24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Robust Multi-Instance Learning with Stable Instances

Weijia Zhang[1] and **Lin Liu**[1] and **Jiuyong Li**[1]

Abstract. Multi-instance learning (MIL) deals with tasks where data is represented by a set of bags and each bag is described by a set of instances. Unlike standard supervised learning, only the bag labels are observed whereas the label for each instance is not available to the learner. Previous MIL studies typically follow the i.i.d. assumption, that the training and test samples are independently drawn from the same distribution. However, such assumption is often violated in real-world applications. Efforts have been made towards addressing distribution changes by importance weighting the training data with the density ratio between the training and test samples. Unfortunately, models often need to be trained without seeing the test distributions. In this paper we propose possibly the first framework for addressing distribution change in MIL without requiring access to the unlabeled test data. Our framework builds upon identifying a novel connection between MIL and the potential outcome framework in causal effect estimation. Experimental results on synthetic distribution change datasets, real-world datasets with synthetic distribution biases and real distributional biased image classification datasets validate the effectiveness of our approach.

1 Introduction

Multi-instance learning (MIL) [5] deals with tasks where the data is consisted of a set of bags and each bag contains a set of instances. Originally, MIL was proposed for drug activity prediction, where the goal is to predict whether a new molecule is qualified to make drug. Each molecule can have many different low-energy shapes; however, biochemists only know whether it is qualified for drug making at the molecule level without knowing which specific shape of the molecule is qualified. To solve this problem, [5] proposed the multi-instance learning framework where each molecule should be modeled as a bag, and the low-energy shapes of the molecule consist its instances.

Unlike traditional supervised learning where each sample is associated with a label, in MIL only the bag labels are available whereas instance labels are unknown. The relationships between the instance labels and the bag labels are defined by multi-instance assumptions. In this paper, we focus on the widely used standard multi-instance assumption [7]: a bag is labeled as positive if it contains at least one positive instance, and labeled as negative if otherwise. The main goal of the majority of MIL algorithms is to predict the labels of unseen test bags. Drug activity prediction asside, the problem of MIL arises naturally in many applications where label is difficult to obtain, including text categorization [2], web index page recommendation [27], compute-aided diagnoses, and image classification [4].

Most multi-instance learning methods assume that the training and test data are drawn independently from an identical distribution. However, such assumption is frequently violated in real-world tasks [19]. Distribution change happens due to multiple reasons, i.e., when the training and test data are collected during different times or from different locations. Consider the example of an image classification task in Figure 1 where the task is to train a classifier for dogs. The training images are collected during summer where the backgrounds often contain grass; however, the test samples are collected during winter where the backgrounds are mostly snowy [25]. Without accommodating the distribution change, standard supervised methods have a tendency to predict images with grass as positive and images with dogs in snow as negative because of the distribution difference between the training and test samples [9].

Addressing the discrepancy between training and test distributions in standard supervised learning has attracted much attention during the past decades. Many algorithms have been proposed to solve the problem [3, 20]. Among these approaches the covariate shift setting, where the marginal distribution of samples changes but the conditional distribution of the class label conditioning on the samples do not change, has attracted the most attention [23, 21]. Unfortunately, most existing studies focused on single-instance setting and it has been shown that single-instance techniques for handling distribution change are not effective in multi-instance learning [25].

Several MIL methods have been proposed to address distribution change under the covariate shift assumption by utilizing the unlabeled test data to estimate the importance weights between the test and training samples, and incorporating separate weights address the bag-level and instance-level distribution change in MIL [25, 26]. Unfortunately, in many real-world scenarios classifiers often need to be trained without giving access to the test samples, which renders the existing distribution change MIL methods inapplicable.

In this paper, we tackle distribution change in MIL by grouping instances into three categories: causal instances (i.e., dog in Figure 1), noisy instances (grass) and negative instances (other background instances). We assume that the probability of the bag label conditioning on the causal instances remains unchanged across the training and test data. Different from the covariate shift assumption, both the probabilities of the bag label conditioning on the noisy and negative instances, and the marginal distributions of all three types of instances can change under our assumption.

By considering adding an instance to a bag as a treatment, we propose that causal instances can be distinguished from noisy and negative instances in MIL by estimating the causal effect of an instance on the bag label under the potential outcome framework [13]. Under the standard MIL assumption [7], adding a causal instance to a negative bag would change the label of the bag; however, the bag label would remain unchanged if the instance is noisy or negative. In other words, causal instances have greater estimated treatment effects than noisy and negative instances. Since causal instances are less affected by the distribution shift between training and test sets, using only on the causal instances will increase the robustness of a classifier.

[1] University of South Australia, Australia, email: {weijia.zhang, lin.liu, jiuyong.li}@unisa.edu.au

Inspired by this motivation, we present possibly the first work, coined the StableMIL framework, to address distribution change in MIL without requiring access to the unlabeled test data. We conduct experiments to evaluate StableMIL on synthetic datasets, real-world text and image classification tasks with synthetic distribution biases, and real-world biased image classification dataset. Results have shown that without accessing the test distribution, StableMIL significantly outperforms state-of-the-art algorithms and performs similarly to MIL distribution change methods that have access to the test data.

The rest of this paper is organized as follows. We review related work in Section 2, and present the proposed StableMIL framework in Section 3. Then we report the experimental results in Section 4 and we conclude the paper in Section 5.

2 Related work

Multi-instance learning was first proposed for drug activity prediction [5]. Since then, many algorithms have been proposed to solve the MIL problem, which can be roughly divided into two categories: one group of methods aim to directly solve the MIL problem in either instance level [2] or bag level [28], and another category of methods transform MIL into single instance learning via bag embedding, among which MILES [4] is a typical representative, and many methods [8, 22] have been proposed thereafter following this paradigm.

Distribution change has been studied extensively in single-instance learning, among which the covariate shift setting [19, 11] has attracted the most attention. In covariate shift, the marginal training distribution $P_{tr}(X)$ is different from the marginal test distribution $P_{te}(X)$, but the conditional distribution $P(y|X)$ remains stable across training and test samples. A common approach for addressing covariate shift is importance weighting, which assigns each training instance x with a weight $w(x) = P_{te}(X)/P_{tr}(X)$ to diminish the discrepancy of training and test marginals [20].

