Pattern Recognition 47 (2014) 2091–2103

Contents lists available at ScienceDirect

Pattern Recognition

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

Maximum margin multiple-instance feature weighting

Jing Chai [a,*], Hongtao Chen [a], Lixia Huang [a], Fanhua Shang [b]

[a] College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, PR China
[b] School of Electronic Engineering, Xidian University, Xi'an 710071, PR China

ARTICLE INFO

Article history:
Received 9 March 2013
Received in revised form
3 December 2013
Accepted 20 December 2013
Available online 3 January 2014

Keywords:
Feature weighting
Multiple-instance learning
Maximum margin
Coordinate ascent

ABSTRACT

Feature weighting is of considerable importance in machine learning due to its effectiveness to highlight relevant components and suppress irrelevant ones. In this paper, we focus on the feature weighting problem in a specific machine learning area: multiple-instance learning, and propose maximum margin multiple-instance feature weighting (M3IFW) to seek large classification margins in the weighted feature space. The designed M3IFW algorithm can be applied to both standard binary-class multiple-instance learning and the corresponding multi-class learning, and we abbreviate them to B-M3IFW (binary-class M3IFW) and M-M3IFW (multi-class M3IFW), respectively. Both B-M3IFW and M-M3IFW contain three kinds of unknown variables, i.e., positive prototypes, classification margins, and weighting coefficients. We utilize the coordinate ascent algorithm to update the three kinds of unknown variables, respectively and iteratively, and then perform classifications in the weighted feature space. Experiments conducted on synthetic and real-world datasets empirically demonstrate the effectiveness of M3IFW in improving classification accuracies.

1. Introduction

The main difference between multiple-instance learning and traditional supervised learning is that in supervised learning, class labels are attached to instances and the goal is to predict the class labels of unseen instances, whereas in multiple-instance learning, class labels are attached to bags (a set of instances is termed as a bag) and the goal is to predict the class labels of unseen bags. In standard multiple-instance learning, there are two classes in total, i.e., the positive class and the negative class. Note that the terminologies of "positive" and "negative" both define the class labels of given objects from two levels, i.e., the instance level and the bag level. In the instance level, given an instance, if it belongs to the positive class, we term it as a positive instance; otherwise we term it as a negative instance. In the bag level, given a bag, if it contains at least one positive instance, we term it as a positive bag; otherwise it is termed as a negative bag, which means all instances in a negative bag are negative ones. Based on the above definition, it is noted that the class label of a given negative bag and the class labels of instances in this negative bag are consistent, because all instances in a negative bag are negative ones. However, the class label of a given positive bag and that of instances in this positive bag are usually inconsistent, because a positive bag may contain both positive and negative instances simultaneously. Hence, there are class-label ambiguities for instances in positive bags, because their class labels are unclear, i.e., they may be either positive or negative.

The terminology "multiple-instance learning" was originally proposed by Dietterich et al. [1] when they were investigating the drug activity prediction problem. In their seminal work, Dietterich et al. considered the problem of predicting whether a candidate drug molecule binds to the target protein or not. Actually, a molecule may take on many different shapes by rotating its internal bonds, and if any of these shapes conforms closely to the structure of the binding site, the candidate molecule binds to the target protein. As a result, if we treat each shape of a molecule as an instance and each molecule which may take on many different shapes as a bag, we can easily see that drug activity prediction is a typical multiple-instance learning problem. Besides drug activity prediction, multiple-instance learning appears in many other areas, such as image categorization [2,3], image retrieval [4,5], protein sequence classification [5], stock selection [6], text classification [6,7], computer aided diagnosis [8,9], security application [10].

Standard multiple-instance learning consists of only two classes, i.e., the positive class and the negative class. However, with the quick development of multiple-instance learning, its application area has been extended from the binary-class case to the multi-class case [3,11]. In multi-class multiple-instance learning, for each given class, if any instance in a bag represents the class label of this class (i.e., the instance is positive for the given class), the bag is positive for this class, otherwise the bag is

* Corresponding author. Tel.: +86 15035101386; fax: +86 351 6010029.
E-mail address: jingchai@aliyun.com (J. Chai).

0031-3203/$ - see front matter © 2014 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.12.009

negative for this class. Note that in multi-class multiple-instance learning, usually we do not define the specific negative bags for a given class, because positive bags for some class can be treated as negative bags for other classes, e.g., positive bags for class c are simultaneously negative bags for classes except for c. Moreover, note that the multi-class multiple-instance learning problem is different from the multiple-instance multiple-label learning one [12], although there are multiple (larger than two) classes for both problems and they are kind of similar from this point of view. For multi-class multiple-instance learning, each bag is attached to only one class label, whereas for multiple-instance multiple-label learning, each bag may have more than one class labels. In this paper, we focus our study on binary-class and multi-class multiple-instance learning. Multiple-instance multiple-label learning is out of the scope of the current paper.

