Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence

Multi-Instance Learning with Distribution Change*

Wei-Jia Zhang and **Zhi-Hua Zhou**
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
{zhangwj,zhouzh}@lamda.nju.edu.cn

Abstract

Multi-instance learning deals with tasks where each example is a bag of instances, and the bag labels of training data are known whereas instance labels are unknown. Most previous studies on multi-instance learning assumed that the training and testing data are from the same distribution; however, this assumption is often violated in real tasks. In this paper, we present possibly the first study on multi-instance learning with distribution change. We propose the MICS approach by considering both bag-level and instance-level distribution change. Experiments show that MICS is almost always significantly better than many state-of-the-art multi-instance learning algorithms when distribution change occurs; and even when there is no distribution change, their performances are still comparable.

Introduction

Multi-instance learning deals with learning tasks where each example is a *bag* of instances, and the bag labels of training data are known whereas instance labels are unknown. Among the many possible assumptions (Foulds and Frank 2010), it is usually assumed that a bag is positive if there is at least one positive instance; thus, negative bags contain only negative instances. For example, in image retrieval, we can treat each image as a bag and every possible patch in the image as an instance. Therefore, if there exists a patch which contains the object of interest, we can conclude that the whole image is interesting to the user. It is clear that such framework is helpful in learning tasks with complicated data objects; consequently, it is not strange that a considerable amount of literature has contributed to multi-instance learning (Zhang and Goldman 2001; Andrews, Tsochantaridis, and Hofmann 2003; Zhou and Xu 2007; Zhang et al. 2009; Wang et al. 2011)

It is noteworthy that almost all previous studies on multi-instance learning assumed that the training and testing data are under the same distribution. This assumption significantly reduces the difficulty of multi-instance learning studies; however, it is often violated in real-world tasks. Actually, the discrepancy between training and test distributions has attracted much attention during the past decade. Many algorithms have been proposed to solve the problem including manifold-based algorithms (Zhu et al. 2003), feature representation methods (Blitzer, McDonald, and Pereira 2006; Ben-David et al. 2007; Pan et al. 2009), covariate shift approaches (Zadrozny 2004; Sugiyama et al. 2008; Kanamori, Hido, and Sugiyama 2009), etc. Unfortunately, all these studies focused on single-instance data.

Single-instance techniques for handling distribution change can hardly be applied to multi-instance learning directly, and even its extension is non-trivial. This is because in multi-instance learning the distribution change can occur at bag-level, instance-level, or both. For example, Figure 1 illustrates the possible situations of distribution change if we regard the whole image as a bag. Furthermore, previous techniques assumed that the examples are i.i.d whereas instances in the same bag should not be regarded as i.i.d. in multi-instance learning (Zhou, Sun, and Li 2009). It is clearly evident that in order to address distribution change in multi-instance learning, we need to handle bag-level as well as instance-level distribution change, and consider the non-i.i.d. issue for instances in the same bag.

In this paper, we present possibly the first study on multi-instance learning with distribution change. Considering that covariate shift has attracted the greatest number of studies in single-instance distribution change, we focus on covariate shift in multi-instance learning in this paper. That is, we consider that the distribution of bags $P(X)$ and instances $P(\mathbf{x})$ change, but the corresponding conditional probabilities $P(y|X)$ and $P(.|\mathbf{x})$ do not change. Experiments show that when distribution change occurs, our MICS (Multi-Instance Covariate Shift) approach is significantly better than many state-of-the-art multi-instance learning algorithms and extensions of single-instance covariate shift techniques. Even when training and testing examples follow the same distribution, the performance of MICS is still comparable to other state-of-art multi-instance algorithms. In other words, users can simply deploy our MICS approach, with the expectation that it will achieve good performance no matter whether there are distribution change or not.

The rest of the paper is organized as follows. We start by a brief review of some related work. Then we present our MICS approach. After that we report our experiment results, which is followed by the conclusion.

*This research was supported by NSFC (61333014, 61105043).

(a) Positive training example: a red fox in grass.

(b) Bag-level distribution change: wolves in negative bags.

(c) Instance-level change: a gray fox in grass.

(d) Instance-level change: a red fox in snow.

Figure 1: Illustration of the possible distribution changes in multi-instance learning. The learning target is "fox". 1) A training image showing a red fox in grass. 2) Bag-level distribution change: testing images are foxes and other animals. 3) Instance-level distribution change: testing images are gray foxes, not red ones. 4) Instance-level distribution change: testing images are foxes in snow, not in green grass.