Distribution robust supervised learning (DRSL) has been attracting research interest during the last few years [17, 10]. These methods minimize the empirical risk with regard to the worst case test distribution within a specific uncertainty set considered by the algorithm. Unfortunately, none of them considers the setting of multi-instance learning, and directly applying single instance distribution change methods to multi-instance learning is difficult with studies shown that trivial extensions do not improve the performances [26]. Recently, several works have proposed that building a classifier based on causal relationships lead to more stable predictions than classifiers purely based on correlations in single-instance learning [14].

Several MIL algorithms have been proposed to address distribution change by utilizing the test distribution to calculate importance weight. MICS [25] solves the problem by modeling the bag-level and instance-level distribution changes separately and integrate the weights for training. MIKI [26] proposes that the distributions of positive instances should be modeled explicitly to address shifts of the positive concepts.

Several algorithms tackle the multi-instance classification problem by directly identifying the positive instances in the multi-instance bags [18, 12]. StableMIL can also be used to identify the positive instance; however, StableMIL is fundamentally different from previous algorithms because existing methods do not differentiate causality from association, and thus will suffer performance degeneration when the training and test distributions change.

To the best of our knowledge, there is no existing work which considers the link between MIL and causal learning, and to solve the distribution change problem under multi-instance learning setting without requiring the access of unlabeled test samples during the training process.

3 Stable Multi-instance Learning Framework

3.1 Notations

Let $\mathcal{X} = \mathbb{R}^d$ denote the instance space and $\mathcal{Y} = \{0, 1\}$ denote the label space. The learner is given a data set with m training bags $\mathcal{B}_{tr} = \{(X_1, y_1), \cdots, (X_i, y_i), \cdots, (X_m, y_m)\}$, where $X_i = \{\boldsymbol{x}_{i1}, \cdots, \boldsymbol{x}_{ij}, \cdots, \boldsymbol{x}_{x_{in_i}}\}$ is a bag of instances. For the simplicity of notation, we assume that the bags contain the same number of instances, i.e., $n_i = n$ for all i. Given training data $\mathcal{D}$, we denote the set containing all positive (negative) bags by $\mathcal{B}^+$ ($\mathcal{B}^-$). The number of positive (negative) bags is denoted by m^+ (m^-).

We consider the widely accepted "standard multi-instance learning assumption" [7] throughout our discussion. Formally, the standard MIL assumption can be described as:

Assumption 1 *A multi-instance bag is negative if and only if all of its instances are negative, and a bag is positive if at least one of its instances are positive: i.e.,*

$$\phi(X_i) \triangleq f(h(\boldsymbol{x}_{i1}), h(\boldsymbol{x}_{i1}), \cdots, h(\boldsymbol{x}_{in})), \tag{1}$$

where ϕ is the bag labeling hypothesis, f is the boolean OR function, and $h \in \mathcal{H}$ is a hypothesis for instances in $\mathcal{X}$.

With

3.2 The framework

The heart of this work is the claim that there exists a connection between multi-instance learning and randomized experiments in the causal inference literature [13]. Generally speaking, an experiment can be used for determining whether a causal relationship exists between a binary *treatment* variable and an *outcome* variable. In other words, whether the action of changing the value of the treatment would affect the value of the outcome.

With the standard multi-instance assumption, let us consider *adding instance $\boldsymbol{x}$ to a bag X_j* as the action of treatment and the label Y of the bag as the outcome, then the causal relationship between instance $\boldsymbol{x}$ and label Y can be determined by whether the treatment changes the label from 0 to 1. In other words, if instance $\boldsymbol{x}$ has causal relationship with Y, adding it to a negative bag would flip the label from negative to positive. On the other hand, if instance $\boldsymbol{x}$ is not causally related to Y, adding $\boldsymbol{x}$ will not change the label. Formally we state the definition of *causal instance* as:

Definition 2 *(**Causal instance**) An instance $\boldsymbol{x}_{ik}$ from a positive bag X_i^+ is a causal instance with regard to the bag label Y if for any negative bag $X_j \in \mathcal{B}^-$, it satisfies*

$$\phi^*(\boldsymbol{x}_{ik} \cup X_j) = 1, \tag{2}$$

where ϕ^* denotes an oracle bag classifier which always return the correct label of a multi-instance bag, and $\boldsymbol{x}_{ik} \cup X_j$ denotes a treated bag containing the instance of interest $\boldsymbol{x}_{ik}$ along with all the instances in the pre-treatment bag X_j.

With Definition 2, we can now group the instances in a multi-instance bag into three categories: for the first category, the conditional expectation of the label has a non-zero dependence on the instances and the dependence does not change when other instances

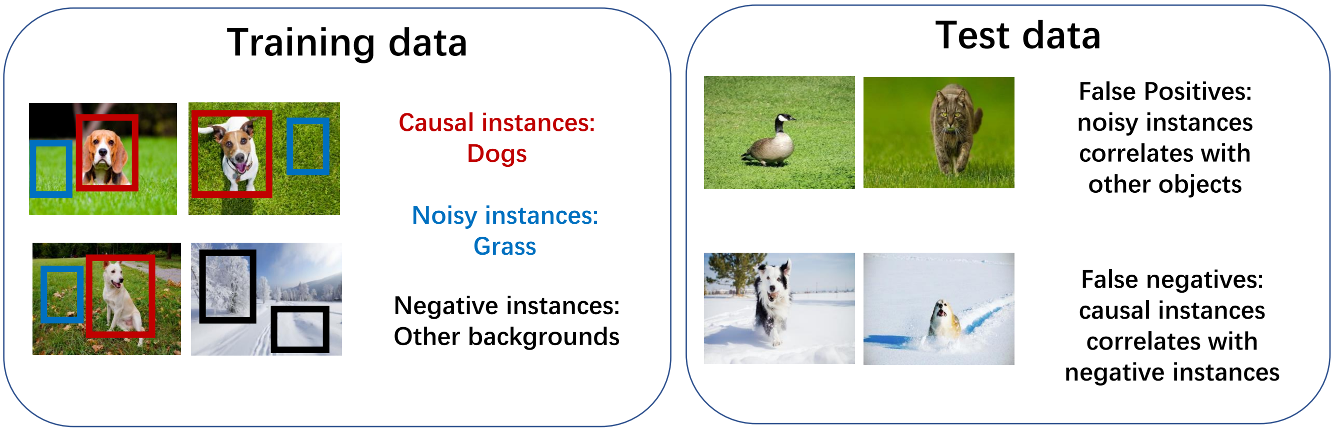


Figure 1: (Best viewed in colour) Causal instances (red) are dogs. In training data, noisy instances (blue) are grass and are highly correlated with the causal instances; negative instances (black) are other random backgrounds which are negatively correlated with the causal instances. In test data, noisy instances correlate with other objects such as cats and produce false positives, causal instances correlate with negative instances and produce false negatives.

are added to the conditional set, we call these *causal instances*. For example, dogs are causal instances of photos labeled as animals and the relationship will not change between the training and the test distributions (Figure 1). For the second category of instances, we term them as *noisy instances*. Noisy instances are correlated with either the causal instances, the bag label, or both, but do not themselves have causal relationships with the label. When conditioned on the full set of causal instances, noisy instances are independent of the bag label. For example, grasses and snows are noisy instances of an animal photo. Although they are highly correlated with the causal instances and the label, their correlations is vulnerable to changes during the collection of the training and test data. The third category of instances are termed similar as in existing literature as *negative instances*, which contains instances that are not significantly correlated to the label, i.e., random background objects without any significant correlation to the bag label.

