Learning a Distance Metric from Multi-instance Multi-label Data

Rong Jin[1] Shijun Wang[2] Zhi-Hua Zhou[3]
[1]Dept. of Computer Science & Engineering, Michigan State University, East Lansing, MI 48824
[2]Dept. of Radiology & Imaging Sciences, National Institutes of Health, Bethesda, MD 20892
[3]National Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093, China
rongjin@cse.msu.edu wangshi@cc.nih.gov zhouzh@lamda.nju.edu.cn

Abstract

Multi-instance multi-label learning (MIML) refers to the learning problems where each example is represented by a bag/collection of instances and is labeled by multiple labels. An example application of MIML is visual object recognition in which each image is represented by multiple key points (i.e., instances) and is assigned to multiple object categories. In this paper, we study the problem of learning a distance metric from multi-instance multi-label data. It is significantly more challenging than the conventional setup of distance metric learning because it is difficult to associate instances in a bag with its assigned class labels. We propose an iterative algorithm for MIML distance metric learning: it first estimates the association between instances in a bag and its assigned class labels, and learns a distance metric from the estimated association by a discriminative analysis; the learned metric will be used to update the association between instances and class labels, which is further used to improve the learning of distance metric. We evaluate the proposed algorithm by the task of automated image annotation, a well known MIML problem. Our empirical study shows an encouraging result when combining the proposed algorithm with citation-kNN, a state-of-the-art algorithm for multi-instance learning.

1. Introduction

Distance metric learning aims to learn a distance metric from the training data that tries to maintain the class information of examples by their distances, i.e., examples sharing the same class are close to each other while examples from different classes are separated by a large distance. During the past few years, a large number of studies are devoted to distance metric learning [17]. Most of them assume that every training instance is labeled by a single class label. Multi-instance multi-label learning (MIML) [23] is a recent framework for learning ambiguous data which finds application in a wide range of real-world tasks [18, 19, 20, 23]. Unlike conventional setup of supervised learning where each instance is labeled by a single-class label, MIML refers to the learning problems where each example is represented by a bag/collection of instances and is assigned to multiple classes. In this paper, we consider the problem of learning a distance metric from multi-instance multi-label data. The main challenge arises from the fact that the class labels are assigned to each bag, not each instance. As a result, it is unclear, which instance in a bag is associated with which class label assigned to the bag. This unknown association between instances and class labels makes it difficult to directly apply the existing algorithms for distance metric learning. In this paper, we present an iterative algorithm for multi-instance multi-label distance metric learning. It alters between two steps, i.e.,

- estimating the association between instances in bags and the class labels assigned to bags, and
- learning a distance metric from the estimated association between instances and class labels.

Our empirical study with automatic image annotation, a typical MIML problem [18], shows encouraging results of classification when combining the proposed distance metric learning algorithm with the citation-kNN algorithm, a famous algorithm for multi-instance learning.

The rest of the paper is organized as follows. Section 2 briefly reviews metric learning and multi-instance multi-label learning. Sections 3 and 4 formulate the MIML metric learning problem and present an iterative algorithm for the related optimization problem. Experimental results are discussed in Section 5. Section 6 concludes this paper.

2. Related Work

The objective of metric learning is to learn an optimal mapping, either linear or nonlinear, in the original feature space or the reproducing kernel Hilbert space, from training data. Existing approaches can be classified into the categories of unsupervised metric learning and supervised metric learning, depending on whether or not label or side-information is used to learn the optimal metric. Principal component analysis, locally linear embedding (LLE) [9], ISOMAP [11], etc. are typical unsupervised metric learning methods. Most of the algorithms for supervised metric learning are designed to learn either from the class label information or from the side information that is usually cast in the form of pairwise constraints (i.e., must-link constraints and cannot-link constraints) . In the seminar work of Xing et al. [15], the authors proposed to learn a distance metric from pairwise constraints. The optimal metric is found to minimize the distances between data points in must-link constraints and simultaneously maximize the distances between data points in cannot-link constraints. After that, a number of methods and criteria are proposed from supervised metric learning. For example, Weinberger et al. [14] proposed the maximum-margin nearest neighbor (LMNN) classifier that learns an optimal metric for kNN classifiers in a maximum margin framework. Relevance component analysis [10] is another popular approach for supervised metric learning. Data points in the same classes are grouped into the so-called chunklets, and the distance metric is computed based on the covariance matrix of each chunklet. For more information about metric learning, we refer the readers to a recent survey [17].

