Neurocomputing ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Neurocomputing

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

A fast Markov chain based algorithm for MIML learning

Michael K. Ng [a], Qingyao Wu [b,c], Chenyang Shen [a,*]

[a] Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong
[b] School of Software Engineering, South China University of Technology, Guangzhou, PR China
[c] State Key Laboratory for Novel Software Technology, Nanjing University, P.R. China

ARTICLE INFO

Article history:
Received 7 November 2015
Received in revised form
7 June 2016
Accepted 9 August 2016
Communicated by Xiang Xiang Bai

Keywords:
Multi-Instance Multi-Label Learning
Markov Chains
Iterative Method
Labels Propagation

ABSTRACT

Multi-instance multi-label (MIML) learning is one of challenging research problems in machine learning. In the literature, there are several methods for solving MIML problems. However, they may take a long computational time and have a huge storage cost for large MIML data sets. The main aim of this paper is to propose and develop an efficient Markov Chain learning algorithm for MIML problems, especially for data represented by non-negative features. Our idea is to perform labels classification iteratively through two Markov chains constructed by using objects and features respectively. The classification of objects can be obtained by using labels propagation via training data in the iterative method. Moreover, we demonstrate that the proposed method can be formulated by considering normalized linear kernel. Because linear kernel function is explicit and separable, it is not necessary to compute and store a huge affinity matrix among objects/instances compared with the use of other kernel functions. Therefore, both the storage and computational time of the proposed algorithm are very efficient. Experimental results are presented to show that the classification performance of the proposed method using normalized linear kernel function is about the same as those using the other kernel functions, while the required computational time is much less, which together suggest that the linear kernel can be good enough for MIML problem.

Also experimental results on some benchmark data sets are reported to illustrate the effectiveness of the proposed method in one-error, ranking loss, coverage and average precision, and show that it is competitive with the other MIML methods.

1. Introduction

Multi-instance multi-label (MIML) problem is one of challenging research problems in machine learning [33]. In the traditional single-label classification problem, an object is described by one single instance and it only belongs to one single class. In the MIML problem, an object is represented by multiple instances and it can be characterized by more than one categories or labels. Compared to traditional classification problem, it is usually more suitable to formulate complex real problems such as image classification, text mining and gene functional prediction into MIML learning problem: multiple patches contained in real image often refer to different semantic meanings; different sections of a document usually correspond to diverse topics in text categorization; multiple segments encoded in one DNA sequence frequently response to distinct biological functions. For more detailed information on applications of MIML learning, please refer to [33] and its references.

1.1. Existing algorithms

There are many algorithms that have been developed for solving MIML problems. Among them we will mainly focus on the comparison between several state-of-art algorithms and the proposed algorithm. Let us briefly review them:

1. **MIML-SVM** [33,34]: This algorithm is based on support vector machine. The approach is motivated by solving Multi-label (ML) learning problems [5]. The main idea is to transform the MIML problem into single instance Multi-label problem by using the *k*-medoids algorithm, and then employ support vector machine (SVM) to predict classes for unlabeled objects. The computational complexity of the MIML-SVM method is huge when the number of training samples is large. However, when the number of training samples is small (i.e., there is limited information for learning), the classification accuracy may be low.
2. **MIML-*k*NN** [30]: This algorithm is based on *k*-nearest neighbor (*k*NN) method [25]. Similar to MIML-SVM, the approach is to consider ML learning problems. The MIML-*k*NN method

* Corresponding author.
E-mail address: 12466743@life.hkbu.edu.hk (C. Shen).

http://dx.doi.org/10.1016/j.neucom.2016.08.033
0925-2312/

employs Hausdorff metric [8] to measure the distance between each object which is represented by a bag of instances, and then solve the classification task by using *k*NN. Again the computational cost of the MIML-*k*NN is large when we deal a large MIML data set.

3. **MIML-Boost** [33,34]: This algorithm is different from the MIML-SVM and MIML-*k*NN. The approach is to decompose the MIML problem into several Multi-Instance (MI) learning problems [19]. For each MI learning problem, it can be solved by the MI-boosting algorithm [24] which is a supervised learning algorithm. However, the boosting algorithm can take a huge amount of computational time when we deal with a large number of ML problems. In this paper, we do not compare proposed algorithm with MIML-Boost algorithm due to the time limit.
4. **M^3**MIML [31]: This algorithm is based on optimization approach by exploiting the relationship between instances and labels. For each class label, the M^{3}MIML algorithm first solves a linear regularized optimization problem such that a maximum prediction for all instances can be generated. Then the prediction results are combined together to construct the final classification results for all the objects. According to the numerical results, the M^{3}MIML algorithm is more efficient than the MIML-Boost algorithm, but it is still more expensive than the MIML-SVM and MIML-*k*NN algorithms.
5. **Markov-MIML** [20,26]: This algorithm is based on Markov chain framework. In the algorithm, information of nearest neighbors for all instances is utilized to predict labels of testing samples. The training step is formulated as random walk process with restart. Experimental results on benchmark data sets have shown that both classification results and computational time by the Markov-MIML algorithm are competitive with the above mentioned MIML algorithms. However, the main bottleneck of the algorithm is required to build a huge affinity matrix for Markov chain construction. When the numbers of training and testing instances are large, the construction cost would be very expensive.