The performances of existing MIL methods degenerate when distribution changes because they do not differentiate causal instances from noisy instances. When the training and test distributions differ, the correlations between noisy instances and the label will not be consistent. Therefore, models built by existing methods will be mislead by spurious correlations that only exist in the training data but are not valid in the test data. On the other hand, a multi-instance classifier based on causal instances will achieve a more stable performance because causal relationships are not affected by the distribution changes in the training and test distributions.

3.2.1 Learning Causal Instances from Experiment

The key of building a distributional robust multi-instance classifier lies in differentiating causal instances from noisy ones. In this section, we will discuss the identification of causal instances from an ideal experiment setting.

By Definition 2, instances from negative bags cannot have causal relationships with the label. Therefore, we only need to consider the instances from the positive bags. Let $\bigcup \mathcal{B}^+$ denote the candidate instance pool which contains all the instances from the positive bags and $\boldsymbol{x}_k \in \bigcup \mathcal{B}^+$ denote a candidate instance. For the sake of conciseness, subscript k will be dropped when the context is clear. To determine whether $\boldsymbol{x}$ is a causal instance, we need to estimate the causal effect of $\boldsymbol{x}$ on the bag label Y which can be defined as the difference between the expected label of a bag if it was treated minus the expected label of the bag if it were not treated:

$$\tau(\boldsymbol{x}) = \mathbb{E}[Y(T=1)] - \mathbb{E}[Y(T=0)]. \tag{3}$$

Here we use $Y(T=1)$ to denote the potential bag label if it were treated, i.e., the candidate instance $\boldsymbol{x}$ is present in the bag; and we use $Y(T=0)$ to denote the potential bag label if the bag were not treated, i.e., $\boldsymbol{x}$ is not present in the bag.

Given a set of multi-instance bags, we can always obtain matched pairs of treated and untreated bags by adding the candidate instance $\boldsymbol{x}$ to a bag (if $\boldsymbol{x}$ is not in the pre-treatment bag) or removing $\boldsymbol{x}$ from a bag (if $\boldsymbol{x}$ is in the pre-treatment bag). Therefore, the causal effect can be estimated using the difference in the expectation of the realized outcomes provided by the data and the oracle classifier:

$$\tau(\boldsymbol{x}) = \mathbb{E}[Y^*|T=1] - \mathbb{E}[Y^*|T=0], \tag{4}$$

where Y^* denotes the bag label after the treatment. Combining with the standard multi-instance assumption, we can obtain the following theorem:

Theorem 1 *The causal effect of an instance $\boldsymbol{x}$ on the bag label Y obtained from an ideal experiment is*

$$\tau(\boldsymbol{x}) = P(Y=0) \cdot \mathbb{E}[Y^*|Y=0, T=1] + const, \tag{5}$$

where const is the probability of that a bag contains one and only one positive instance with the instance being $\boldsymbol{x}$.

Proof: Utilizing the tower property of conditional expectation, we have

$$\begin{aligned}\tau(\boldsymbol{x}) &= \mathbb{E}[Y^*|T=1] - \mathbb{E}[Y^*|T=0] \\ &= \mathbb{E}[\mathbb{E}(Y^*|Y, T=1)] - \mathbb{E}[\mathbb{E}(Y^*|Y, T=0)] \\ &= \sum_{i=0}^{1} \mathbb{E}[Y^*|Y=i, T=1] P(Y=i) \\ &- \sum_{i=0}^{1} \mathbb{E}[Y^*|Y=i, T=0] P(Y=i).\end{aligned}$$

From the Standard MI assumption, adding any instance to a positive bag or removing any instance from a negative bag will not change the bag label. Therefore we have, $\mathbb{E}[Y^*|Y=1,T=1]=1$ and $\mathbb{E}[Y^*|Y=0,T=0]=0$. Accordingly,

$$\begin{aligned}\tau(\boldsymbol{x}) &= \mathbb{E}[Y^*|Y=0,T=1]\cdot P(Y=0)\\ &\quad - \mathbb{E}[Y^*|Y=1,T=0]\cdot P(Y=1) + P(Y=1).\end{aligned}$$

There exists two possibilities when a positive bag is under the control treatment: the pre-treatment bag contains positive instances other than $\boldsymbol{x}$ (may and may not contains $\boldsymbol{x}$ itself), the bag contains $\boldsymbol{x}$ and only $\boldsymbol{x}$ as its positive instance. Under the standard multi-instance assumption, the expectation is 1 and 0 for the two scenarios, respectively. Let p denote the probability of the second scenario, we can write $\mathbb{E}[Y^*|Y=1,T=0]=1-p$. Therefore, we have

$$\tau(\boldsymbol{x}) = \mathbb{E}[Y^*|Y=0,T=1]\cdot P(Y=0) + p\cdot P(Y=1)$$

□

Theorem 1 indicates that the causal effect of an instance can be characterized by the expected treated bag label after adding the instance to a negative bag. It is safe to ignore the constant term during the estimation because p would be small for causal instances and $p=0$ for non-causal instances. Additionally, if we assume the multi-instance bags are distributions over instances, we have $p=0$ for all instances since any $\boldsymbol{x}$ almost surely does not appear in any particular positive bag [6].

3.2.2 Learning Stable Instances from Data

Our proposed framework for stable multi-instance learning is inspired by the above discussion of learning the causal instances. However, since the oracle classifier does not exist, the instances identified using a surrogate empirical classifier may not be causal. We hence refer to them as stable instances to avoid misconceptions.

