Multiple Instance Learning for Unilateral Data*

Xijia Tang, Tingjin Luo✉, Tianxiang Luan, and Chenping Hou✉

National University of Defense Technology, Changsha, China
la__corse@163.com, tingjinluo@hotmail.com,
luantx@163.com, hcpnudt@hotmail.com

Abstract. Multi-instance learning (MIL) is a popular learning paradigm rooted in real-world applications. Recent studies have achieved prominent performance with sufficient annotation data. Nevertheless, acquisition of enough labeled data is often hard and only a little or partially labeled data is available. For example, in web text mining, the concerning bags (positive) is often rare compared with the unrelated ones (negative) and unlabeled ones. This leads to a new learning scenario with little negative bags and many unlabeled bags, which we name it as unilateral data. It is a new learning problem and has received little attention. In this paper, we propose a new method called Multiple Instance Learning for Unilateral Data (MILUD) to tackle this problem. To utilize the information of bags fully, we consider statistics characters and discriminative mapping information simultaneously. The key instances of bags are determined by the distinguishability of mapped samples based on fake labels. Besides, we also employed a empirical risk minimization loss function based on the mapping results to learn the optimal classifier and analyze its generalization error bound. The experimental results show our method outperforms other existing state-of-art methods.

Keywords: multi-instance learning · negative and unlabeled data learning · bag mapping · classification.

1 Introduction

Multiple instance learning (MIL) is one of the popular learning paradigms in the practical applications. In MIL, the annotations of data were only assigned to the bags. Therefore, MIL methods are able to deal with the classification problems with label ambiguity and reduce the requirements of the label information. Due to their excellent characteristics and outstanding performance, they have been widely used in many practical applications, such as drug activity prediction[9], image classification [1] and text recognition [4] etc.

In literature, there are a lot of methods proposed to solve MIL problem. These methods can be divided into three categories: instance-level methods,

* This work was supported by the NSF of China under Grants No. 61922087, 61906201 and 62006238, NSF of Hunan Province under Grant No. 2020JJ5669, and the NSF for Distinguished Young Scholars of Hunan Province under Grant No. 2019JJ20020. Chenping Hou and Tingjin Luo are the corresponding authors.

bag-level methods and embedding approaches by different characteristics [1]. The first group is the instance-level methods [9, 15, 13, 19, 5], which all of bags split up into the instances and then learn the optimal classifier based on all instances. Ins-KI-SVM [15] is one of their representative methods and proposed to build the instance classifier by maximizing the margin between the selected key instances. The second group is bag-level methods [17, 11, 12, 8, 2], which build the classification model of the bags. Citation-kNN [17] and MI-Kernel [11] are two representatives of traditional bag-level methods. Citation-kNN [17] adapts kNN to MIL problem, which not only take the neighbors of bag into account but also samples that count bag as neighbor. Smola et al. [11] proposed MI-Kernel to obtain the high dimensional mapping of each bag via set kernels and learn a linear model by SVM. Last group is embedding methods [7, 6, 14, 10, 5, 18], whose main idea is to extract particular kinds of information for each bag in the new latent feature space. After mapping, MIL problem is transferred as a classical supervised problem. Wu et al. [18] proposed a discriminative mapping approach named as MILDM, which maps bags into the latent feature space via discriminative instance pool.

Although existing methods can handle the MIL problem well, most of them require the entire accurate label information of all bag data. However, acquisition of enough labeled data is often hard and only a little partially labeled data is available in practice. For example, in web text mining, the concerning bags (positive) is often rare compared with the unrelated ones (negative) and unlabeled ones. In the spam identification, the feature of spam (positive) will change to evade our shielding. Traditional classifier needs to continuously update the positive train set to ensure its effectiveness, while the NU classifier only needs to update the unlabeled data, which cost much less. This leads to a new learning scenario with little negative bags and many unlabeled bags, which we name it as unilateral data. As far as we know, this problem has so far been little studied and cannot be suitable for existing multiple instance methods. Bao et al. [3] proposed a convex classification method called PU-SKC which solves the MIL problem in PU scenario based on set kernel. However, set kernel can only extract maximum and minimum of the instances features in bag. Since this strategy is relatively simple and only extracts the training set information from data level, the performance of classifier cannot be guaranteed during the training process.

