Pattern Recognition 76 (2018) 175–190

Contents lists available at ScienceDirect

Pattern Recognition

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

A framework for dynamic classifier selection oriented by the classification problem difficulty

André L. Brun[a], Alceu S. Britto Jr.[a,b,*], Luiz S. Oliveira[c], Fabricio Enembreck[a], Robert Sabourin[d]

[a] Pontifícia Universidade Católica do Paraná (PUCPR), Curitiba, PR, Brazil
[b] Universidade Estadual de Ponta Grossa (UEPG), Ponta Grossa, PR, Brazil
[c] Universidade Federal do Paraná (UFPR), Curitiba, PR, Brazil
[d] École de technologie supérieure (ÉTS), Université du Québec, Montreal, QC, Canada

ARTICLE INFO

Article history:
Received 28 March 2017
Revised 26 August 2017
Accepted 30 October 2017
Available online 31 October 2017

Keywords:
Multiple classifier systems
Classifier pool generation
Dynamic classifiers selection
Classification problem difficulty

ABSTRACT

This paper describes a framework for Dynamic Classifier Selection (DCS) whose novelty resides in its use of features that address the difficulty posed by the classification problem in terms of orienting both pool generation and classifier selection. The classification difficulty is described by meta-features estimated from problem data using complexity measures. Firstly, these features are used to drive the classifier pool generation expecting a better coverage of the problem space, and then, a dynamic classifier selection based on similar features estimates the ability of the classifiers to deal with the test instance. The rationale here is to dynamically select a classifier trained on a subproblem (training subset) having a similar level of difficulty as that observed in the neighborhood of the test instance defined in a validation set. A robust experimental protocol based on 30 datasets, and considering 20 replications, has confirmed that a better understanding of the classification problem difficulty may positively impact the performance of a DCS. For the pool generation method, it was observed that in 126 of 180 experiments (70.0%) adopting the proposed pool generator allowed an improvement of the accuracy of the evaluated DCS methods. In addition, the main results from the proposed framework, in which pool generation and classifier selection are both based on problem difficulty features, are very promising. In 165 of 180 experiments (91.6%), it was also observed that the proposed DCS framework based on the problem difficulty achieved a better classification accuracy when compared to 6 well known DCS methods in the literature.

1. Introduction

Many researchers have focused on Dynamic Classifier Selection (DCS), and have produced interesting solutions. The main difference between the researchers' approaches lies in the criterion adopted in selecting the classifier(s) from the pool. Usually, this selection is based on the concept of classifier competence, which is most commonly estimated over a region of the feature space defined as the neighborhood of the test pattern on a validation set. In [1], a proposed taxonomy organizes the DCS methods taking into account the criterion applied to compute the classifiers' competence. In their view, we may organize them in two main groups: methods based on the sole competence of the classifiers in the pool, and methods in which the interaction between the classifiers is considered. Regardless of the large number of different criteria available to measure the competence of the classifiers in the pool, one common thread running through them is the use of accuracy-based competence analysis, which is carried out over the feature or decision space.

In such a context, it is known that the pool in which the classifier selection is executed also plays an important role in the DCS performance. However, little effort has been dedicated to investigating new strategies to create a pool well-suited for DCS-based methods. Diversity is always expected irrespective of whether a homogeneous or a heterogeneous pool is used. The most popular techniques for pool generation are Bagging [2], Boosting [3] and Random Subspaces (RSS) [4]. With the exception of Boosting, in which future weak classifiers focus more on the examples that previous weak classifiers misclassified, these techniques usually ma-

* Corresponding author at Pontifícia Universidade Católica do Paraná (PUCPR), Curitiba, PR, Brazil: .

E-mail addresses: abrun@ppgia.pucpr.br (A.L. Brun), alceu@ppgia.pucpr.br (A.S. Britto Jr.), lesoliveira@inf.ufpr.br (L.S. Oliveira), fabricio@ppgia.pucpr.br (F. Enembreck), robert.sabourin@etsmtl.ca (R. Sabourin).

https://doi.org/10.1016/j.patcog.2017.10.038
0031-3203/

nipulate the data for training weak and diverse classifiers in a random fashion.

To the best of our knowledge, there is no DCS method oriented by the classification problem properties. A DCS in which the pool is generated to provide a better compromise with the criterion used for classifier selection. More than simply classifier accuracy-based competence, we are talking here about the ability of each classifier in the pool to deal with a specific kind of problem. This idea is based on works that attempt to find the best learning method for a specific classification problem, taking into account its difficulty [6–8]. Similarly, if we consider the space of a classification problem as commonly composed of subproblems with different levels of difficulty, the best case scenario would be to have a well-suited classifier for each subproblem. Thus, the most promising classifier for a given test instance could be the one trained on a similar subproblem, i.e., a subproblem with a similar level of difficulty as that estimated in the neighborhood of the test instance. The neighborhood of the test instance could be used to specify the kind of subproblem to which it belongs. It would appear reasonable to believe that a classifier trained on a similar subproblem is able to deal with the given test instance. Nevertheless, in such a DCS-based method, the pool generated must be able to provide a better coverage of the problem complexity space, but the methods available in the literature are not suitable for creating classifiers covering different regions of this space.