To identify the stable instances, first we train a multi-instance classification algorithm $\mathcal{A}$ with the training data and use A to denote the classifier that $\mathcal{A}$ returns. For each candidate instance $\boldsymbol{x}$, we then construct a set of bags which contains m^- number of "treated" bags from the original negative bags. The treated bags are constructed by adding the candidate instance $\boldsymbol{x}$ into the negative bags X_i^- as $X_i^{\boldsymbol{x}} = \boldsymbol{x} \bigcup X_i^-$, for $i=1,\cdots,m^-$. For each treated bag, we use the previously trained classifier A to predict its label. Finally we use the average of predicted labels $A(X_i^{\boldsymbol{x}})$ on the treated bags to estimate the expectation term in Equation 5 as

$$\hat{\tau}(\boldsymbol{x}) = \frac{1}{m^-}\sum_{i=1}^{m^-} A(X_i^{\boldsymbol{x}}). \tag{6}$$

After Equation 6 has been estimated for all the candidate instances, we choose the instances with a score s higher than τ as stable instances. Note that here we only select the stable instances to be included in $\mathcal{C}$ and we exclude all negative instances. Although it has been shown that in i.i.d. MIL negative instances can be helpful to the learner [8], in distribution change negative instances can still be misleading since the causal instances may correlate with negative instances in the test data. We summarize the procedure of StableMIL for learning the stable instances in Algorithm 1.

Even when considered from a non-causal perspective, StableMIL is more robust to distribution change than standard MIL methods. Let us assume that the training bags in Figure 1 consist 40% of images with dog in grass background, 10% of images with dog in snow background, 10% of grass images, 10% of snow images, and the rest are other generic negative images. In the procedure of Algorithm 1, grass and snow instances will have a lower chance of flipping the label when added to negative bags than the dog instances: dog instances exist in 100% of positive bags; however, grass instances only exist in 80% of the positive bags but also exist in 20% of the negative bags, and snow instances only exist in 20% of the positive images but also exist in 20% of the negative ones. Therefore, stable instances can be differentiated from noisy instances.

Algorithm 1 Stable Multi Instance Learning

```
Require: Training bags B_tr = {(X_i^tr, y_i)}_{i=1}^m, multi-instance algorithm A, threshold τ
Ensure: Stable Instance Pool C
Train A using data in B_tr
for every x ∈ ∪B^+ do
    s ← 0
    for every X_i in X^- do
        s = s + A(x ∪ X_i^-)
    end for
    s ← s/m^-
    if s ≥ τ then
        C = x ∪ C
    end if
end for
for every X ∈ B_tr do
    for every x ∈ C do
        d(X_i, x) = max_{x_ij} exp(−λ||x_ij − x||^2)
    end for
    z_i = [d(X_i, x_1), ··· , d(X_i, x_q)], x_1, ··· , x_q ∈ C
end for
Train classifier L with the embedded feature vectors z_i
return L
```

3.2.3 Bag Embedding and Embedded Classification

After learning the stable instance set C, we used bag embedding [4, 8, 26] to map the bags into single instance representation using the stable instances, and then construct our classifier.

The multi-instance bag embedding is performed based on the similarity between the bags and each of the instances in $\mathcal{C}$. The similarity between a bag and an instance can be measured using the following function:

$$d(X_i,\boldsymbol{x}) = \max_{\boldsymbol{x}_{ij}\in X_i} \exp(-\lambda\|\boldsymbol{x}_{ij}-\boldsymbol{x}\|^2) \tag{7}$$

where λ is the scaling parameter and can be chosen automatically by local scaling [24]. The intuition is that positive bags should have high similarities with at least one instance in $\mathcal{C}$, and negative bags should have low similarities with all instances in $\mathcal{C}$.

As a result, the embedded feature vector for bag X_i is a q-dimensional vector which is consisted of the concatenated bag-to-instance similarities:

$$\boldsymbol{z}_i = [d(X_i,\boldsymbol{x}_1),\cdots,d(X_i,\boldsymbol{x}_j),\cdots,d(X_i,\boldsymbol{x}_q)], \tag{8}$$

where $\boldsymbol{x}_j \in \mathcal{C}$ and q is the cardinality of set $\mathcal{C}$. After the bag embeddings are calculated, any standard machine learning algorithm can be used for training the model. In our experiments, we use SVM with RBF kernel for fairness of comparison since it is also used as the classifier for many multi-instance algorithms based on the idea of bag embedding.

Table 1: Testing accuracy (%, mean ± std.) on synthetic datasets. The highest average accuracy is marked in bold.

	State-of-the-art multi-instance learning methods				Distribution change multi-instance methods		Proposed method
	miSVM	MILES	miGraph	miFV	MICS	MIKI	StableMIL
Setting 1	79.8 ± 2.0	83.6 ± 1.6	90.7 ± 3.9	76.4 ± 3.8	91.4 ± 4.9	87.3 ± 2.5	**93.9 ± 2.3**
Setting 2	72.7 ± 2.5	71.0 ± 2.8	70.7 ± 3.0	68.3 ± 3.6	70.5 ± 5.9	72.9 ± 2.3	**78.9 ± 3.5**

3.2.4 Time Complexity Analysis

StableMIL is scalable because the identification of stable instances only requires predicting the label of the constructed treated bags, while the training of bag-level classifier only uses the original multi-instance bags. We include a detailed complexity analysis as follows. Without loss of generality and for the simplicity of analysis, we assume that each bag has the same number of n instances and the number of positive/negative bags are balanced. In the training phase, StableMIL first trains a multi-instance bag classifier A using the original training bags. Suppose that we use miFV as the bag classifier [22], the complexity of $O(nmd)$ plus the cost of training a linear SVM. Then we use the trained miFV model to predict the label of the constructed bags, for which the complexity is $O(n^2md)$. Note that here the quadratic term only depends on the number of instances n which is usually significantly smaller than the number of bags m, and the predictions can be easily parallelized. Afterwards, q stable instances are selected and the bag-mapping is performed with regard to the stable instances with complexity of $O(qnmd)$. Now the remaining computation for training StableMIL is training a SVM using n sample of q dimensional feature vectors. To sum up, the time complexity of StableMIL is $O(n^2md)$ plus training cost of two SVMs.