Multi-instance learning (MIL) was first formulated by Dietterich et al. [4] in the study of drug activity prediction. Maron and Lozano-Pérez [8] proposed the DD algorithm which tries to search for a point in the feature space with the maximum diverse density. This algorithm was further extended by introducing expectation maximization (EM) algorithm for estimating which instance(s) in a bag is responsible for the assigned class label [21]. As a natural extension of the classical k nearest neighbor (k-NN) classifier, citation-kNN was proposed by Wang and Zucker [13], in which a Hausdorff distance is used to measure the distance between bags, and both 'citers' and 'references' are considered in calculating neighbors. Later, kernel methods for MIL were developed [1, 3, 7], as well as ensemble methods [12, 16, 22].

Multi-instance multi-label learning (MIML) generalizes MIL by allowing each bag to be assigned to multiple class labels. It was first proposed in [23], and was shown to be useful for tasks involving ambiguous data objects such as image classification and text categorization in which objects are naturally described by multiple instances and associated with multiple class labels simultaneously. In [23], two classical supervised learning algorithms, AdaBoost and SVM, were adapted to MIML. A more efficient SVM algorithm for MIML was proposed in [20].

As pointed out before, the key challenge of MIML distance metric learning arises from the unknown association between the instances in bags and the class labels assigned to the bags, which prevents the direct application of the existing algorithms for supervised metric learning. We also want to point out that decomposing a multi-label task into a set of binary tasks usually results in a suboptimal solution due to the neglect of the correlation among classes [6]. To our best knowledge, this is the first study devoted to learning a distance metric from multi-instance multi-label data.

3. Metric Learning from MIML Data

We first introduce the basic of multi-instance multi-label learning, followed by the definition of distance between bags and the design of the objective function for MIML distance metric learning.

3.1. Multi-Instance Multi-Label Learning

Let m and n denote the number of class labels and the number of training examples, respectively. We denote by $\mathcal{D} = \{(X_i, y_i), i = 1, \ldots, n\}$ the labeled examples that are used for training distance metrics. Each $X_i = (x_i^1, \ldots, x_i^{n_i})$ is a bag of n_i instances, and every instance $x_i^j \in \mathbb{R}^d$ is a vector of d dimensions. Every class assignment $y_i \in \{0, 1\}^m$ is a binary vector, with $y_i^k = 1$ indicating bag X_i is assigned to class c_k and $y_i^k = 0$ otherwise. Assume (a) Bag X is assigned to class $c \Longleftrightarrow$ at least one instance in X belongs to c, and (b) Bag X is not assigned to class $c \Longleftrightarrow$ no instance in X belongs to c.

3.2. Distance Between Bags

Given two instances x_1 and x_2, the Mahalanobis distance is define as $d(x_1, x_2) = |x_1 - x_2|_A^2 = (x_1 - x_2)^\top A(x_1 - x_2)$ where $A \in \mathbf{S}_+^{d \times d}$ is the distance metric to be learned ($\mathbf{S}_+^{d \times d}$ is the space of all $d \times d$ positive-semi definite matrices). To develop a metric learning algorithm for MIML data, we define the distance between two bags X_i and X_j as the minimum distance among the instances in the two bags, i.e.,

$$D(X_i, X_j) \quad = \quad \min_{1 \le k \le n_i, 1 \le l \le n_j} |x_i^k - x_j^l|_A^2 \,. \qquad (1)$$

The above definition indicates that the relationship between two bags is dictated by the shortest distance between instances in the two bags. This is reasonable since we assume that most instances in a bag are irrelevant to the target classes. The following proposition provides an alternative form of the bag distance in (1), which is useful for deriving the optimization algorithm later on.

Proposition 1. *The bag distance defined in (1) is equivalent to the following expression*

$$D(X_i, X_j) = \min_{q_i \in \Delta_{n_i}, q_j \in \Delta_{n_j}} \sum_{k=1}^{n_i} \sum_{l=1}^{n_j} q_i^k q_j^l |x_i^k - x_j^l|_A^2 \qquad (2)$$

where $\Delta_n = \{q \in \mathbb{R}_+^n | \sum_{k=1}^n q^k = 1\}$ and $\mathbb{R}_+^n$ is a vector space whose items are positive numbers or zero.