Related Work

Multi-instance learning originated from the investigation of drug activity prediction (Dietterich, Lathrop, and Lozano-Pérez 1997). Since then, many algorithms have been developed. To name a few, Diverse Density and EM-DD (Maron and Ratan 1998; Zhang and Goldman 2001), boosting and resampling methods MIBoosting (Xu and Frank 2004), SMILe (Doran and Ray 2013), large-margin and kernel methods MI-Kernel (Gärtner et al. 2002), mi-SVM and MI-SVM (Andrews, Tsochantaridis, and Hofmann 2003), MissSVM (Zhou and Xu 2007), PPMM (Wang, Yang, and Zha 2008), M^{3}IC (Zhang et al. 2009), eMIL (Krummenacher, Ong, and Buhmann 2013), Multi-instance learning techniques have been applied to diverse applications such as image classification and retrieval (Maron and Ratan 1998), text categorization (Andrews, Tsochantaridis, and Hofmann 2003), computer-aided medical diagnosis (Fung et al. 2007), etc. It is noteworthy that all previous studies of multi-instance learning assumed that the training and testing examples are drawn from the same distribution.

Distribution change between training and testing examples is a common phenomenon in real-world machine learning applications. In covariate shift, it is assumed that the distributions of data examples $P(\mathbf{x})$ differ between training and testing data, but the conditional distribution of the class label given the examples $P(y|\mathbf{x})$ stays the same. In this situation, minimizing empirical risk on training data will cause the learned model to be fitted better to regions with high training density, but we actually want the model to be fitted better to regions with high testing density. Covariate shift techniques aim to solve this problem by incorporating the importance weights $w(x) = p_{test}(x)/p_{train}(x)$ into the training phase of single-instance learning algorithms. weighting the training examples according to the importance weight $w(\mathbf{x}) = p_{test}(\mathbf{x})/p_{train}(\mathbf{x})$. Many algorithms have been proposed to estimate the above weight, such as Kernel Density Estimation, Kernal Mean Matching (Huang et al. 2007), KLIEP (Sugiyama et al. 2008), LSIF and uLSIF (Kanamori, Hido, and Sugiyama 2009), etc. Unfortunately, all previous studies on covariate shift focus on single-instance learning.

The MICS Approach

Notations

Before presenting the details, we introduce our notations. Let $\mathcal{X}$ denote the instance space and $\mathcal{Y}$ denote the label space. The learner is given a set of m training examples $\mathcal{D}_{train} = \{(X_1^{tr}, y_1), \ldots, (X_i^{tr}, y_i), \ldots, (X_m^{tr}, y_m)\}$, where $X_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{iu}, \ldots, \mathbf{x}_{in_i}\} \subseteq \mathcal{X}$ is called a bag with n_i instances and $y_i \in \mathcal{Y} = \{-1, +1\}$ is the label of X_i. Here $\mathbf{x}_{iu} \in \mathcal{X}$ is an instance $\mathbf{x}_{iu} = [x_{iu1}, \ldots, x_{iul}, \ldots, x_{iud}]'$, in which x_{iul} is the value of $\mathbf{x}_{iu}$ at the l-th attribute, and d is the number of attributes. In standard multi-instance assumption, if there exists $t \in \{1, \ldots, n_i\}$ such that $\mathbf{x}_{it}$ is a positive instance, then X_i is a positive bag and thus $y_i = +1$; if all instances in X_i are negative, $y_i = -1$. Yet the value of index t is unknown. In covariate shift, the learner is also given a set of n unlabeled testing bags $\mathcal{D}_{test} = \{X_1^{te}, \ldots, X_n^{te}\}$. The goal is to generate a good learner to predict the labels for the bags in $\mathcal{D}_{test}$ based on the labeled training data and unlabeled testing data. When discussing distribution change, we use subscripts *train* and *test* to denote the distribution of training and testing examples, respectively.

The Goal

In single-instance learning, algorithms treat each instance as an example. Accordingly, covariate shift techniques in single-instance learning aim to estimate the importance weight for each instance; then incorporate the estimated weights into single-instance learning algorithms. Thus, a straightforward way to deal with multi-instance covariate shift is to consider the instances in multi-instance bags as i.i.d samples, estimate the importance weights $w(\mathbf{x}) = p_{test}(\mathbf{x})/p_{train}(\mathbf{x})$ for the instances in training bags with respect to the instances in testing bags by directly applying importance weighting techniques in single-instance learning, and then incorporate the estimated weights to existing multi-instance learning algorithms. However, such a strategy not only neglects the fact that instances in a multi-instance bag are usually not i.i.d, but also ignores the possibility that distribution change can happen both at bag-level and

at instance-level in multi-instance learning. Before presenting our MICS approach, we start by taking a closer look at the different possibilities of distribution change in multi-instance learning.