4 Empirical validation

In this section, we empirically validate the performance of StableMIL. Firstly, we compare StableMIL with benchmark multi-instance algorithms including miSVM [2], MILES [4], miGraph [28], and miFV [22]. Secondly, we compare with distribution change MIL methods that utilize the test distribution, including MICS [25] and MIKI [26]. Hyper parameters for the compared methods are selected either as the default recommended by the original authors (for MIKI) or using cross-validated grid search on the training samples (for other algorithms).

We use miFV as the base classifier of StableMIL and SVM with Gaussian kernel for the classification after instance embedding. Although using instance level classifiers is more theoretical consistent with the standard MI assumption. However, since it has been shown that instance-level MIL methods generally perform significantly worse than bag-level methods [1], we choose to use a state-of-the-art bag-level classifier miFV because we want the base classifier to be as accurate as possible.

We use SVM with Gaussian kernel as the classifier of StableMIL after the multi-instance bags has been transformed to a single instance after stable instance embedding. A reason for choosing Gaussian SVM is because it is also used in the compared methods (except miFV which uses linear SVM). Thus, it can be seen that the improvement is due to identification of the causal instances by StableMIL. We tested using linear kernel instead of Gaussian, the trend remains similar, but the accuracies are slightly lower than Gaussian for all compared algorithms.

For choosing the stable instances threshold τ of StableMIL, we first split the negative bags in the training data into two equally-sized parts. Then we construct a set of treated bags by adding the instances from the first part to the bags from the second part of the negative training bags. After estimating the scores s_{neg} for the constructed bags, we use the third quartile value of s_{neg} for parameter τ. Other parameters for StableMIL are selected as same as those of miFV.

4.1 Synthetic Data

We first evaluate StableMIL with synthetic data. The instances in the multi-instance bags are generated by sampling from four distinct multivariate Gaussian distributions: the first one for the causal instances, the second one for the noisy instance, the third one and the fourth one for the negative instances.

For example, let us suppose the task is to classify dog and the four distributions describe "dog", "grass", "snow" and "cat", respectively. In the training data most of the positive bags describe "dog on grass" with only a small number of bags describe "dog on snow"; for the negative bags, most of them describe "snow" and the rest are "grass" or "cat". In the test data, the bias is reversed: most positive bags are "dog on snow" while the majority of negative bags are "grass" or "cat on grass".

More specifically, the distribution changes between the training and test datasets are generated using a biased sampling procedure similar to those used in relevant literature [23, 25, 26]. We define a selection variable s_i, where $s_i = 1$ indicates the i-th bag is selected into the training set and $s_i = 0$ indicates the bag falls into the test set. Denote the positive and negative instances as $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$, then for positive bags the sampling rules are

$$Pr(s_i = 1|\boldsymbol{x}^+ \in \mathcal{P}, \boldsymbol{x}^- \in \mathcal{N}_1) = a.$$
$$Pr(s_i = 1|\boldsymbol{x}^+ \in \mathcal{P}, \boldsymbol{x}^- \in \mathcal{N}_2) = 1 - a.$$

For negative bags the rules are

$$Pr(s_i = 1|\boldsymbol{x}^- \in \mathcal{N}_1) = 1 - a$$
$$Pr(s_i = 1|\boldsymbol{x}^- \in \mathcal{N}_j) = a. \text{ for } j = 2, 3$$

Here $\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{N}_1$ and $\mathcal{N}_2$ are multivariate Gaussian distributions where the positive and negative instances are sampled from. The values of sampling ratio a are uniformly selected from $a \in [0.65, 0.95]$ at each simulation. Two different settings of simulated datasets are generated and they differ in the dimensionality of the instances. In Setting 1 each instance is a 3-dimensional feature vector while the instances in Setting 2 are 100-dimensional feature vectors.

We repeated experiments for 30 times by using the sampling procedure to generate training and test sets and report the simulation results in Table 1. Firtly, the proposed StableMIL method achieves significantly better performance than the compared i.i.d. MIL methods. Secondly, when compared with distribution change based MIL algorithms which has access to the unlabeled test samples, StableMIL also achieves better performance without seeing the samples.

4.2 MNIST Dataset

Next we evaluate StableMIL for distribution change in image classification using the multi-instance MNIST dataset. We generate 200

Table 2: Testing accuracy (%, mean ± std.) on MNIST. The highest average accuracy is marked in bold. •/◦ indicates that StableMIL is significantly better/worse than the compared methods (paired *t*-tests at 95% significance level). The last row summarizes the Win/Tie/Lose counts of StableMIL versus other methods.

	State-of-the-art multi-instance learning methods				Distribution change multi-instance methods		Proposed method
	miSVM	MILES	miGraph	miFV	MICS	MIKI	StableMIL
6 and 9	59.3 ± 10.3•	77.7 ± 7.6•	78.5 ± 10.8•	70.8 ± 12.7•	80.0 ± 10.2•	**92.8 ± 7.9**	91.9 ± 7.1
0 and 6	54.9 ± 7.1•	79.2 ± 10.6•	75.5 ± 11.5•	75.0 ± 12.8•	75.6 ± 11.3•	**94.6 ± 4.3**◦	93.3 ± 4.6
0 and 8	53.6 ± 9.8•	72.8 ± 9.5•	74.3 ± 10.9•	70.0 ± 15.0•	75.4 ± 11.2•	**94.7 ± 4.3**◦	91.8 ± 8.8
0 and 9	56.9 ± 10.4•	71.9 ± 10.0•	74.6 ± 11.0•	70.5 ± 10.8•	74.0 ± 10.3•	95.4 ± 2.5	**95.8 ± 3.3**
4 and 7	61.4 ± 13.0•	68.9 ± 10.5•	74.5 ± 11.3•	71.7 ± 13.9•	74.6 ± 10.7•	**94.8 ± 7.7**	94.6 ± 6.2
1 and 7	58.4 ± 11.2•	90.5 ± 7.1•	77.5 ± 11.3•	76.0 ± 10.8•	70.0 ± 11.6•	92.4 ± 5.7•	**95.4 ± 5.0**
2 and 7	57.9 ± 11.3•	59.3 ± 5.9•	75.2 ± 12.0•	70.4 ± 13.8•	78.6 ± 10.2•	**92.1 ± 10.4**	90.1 ± 5.6
3 and 6	57.6 ± 8.9•	61.6 ± 8.1•	77.6 ± 12.2•	68.8 ± 10.9•	76.1± 10.2•	**95.0 ± 3.0**◦	92.6 ± 7.9
6 and 8	57.9 ± 11.3•	76.0 ± 7.7•	76.3 ± 11.3•	71.0 ± 14.3•	74.1 ± 10.5•	**92.7 ± 5.9**	92.1 ± 5.6
2 and 4	58.5 ± 8.9•	60.0 ± 6.6•	72.4 ± 10.3•	71.7 ± 12.2•	74.1 ± 11.8•	**94.7 ± 3.4**◦	92.0 ± 6.5
W/T/L	10/0/0	10/0/0	10/0/0	10/0/0	10/0/0	1/5/4	

positive and 200 negative multi-instance bags where each bag contains an average of 20 instances (digits). The feature for each instance is extracted using LeNet-5 [15]. A bag is labeled as positive if it contains a specific digit (i.e., "1"), we use a visually similar digit as an analogy of the "cat" negative concept (i.e., "1" and "7"), and separate the rest of the digits into two groups as the other negative concepts. The biased sampling procedure is performed similarly to the last section. The reported results are averaged over repeating the biased sampling procedure for 30 times.