Given the distance between bags defined in (1), a straightforward approach is to extend the conventional approaches for distance metric learning to MIML data. For instance, searching for the distance metric A that minimizes the distance between bags in the same classes, and maximizes the distance between bags from different classes. This is however insufficient when each bag is assigned to multiple classes simultaneously, because two bags could share some common classes and in the meantime differ in the assignment of other classes. To address this challenge, we propose to combine data clustering with metric learning. In particular, we introduce multiple centers for each class. For each class c_l ($l = 1, \ldots, m$), we introduce K centers, denoted by $Z_l = \{z_l^i\}$ ($i = 1, \ldots, K$) where $z_l^i \in \mathbb{R}^d$ is a center for class c_l. We further introduce notation $Z = (Z_1, \ldots, Z_m)$ to include the centers of all classes. Since the centers of a class c_l are represented by bag Z_l, we can measure the distance between a bag X_i and a class c_j by the distance between bags X_i and Z_j, i.e.,

$$d(X_i, c_j) = D(X_i, Z_j) = \min_{1 \le k \le n_i, 1 \le l \le K} |x_i^k - z_j^l|_A^2 . \quad (3)$$

Similarly, we define the distance between two classes c_i and c_j by the distance between the two corresponding bags Z_i and Z_j, i.e., $D(Z_i, Z_j) = \min_{1 \le k, l \le K} |z_i^k - z_j^l|_2^2$.

3.3. Objective Function

With the defined distance measure between two bags, we now examine the principle of constructing an objective function for MIML distance metric learning. In particular, we consider the following principle to learn optimal distance metrics from MIML data: (I) minimizing the distance between each bag and its assigned classes, and (II) maximizing the distance between classes. We thus follow the idea of Rayleigh ratio, which is widely used in discriminant analysis, to construct the objective function as the ratio between the two factors, i.e.,

$$\min_{\text{tr}(A)=r, A \succeq 0, Z} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j D(X_i, Z_j)}{\sum_{i,j=1}^m D(Z_i, Z_j)(1 - \delta(i,j))} \quad (4)$$

where $r \in \mathbb{N}$ is an integer constant. Note that the constraint $\text{tr}(A) = r$ is introduced to avoid the scaling invariance of the objective function, and will only affect the learned distance metric by a constant factor. To facilitate our computation, we further restrict A to be constructed by a set of orthonormal vectors $\{w_i\}_{i=1}^r$, i.e., $A = \sum_{i=1}^r w_i w_i^\top$ where $w_i^\top w_j = \delta(i,j)$. The resulting optimization problem becomes

$$\min_{A \in \Lambda_r, Z} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j D(X_i, Z_j)}{\sum_{i,j=1}^m D(Z_i, Z_j)(1 - \delta(i,j))} \quad (5)$$

where $\Lambda_r = \{A = WW^\top | W^\top W = I_r, W \in \mathbb{R}^{d \times r}\}$.

4. Optimization Strategy

In this section, we discuss the strategy for solving the optimization problem in (5). We first simplify the distance function in (2), followed by the algorithm for optimization.

4.1. Simplifying Distance Function

First, we have the following proposition to rewrite the distance function in (2).

Proposition 2. *The distance function in (2) is equivalent to*

$$D(X_i, X_j) = \min_{Q \in \Pi(n_i, n_j)} \sum_{k=1}^{n_i} \sum_{l=1}^{n_j} Q_{k,l} |x_i^k - x_j^l|_A^2 , \quad (6)$$

where

$$\Pi(n, m) = \{Q \in [0,1]^{n \times m} : tr(Q\mathbf{1}) = \mathbf{1}, rank(Q) = 1\} .$$

The proof of the above proposition can be found in a longer version of the paper. Optimization under the constraint of rank is usually NP-hard. Given the result in Proposition 2 and in order to make it computationally tractable, we then simplify the definition of bag distance by dropping the rank constraint, which results in the following simplified definition

$$D(X_i, X_j) = \min_{\substack{Q \in \mathbb{R}_+^{n_i \times n_j} \\ \text{tr}(Q\mathbf{1}) = \mathbf{1}}} \sum_{k=1}^{n_i} \sum_{l=1}^{n_j} Q_{i,j} |x_i^k - x_j^l|_A^2 . \quad (7)$$