To represent the classification problem difficulty, we may extract features from the problem data using complexity measures. It is worth noting that the complexity, or difficulty, here involves more than just the quantities of instances, classes and features. It considers intrinsic characteristics of a classification problem, which can be obtained by means of complexity measures applied on the problem data. For instance, there are measures of difficulty based on overlap between classes, on the behavior of the edges between classes, on the class spatial distribution, and so on.

Our first hypothesis is that DCS can be done based on the classification problem difficulty, i.e., by selecting a classifier trained on a subproblem showing a similar level of difficulty as that of the neighborhood of the test instance. In our previous work [9], we observed that the adoption of data complexity features in the process of evaluating the skill of each classifier, given a test instance, may contribute to improve the performance of the classifier selection process. Deviating from that work, here we propose a complete DCS framework to investigate the impact of using problem complexity information not only in the selection process, but also for pool generation. Thus, an important hypothesis is evaluated, which is related to a better compromise between pool generation and classifier selection in a DCS method. In fact, it is expected that a pool of classifiers covering the problem complexity space adequately, i.e., that is trained on data subsets that are diverse in terms of level of difficulty, may provide better classification performance for a DCS, mainly when the selection of classifiers is also based on the problem difficulty.

In summary, more than just proposing a new framework for DCS, we intend to answer the following research questions: (a) Could a pool generated considering the difficulty of the classification problem provide gains in terms of classification performance by covering the problem space better?; (b) What is the impact, in terms of accuracy, of using the classification problem difficulty to drive both pool generation and classifier selection of a DCS-based method? We answered these questions by means of an experimental protocol composed of 30 datasets of classification problems with different levels of difficulty. We compared the results obtained with 6 DCS-based methods of the literature. The experiments showed that the strategy of generating and selecting classifiers based on the problem difficulty is very promising. The proposed DCS provides a better compromise between pool generation and classifier selection processes. In addition, similar experiments have shown that the proposed pool generation has a positive impact on the performance of DCS methods.

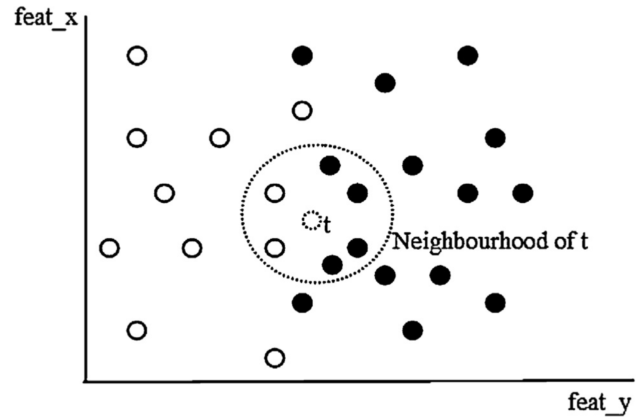

Fig. 1. Concept of competence estimated in a local region of feature space, defined as the neighborhood of the test instance in a validation set.

The remaining of this manuscript is divided into 6 sections. The Section 2 presents the main related works. Section 3 summarizes some basic concepts and definitions needed to understand the proposed DCS framework. Section 4 describes the proposed framework, detailing its generation and selection phases, while Section 5 presents the experimental protocol and corresponding results. Finally, Section 6 presents the conclusion and future work directions.

2. Related works

Various methods for dynamic selection of classifiers are available in the literature. Basically, the difference between them is at the level of the criterion used to define the competence of the classifiers for each test instance in the selection process. Fig. 1 illustrates the concept of competence estimation. A local region of the feature space, usually represented by the neighborhood of the test instance in a validation set, is used to estimate the criterion adopted.

It is common to find competence measures based on accuracy (overall or class-based) [10,11], ranking of classifiers [12], probabilistic measures [11,13], behavior of the classifiers computed on their output profiles [14], Oracle-based criteria [15,16], etc. In addition, some measures take into account group-based information such as ambiguity [19], diversity [17,18], or data handling theory like in [20].

We selected six of the preceding important contributions to the literature to implement in our experimental protocol, with 4 being single classifier selection methods, and 2 being ensemble selection methods. From [10], we have implemented 2 methods, the Overall Local Accuracy (OLA) and the Local Class Accuracy (LCA). The first calculates the classifier competence as the percentage of the correct recognition of the neighbors of the test instance in the feature space, while the second computes it as the percentage of correct classifications within the test instance neighborhood, but considering only those examples where the classifier has given the same class as the one it gives for the test instance. The other 2 single classifier selection methods were implemented from [13], the A Priori (APRI) and A Posteriori (APOS) methods. In the APRI method, a classifier is selected based on its class posterior probability estimated in the neighborhood of the test instance. This probability is weighted by the Euclidian distance between the test instance and each neighbor. Unlike in the APRI, the APOS method takes into account the class assigned by the classifier to the test instance.