Applied Soft Computing 68 (2018) 651–666

Contents lists available at ScienceDirect

Applied Soft Computing

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

Local sets for multi-label instance selection

Álvar Arnaiz-González, José-Francisco Díez-Pastor, Juan J Rodríguez, César García-Osorio*

Escuela Politécnica Superior, Universidad de Burgos, 09006 Burgos, Spain

A R T I C L E I N F O

Article history:
Received 28 July 2017
Received in revised form 14 March 2018
Accepted 5 April 2018
Available online 17 April 2018

Keywords:
Multi-label classification
Data reduction
Instance selection
Nearest neighbor
Local set

A B S T R A C T

The multi-label classification problem is an extension of traditional (single-label) classification, in which the output is a vector of values rather than a single categorical value. The multi-label problem is therefore a very different and much more challenging one than the single-label problem. Recently, multi-label classification has attracted interest, because of its real-life applications, such as image recognition, bioinformatics, and text categorization, among others. Unfortunately, there are few instance selection techniques capable of processing the data used for these applications. These techniques are also very useful for cleaning and reducing the size of data sets.

In single-label problems, the local set of an instance **x** comprises all instances in the largest hypersphere centered on **x**, so that they are all of the same class. This concept has been successfully integrated in the design of Iterative Case Filtering, one of the most influential instance selection methods in single-label learning. Unfortunately, the concept that was originally defined for single-label learning cannot be directly applied to multi-label data, as each instance has more than one label.

An adaptation of the local set concept to multi-label data is proposed in this paper and its effectiveness is verified in the design of two new algorithms that yielded competitive results. One of the adaptations cleans the data sets, to improve their predictive capabilities, while the other aims to reduce data set sizes. Both are tested and compared against the state-of-the-art instance selection methods available for multi-label learning.

1. Introduction

Single-label classification is a predictive data mining task that consists of assigning a label to an instance for which the label is unknown. Multi-label classification presents a similar task, although the difference is that the instances have a collection of labels, known as a labelset, rather than only one label. The maximum size of the labelset is determined by the number of different labels in the data set. The aforementioned labelset concept can also be considered as a sequence of binary output attributes (as many attributes as there are labels in the whole data set). Each attribute indicates whether the corresponding label is applicable to the instance. Only one of the attributes is active in single-label problems, while several attributes may be active in multi-label problems [22]. In other words, the labels in multi-label learning are not mutually exclusive [37]. This feature implies a harder and more challenging problem, due to the high relevance of the relations between the different labels [50].

Following [51], let us look at a more formal description of the problem. Consider an input space, $\mathcal{X}$, so that $\mathcal{X} = \mathbb{R}^d$ is the domain of instances (where d is the number of features); and, $\mathcal{Y}$ is the output space, so that $\mathcal{Y} = \{1, 2, \ldots, Q\}$ is the finite set of labels. Let $X = \{(\mathbf{x}_1, \omega_1), \ldots, (\mathbf{x}_n, \omega_n)\}$ denote a multi-label data set, where $\mathbf{x}_i \in \mathcal{X}$ and $\omega_i \subseteq \mathcal{Y}$, are i.i.d. drawn from an unknown distribution. The aim of multi-label classification is to try to find the function $f : \mathcal{X} \rightarrow 2^{\mathcal{Y}}$, which optimizes some of the multi-label metrics.

The origins of this field were, mainly, document [1,11,26] and music categorization [6,44] (where a document can simultaneously cover different topics and a musical piece can comprise different styles) and image recognition [5] (where different objects appear together in the same picture). However, the usefulness of multi-label classification is not limited to the above-mentioned fields, as it is also very valuable in many other real-world applications, such as genetics/biology, map labeling, and marketing, etc. As a result, there is a fast-growing interest in multi-label techniques in the data mining community [48]. This tendency becomes even

[*] Corresponding author.
E-mail addresses: alvarag@ubu.es (Á. Arnaiz-González), jfdpastor@ubu.es (J.-F. Díez-Pastor), jjrodriguez@ubu.es (J.J. Rodríguez), cgosorio@ubu.es (C. García-Osorio).

https://doi.org/10.1016/j.asoc.2018.04.016
1568-4946/© 2018 Elsevier B.V. All rights reserved.

clearer, if more recent publications on this topic are considered [22,31,39–41,53].