In this paper, we propose a new method called Multiple Instance Learning for Unilateral Data (MILUD) to tackle this problem. Specifically, to utilize the information of bags fully, we consider statistics characters and discriminative mapping information simultaneously to make the bags in different classes more separable. To preserve more data information of bags, the key instances of bags are selected by maximizing the distances between bags in different classes. After bag mapping, we propose to use the convex NU empirical risk loss to learn the optimal classifier for unilateral data problem. Besides, the generalization error bound of our method are provided. Finally, the extensive experimental results show our proposed method achieves better performance than other existing state-of-art methods. The contributions of this paper can be summarized as follows:

- To the best of our knowledge, our method MILUD is the first one proposed to solve the unilateral MIL data problem.
- We propose a novel method by incorporating the statistics characters with the discriminative mapping information to enhance its performance. Besides, we select the key instances to preserve more discriminate embedded features of bags by maximizing the distances between bags in different classes.
- The extensive experimental results on multiple public datasets verify the effectiveness of our proposed method and analyse the effect of the quantity of labeled and unlabeled data.

2 The Proposed Method

The problem of NU learning in MIL scenario comes from real life and have rarely been studied. For example, when the probability distribution of positive data changes frequently and the negative part remains constant. In this case, cost of updating the unlabeled train set is much less than the positive train set, which leads to a NU learning problem. Compared with traditional learning, difficulties in this scenario are mainly reflected in two aspects. First, to extract the information of positive and negative categories from only unlabeled and negative data. It indicates that the emphasis of NU learning is how to fully utilize unlabeled data. Second, labels of positive bags cannot express the labels of specific instances in it. This characteristic makes the unlabeled data, which is the focus point in NU learning, become ambiguous.

In this paper, we propose a novel method called MILUD, which can overcome the inexact information in NU-MIL problem. The mean ideas of our approach are corresponding to the two difficulties. We first use statistical and discriminative feature to fully extract the effective information simultaneously. After the NU-MIL problem be convert to a NU learning problem, train the classifier by minimize the empirical risk loss on the basis of dataset distribution.

Before presenting the details of our method, we describe the notations used in this paper. Denote $B_p^N = \{\boldsymbol{x}_{p_1}^N, \boldsymbol{x}_{p_2}^N, \cdots, \boldsymbol{x}_{p_i}^N\}$, $B_q^U = \{\boldsymbol{x}_{q_1}^U, \boldsymbol{x}_{q_2}^U \cdots, \boldsymbol{x}_{q_j}^U\}$ denote p_i and q_j instances in bag B_p^N and B_q^U, respectively, where $\boldsymbol{x}_{p_i}^N$, $\boldsymbol{x}_{q_j}^U \in \mathbb{R}^d$. The negative and unlabeled training sets are defined as $D^N = \{B_1^N, B_2^N, \cdots, B_{Nn}^N\}$ and $D^U = \{B_1^U, B_2^U, \cdots, B_{Nu}^U\}$, N_n and N_u represent the number of negative and unlabeled bags in D^N and D^U.

2.1 Bag Mapping

Bag mapping solves the label ambiguous problem in MIL by extracting feature mainly from two perspectives: data-based and label-based. However, in NU-MIL scenario, label-based mapping usually leads to unsatisfied performance, because labels of the unlabeled samples are invisible. Therefore, we consider the data-based mapping features first, which can be extracted without label information. Statistical feature is a great choice to be the first part of our mapping. It complements the shortcomings of label-based features and possesses low computational

cost. Motivated by the idea of set kernel [11], we calculate statistic $\boldsymbol{s}_{3m}(B)$ which are consist of the maximum, minimum and mean values of each dimension $x^{(i)}$ of the instances $\boldsymbol{x}$ in bag B by the equation below:

$$\boldsymbol{s}_{3m}(B) = \left[\min_{\boldsymbol{x}\in B} x^{(1)}, \cdots, \min_{\boldsymbol{x}\in B} x^{(d)}, \max_{\boldsymbol{x}\in B} x^{(1)}, \cdots, \max_{\boldsymbol{x}\in B} x^{(d)}, \overline{x^{(1)}}, \cdots, \overline{x^{(d)}}\right]. \tag{1}$$