We report the results Table 2. When the distribution bias between training and test samples exists, the performance of StableMIL is significantly better on all datasets when compared with state-of-the-art MIL algorithms. Moreover, it can be seen that the standard deviation of the accuracies in StableMIL is much lower than the compared methods. This indicates that StableMIL is less susceptible to the variations between the training and test distributions.

If we consider the fact that MILES is a bag embedding based MIL algorithm using the union of the instances in positive bags and StableMIL performs bag embedding using only the stable instances, the superior performance of StableMIL over MILES indicates that the stable instances selected by StableMIL are indeed useful for stable prediction across unknown test distributions.

When compared to distribution change methods, the performance of StableMIL is similar to MIKI (Wilcoxon rank-sum test at $p = 0.05$ indicates the performances of StableMIL and MIKI are not statistically significant); however, MIKI needs to access the test distribution whereas StableMIL achieves competitive accuracy using only the training data.

4.3 20 Newsgroups Dataset

To evaluate the performance of StableMIL for text categorization tasks, we utilize the *20 Newsgroups corpus* [28]. It contains paragraphs belonging to 20 different news topics. We use each paragraph as an instance and construct each multi-instance bag with an average of 20 instances. The feature for each instance is represented by the top 200 TF-IDF features. A bag is labeled as positive if it contains a specific topic (i.e., "graphics"), we use a semantically similar topic as an analogy of the "cat" negative concept (i.e., "graphics" and "operating system"), and separate the rest of the topics into two groups to use them as the other negative concepts. Again, the biased sampling procedure is performed similarly to the last two experiments. The reported results are averaged over repeating the biased sampling procedure for 30 times.

The results on the 20 Newsgroup corpus text categorization task are shown in Table 3. StableMIL performs significantly better than all compared state-of-the-art MIL algorithms. When compared to distribution change MIL algorithms, the results are similar to those on the MNIST dataset: the performance differences of StableMIL and MIKI are not significant (Wilcoxon rank-sum test $p = 0.05$).

4.4 Real-world Distribution Biased Dataset

Recently, a large-scale benchmark dataset NICO, designed for benchmarking distribution-biased image classification has been released on ArXiv [9]. Using this data, we evaluate StableMIL on two real-world biased image classification tasks.

The first task is to distinguish dog vs non-dog images where the training and test samples are biased because of the different seasons during which the photos were taken. The training set contains mostly dog images taken in winter with snowy background, and a small portion of dog images taken in summer with grassy background. In the test samples the bias is reversed with significantly more dog on grass images than dog on snow images. Additionally, the negative images in the test samples also contain more images with grass background than images with snow background, and some images of other animals on grass. The second task is similar to the first one but the training and test sets are biased due to the different locations where the photos are taken. Specifically, the training set contains mostly dog images in urban environment, including in cage, at home and in street, etc., and less dog images in natural, including dogs in river, grass, and snow, etc. The test samples contain more images of dogs in nature environment and less dogs in urban images.

To extract multi-instance bags from the images, we use a Mask-RCNN pre-trained on Microsoft COCO dataset [16] to generate the bounding-boxes and the region in each bounding-box is extracted as an instance. Additionally, the backgrounds are also extracted as instances. The feature vectors for the instances are extracted using an unsupervised convolutional auto-encoder.

The results (Table 4) show that the performances of StableMIL and MIKI are comparable. For other compared methods, StableMIL performs significantly better. Although MIKI performs better than StableMIL on these datasets, the critical drawback of MIKI is that it requires accessing the test samples during training whereas Sta-

Table 3: Testing accuracy (%, mean ± std.) on 20 Newsgroup. The highest average accuracy is marked in bold. •/○ indicates that StableMIL is significantly better/worse than the compared methods (paired *t*-tests at 95% significance level). The last row summarizes the W/T/L counts.