Using the distance function (7), we can rewrite the optimization problem in (5). We introduce $Q^{(i,j)}$ for measuring the distance between a bag X_i and a class label c_j, and $P^{(i,j)}$ for measuring the distance between two class labels c_i and c_j. The resulting optimization problem becomes

$$\min_{A \in \Lambda_r, Q, P, Z} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} \sum_{l=1}^K Q_{k,l}^{(i,j)} |x_i^k - z_j^l|_A^2}{\sum_{i,j=1}^m (1 - \delta(i,j)) \sum_{k,l=1}^K P_{k,l}^{(i,j)} |z_i^k - z_j^l|_A^2} \quad (8)$$

$$\text{s. t.} \quad Q^{(i,j)} \in \mathbb{R}_+^{n_i \times n_j}, Q^{(i,j)}\mathbf{1} = \mathbf{1}, i, j = 1, \ldots, n$$
$$P^{(i,j)} \in \mathbb{R}_+^{K \times K}, P^{(i,j)}\mathbf{1} = \mathbf{1}, i, j = 1, \ldots, K$$

4.2. Alternating Optimization

We present an alternating optimization algorithm for (8). In particular, we divide the variables into three groups A, $\{Q, P\}$, and Z. We optimize each group of variables with the other groups of variables fixed. It is noticeable that our optimization problem is more challenging than common non-convex optimization problems since each step of alternating optimization requires solving a non-convex optimization problem.

Optimizing $\{Q, P\}$ with A and Z fixed It is straightforward to verify that for each bag X_i and each of its assigned class c_j (i.e., $y_i^j = 1$), we have the following optimal solution for $Q^{(i,j)}$:

$$Q_{k,l}^{(i,j)} = \begin{cases} 1 & (k,l) = \underset{1 \le k' \le n_i, 1 \le l' \le K}{\arg\min} |x_i^{k'} - z_j^{l'}|_A \\ 0 & \text{otherwise} \end{cases}$$

Similar, for any two class labels c_i and c_j, we have the following optimal solution for $P^{(i,j)}$:

$$P_{k,l}^{(i,j)} = \begin{cases} 1 & (k,l) = \underset{1 \le k', l' \le K}{\arg\min} |z_i^{k'} - z_j^{l'}|_A \\ 0 & \text{otherwise} \end{cases}$$

Optimizing A with $\{Q, P\}$ and Z fixed The corresponding optimization problem is

$$\min_{A \in \Lambda_r} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} \sum_{l=1}^K Q_{k,l}^{(i,j)} |x_i^k - z_j^l|_A^2}{\sum_{i,j=1}^m (1 - \delta(i,j)) \sum_{k,l=1}^K P_{k,l}^{(i,j)} |z_i^k - z_j^l|_A^2} \quad (9)$$

Note that the above problem is not a convex optimization since (a) the objective function is a linear fraction function and therefore is non-convex, and (b) domain Λ_r is non-convex. Here, we present an efficient approach to solve (9) by using the Rayleigh ratio. We define

$$U = \sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} \sum_{l=1}^K Q_{k,l}^{(i,j)} (x_i^k - z_j^l)(x_i^k - z_j^l)^\top$$

$$V = \sum_{i,j=1}^m (1 - \delta(i,j)) \sum_{k,l=1}^K P_{k,l}^{(i,j)} (z_i^k - z_j^l)(z_i^k - z_j^l)^\top .$$

The following theorem shows the optimal solution to (9).

Theorem 1. *The problem in (9) is equivalent to*

$$\min_{W \in \mathbf{R}^{d \times r}, W^\top W = I_r} \frac{tr(W^\top U W)}{tr(W^\top V W)} \quad (10)$$

The optimal solution to $W = (w_1, \ldots, w_r)$ is the first r principal eigenvectors of the generalized eigenvector problem $V w_i = \lambda U w_i$.

