática y Computación, Universidad Pública de Navarra, Pamplona, Spain
[c] Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Technology, Wrocław, Poland
[d] Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
[e] Faculty of Computing and Information Technology - North Jeddah, King Abdulaziz University, Jeddah, 21589, Saudi Arabia

ARTICLE INFO

Article history:
Received 4 October 2017
Revised 19 April 2018
Accepted 13 May 2018

Keywords:
Machine learning
Classifier ensemble
One-class classification
Multi-class decomposition
Dynamic classifier selection
Ensemble pruning

ABSTRACT

In this paper we deal with the problem of addressing multi-class problems with decomposition strategies. Based on the divide-and-conquer principle, a multi-class problem is divided into a number of easier to solve sub-problems. In order to do so, binary decomposition is considered to be the most popular approach. However, when using this strategy we may deal with the problem of non-competent classifiers. Otherwise, recent studies highlighted the potential usefulness of one-class classifiers for this task. Despite not using all the available knowledge, one-class classifiers have several desirable properties that may benefit the decomposition task.

From this perspective, we propose a novel approach for combining one-class classifiers to solve multi class problems based on dynamic ensemble selection, which allows us to discard non-competent classifiers to improve the robustness of the combination phase. We consider the neighborhood of each instance to decide whether a classifier may be competent or not. We further augment this with a threshold option that prevents from the selection of classifiers corresponding to classes with too little examples in this neighborhood.

To evaluate the usefulness of our approach an extensive experimental study is carried out, backed-up by a thorough statistical analysis. The results obtained show the high quality of our proposal and that the dynamic selection of one-class classifiers is a useful tool for decomposing multi-class problems.

1. Introduction

Classification problems involving multiple classes are present in many real-world scenarios. Nevertheless, with increasing number of classes the complexity of a problem grows up, since the establishment of the decision boundaries becomes more difficult due to the greater overlapping.

To alleviate this problem we may divide the original multi-class task into several binary classification problems that can be addressed by any classifier. These approaches are commonly known as decomposition strategies [1] and they are widely applied to solve multi-class classification problems. They are not only useful to deal with multiple classes using binary classifiers but also to increase the classification performance of classifiers with inherent multi-class support [2]. Among these types of strategies, there are two that stand out due to their simplicity and accuracy. They are known as One-vs-One (OVO) [3] and One-vs-All (OVA) [4] strategies, which can be included in the broader concept of Error Correcting Output Codes (ECOC) [5] framework. Their major difference is the way in which the original problem is decomposed: in OVO, a classifier is learned for each pair of classes, whereas in OVA, only one classifier per class is learned. As a consequence, their aggregation phases also differ, where the outputs of the classifiers are combined. This phase is a key component in these strategies [2] (as well as in any ensemble method [6,7]). In classical decomposition, OVO is usually considered to be more accurate than OVA, which is usually attributed to the fact that it creates simpler problems with less instances and does not produce imbalanced datasets as OVA does [8]. On the contrary, OVO has the problem known as

* Corresponding author.
E-mail addresses: bkrawczyk@vcu.edu, bartosz.krawczyk@pwr.wroc.pl (B. Krawczyk), mikel.galar@unavarra.es (M. Galar), michal.wozniak@pwr.edu.pl (M. Woźniak), bustince@unavarra.es (H. Bustince), herrera@decsai.ugr.es (F. Herrera).

https://doi.org/10.1016/j.patcog.2018.05.015
0031-3203/

non-competence [9,10], which is produced by the fact that classifiers are only competent on the pair of classes that they have been trained on, even though they are used to classify an instance belonging to any of the classes.

Traditionally, decomposition strategies have been focused on the usage of binary classifiers. Nevertheless, it has been shown that taking advantage of other classifier learning paradigms, the performance obtained in some specific problems can also be boosted [11,12]. Especially, this is the case of One-Class Classification (OCC) paradigm [13]. Basically, the objective of OCC learning is to train a classifier considering only examples from a target class, whereas examples from the other class (or classes) are left aside (they may be insufficient for a proper representation or too difficult to gather). Once the classifier is learned, new examples are classified into the target class or out of it.

Addressing the multi-class problems with OCC classifiers offers several advantages [12], mainly when the number of classes increases and when challenges embedded in the nature of data, such as multi-class imbalance or label noise, are present [11,14].

There are some important issues that must be taken into consideration in the framework of multi-class decomposition using OCC.