The maximum and minimum value describe the boundary of bags, and the average value can further give the outlier information of bags. Based on Eq.(1), the set kernel $\widetilde{\boldsymbol{k}}_{3m}$ can be calculated by

$$\widetilde{\boldsymbol{k}}_{3m}(B, C) = \widetilde{\boldsymbol{k}}\left(\boldsymbol{s}_{3\mathrm{m}}(B), \boldsymbol{s}_{3\mathrm{m}}(C)\right), \tag{2}$$

where $\widetilde{\boldsymbol{k}}$ can be any kernel function and it is a basic mapping derived from set kernel based on the limit information. Now we can map the bags to single instances by kernel centers $\{C_1, C_2, \cdots, C_M\}$ and $\widetilde{\boldsymbol{k}}_{3m}(B, C)$ mentioned in Eq.(2):

$$\boldsymbol{\Phi}_{3m}(B) := \left[\widetilde{\boldsymbol{k}}_{3\mathrm{m}}(B, C_1), \cdots, \widetilde{\boldsymbol{k}}_{3\mathrm{m}}(B, C_M)\right]^{\top} \tag{3}$$

However, the statistical features do not make use of label information, which means the extraction of dataset information is insufficient. This is why we adopte discriminative feature, which is a excellent label-based mapping strategy. The discriminative feature should meet the following two conditions:

- bags with same label are as similar as possible in mapping feature space;
- bags with different labels are as diverse as possible in mapping feature space.

These two conditions guarantees the separability of samples in the mapping space, because it makes labels directly linked to the classification performance. For PN data, training process is carried out on the existing of two categories of label. However, there is only label information of negative data for NU data, which makes it impossible to maximize the distance between different data directly. By mapping process, when keeping negative data far away positive data, it also makes negative data far away from itself. Therefore, it is not feasible to map data by only discriminative features in NU-MIL scenarios.

To solve the discriminative feature extraction problem of NU dataset thereby further obtain more complete train set information, we separate the "most likely positive samples" from unlabeled data. The specific method is to train a preliminary classifier by Eq.(3), and on this basic classifier we can give each unlabeled bag a fake label. Here we directly use the result of this strategy. The unlabeled data D^N is divided into two parts: D^{UP} and D^{UN}, which are the positive and negative bags identified from unlabeled bags. Denote $D^{N'} = \{D^N, D^{UN}\}$ and $D^{P'} = \{D^{UP}\}$ as the new dataset based on fake labels, which is the relatively reliable information extracted from the unilateral data.

Since the label-based extraction problem is solved, next we'll explain the step details of discriminative feature. Denote $\boldsymbol{\Phi}_{DIP}(B)$ as a mapping rule based on instance similarity $\boldsymbol{s}$:

$$\boldsymbol{\Phi}_{DIP}(B) = \left[\boldsymbol{s}\left(B, \boldsymbol{x}_1^{\phi}\right), \cdots, \boldsymbol{s}\left(B, \boldsymbol{x}_m^{\phi}\right)\right]^{\top}, \tag{4}$$

where $\boldsymbol{s}\left(B, \boldsymbol{x}_k^{\phi}\right) = \max_{\boldsymbol{x}_l \in B} \exp\left(-\left\|\boldsymbol{x}_l - \boldsymbol{x}_k^{\phi}\right\|^2 / \sigma^2\right)$, which can be viewed as the similarity between bag B and $\boldsymbol{x}_k^{\phi}$. $\boldsymbol{x}_l$ is the l-th instance in the bag B, $\boldsymbol{x}_k^{\phi}$ is the instance used for mapping and m is the number of it. The m most discriminative instances used for mapping make up a collection,which called discriminative instance pool (DIP): $\mathcal{P} = \left\{\boldsymbol{x}_1^{\phi}, \cdots, \boldsymbol{x}_m^{\phi}\right\}, \boldsymbol{x}_k^{\phi} \in \left[B^{N'}, B^{U'}\right]$, where $\boldsymbol{x}_k^{\phi}$ is the k-th element in $\mathcal{P}$. The strategy of processing NU data by DIP is similar to the idea in MILDM [18]. We can map a bag into single instance by the DIP. In order to find the m instances that make the mapped samples most separable, denote $\mathcal{J}(\mathcal{P})$ as objective function:

$$\mathcal{J}(\mathcal{P}) = \frac{1}{2}\sum_{i,j} K_{\mathcal{P}}\left(B_i, B_j\right) Q_{i,j},\ Q_{i,j} = \begin{cases} -1/|A|, y_i y_j = 1; \\ 1/|B|, y_i y_j = -1, \end{cases} \tag{5}$$

where $K_{\mathcal{P}}\left(B_i, B_j\right)$ denotes the distance between B_i and B_j after being mapped, $Q_{i,j}$ is weight factor, $|\cdot|$ is the number of elements in the set, and y_i, y_j are the labels of B_i and B_j, respectively. A and B are denoted as:

$$A = \{(i,j) \mid y_i y_j = 1\}, B = \{(i,j) \mid y_i y_j = -1\}. \tag{6}$$

By optimizing $\mathcal{J}(\mathcal{P})$, the goal of minimizing the distance between samples from same category and maximizing the distance between different ones' can be directly achieved. The function $K_{\mathcal{P}}\left(B_i, B_j\right)$ is used to describe the distance between B_i and B_j denoted as:

$$K_{\mathcal{P}}\left(B_i, B_j\right) = \left\|B_i^{\phi} - B_j^{\phi}\right\|^2 = \left\|\mathcal{I}_{\mathcal{P}} B_i^{\phi_x} - \mathcal{I}_{\mathcal{P}} B_j^{\phi_x}\right\|^2, \tag{7}$$

where $\mathcal{I}_{\mathcal{P}}$ denotes a diagonal matrix, if x_k belongs to the discriminative instance pool, the k-th diagonal element in $\mathcal{I}_{\mathcal{P}}$ is 1, otherwise it is 0. Based on $\mathcal{I}_{\mathcal{P}}$, we can choose mapped features by discriminative instance pool from all the instances.

To sum up, when maximizing $\mathcal{J}(\mathcal{P})$, we diminish the distances between instances from same class of bags and enlarge distances between different classes instances. So our next target is maximizing $\mathcal{J}(\mathcal{P})$ to get $\mathcal{P}_*$:

$$\mathcal{P}_* = \arg\max \mathcal{J}(\mathcal{P}), \quad \text{s.t.} \quad |\mathcal{P}| = m. \tag{8}$$

Let $\mathcal{X}_{\phi} = [B_1^{\phi_x}, \cdots, B_n^{\phi_x}] = [\phi_1, \cdots, \phi_p]^{\top}$ and $n = |D^{N'}| + |D^{P'}|$. $L = D - Q$ is a Laplacian matrix, where diagonal matrix D satisfy $D_{i,i} = \sum_j Q_{ij}$. $\mathcal{J}(\mathcal{P})$ can be rewritten as:

$$\begin{aligned} \mathcal{J}(\mathcal{P}) &= \frac{1}{2}\sum_{i,j}\left\|\mathcal{I}_{\mathcal{P}} B_i^{\phi_x} - \mathcal{I}_{\mathcal{P}} B_j^{\phi_x}\right\|^2 Q_{i,j} \\ &= \sum_i \left(B_i^{\phi_x}\right)^{\top} \mathcal{I}_{\mathcal{P}}^{\top} \mathcal{I}_{\mathcal{P}} B_i^{\phi_x} D_{i,i} - \sum_{i,j} \left(B_i^{\phi_x}\right)^{\top} \mathcal{I}_{\mathcal{P}}^{\top} \mathcal{I}_{\mathcal{P}} B_j^{\phi_x} Q_{i,j} \\ &= \operatorname{tr}\left(\mathcal{I}_{\mathcal{P}}^{\top} \mathcal{X}_{\phi} L \mathcal{X}_{\phi}^{\top} \mathcal{I}_{\mathcal{P}}\right) = \sum_{x_k^{\phi} \in \mathcal{P}} \phi_k^{\top} L \phi_k. \end{aligned} \tag{9}$$