Besides what we have already mentioned, there are still many other MIML learning algorithms that have been developed in the literature such as [13,32,15,2,22,3,9]. Readers who are interested may refer to these references for more details.

1.2. The proposal

The main issue in the design of MIML algorithm is how to handle a large MIML data set effectively and efficiently such that both storage and computational cost can be reduced in the learning process. The main contribution of this paper is to propose an Object-Feature MIML learning algorithm targeting on non-negative data to evaluate the importance of relationships between a set of labels associated with objects of multiple instances. The proposed algorithm computes the ranking score of importance of different labels (label-object ranking) as labeling indicator for each object. Instead of the construction of affinity matrix among objects, we make use of instance-feature matrix (the input MIML data) to construct two Markov chains. One Markov chain is to determine the transition probabilities from features to instances, and the other Markov chain is to determine the transition probabilities from instances to features. The construction cost of the two Markov chains is significantly less than that of the MIML-Markov method. By making use of the object-instance relation matrix, we can design a simple iterative scheme to govern ranking score of labels associated with objects and relevance score of features in the MIML problem. With label information from labeled objects, all objects then spread their label ranking scores to their neighbors based on the two Markov chains in a random walk. The spread process is repeated until a global steady state is reached. We are able to show that the proposed algorithm is actually seeking for global optimal solution for a convex minimization problem. We further demonstrate that the proposed method can be formulated by consider normalized linear kernel. Because linear kernel function is explicit and separable, it is not necessary to compute and store a huge affinity matrix among objects/instances compared with the use of other kernel functions. Both the storage and computational time of the proposed algorithm are very efficient. Experimental results are presented to show that the classification performance of the proposed method using normalized linear kernel function is about the same as those using the other kernel functions, but the computational time of the proposed method Also experimental results on two MIML text and image data sets have illustrated that the proposed method is very efficient while the classification performance is still competitive with the other MIML algorithms: Markov-MIML, MIML-*k*NN, MIML-SVM and M^{3}MIML.

The rest of this paper is organized as follows. In Section 2, we present the proposed fast MIML method. In Section 3, we show a preprocessing step to adapt features with negative data for the proposed Markov chain method. In Section 4, experimental results will be reported and discussed. Finally, some concluding remarks will be given in Section 5.

2. The proposed method

Suppose there are multiple instances among objects in MIML data and $\mathbf{x}_i^{(j)}$ is the d-dimensional feature vector for the j-th instance of the i-th object. The MIML data can be represented by an d-by-n instance-feature matrix as follows:

$$\mathbf{X} = \left[\underbrace{\mathbf{x}_1^{(1)} \cdots \mathbf{x}_{n_1}^{(1)}}_{\text{1st object}} \middle| \underbrace{\mathbf{x}_1^{(2)} \cdots \mathbf{x}_{n_2}^{(2)}}_{\text{2nd object}} \middle| \cdots \middle| \underbrace{\mathbf{x}_1^{(m)}, \ldots, \mathbf{x}_{n_m}^{(m)}}_{\text{mth object}} \right],$$

where n is the total number of instances in the MIML data, i.e., $n = \sum_{i=1}^{m} n_i$. Each column contains the features of each instance.

2.1. The Markov-MIML Algorithm

In the Markov-MIML algorithm [26], the first step is to construct the affinity matrix among instances. In particular, the affinity between the s-th instance of the i-th object and the t-th instance of the j-th object is calculated as follows:

$$a_{i,j,s,t} = \exp\left[\frac{-\| \mathbf{x}_s^{(i)} - \mathbf{x}_t^{(j)} \|_2^2}{2\sigma^2} \right].$$

Here $\|\cdot\|_2$ is the Euclidean distance and σ^2 is the Gaussian kernel parameter. Then the n-by-n affinity matrix $\mathbf{A} = [\mathbf{A}_{i,j}]$ where the (i, j)-th block is an n_i-by-n_j matrix $\mathbf{A}_{i,j} = [a_{i,j,s,t}]$ with $s = 1, \ldots, n_i$ and $t = 1, \ldots, n_j$ is obtained. The instance-to-object-relation matrix is also a block diagonal matrix $\mathbf{B} = [\mathbf{B}_{i,j}]$ where the (i,j)-th block is a zero matrix except $i=j$. For the (i,i)-th block, $\mathbf{B}_{i,i}$ is a 1-by-n_i matrix where all its entries are equal to 1. This block indicates the relation between the i-th object and its association instances. The size of $\mathbf{B}$ is m-by-n. The next step is to transfer the affinity information among instances into object level by utilizing instance-to-object-relation matrix:

$$\mathbf{S} = \mathbf{B}\mathbf{A}\mathbf{B}^T.$$

Here $\mathbf{S}$ is the similarity matrix among objects which can be treated as nearest neighbors affinity for all the objects. By normalizing

Please cite this article as: M.K. Ng, et al., A fast Markov chain based algorithm for MIML learning, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.08.033