Optimizing Z with A and $\{Q, P\}$ fixed The corresponding optimization problem becomes

$$\min_{Z} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} \sum_{l=1}^K Q_{k,l}^{(i,j)} |x_i^k - z_j^l|_A^2}{\sum_{i,j=1}^m (1 - \delta(i,j)) \sum_{k,l=1}^K P_{k,l}^{(i,j)} |z_i^k - z_j^l|_A^2} \quad (11)$$

Again, the above problem is non-convex. In order to efficiently solve (11), we first have the following proposition.

Proposition 3. *Problem in (11) is equivalent to the following optimization problem*

$$\min_{\lambda \ge 0} \lambda \quad \textit{s. t. } \exists Z \; f(\lambda, Z) = 0 \quad (12)$$

where $f(\lambda, Z) = \phi(Z) - \lambda\varphi(Z)$, and $\phi(Z)$ and $\varphi(Z)$ are defined as

$$\phi(Z) = \sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} \sum_{l=1}^K Q_{k,l}^{(i,j)} |x_i^k - z_j^l|_A^2$$

$$\varphi(Z) = \sum_{i,j=1}^m (1 - \delta(i,j)) \sum_{k=1}^K \sum_{l=1}^K P_{k,l}^{(i,j)} |z_i^k - z_j^l|_A^2$$

Given the optimization problem in (12), a straightforward approach is to convert (12) into a sequence of feasibility problems. More specifically, we consider a bisection approach for finding the optimal value for λ. We maintain the largest and the smallest values for λ, denoted by $\lambda_{\max}$ and $\lambda_{\min}$. In each iteration of bi-search, we set $\lambda = (\lambda_{\max} + \lambda_{\min})/2$, and try to solve the feasibility problem $\exists Z f(\lambda, Z) = 0$. This is equivalent to show (a) $\max_Z f(\lambda, Z) \ge 0$ and (b) $\min_Z f(\lambda, Z) \le 0$. If the feasibility problem is satisfied, we have $\lambda_{\max} = \lambda$; otherwise $\lambda_{\min} = \lambda$. Details of this algorithm can be found in a longer version of the paper.

Below we discuss a computationally more efficient approach for (12) when each class c_j is represented by a single center z_j. Given that each class has a single center, we simplify (12) as

$$\min_{Z} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} Q_k^{(i,j)} |x_i^k - z_j|_A^2}{\sum_{i,j=1}^m |z_i - z_j|_A^2} \quad (13)$$

We define $\hat{x}_i^k = W^\top x_i^k$ and $\hat{z}_j = W^\top z_j$ and write (13) as

$$\min_{\hat{Z}} \frac{\sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} Q_k^{(i,j)} |\hat{x}_i^k - \hat{z}_j|^2}{\sum_{i,j=1}^m |\hat{z}_i - \hat{z}_j|^2} \quad (14)$$

Let $\hat{Z}'$ be the current solution for $\hat{Z}$, and our goal is to reduce the objective in (14) with a new solution $\hat{Z}$. We thus consider an relaxed problem of (14) as

$$\min_{\hat{Z}} \sum_{i=1}^n \sum_{j=1}^m y_i^j \sum_{k=1}^{n_i} Q_k^{(i,j)} |x_i^k - \hat{z}_j|^2 \quad (15)$$
$$\text{s. t. } \sum_{i,j=1}^m |\hat{z}_i - \hat{z}_j|^2 \ge t$$

where $t = \sum_{i,j=1}^m |\hat{z}_i' - \hat{z}_j'|^2$.

Proposition 4. *Let Z' be the existing solution for Z, and $\tilde{Z}$ be the solution that optimize (16). Let $\mathcal{L}(Z)$ denote the objective function in (12), i.e., $\mathcal{L}(Z) = \phi(Z)/\varphi(Z)$. We have $\mathcal{L}(\tilde{Z}) \le \mathcal{L}(Z')$.*

The above proposition indicates that the new solution obtained by optimizing (16) will guarantee to reduce the objective function in (12). Below we describe a coordinate descent approach for solving (16).