In multi-instance learning, the first situation of distribution change is possible to happen at bag-level. The bag distributions of training and testing sets are different, $P_{train}(X) \neq P_{test}(X)$, whereas the conditional distributions of label given the bag stay unchanged, $P_{train}(y|X) = P_{test}(y|X)$. For example, in a image classification task where positive examples in training set are images with red fox (Figure 1.a) and negative examples are mostly taken in urban area(i.e., car, human, beaches, buildings, etc.), while the examples in testing set are positive images of foxes and many images of wolves (Figure 1.b). In this case, the classifier built solely on the training set would try to discriminate animals images and non-animal images. However, in order to achieve good performance on testing set, we actually want the classifier to emphasize on discriminating foxes and other animal images such as wolves.

The second situation of multi-instance distribution change is possible to happen at instance-level. Formally, the distributions of instances in the bags change, $P_{train}(\mathbf{x}) \neq P_{test}(\mathbf{x})$; whereas the conditional probability do not change, $P_{train}(.|\mathbf{x}) = P_{test}(.|\mathbf{x})$, implying that whether an instance is relevant or irrelevant to the user's interest stays unchanged between training and testing sets. For example, instance-level distribution change happens when the data were collected near a habitat of red fox, and thus the positive images in training set are mostly red foxes (Figure 1.a); while the testing examples were collected near a gray fox habitat, where the positive images are mostly gray foxes (Figure 1.c). It is also possible that the training images were mostly taken in summer, where the positive images were mostly foxes in grass (Figure 1.a); but the testing images were taken in winter, where the positive images were mostly presented with snowy background (Figure 1.d). Note that the distribution change of single-instance learning lies in the example-level as each example is represented by a single feature vector, whereas the instance-level distribution change of multi-instance learning occurs at the level of building blocks of an example, as each example is a set of instances.

It is noteworthy that the bag-level and instance-level distribution change may take place simultaneously, making the situation of distribution change in multi-instance learning much more complicated than single-instance learning.

The Solution

We now propose our MICS approach. To handle the bag-level distribution change, we treat each multi-instance bag as an entity and estimate the importance weight of bag $W(X)$ by the following linear model

$$\hat{W}(X) = \sum_{p=1}^{b} \alpha_p \varphi_p(X), \tag{1}$$

where α_p are parameters to be learned from data and $\varphi_p(X)$ are the basis functions such that $\varphi_p(X) \geq 0$ for all $X \in \mathcal{D}$ and $p = 1, \ldots, b$.

There are many existing techniques to estimate the parameters α_p. In this paper, we estimate them by minimizing the least square loss between $\hat{W}(X)$ and $W(X)$ as follows:

$$\begin{aligned} L_0(\alpha) &= \frac{1}{2}\int (\hat{W}(X) - W(X))^2 p_{tr}(X) dX \\ &= \frac{1}{2}\int (\hat{W}(X)^2 p_{tr}(X) - 2\hat{W}(X)W(X)p_{tr}(X) \\ &\quad + W(X)^2 p_{tr}(X)) dx, \end{aligned} \tag{2}$$

Given a training set, the third term in Eq.2 is a constant and therefore can be safely ignored during optimization. Let us denote the first two terms of Eq.2 by L, since $W(X) = p_{te}(X)/p_{tr}(X)$, we have:

$$\begin{aligned} L(\alpha) &= \frac{1}{2}\int \hat{W}(X)^2 p_{tr}(X) dX - \int \hat{W}(X) p_{te}(X) dX \\ &= \frac{1}{2}\sum_{p,p'=1}^{b} \alpha_p \alpha_{p'} \left(\int \varphi_p(X)\varphi_{p'}(X) p_{tr}(X) dX \right) \\ &\quad - \sum_{p=1}^{b} \alpha_p \left(\int \varphi_p(X) p_{te}(X) dX \right) \\ &= \frac{1}{2}\alpha^\top \hat{H} \alpha - \hat{h}^\top \alpha, \end{aligned} \tag{3}$$

where $\hat{H}$ be the $b \times b$ matrix with the (p, p')th-element as $h_{p,p'} = \sum_{i=1}^{m} \varphi_p(X_i^{tr})\varphi_{p'}(X_i^{tr})/m$ and h be the b-dimensional vector with the p-th element as $\hat{h}_p = \sum_{j=1}^{n} \varphi_p(X_j^{te})/n$. From Eq.3, we can estimate the coefficients $\{\alpha_p\}_{p=1}^{b}$ by solving the following optimization problem:

$$\min_{\alpha \in \mathbb{R}^b} [\frac{1}{2}\alpha^\top \hat{H} \alpha - \hat{h}^\top \alpha + \frac{\lambda}{2}\alpha^\top \alpha]. \tag{4}$$