	State-of-the-art multi-instance learning methods				Distribution change multi-instance methods		Proposed method
	miSVM	MILES	miGraph	miFV	MICS	MIKI	StableMIL
gra.os	56.2 ± 4.9•	61.2 ± 4.2•	68.4 ± 2.9•	61.6 ± 6.6•	70.3 ± 5.3•	68.5 ± 6.7•	**71.4 ± 3.5**
gra.ibm	57.8 ± 3.7•	57.7 ± 4.2•	64.0 ± 3.3•	57.8 ± 4.7•	62.7 ± 4.0•	65.7 ± 5.3•	**68.6 ± 4.0**
mac.win	56.6 ± 5.0•	61.1 ± 3.8•	67.5 ± 3.5•	67.7 ± 6.5•	68.0 ± 3.0•	69.7 ± 4.8•	**77.2 ± 2.3**
os.mac	54.7 ± 4.9•	59.7 ± 3.2•	61.7 ± 3.5•	58.0 ± 4.8•	62.8 ± 3.6•	**65.8 ± 4.3**○	64.8 ± 3.7
os.win	63.8 ± 6.0•	64.6 ± 6.3•	**74.7 ± 3.2**○	61.8 ± 7.3•	70.2 ± 2.8•	72.2 ± 5.6	72.0 ± 3.2
auto.baseball	54.8 ± 4.6•	60.8 ± 4.2•	55.3 ± 4.5•	59.6 ± 4.0•	58.5 ± 4.9•	**64.0 ± 4.5**○	62.8 ± 4.0
auto.moto	54.5 ± 5.0•	65.5 ± 7.5•	59.5 ± 3.4•	61.7 ± 5.2•	60.4 ± 4.2•	65.6 ± 6.0•	**68.3 ± 3.6**
baseball.hockey	69.2 ± 8.0•	69.6 ± 3.6•	74.1 ± 3.7	72.2 ± 6.1•	73.2 ± 3.6	**75.4 ± 5.7**○	73.6 ± 6.0
moto.baseball	54.2 ± 3.8•	56.8 ± 3.9•	50.4 ± 3.4•	56.0 ± 4.1•	55.3 ± 3.1•	58.2 ± 6.9•	**61.2 ± 2.9**
moto.hockey	54.4 ± 4.5•	59.2 ± 2.7	52.5 ± 3.2•	58.0 ± 4.6•	58.0 ± 3.2•	60.1 ± 4.7	60.0 ± 3.0
sci.crypt.elec	56.5 ± 8.6•	56.5 ± 5.2•	57.7 ± 4.8•	54.5 ± 4.0•	60.0 ± 3.0•	61.4 ± 3.5•	**63.4 ± 2.9**
crypt.med	52.0 ± 3.9•	57.1 ± 4.1•	52.0 ± 3.8•	55.3 ± 4.6•	55.6 ± 4.0•	61.8 ± 5.7•	**74.5 ± 5.7**
crypt.space	54.8 ± 6.5•	58.1 ± 2.7•	58.2 ± 1.8•	57.3 ± 5.0•	59.2 ± 4.3•	66.1 ± 5.6•	**69.4 ± 3.4**
elec.space	51.6 ± 2.9•	57.4 ± 4.2•	49.8 ± 3.6•	54.5 ± 3.7•	52.3 ± 3.0•	**61.9 ± 6.8**○	59.8 ± 4.0
med.space	52.1 ± 4.0•	57.2 ± 4.1•	49.7 ± 2.7•	54.7 ± 3.8•	54.3 ± 4.0•	58.4 ± 9.4•	**58.8 ± 6.5**
guns.mideast	58.3 ± 7.8•	54.6 ± 2.1•	59.1 ± 3.6•	58.3 ± 4.9•	60.0 ± 3.1•	60.8 ± 5.4•	**62.1 ± 3.3**
guns.misc	55.6 ± 6.8•	58.7 ± 3.5•	60.0 ± 4.2•	58.5 ± 5.8•	60.3 ± 3.9•	62.5 ± 7.1•	**64.4 ± 2.2**
mideast.misc	55.7 ± 3.2•	57.6 ± 2.9•	63.6 ± 3.2•	58.2 ± 5.3•	62.6 ± 3.7•	**66.3 ± 4.7**○	65.7 ± 2.2
mideast.religion	58.0 ± 5.8•	56.5 ± 5.1•	61.2 ± 3.8	60.0 ± 5.6•	61.6 ± 3.1	**64.4 ± 7.0**○	61.5 ± 3.8
religion.guns	53.5 ± 3.9•	56.7 ± 2.4•	60.4 ± 6.1	58.0 ± 5.6•	58.6 ± 3.6•	58.7 ± 3.5•	**60.9 ± 3.9**
W/T/L	20/0/0	19/1/0	16/3/1	20/0/0	18/2/0	12/2/6	

Table 4: Testing accuracy (%) on two real-world distributionally biased image classification tasks from the NICO dataset.

Dataset	mi-SVM	MILES	miGraph	miFV	MICS	MIKI	StableMIL
Season	71.0	76.2	75.3	74.5	76.0	**82.8**	81.9
Location	66.5	73.5	72.0	73.2	74.1	**79.8**	78.0

bleMIL does not have this requirement.

Table 5: Testing accuracy (%) on benchmark datasets. The highest average accuracy is marked in bold (paired *t*-tests at 95% significance level).

Dataset	miSVM	MILES	miGraph	miFV	MICS	MIKI	StableMIL
Musk1	87.4	84.2	**88.9**	87.5	88.0	88.2	87.6
Musk2	83.6	83.8	**90.3**	86.1	90.0	91.0	90.0
Elephant	82.0	**89.1**	86.8	82.4	86.0	85.0	84.2
Fox	58.2	**76.0**	61.1	58.9	72.7	66.5	63.8
Tiger	78.9	**86.0**	85.9	79.0	86.0	83.0	78.6

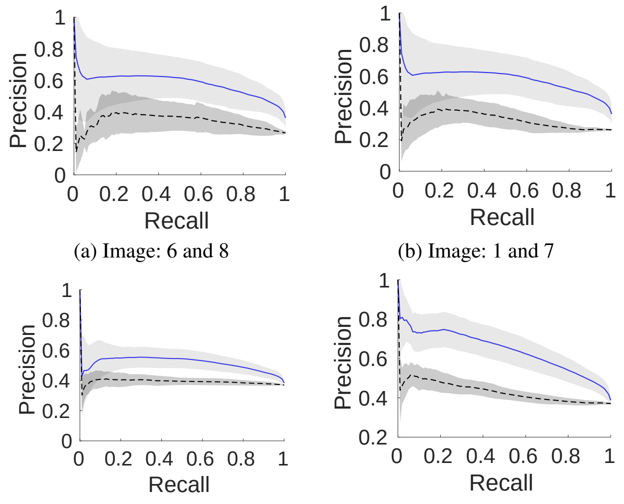
4.5 Classic MIL Benchmark Datasets

StableMIL can be used without prior knowledge of whether distribution change has occurred. We evaluate StableMIL with classic MIL benchmark non-distributional datasets using repeated 10-fold cross-validation. Each experiment has been repeated for 10 times and the average accuracy are reported in Table 5. The performance of StableMIL is similar to the compared state-of-the-art methods. The results indicate that although StableMIL is designed to produce robust predictions across different training and test distributions, it performs competitively to the state-of-the-art when there is no change between the training and test distribution. This is reasonable, since the causal instances are useful for prediction regardless of whether distribution change has occurred.

It is worth noting that StableMIL performs worse than other methods on the "Tiger" dataset. This is because in StableMIL only the causal instances are used for classification and all other instances are discarded during bag embedding. This is preferable in a distributional biased setting, since relying on the non-causal instances will mislead the classifier as their distributions change between the training and test data. However, if the training and test data are from the same distribution, including correlated and negative instances is beneficial for two reasons: firstly, it encourages the decision boundary to be positioned away from the negative instances; secondly, correlated instances can be used as "surrogates" of causal instances for finding the positive concept if the distributions do not change between the training and test samples. To the best of our understanding, this is the reason why StableMIL does not excel in the experiments of Section 4.5 where the results are obtained from repeated cross-validation and the distributions are not biased.