By fixing $\hat{z}_j, j \neq l$ except $\hat{z}_l$, we have the following optimization problem for $\hat{z}_l$:

$$\min_{\hat{z}_l} \; a|\hat{z}_l|^2 - 2\hat{z}_l^\top v + h \quad \text{s. t. } |\hat{z}_l|^2 - 2\hat{z}_l^\top u \geq s \qquad (16)$$

where

$$\begin{aligned} u &= \frac{1}{m-1}\sum\nolimits_{j\neq l} \hat{z}_j \\ s &= \frac{1}{2(m-1)}\left(t - \sum_{j\neq l}\sum_{k\neq l} |\hat{z}_j - \hat{z}_k|^2 - 2\sum_{j\neq l}|\hat{z}_j|^2\right) \\ v &= \sum\nolimits_{i=1}^{n}\sum\nolimits_{k=1}^{n_i} y_i^l Q_k^{(i,l)} \hat{x}_i^k \\ h &= \sum\nolimits_{i=1}^{n}\sum\nolimits_{k=1}^{n_i} y_i^l Q_k^{(i,l)} |\hat{x}_i^k|^2 \\ a &= \sum\nolimits_{i=1}^{n}\sum\nolimits_{k=1}^{n_i} y_i^l Q_k^{(i,l)} \end{aligned}$$

It is important to note that (16) is a non-convex optimization problem since the constraint $|\hat{z}_l|^2 - 2\hat{z}_l^\top u \geq s$ is a non-convex constraint. We can solve the optimization problem in (16) via the S-procedure [2].

Theorem 2. *The optimal solution to (16) is*

$$\hat{z}_l = \frac{v - \lambda u}{a - \lambda} \qquad (17)$$

where

$$\lambda = \begin{cases} a - \min\left(\frac{|v-au|}{\sqrt{s+|u|^2}}, a\right) & s + |u|^2 \leq 0 \\ 0 & \textit{otherwise} \end{cases}$$

The proof of the above theorem can be found in a longer version of the paper.

5. Experiments

5.1. Data and Settings

To validate our method, we evaluate it on the task of automated image annotation. We use the same image data set which had been used by Duygulu et al. in [5]. It includes $4,500$ train images and 500 test images selected from the COREL image data set. Each image was segmented into no more than 10 regions by Normalized Cut, and each region was represented by 36 visual features. A K-means clustering algorithm was applied to quantize the image regions into 500 blobs. A total of 371 keywords was assigned to $5,000$ images. In our experiment, we only consider the first 20 most popular keywords since most of keywords are only used for annotating a few images. This selection results in a total $3,947$ training images and 444 test images.

The focus of this study is to evaluate the efficacy of the proposed algorithm for learning a distance metric from multi-instance multi-label data. To this end, we first learn a distance metric from the training images, and the learned distance metric is then used by the citation-kNN algorithm [13] to annotate the test images. We extends the classical kNN classifier, which is originally designed for multi-instance learning, to MIML learning. This is achieved by measuring the distance between two bags X_i and X_j with a Hausdorff distance that is defined as

$$H(X_i, X_j) = \min_{1\leq k\leq n_i} \max_{1\leq l\leq n_j} |x_i^k - x_j^l|_A \,, \qquad (18)$$

where A is the metric learned by the proposed algorithm. To determine the class labels for a given test example, citation-kNN considers both references and citers. Given a test bag X, we define its references as the R nearest bags in the training set, and its citers as the training bags for which X is its C nearest neighbors. The class labels of X is decided by a majority vote of the R reference bags and the C citing bags. Using the citation-kNN, we measure the quality of the learned distance metric by the annotation accuracy of citation-kNN. Finally, for the proposed algorithm, we set the number of centers for each class to be one, i.e., $K = 1$, and the number of iterations to be ten, mainly for the computational efficiency.

To measure the MIML learning performance, we adopt three different metrics used in [23]. Assume we have n_t test bags. Given a test bag X_i that is labeled by $y_i \in \{0,1\}^m$, we denote by $f(X,l)$ the score of class c_l for X computed by the citation-kNN algorithm, with $f(X,l) > 0$ indicating that X should be assigned to c_l. We further denote by $rank_f(X,l)$ the rank of class c_l for bag X. Using these notations, the three metrics are defined as follows:

- *One-error* measures the performance by considering the top-ranked proper label of the test bag according to

$$\text{oneerror} = \frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} I\left(y_i^{l_i} = 0\right),$$

where $l_i = \arg\max_{l\in[1,m]} f(X_i, l)$. For single-label problems, it reduces to the ordinary classification error.