It is evident that the solution of (4) can be analytically computed as $\alpha = (\hat{H} + \lambda I_b)^{-1}\hat{h}$, where I_b is the b-dimensional identity matrix, therefore the parameters can be efficiently estimated. It is noteworthy that in single-instance learning the basis functions $\varphi_p(x)$ can be chosen as a fixed number of Gaussian kernels centered at test points (Sugiyama et al. 2008; Kanamori, Hido, and Sugiyama 2009), but here we need to use a kernel to capture the similarity information between multi-instance bags. In this paper we use the MI-Kernel (Gärtner et al. 2002) as follows:

$$k_{MI}(X_i, X_j) = \sum_{u=1}^{n_i}\sum_{v=1}^{n_j} k_g(\mathbf{x}_{iu}, \mathbf{x}_{jv}) \tag{5}$$

where $k_g(\mathbf{x}_{iu}, \mathbf{x}_{jv}) = exp(-\gamma||\mathbf{x}_{iu} - \mathbf{x}_{jv}||^2)$ is a RBF kernel and γ is the kernel width.

We now discuss how to handle instance-level distribution change. If we treat instances in the same bag as i.i.d. samples, the instance-level distribution change can be simply handled by estimating $w_{i.i.d.}(\mathbf{x}_{iu}) = p_{te}(\mathbf{x}_{iu})/p_{tr}(\mathbf{x}_{iu})$ with single-instance covariate shift techniques. However,

previous studies (Zhou, Sun, and Li 2009) disclosed that an important property of multi-instance learning lies in the fact that the instances in the same bag are usually related, and they should not be treated as i.i.d. samples. If we simply treat instances in the same bag as i.i.d. samples, then there is no need to consider multi-instance learning(Zhou and Xu 2007). Thus, when addressing the instance-level distribution change, we need to consider the non-i.i.d. issue.

As (Zhou, Sun, and Li 2009) shows, ϵ-graph is a simple yet effective way for considering the relations between instances in the same bag. Inspired by their work, we use ϵ-graph to help estimate the instance-level importance weights. For each bag X_i, we construct an ϵ-graph $G_i = (V_i, E_i)$, where every instance in X_i is a node of G_i. We then compute the distance of every pair of nodes by the Euclidean distance $d(\mathbf{x}_{iu}, \mathbf{x}_{iv}) = ||\mathbf{x}_{iu} - \mathbf{x}_{iv}||_2$. If the distance between $\mathbf{x}_{iu}$ and $\mathbf{x}_{iv}$ is smaller than ϵ, we establish an edge between them. Since the graph is constructed by thresholding the edges with ϵ, we treat each edge in graph G_i as equally weighted. Let D_i denote the degree matrix of G_i and d_{iu} denote the uth diagonal element of D_i. We estimate the instance-level importance weight $w(\mathbf{x}_{iu})$ for the u-th instance in bag X_i by

$$w(\mathbf{x}_{iu}) = w_{i.i.d.}(\mathbf{x}_{iu})/(d_{iu} + 1). \quad (6)$$

To understand the intuition of w, consider that with the constructed ϵ-graph, nodes in the same clique can be regarded as related to each other. For a bag X_i, the u-th diagonal element d_{iu} represents the number of instances related to the u-th instance. Thus, if all nodes in G_i are not connected to each other, then none of the instances belongs to the same clique and each instance is treated equally; if the nodes in G_i are clustered into cliques, the contribution of the instances to its concept is related to the number of nodes in each clique; if all the nodes of G_i are connected to each other, thus all instances in the bag belong to the same concept and each instance contributes identically.

After we estimate the bag-level weights $W(X)$ and instance-level weights $w(\mathbf{x})$, there are many ways to incorporate them into existing multi-instance algorithms. For example, for multi-instance algorithms directly deal with instances, we can incorporate the weights as $w_{iu} = W(X_i) \cdot w(\mathbf{x}_{iu})$; for multi-instance algorithms that transform bags into entities and classify the transformed ones, the instance-level weights can be incorporated in the bag transformation step and bag-level weights in the classification step. In this paper, we simply modify the multi-instance kernel in Eq.5 to incorporate the weights as follows:

$$k_{MICS}(X_i, X_j) = \sum_{u=1}^{n_i} \sum_{v=1}^{n_j} \omega_{iu} \cdot \omega_{jv} \cdot k_g(\mathbf{x}_{iu}, \mathbf{x}_{jv}), \quad (7)$$

where $\omega_{iu} = W(X_i) \cdot w(\mathbf{x}_{iu}), \omega_{jv} = W(X_j) \cdot w(\mathbf{x}_{jv})$. As can be seen, we use a very straightforward method to incorporate the instance-level and bag-level important weights, and it is very likely that there are many advanced techniques that can do better than such a simple strategy of weight incorporation. However, in the next section we can see that the proposed MICS method has already worked well with such a simple solution, implying that we can get even better performance by considering better weight incorporation strategies; this is a future issue to be explored.