The greatly increasing applications make multiple-instance learning more and more popular in the machine learning community, and hence, many representative multiple-instance learning algorithms, e.g., ID-APR [1], Diverse Density (DD) [2] and EM-DD [13], Bayesian-KNN and Citation-KNN [14], MI-SVM and mi-SVM [6], MI-Kernel [15], MIGraph and miGraph [11], MIForests [16], MIRVM [17], MIDR [18], and the classifier combining algorithm γ-Rule [19], have been proposed to cope with various multiple-instance learning tasks.

Feature weighting, which assigns different coefficients to different features with each coefficient indicating the relative importance of the corresponding feature to the given learning task (e.g., classifications), has been studied by machine learning researchers and by which we may expect to obtain some kind of performance improvements. Several representative feature weighting algorithms, such as RELIEF [20], I-RELIEF [21,22], LESS [23], and LMFW [24], have obtained very promising learning performances in various applications.

Feature weighting has very close relationship with another two kinds of data preprocessing transformations: feature extraction [25] and feature selection [26,27], because all the three transformations try to mine the intrinsic information related to the given learning task and aim to improve the learning performances via this information. However, the ways of realizing the above three transformations are different. In feature weighting, we first endow each feature with a nonnegative coefficient to denote the relative importance of this feature to the learning task, and then utilize the weighted data to replace the original data in the following learning task. Feature selection can be treated as a special case of feature weighting, i.e., in feature selection the endowed coefficients cannot be arbitrary nonnegative values but have to be binary ones (0 or 1). In feature extraction, we first transform data to a linear or nonlinear feature space, and then utilize the data in the transformed space to operate the following learning task. Compared with feature weighting and feature selection, feature extraction has more degrees of freedom, because the transformed space of feature extraction may be a totally different space from the original one (e.g., it may be either a linear lower-dimensional subspace or a nonlinear kernel space, and for both cases features in the transformed space and features in the original space are totally different), whereas the transformed space of either feature weighting or feature selection is still the same to the original one, since features in the weighted or selected space still correspond to the original features.

Similar as many other machine learning areas (e.g., supervised learning, unsupervised learning, and semi-supervised learning), the feature weighting problem exists in multiple-instance learning as well. In particular, different features usually contribute differently in separating positive and negative bags. On the one hand, some features may contain intrinsic discriminative information and can help to improve the discrimination of heterogeneous bags, and hence, they are relevant to classifications and can be termed as relevant features; on the other hand, some features may only contain redundant and noisy information which are useless or even harmful to the discrimination, and hence, they are irrelevant to classifications and usually termed as irrelevant features. If we can effectively find out which features are relevant and which features are irrelevant, and then perform feature weighting by simultaneously highlighting the relevant ones and suppressing the irrelevant ones, we may expect to obtain improved classification accuracies.

In this paper, we focus our research on multiple-instance feature weighting, i.e., designing feature weighting algorithms that are suitable to multiple-instance learning tasks. In particular, we adopt the popular maximum margin principle to design our feature weighting algorithm, and thus term it as maximum margin multiple-instance feature weighting (M3IFW). We utilize the coordinate ascent algorithm to update all three kinds of unknown variables in M3IFW (i.e., positive prototypes, classification margins, and weighting coefficients), respectively and iteratively, and then perform feature weighting by transforming data from the original space to the weighted space, and finally utilize the weighted data to operate classifications.

Note that the designed M3IFW algorithm contains two different versions, of which one is applicable for the standard binary-class multiple-instance learning, while the other one is applicable for multi-class applications. To make a distinction, we refer to them as binary-class M3IFW and multi-class M3IFW, respectively. For convenience, we abbreviate binary-class M3IFW to B-M3IFW, and abbreviate multi-class M3IFW to M-M3IFW.

The rest of this paper is organized as follows. In Section 2, we introduce some related work and discuss their relationships and inspirations to our work. In Section 3, we introduce the design work of B-M3IFW. In Section 4, we discuss the optimization process of B-M3IFW in detail. In Section 5, we introduce how to extend B-M3IFW to M-M3IFW, and then give a brief discussion of the optimization of M-M3IFW, which is very similar to that of B-M3IFW. The optimality of B-M3IFW and M-M3IFW is discussed in Section 6. In Section 7, first we conduct experiments on two synthetic datasets to operate performance evaluations on B-M3IFW, and then compare B-M3IFW and M-M3IFW with their competing algorithms on benchmark datasets and the Corel Dataset, respectively. Finally, we give concluding remarks and discuss the future work in Section 8.

2. Related work

In this section, we give an introduction of some related work and discuss their relationships to our work, with the expectation of revealing the inspiration of related work to our work and giving further understandings on both our and related work. Note that in following discussions (both this section and the following ones), sometimes we do not distinguish B-M3IFW from M-M3IFW and just term them as M3IFW for convenience, unless it is prone to cause misunderstandings and we have to make clear distinctions.

2.1. Single-instance learning algorithms

In this subsection we give a brief discussion of several representative single-instance learning algorithms: LDA [28], MMC [29], I-RELIEF [21,22], and SVM-RFE [30], of which LDA and MMC are for feature extraction, I-RELIEF is for feature weighting, and SVM- RFE is for feature selection.

2.1.1. LDA and MMC

LDA and MMC are two supervised feature extraction algorithms which project data onto some lower-dimensional subspaces and