4.6 Causal Instances Identification

We investigate in what degree the stable instances identified by StableMIL are indeed the instances that causes the bag label using both the MNIST and 20 Newsgroup datasets. Since the total number of positive instances and negative instances are usually imbalanced, we use precision and recall instead of accuracy, and report the Precision-Recall (PR) curves for comparison. Figure 2 shows the PR-curves (additional figures are included in the supplementary material due to space limit) of StableMIL comparing with miVF [18], a MIL algorithm designed for identifying the positive instances in the bags. From the figure we see that StableMIL is effective in identifying the true positive instances, and it consistently outperforms miVF.



(a) Image: 6 and 8 (b) Image: 1 and 7

(c) Text: mac.win (d) Text: guns.mideast

Figure 2: The PR curves of causal instances identification (plotted against the stable instance threshold) of StableMIL (blue solid curves) and miVF (black dashed curves). The shaded area indicates the confidence interval.

5 Conclusion

In this paper, we identify an intrinsic connection between multi-instance learning and causal inference. Inspired by this connection, we propose a novel MIL framework towards robust classification in distributional biased data in without the requirement of accessing the test samples. Experiments show that the proposed Stable MIL framework is significantly less sensitive to the distribution changes between the training and test data than existing MIL algorithms, and it performs competitively with state-of-the-art multi-instance distribution change methods *without* the requirement of seeing the unlabeled test data during training.

We have focused on the standard multi-instance assumption and the covariate shift setting in this work. Future work includes investigating the extension of StableMIL to more relaxed distribution change settings, and exploring the link between other MIL assumptions and causal inference.

Acknowledgment This research is partially funded by Australian Research Council Discovery Project (DP170101306).

REFERENCES

[1] Jaume Amores, 'Multiple instance classification: Review, taxonomy and comparative study', *Artificial Intelligence*, **201**, 81–105, (2013).
[2] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann, 'Support vector machines for multiple-instance learning', in *Proceedings of the 15th International Conference on Neural Information Processing Systems*, pp. 577–584, (2002).
[3] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira, 'Analysis of representations for domain adaptation', in *Advances in Neural Information Processing Systems 19 (NIPS 2006)*, pp. 137–144, (2006).
[4] Yixin Chen, Jinbo Bi, and J.Z. Wang, 'MILES: Multiple-instance learning via embedded instance selection', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(12), 1931–1947, (2006).
[5] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez, 'Solving the multiple instance problem with axis-parallel rectangles', *Artificial Intelligence*, **89**(1-2), 31–71, (1997).
[6] Gary Doran and Soumya Ray, 'Multiple-instance learning from distributions', *Journal of Machine Learning Research*, **17**(128), 1–50, (2016).
[7] James Foulds and Eibe Frank, 'A review of multi-instance learning assumptions', *The Knowledge Engineering Review*, **25**(1), 1, (2010).
[8] Zhouyu Fu, A Robles-Kelly, and Jun Zhou, 'MILIS: Multiple instance learning with instance selection', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **33**(5), 958–977, (2011).
[9] Yue He, Zheyan Shen, and Peng Cui, 'NICO: A dataset towards non-i.i.d. image classification', *ArXiv*, (June 2019).
[10] Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama, 'Does distributionally robust supervised learning give robust classifiers?', in *Proceedings of the 35th International Conference on Machine Learning*, pp. 2029–2037, (2018).
[11] Jiayuan Huang, Alexander J. Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Scholkopf, 'Correcting sample selection bias by unlabeled data', in *Proceedings of the 19th International Conference on Neural Information Processing Systems*, (2006).
[12] Maximilian Ilse, Jakub Tomczak, and Max Welling, 'Attention-based deep multiple instance learning', in *Proceedings of the 35th International Conference on Machine Learning*, pp. 2127–2136, (2018).
[13] Guido W. Imbens and Donald B. Rubin, *Causal Inference for Statistics, Social, and Biomedical Sciences*, Cambridge University Press, 2015.
[14] Kun Kuang, Peng Cui, Susan Athey, Ruoxuan Xiong, and Bo Li, 'Stable prediction across unknown environments', in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1617–1626, (2018).
[15] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE*, **86**(11), 2278–2324, (1998).
[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick, 'Microsoft COCO: Common objects in context', in *Computer Vision – ECCV 2014*, 740–755, Springer International Publishing, (2014).
[17] Anqi Liu and Brian D. Ziebart, 'Robust classification under sample selection bias', in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 37–45, (2014).
[18] Guoqing Liu, Jianxin Wu, and Zhi-Hua Zhou, 'Key instance detection in multi-instance learning', in *Proceedings of the 4th Asian Conference on Machine Learning*, pp. 253–268, (2012).
[19] Hidetoshi Shimodaira, 'Improving predictive inference under covariate shift by weighting the log-likelihood function', *Journal of Statistical Planning and Inference*, **90**(2), 227–244, (2000).
[20] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe, 'Direct importance estimation with model selection and its application to covariate shift adaptation', in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, pp. 1433–1440, (2007).
[21] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori, *Density Ratio Estimation in Machine Learning*, Cambridge University Press, 2009.
[22] Xiu-Shen Wei, Jianxin Wu, and Zhi-Hua Zhou, 'Scalable algorithms for multi-instance learning', *IEEE Transactions on Neural Networks and Learning Systems*, **28**(4), 975–987, (2017).
[23] Bianca Zadrozny, 'Learning and evaluating classifiers under sample selection bias', in *Proceedings of the 21st International Conference on Machine learning*, pp. 114–121, (2004).
[24] Lihi Zelnik-Manor and Pietro Perona, 'Self-tuning spectral clustering', in *Proceedings of the 17th International Conference on Neural Information Processing Systems*, pp. 1601–1608, (2004).
[25] Weijia Zhang and Zhi-Hua Zhou, 'Multi-instance learning with distribution change', in *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pp. 2184–2190, (2014).
[26] Ya-Lin Zhang and Zhi-Hua Zhou, 'Multi-instance learning with key instance shift', in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3441–3447, (2017).
[27] Zhi-Hua Zhou, Kai Jiang, and Ming Li, 'Multi-instance learning based web mining', *Applied Intelligence*, **22**(2), 135–147, (2005).
[28] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li, 'Multi-instance learning by treating instances as non-i.i.d. samples', in *Proceedings of the 26th International Conference on Machine Learning*, pp. 1249–1256, (2009).