Experiments

In this section, we empirically evaluate the performance of the proposed MICS approach. We compare MICS with a set of state-of-the-art multi-instance learning algorithms. Firstly, considering that our MICS is a kind of SVM-style approach, we compare with mi-SVM (Andrews, Tsochantaridis, and Hofmann 2003), which is a famous multi-instance SVM algorithm. Secondly, considering that we have motivated the study by using some image data, whereas MILES (Chen, Bi, and Wang 2006) is known as an effective multi-instance algorithm for image tasks, we include it as a baseline for comparison. Thirdly, since we incorporate our importance weights into Eq.5 proposed by (Gärtner et al.), we also include MI-Kernel as a baseline. Moreover, we have been inspired by miGraph (Zhou, Sun, and Li 2009) to treat instances in the same bag as non-i.i.d. samples, and therefore, we also include miGraph in comparison.

To study whether the direct application of single-instance distribution change techniques can work with multi-instance learning, we compare with some baselines. Note that there was no algorithm for multi-instance distribution change before, and thus, we derive two algorithms. The first one, mi-SVM+, is a variant of mi-SVM working by using the single-instance covariant shift algorithm (Kanamori, Hido, and Sugiyama 2009) to estimate the weights for each instances, and incorporating these weights directly to mi-SVM. The second algorithm, MILES+, is a variant derived from MILES in a similar way.

Considering that in the study of distribution change, we need to get access to the test data (otherwise we could not estimate the difference between training/testing distribution), we believe that we should compare with semi-supervised multi-instance learning algorithms that exploit information in the unlabeled testing data. For this purpose, we compare with MissSVM (Zhou and Xu 2007), a semi-supervised multi-instance learning algorithm. During the experiments, the parameters are selected via 5-folds cross validation on the training data.

Experiments on Text Data Sets

First, we perform experiments on text data sets based on the 20 Newsgroups corpus popularly used in text categorization. Each of the 20 news categories corresponds to a data set. Similar to (Settles, Craven, and Ray 2007; Zhou, Sun, and Li 2009), here each positive bag contains approximately 3% posts drawn from the target category, whereas the other instances in positive bags and the instances in negative bags are drawn from non-target categories. Each instance is a post in the corpus represented by the top 200 TFIDF features. We then use a deliberately biased sampling procedure to separate the examples into disjoint training and testing sets. The biased selection scheme is similar as the one used in (Zadrozny 2004; Huang et al. 2007). In detail, we define a random variable s_i

Table 1: Testing accuracy (%, mean ± std.) on text categorization tasks. The best performance and its comparable results (paired t-tests at 95% significance level) are bolded. The last row shows the win/tie/loss counts of MICS versus other methods.