- *Coverage* measures the performance by considering the lowest-ranked proper label of the test bag according to

$$\text{coverage} = \frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} \max_{l:y_i^l=1} rank_f(x,l) - 1$$

The smaller the coverage, the better the performance.

- *Average precision* measures the performance by considering all proper labels of the test bag according to

$$\text{avgprec} = \frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} \frac{1}{\sum_{l=1}^{m} y_i^l}\sum\nolimits_{l:y_i^l=1} \frac{b_f(X_i,l)}{rank_f(X_i,l)}$$

Table 1. Annotation performance of citation-kNN on the COREL image dataset. ↓: the lower the metric, the better the performance; ↑: the larger the metric, the better the performance. ML denotes MIML distance metric learning.

	One-error (↓)		Coverage (↓)		Avg. Precision (↑)	
	without ML	with ML	without ML	with ML	without ML	with ML
R=5,C=5	0.696	0.583	6.869	6.191	0.436	0.504
R=10,C=10	0.676	0.565	6.441	5.847	0.459	0.524
R=15,C=15	0.640	0.586	6.110	5.574	0.483	0.527
R=20,C=20	0.633	0.570	6.000	5.507	0.490	0.535

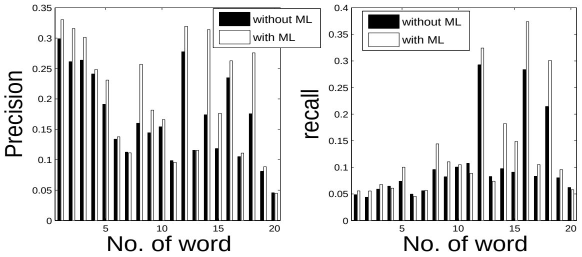

Figure 1. Average precision (left) and recall (right) of each keyword for citation-kNN with and without using metric learning.

where $b_f(X_i, l)$ measures the number of assigned class labels that are ranked before c_l, i.e.,

$$b_f(X_i,l) = \sum_{l':y_i^{l'}=1} I(rank_f\left(X_i, l^{'}\right) \leq rank_f\left(X_i,l\right))$$

5.2. Results

Table 1 summarizes the performance of citation-kNN on test set with four different configurations generated by varying R and C. By comparing the results obtained using the learned metric to those without using the learned metric, we can find that the learned metric is indeed able to significantly improve the performance of citation-kNN. This suggests that the proposed algorithm is effective in identifying appropriate distance metrics for training examples.

To examine the effect of metric learning on the prediction of different keywords, in Figure 1 we show the average precision and recall for each word in the test set. In this study, we set $R = C = 10$ for citation-kNN. From Figure 1 we observe that for average precision, by using the learned distance metric, the performance of citation-kNN is improved by 16 out of 20 keywords; for average recall, the performance is improved for 14 out of 20 keywords. We thus verify that the proposed algorithm is able to learn appropriate distance metrics from MIML data.

Figure 2 shows some example test images and the nearest images identified by citation-kNN with/without MIML distance metric learning. We clearly observe that by using metric learning, the nearest neighbors are semantically more relevant to the test images than without using metric learning, which further validates the efficacy of the proposed algorithm.

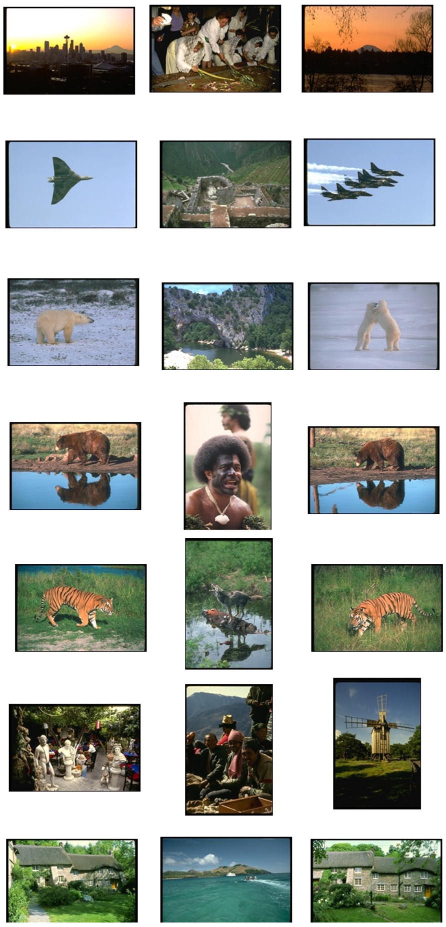

Figure 2. Comparisons of nearest images identified by citation-kNN with/without metric learning. The first column shows some test images; the second/third columns show the nearest reference image in training set identified by citation-kNN without/with metric learning, respectively.