As multi-label classification can be considered as an extension of single-label classification, it is natural to try to solve the first task using the methods available for the second. Unfortunately, the use of single-label algorithms for multi-label learning is not straightforward. The common approach is the modification of algorithms or the transformation of data sets. Multi-label algorithms are usually grouped into the following three main categories [30]:

- Algorithm adaptation: as its name suggests, this approach consists in modifying the learning methods, so that the algorithm can process multi-label data sets, capturing all the dependencies and the internal relationships that exist among the labels.
- Problem-transformation algorithms: the original multi-label data set is used to obtain multiple single-label data sets. In this way, it is possible to use any one of the hundreds of existing single-label learning algorithms. A model is built for each single-label data set and all of the models are then combined, to obtain the multi-label assignation. These methods are usually grouped into three categories: binary relevance, label power-set, and pair-wise methods.
- Ensemble methods: working on top of problem transformation approaches, different subsets are used by the ensemble to train single-label learning algorithms, which are combined by means of ensemble techniques. Some ensemble methods can even process multi-label data sets without any change.

Due to limitations on space, the details of these algorithms are not given, although they may be consulted in the following reviews [20,22,30,40].

A common problem of real data sets is their volume (big or massive in many domains), as well as the presence of noise and anomalies that complicates the learning process. On the one hand, data-set size is a challenge, because not all algorithms (in terms of time or memory) scale properly when the number of instances is really high. On the other hand, the data acquisition process is usually prone to anomalies and noise, which can lead to inaccurate models [24]. This aspect is specially important for algorithms that are highly sensitive to noise, such as *k*NN [43].

Feature selection and instance selection are both popular pre-processing techniques for the removal of unnecessary and even harmful information in data sets. In feature selection, an attempt is made to select the most relevant features/attributes of the data sets, while instance selection attempts to find the most useful subset of instances. Both techniques have been researched in depth for single-label tasks [12,17,28]. However, whilst feature selection has been actively researched in the field of multi-label learning [29,33,37,48], instance selection has not received the same attention and only very few instance selection methods have been made available to date [9,25]. In this paper, new proposals are presented that are based on adapting the concept of the local set[1] to multi-label data sets.

The concept of the local set has been used for designing several instance selection methods in single-label [7,27]. Moreover, according to [18], Iterative Case Filtering (ICF for short) [8], based on the local set concept, is considered one of the most influential instance selection methods in single-label learning. Hence, this concept is of special interest when considering the design of new algorithms for multi-label instance selection.

The main contributions of this paper are:

- The definition of the local set concept in the context of multi-label data sets.
- The proposal that defines two new instance selection methods, based on the adaptation of single-label classification algorithms to multi-label learning: LSBo and LSSm [27].
- The experimental evaluation of the new algorithms. The new methods are compared with the few existing algorithms, and for the first time, an experimental study is made in which these existing algorithms are compared to each other.

The paper is structured as follows: in Section 2 instance selection techniques for both single-label and multi-label classification will be introduced; the local set concept and the instance selection algorithms that use it will be explained in detail in Section 3; in Section 4, the new definition of local sets in the context of multi-label data sets will be presented; the experimentation details and the results will be reported in Section 5. Lastly, the conclusions and a discussion on further research will be presented in Section 6.

2. Instance selection

It is well known that real-world data sets may include harmful, irrelevant, and redundant instances, due to measurement error and other issues. Instance selection methods represent an attempt to surmount this problem, by selecting a subset of instances of the original data set[2]. Their aim is to clean the data sets by removing noise, irrelevant instances, anomalies, etc. It is not a novel discipline, as the first works in the area date back to the late 1960s [21]. These techniques were initially designed to be used with *k*-nearest neighbors, but the selection of the best set of instances has proven its worth with many others classifiers. The reduction of data set size has the advantage of a reduction in classification times (in lazy-algorithms) and a reduction in the training time (in eager-algorithms) [4].

The reduction of the number of instances can be achieved by two main approaches: instance selection and prototype generation. Prototype generation uses information from original instances to create new ones that replace the old instances. In contrast, instance selection finds the best subset of instances of the original data set. The rest of this paper is focused on instance selection, although we would recommend the paper by Triguero et al. [38] to readers interested in prototype generation.

Instance selection methods are usually focused on boundaries between classes [17]. These boundaries clearly determine the classification process and therefore demarcate the areas of interest of the different methods. With regard to the search direction, instance selection algorithms can be grouped into two main categories[3]: incremental algorithms, that start with an empty data set and progressively add instances; and decremental algorithms, that work in the opposite direction, starting with the whole data set and removing those instances that are not relevant.

According to the type of search, instance selection algorithms are grouped into three main families [17]:

- Edition: the objective is to clean noisy instances and outliers from the data sets. Edition algorithms only clean the data sets, without attempting to reduce them in size.
- Condensation: these techniques attempt to shrink the data set size. Their focus is on the instances positioned around the class

[1] A local set is formed by the set of instances included in the largest hypersphere centered on an instance. In which the instances are therefore of the same class [27].

[2] It should be noted that the instance selection process considered in this paper (for noise removal and reduction of dataset size) is different from the instance selection or sample selection techniques used in active learning that try to identify the set of instances that have to be labeled [16].

[3] Although in [17] three more categories are considered: batch, mixed, and fixed.