Dataset	mi-SVM	MILES	MI-Kernel	miGraph	MissSVM	mi-SVM+	MILES+	MICS
alt.atheism	83.0±2.1	70.0±3.3	54.8±3.0	65.5±4.0	78.0±1.4	84.3±1.5	69.4±3.3	**86.8±3.2**
comp.graphics	84.6±1.9	59.2±3.0	51.2±1.8	77.8±1.6	83.2±1.8	83.6±1.3	61.0±3.3	**86.0**±3.4
comp.os.ms-win	76.1±2.2	62.8±3.3	52.4±3.0	70.1±1.5	76.5±2.3	**84.6±1.6**	62.1±3.6	78.8±3.0
comp.sys.ibm.pc	62.6±3.1	62.4±3.9	52.0±2.0	59.5±2.7	68.6±2.5	62.3±2.1	64.5±3.8	**80.4±3.3**
comp.sys.mac	75.7±1.2	64.8±3.8	52.4±3.0	79.4±4.8	72.3±1.8	82.6±1.4	61.7±3.7	**84.4±3.4**
talk.politics.guns	76.8±1.3	68.0±3.4	58.0±3.0	72.3±2.1	78.6±1.2	78.7±1.6	68.5±3.5	**80.4±2.4**
talk.politics.mideast	76.2±2.5	64.8±3.3	50.4±1.9	75.5±2.7	78.8±1.3	**89.3±1.0**	68.0±3.3	83.2±3.2
talk.politics.misc	78.4±1.3	72.4±3.5	51.6±2.4	73.8±3.7	74.6±2.4	**80.3±1.5**	70.6±3.5	78.4±3.4
comp.windows.x	70.4±2.6	64.8±3.7	55.2±3.4	71.0±2.8	72.0±1.9	71.0±2.5	66.7±3.7	**84.4**±3.6
rec.autos	76.7±2.4	59.2±3.7	53.6±3.4	71.8±3.1	70.5±2.0	76.3±1.9	56.6±3.6	**86.0±2.8**
rec.sport.hockey	66.3±1.2	59.2±3.0	52.8±3.4	70.1±2.5	70.5±1.8	60.1±1.8	59.3±3.0	90.8±2.2
sci.crypt	66.4±1.8	69.6±3.6	63.6±3.6	72.1±2.1	75.5±2.4	78.8±1.6	69.3±3.6	**88.4±2.9**
misc.forsale	73.4±2.0	64.8±4.1	55.6±3.0	65.2±1.7	67.4±1.6	**82.5±1.2**	57.1±4.0	82.0±2.4
sci.med	70.2±1.3	64.0±3.2	51.6±2.2	72.1±3.9	75.9±2.0	**88.2±1.2**	63.8±3.1	**88.4±1.6**
sci.electronics	64.6±2.5	55.6±2.8	53.6±2.4	84.0±3.4	68.8±2.5	65.0±2.8	54.9±2.7	**94.0±3.0**
rec.sport.baseball	58.5±1.4	60.8±3.5	54.0±32.	64.7±4.7	63.4±2.4	57.3±1.5	60.6±3.5	**86.0±3.4**
rec.motocycles	65.0±1.9	71.2±3.2	68.8±4.1	75.0±6.0	69.2±1.9	64.0±2.0	71.9±3.3	**86.0±2.5**
soc.religion.christ	50.0±0.0	67.6±3.4	52.0±1.4	59.0±2.6	58.0±2.1	50.0±0.0	67.0±3.4	**82.0±2.3**
sci.space	58.7±3.3	64.4±3.3	59.6±3.5	75.7±3.6	70.2±3.2	58.7±1.6	63.3±3.3	**81.2±2.4**
talk.religion.misc	50.2±0.1	62.0±3.2	52.0±2.2	67.5±3.5	60.5±1.8	50.2±0.5	60.1±3.2	77.2±3.3
MICS: W/T/L	20/0/0	20/0/0	20/0/0	20/0/0	20/0/0	15/1/4	20/0/0	-

for all the bags in the data set, where $s_i = 1$ indicates the ith bag is selected into training set, and $s_i = 0$ otherwise. Since smaller feature values predominate in the unbiased data, we sample according to $P(s_i = 1|\sum_{u,l} x_{iul} \leq 3) = 0.2$ and $P(s_i = 0|\sum_{u,l} x_{iul} > 3) = 0.8$. In other words, the testing bags have higher density of instances with small feature values instances than the training bags, whereas the training bags have higher density of instances with large feature values. Note that in this series of experiments we only consider the bag-level distribution change.

On each data set we repeat the experiments for 25 times using the above described biased sampling strategy to generate random training/testing splits, where half of the data are used for training and the other half are used for testing. The average accuracy with standard deviation are reported in Table 1, with the best results and its comparable ones (paired t-tests at 95% significance level) bolded. We can see that existing multi-instance algorithms fail to achieve good performance when distribution change occurs. By incorporating single-instance covariate shift techniques, mi-SVM+ and MILES+ becomes better than mi-SVM and MILES on only a small number of datasets. It is worth noting that the incorporation of single-instance distribution change techniques does not necessarily lead to better performances; for example, the performance of mi-SVM+ is worse than mi-SVM on *comp.graphics* and *rec.sport.hockey*; the performance of MILES+ is worse than MILES on *comp.sys.mac.hardware* and *misc.forsale*. Except 4 losses to mi-SVM+, MICS outperforms other algorithms on all datasets.

Experiments on Image Data Sets

Then, we perform experiments on a number of COREL image datasets. Each image is regarded as a bag, whereas the single-blob approach (Maron and Ratan 1998) is used to extract regions in the image as instances. Here each bag contains 16 instances described by color features.

In this series of experiments we consider both the bag-level and instance-level distribution changes. We split the images in each class into training and testing sets with similar sizes. We first change the bag-level distribution by putting negative bags from different classes into training and testing sets. Then we change the instance-level distribution by bias sampling the positive instances. For example, in the *fox* data set, the training positive examples mostly includes pictures that contain red fox with snow background, while testing positive examples contain mostly gray foxes with forest and grass background. The negative bags in training set are mostly consist of natural scene images (beach, building, forest) whereas those in testing set are mostly consist of animal images (lion, dog, tiger).