Let $\phi_k^\top L \phi_k = f(\boldsymbol{x}_k^\phi, L)$, the optimal DIP (8) is equivalent to:

$$\mathcal{P}_* = \arg\max \sum_{x_k^\phi \in \mathcal{P}} f(\boldsymbol{x}_k^\phi, L), \quad \text{s.t.} \quad |\mathcal{P}| = m. \tag{10}$$

Since $\mathcal{P}_*$ is composed of the function $f(\boldsymbol{x}_k^\phi, L)$ of all instances $\boldsymbol{x}_k$ in dataset, we can optimize $\mathcal{P}$ by the searching algorithm. Under the premise of satisfying the constraints, update $f(\boldsymbol{x}_k^\phi, L)$ one by one and iterate $\mathcal{P}$. Through each iteration, instance $\boldsymbol{x}_{min}$ corresponding to the minimum value of $f(\boldsymbol{x}_{min}^\phi, L)$ will be removed. The iteration ends until all instances have been traversed.

Now we have two kinds of mapped features based on different rules: statistic rule $\boldsymbol{\Phi}_{3m}(B)$ and discriminative instance pool rule $\boldsymbol{\Phi}_{DIP}(B)$, which two are complementary to each other. The combination of two mappings is defined as

$$\boldsymbol{\Phi}_{cps}(B) := [\boldsymbol{\Phi}_{3m}(B), \boldsymbol{\Phi}_{DIP}(B)]. \tag{11}$$

Composite mapping $\boldsymbol{\Phi}_{cps}(B)$ describe the characteristics of the bag from both statistical features and discriminative features, so that the mapped results can better help the next step of NU classification.

2.2 NU Classification Based on Composite mapping

By the mapping rule in Eq.(11), bags in datasets D_N and D_U can be mapped to single instances. The next step is to get the classifier by constructing the NU loss function. For more convenient representation, we write mapped sample $\boldsymbol{\Phi}_{cps}(B)$ as $\boldsymbol{\varphi}$. In particular, $\boldsymbol{\Phi}_{cps}(B_i^N)$ and $\boldsymbol{\Phi}_{cps}(B_j^U)$ are denoted as $\boldsymbol{\varphi}_i^N$ and $\boldsymbol{\varphi}_j^U$, respectively. Classifier $g(\boldsymbol{\varphi})$ is a linear parametric model:

$$g(\boldsymbol{\varphi}) = \boldsymbol{\omega}^\top \boldsymbol{\varphi} + b. \tag{12}$$

On the basis of all the samples in dataset are independent and identically distributed with probability $p(\boldsymbol{\varphi}, y)$, we intend to train a classifier based on minimizing the empirical risk loss on the training dataset. The loss function is composed of the expectation over risk loss of positive and negative data. However, labels for positive class are inexistent in NU dataset, which means the positive part in loss function is not directly available. In order to train a classifier by only negative and unlabeled data, the risk loss in positive samples can be estimated by negative and unlabeled samples:

$$\theta_P \mathbb{E}_P[l(g(\boldsymbol{\varphi})] = \mathbb{E}_U[l(g(\boldsymbol{\varphi}))] - \theta_N \mathbb{E}_N[l(g(\boldsymbol{\varphi}))], \tag{13}$$

where θ_P and θ_N denote the class-prior probabilities of the positive and negative class, $\theta_P + \theta_N = 1$. $g(\boldsymbol{\varphi})$ is the classifier and $l(z)$ is a loss function. $\mathbb{E}_P[\cdot]$ and $\mathbb{E}_N[\cdot]$ are the expectations over the prior distribution on positive and negative data. Inspired by the similar formulation proposed in [16], specifically, we use loss function $R(g)$ drawn from NU data by the risk:

$$\begin{aligned} R(g) =& \theta_P \mathbb{E}_P[l(g(\boldsymbol{\varphi}))] + \theta_N \mathbb{E}_N[l(-g(\boldsymbol{\varphi}))] \\ =& \theta_N \mathbb{E}_N[\tilde{l}(-g(\boldsymbol{\varphi}))] + \mathbb{E}_U[l(g(\boldsymbol{\varphi}))], \end{aligned} \tag{14}$$