6. Conclusion

In this paper, we study the problem of learning a distance metric from multi-instance multi-labele data. It is significantly more challenging than the conventional setup of distance metric learning because of the difficulty in associating instances in a bag with the class labels assigned to the bag. To address this challenge, we propose an iterative algorithm by alternating between the step of estimating instance-label association and the step of learning distance metrics from the estimated association. Empirical study on automated image annotation shows an encouraging result when combining the proposed method with citation-kNN, a state-of-the-art algorithm for multi-instance learning. Besides citation-kNN, the proposed algorithm for learning distance metrics from MIML data can be combined with the other MIML classifiers in which a distance measure is used as part of classification scheme. We plan to investigate the integration of the proposed algorithms with the other approaches for MIML learning.

Acknowledgements

This research was supported partially by NSF (IIS-0643494), US ARO (W911NF-08-1-0403), NSFC (60635030, 60721002), 863 Program (2007AA01Z169), JiangsuSF (BK2008018), Jiangsu 333 Program and MSRA IST Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of these funding agencies. We also would like to thank Kobus Barnard for providing data used in their ECCV2002 paper.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS 15*, pages 561–568. 2003.

[2] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[3] P.-M. Cheung and J. T. Kwok. A regularization framework for multiple-instance learning. In *ICML*, pages 193–200, 2006.

[4] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple-instance problem with axis-parallel rectangles. *AIJ*, 89(1-2):31–71, 1997.

[5] P. Duygulu, K. Barnard, J. F. G. de Freitas, and D. A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, pages 349–354, 2002.

[6] A. Elisseeff and J. Weston. A kernel method for multi-labelled classification. In *NIPS 14*, pages 681–687. 2002.

[7] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. In *ICML*, pages 179–186, 2002.

[8] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *NIPS 10*, pages 570–576. 1998.

[9] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[10] N. Shental, T. Hertz, D. Weinshall, and M. Pavel. Adjustment learning and relevant component analysis. In *ECCV*, pages 776–792, 2002.

[11] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[12] P. Viola, J. Platt, and C. Zhang. Multiple instance boosting for object detection. In *NIPS 18*, pages 1419–1426. 2006.

[13] J. Wang and J.-D. Zucker. Solving the multi-instance problem: A lazy learning approach. In *ICML*, pages 1119–1125, 2000.

[14] K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS 18*, pages 1473–1480. 2006.

[15] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *NIPS 15*, pages 505–512. 2003.

[16] X. Xu and E. Frank. Logistic regression and boosting for labeled bags of instances. In *PAKDD*, pages 272–281, 2004.

[17] L. Yang and R. Jin. Distance metric learning: A comprehensive survey. Technical report, Department of Computer Science & Engineering, Michigan State University, 2006.

[18] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang. Joint multi-label multi-instance learning for image classification. In *CVPR*, 2008.

[19] M.-L. Zhang and Z.-H. Zhou. Multi-label learning by instance differentiation. In *AAAI*, pages 669–674, 2007.

[20] M.-L. Zhang and Z.-H. Zhou. M^{3}MIML: A maximum margin method for multi-instance multi-label learning. In *ICDM*, pages 688–697, 2008.

[21] Q. Zhang and S. A. Goldman. EM-DD: An improved multi-instance learning technique. In *NIPS 14*, pages 1073–1080. 2002.

[22] Z.-H. Zhou and M.-L. Zhang. Ensembles of multi-instance learners. In *ECML*, pages 492–502, 2003.

[23] Z.-H. Zhou and M.-L. Zhang. Multi-instance multi-label learning with application to scene classification. In *NIPS 19*, pages 1609–1616. 2007.