- When shifting to OCC, decomposition is naturally an OVA approach, but in this case some of the advantages of both OVO and OVA are inherited as well as some disadvantages avoided.
- Despite being an OVA model, no imbalance is created, since only the instances of the target class are considered for learning the classifier. Additionally, the number of base classifiers created is lower than in OVO, and hence a more sparse ensemble is obtained.
- In the original OVA model all classifiers are competent, but in OCC non-competence appears because each classifier has only been trained with the examples of the target class (given the nature of OCC), however, in this case classifiers are aware of the fact that instances may not belong to the target class.
- Addressing multi-class problems by decomposition using OCC is straightforward. Combination methods from both OVO and OVA can be used, since the result of applying OVO and OVA is the same when OCC learners are considered.

Focusing on this framework, our aim with this paper is to further investigate the potential of effective usage of OCC in multi-class problems and to present a new combination strategy for OCC classifiers in decomposition strategies. In the same manner as it occurs with classical decomposition, classifier combination is a key step. A thorough study on OVO and OVA aggregations was carried out in [2]. As a result, a new combination model based on Dynamic Ensemble Selection (DES) [15] was presented to deal with non-competent classifiers in the OVO strategy in [10]. This model was based on the neighboring instances so that classifiers considering pairs of classes that were too far away from the instance to be classified were removed from the aggregation phase. Our proposal is based on this work, whose main contribution was the translation of the DES process from the input space (which is the commonly used model) to the output space in which OVO classifiers are specialized.

The fact that decomposition with OCC classifiers impose the existence of non-competent classifiers even in the OVA approach has led us to a new combination strategy in which we propose the usage of DES in this scenario. Hence, we extend the DES for OVO to the OCC framework. Moreover, we will further develop the original model after showing that it may not exploit all the possibilities owing to the large neighborhood considered to select the competent classifiers, mainly when datasets with many classes are considered. Hence, we will reduce it aiming at obtaining a greater improvement of the performance due to the elimination of classifiers hindering it. We will also show that the management of non-competent classifiers can also help in OCC-based decomposition.

In order to empirically study the usefulness of the DES approach in the OCC-based decomposition, we develop a thorough study including twenty two datasets from KEEL dataset repository [16,17]. Three different OCC classifiers from different paradigms are considered, namely, Support Vector Data Description (SVDD) [18], Parzen Data Description (PDD) [19] and Minimum Spanning Tree [20] (MST). In order to show the usefulness of and flexibility of the proposed DES model, three different combination strategies (maximum of supports, Error-Correcting Output Codes, and Decision Templates) are applied after performing the DES.

Hence, the contributions of this paper can be summarized as follows:

- An insight into the applicability of one-class classifiers to tackle difficult multi-class problems, such as ones characterized by a high number of classes, imbalanced distributions, presence of borderline and noisy instances, as well as class overlapping.
- A new combination method based on DES that allows one to avoid non-competent classifiers and improve the classification performance in the OCC-based decomposition.
- A new threshold-based pruning mechanism extending the proposed DES to reduce the number of classifiers in the final combination when greater number of classes are considered.

The rest of this paper is organized as follows. In Section 2.2, we introduce OCC. Next, Section 2.3 recalls decomposition strategies for binary classifiers. In Section 3 the justification for the usage of OCC classifiers for multi-class problems is given. Afterwards, the DES approach for the OCC-based decomposition is presented, including the new mechanism to further reduce the number of classifiers in the final combination. Section 5 introduces the experimental framework and the corresponding analysis to show the validity of the DES model. Finally, Section 6 concludes this paper.

2. Background

In this section, we recall several concepts that are needed to develop the rest of this work. First, the pattern classification task is described. Afterwards, one-class classification and the usage of decomposition strategies for multi-class problems are introduced.

2.1. Pattern classification

The classifier learning task consists in extracting knowledge from a set of n labeled examples $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. A given ith example is characterized by d-dimensional feature vector $x_i = \left[x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(d)}\right]^T \in \mathcal{X} = \mathcal{X}^{(1)} \times \mathcal{X}^{(2)} \times \ldots \times \mathcal{X}^{(d)}$ and by its label $y_i \in \mathcal{M} = \{\omega_1, \ldots, \omega_M\}$. The aim of a classifier learning is to learn an algorithm capable of predicting this output for new previously unseen examples with good generalization ability.

Therefore, a classifier may be defined as a function Ψ with domain $\mathcal{X}$ and codomain $\mathcal{M}$ such that $\Psi : \mathcal{X} \to \mathcal{M}$. In most of the cases the prediction of the classifier is based on a set of support functions $F_k(x)$ giving a support value for each class k of the problem. The most common way to make the final decision is to use the maximum rule $\Psi(x) = \arg\max_{k \in \mathcal{M}} (F_k(x))$.

2.2. One-Class classification

OCC works under the assumption that in the learning phase only objects from one class are available [21]. This specific class is commonly known as the target class and is denoted by ω_T. For one-class classification, it is the only class to which we have an access during the training phase. For multi-class problems, it is one