On each dataset we repeat the experiments for 25 times and report the average accuracy with standard deviations in Table 2, with the best results and its comparable ones (paired t-tests at 95% significance level) bolded. It is observable that existing multi-instance learning algorithms fail to achieve good performance when distribution change occurs. It is noteworthy that the mi-SVM+ algorithm does not work either, possibly because that now the tasks suffer from both bag-level and instance-level distribution changes. MILES+ improves performance of MILES on some datasets, but it worsen the performance on *sunset* and *text*. Table 2 shows

Table 2: Testing accuracy (%, mean ± std.) on image classification tasks. The best performance and its comparable results (paired t-tests at 95% significance level) are bolded. The last row shows the win/tie/loss counts of MICS versus other methods.

Dataset	mi-SVM	MILES	MI-Kernel	miGraph	MissSVM	mi-SVM+	MILES+	MICS
fox	73.2±0.4	74.4±2.9	74.0±3.3	71.2±2.8	75.8±0.4	73.3±0.2	73.2±3.4	**79.0±0.6**
wolves	74.5±0.2	80.1±2.1	68.8±3.2	75.4±2.4	76.8±0.3	74.5±0.1	**86.5±1.9**	**86.2±1.2**
mountain	94.3±0.2	87.5±2.2	76.0±3.9	92.4±1.5	86.4±0.1	94.4±0.1	96.5±1.4	**98±0.3**
sunset	64.6±0.1	80.4±2.5	76.4±3.8	78.5±3.1	67.5±0.2	64.6±0.1	68.1±1.8	**83.8±0.4**
car	83.4±0.3	81.8±2.9	84.8±3.4	88.4±1.5	78.5±0.3	83.4±0.2	85.3±2.0	**94.0± 0.5**
MICS: W/T/L	5/0/0	5/0/0	5/0/0	5/0/0	5/0/0	5/0/0	4/1/0	-

that on image datasets, our MICS approach achieves the best performance and it is evidently able to handle both bag-level and instance-level distribution changes.

At last, we perform sign-tests and Friedman-tests in conjunction with Bonferroni-Dunn at 95% significance level to determine whether there is any significant difference between the compared algorithms. Both tests show the same results: in terms of all twenty-five datasets mentioned above, MICS is significantly better than all of the compared algorithms.

Experiments on Data Sets without Distribution Change

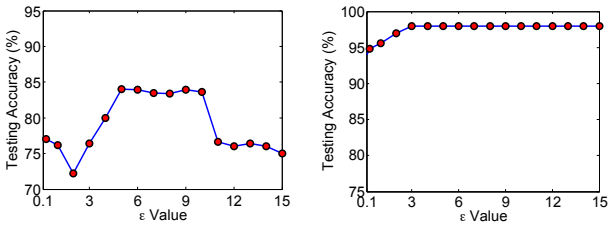
Table 3: Test accuracy (%) on benchmark data sets without distribution change. Performance of the compared algorithms are obtained from related literature (N/A means the result is not reported in the corresponding literature).

Dataset	Musk1	Musk2	Elept	Fox	Tiger
mi-SVM	87.4	83.6	82.0	58.2	78.9
MILES	84.2	83.8	**89.1**	**76.0**	**86.0**
MI-Kernel	82.0	86.8	N/A	N/A	N/A
miGraph	**88.9**	**90.3**	86.8	61.6	85.6
MissSVM	87.6	80.0	N/A	N/A	N/A
MICS	88.0	90.0	86.0	72.7	**86.0**

Currently, there is no reliable techniques available for detecting whether distribution change occurs. Therefore, we evaluate MICS on benchmark datasets where there is no distribution change. Similar to the experimental settings used in previous multi-instance learning studies, we conduct 10-fold cross validations for ten times, and report the average performances in Table 3. Sign-tests and Friedmann-tests in conjunction with Bonferroni-Dunn at 95% significance level show that there are no significant differences between these algorithms over five benchmark datasets. It indicates that, although our MICS approach is designed for multi-instance learning with distribution change, its performance is comparable with state-of-art multi-instance algorithms on datasets without distribution change. As we have mentioned, currently there is no effective routine to detect whether distribution change occurs or not, thus MICS is particularly a good option because it works well no matter whether there is distribution change or not.

Parameter Influence

To study the influence of the ϵ value (used for the ϵ-graph) on the performance of MICS, we perform additional experiments with varied ϵ values. Figure 2 shows some results where each point corresponds to the average performance of the experiments repeated for ten times. We can see that the tuning of ϵ values enables MICS to have good performance, which indicates the effectiveness of considering instances in the bags as non-i.i.d. samples.



(a) Sunset (b) Mountain

Figure 2: Parameter Sensitivity Comparisons

Conclusion

Previous studies on multi-instance learning assumed that the distribution of test set is exactly the same as the distribution of training set, although in real-world tasks their distributions are often different. Directly applying single-instance distribution change techniques to multi-instance learning will not lead to good performance, because in multi-instance learning the distribution change may occur both at the bag-level and instance-level. In this paper, we propose the MICS approach and experiments show that our MICS approach not only performs significantly better than many state-of-art multi-instance learning algorithms when distribution change occurs, but also achieves competitive result when distributions of training and testing examples are the same.

There is much future work to do. To name a few, it will be interesting to study other assumptions of distribution change in multi-instance learning. Employing advanced techniques to handle the non-i.i.d. issue of instances in the same bag may lead to better performance. Moreover, developing refined strategies for incorporating the bag-level and instance-level importance weights might be even more helpful.

References

Andrews, S.; Tsochantaridis, I.; and Hofmann, T. 2003. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems 15*. 561–568.

Ben-David, S.; Blitzer, J.; Crammer, K.; and Pereira, F. 2007. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems 19*. 137–144.

Blitzer, J.; McDonald, R.; and Pereira, F. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 120–128.

Chen, Y.; Bi, J.; and Wang, J. Z. 2006. MILES: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(12):1931–1947.

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

Doran, G., and Ray, S. 2013. SMILe: Shuffled multiple-instance learning. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, 260–266.

Foulds, J., and Frank, E. 2010. A review of multi-instance learning assumptions. *The Knowledge Engineering Review* 25(1):1–25.

Fung, G.; Dundar, M.; Krishnapuram, B.; and Rao, R. B. 2007. Multiple instance learning for computer aided diagnosis. In *Advances in Neural Information Processing Systems 19*. 425–432.

Gärtner, T.; Flach, P. A.; Kowalczyk, A.; and Smola, A. J. 2002. Multi-Instance Kernels. In *Proceedings of the 19th International Conference on Machine Learning*, 179–186.

Huang, J.; Smola, A. J.; Gretton, A.; Borgwardt, K. M.; and Schlkopf, B. 2007. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems 19*. 601–608.

Kanamori, T.; Hido, S.; and Sugiyama, M. 2009. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research* 10:1391–1445.

Krummenacher, G.; Ong, C. S.; and Buhmann, J. 2013. Ellipsoidal multiple instance learning. In *Proceedings of the 30th International Conference on Machine Learning*, 73–81.

Maron, O., and Ratan, A. L. 1998. Multiple-instance learning for natural scene classification. In *Proceedings of the 15th International Conference on Machine Learning*, 341–349.

Pan, S. J.; Tsang, I. W.; Kwok, J. T.; and Yang, Q. 2009. Domain adaptation via transfer component analysis. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 1187–1192.

Settles, B.; Craven, M.; and Ray, S. 2007. Multiple-instance active learning. In *Advances in neural information processing systems 19*. 1289–1296.

Sugiyama, M.; Nakajima, S.; Kashima, H.; Buenau, P. V.; and Kawanabe, M. 2008. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20*. 1433–1440.

Wang, H.; Huang, H.; Kamangar, F.; Nie, F.; and Ding, C. H. 2011. Maximum margin multi-instance learning. In *Advances in Neural Information Processing Systems 24*. 1–9.

Wang, H.-Y.; Yang, Q.; and Zha, H. 2008. Adaptive p-posterior mixture-model kernels for multiple instance learning. In *Proceedings of the 25th International Conference on Machine Learning*, 1136–1143.

Xu, X., and Frank, E. 2004. Logistic regression and boosting for labeled bags of instances. In *Advances in Knowledge Discovery and Data Mining*. Springer. 272–281.

Zadrozny, B. 2004. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21st International Conference on Machine Learning*, 114–121.

Zhang, Q., and Goldman, S. A. 2001. EM-DD: An improved multiple-instance learning technique. In *Advances in Neural Information Processing Systems 14*. 1073–1080.

Zhang, D.; Wang, F.; Si, L.; and Li, T. 2009. M3IC: Maximum margin multiple instance clustering. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 1339–1344.

Zhou, Z.-H., and Xu, J.-M. 2007. On the relation between multi-instance learning and semi-supervised learning. In *Proceedings of the 24th International Conference on Machine learning*, 1167–1174.

Zhou, Z.-H.; Sun, Y.-Y.; and Li, Y.-F. 2009. Multi-instance learning by treating instances as non-i.i.d. samples. In *Proceedings of the 26th International Conference on Machine Learning*, 1249–1256.

Zhu, X.; Ghahramani, Z.; Lafferty, J.; et al. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning*, 912–919.