Algorithm 1 Multiple Instance learning for Unilateral Data (MILUD)

Input: negative dataset D^N and unlabeled dataset D^U
Output: The classifier $(\boldsymbol{\omega}, b)$ for Multiple Instance Data

1: Calculate the statistic $\boldsymbol{s}_{3m}(B^{N(U)})$ by Eq.(1);
2: Compute the set kernel $\widetilde{\boldsymbol{k}}_{3m}(B^{N(U)}, C)$ by Eq.(2);
3: Learn a fake label by Eq. (3) and Eq.(15), $D^U = \{D^{UP}, D^{UN}\}$;
4: Search the optimal DIP set $\mathcal{P}_*$ by solving the problem in (10);
5: Compute the feature $\boldsymbol{\varphi}$ by concatenating $\boldsymbol{\Phi}_{3m}(B)$ and $\boldsymbol{\Phi}_{DIP}(B)$ in Eq.(11);
6: Learn the optimal NU classifier $(\boldsymbol{\omega}, b)$ by optimizing Eq.(16).
7: **return** the optimal model $(\boldsymbol{\omega}, b)$.

where $\tilde{l}(z) = l(z) - l(-z)$ is a composite loss function.

When $\tilde{l}(z)$ satisfies the condition of $\tilde{l}(z) = l(z) - l(-z) = -z$, the minimization of NU loss function Eq.(14) is a convex optimization problem. In this paper, we choose double hinge loss $l_{DH}(z) = \max(-z, \max(0, \frac{1}{2} - \frac{1}{2}z))$ as $l(z)$. $R(g)$ in Eq.(14) is rewritten as:

$$\begin{aligned} R(g) =& \theta_N \mathbb{E}_N[g(\boldsymbol{\varphi}^N)] + \mathbb{E}_U[l(g(\boldsymbol{\varphi}^U))] \\ =& \frac{\theta_N}{N_n} \sum_{i=1}^{N_n} \boldsymbol{\omega}^\top \boldsymbol{\varphi}_i^N + \theta_N b + \frac{\lambda}{2} \boldsymbol{\omega}^\top \boldsymbol{\omega} + \frac{1}{N_u} \sum_{j}^{N_u} l_{DH}(\boldsymbol{\omega}^\top \boldsymbol{\varphi}_j^U + b) \end{aligned} \tag{15}$$

where $\boldsymbol{\omega}^\top \boldsymbol{\omega}$ is the regularization item, λ is the parameter used to adjust $\boldsymbol{\omega}^\top \boldsymbol{\omega}$. The optimization in Eq.(15) can be rewritten with the slack variable $\boldsymbol{\xi}$, which is used to bound the max operators:

$$\begin{cases} \min\limits_{\boldsymbol{\omega}, b, \boldsymbol{\xi}} \frac{\theta_N}{N_n} \mathbf{1}^\top \boldsymbol{\varphi}_i^N \boldsymbol{\omega} + \theta_N b + \frac{1}{N_u} \mathbf{1}^\top \boldsymbol{\xi} + \frac{\lambda}{2} \boldsymbol{\omega}^\top \boldsymbol{\omega} \\ \text{s.t.} \qquad \boldsymbol{\xi} \geq \mathbf{0}, \quad \boldsymbol{\xi} \geq -\boldsymbol{\varphi}_j^U \boldsymbol{\omega} - b\mathbf{1}, \\ \qquad\quad \boldsymbol{\xi} \geq \frac{1}{2}\mathbf{1} - \frac{1}{2}\boldsymbol{\varphi}_j^U \boldsymbol{\omega} - \frac{1}{2} b\mathbf{1}. \end{cases} \tag{16}$$

The problem in Eq.(16) is a quadratic program, which can be solved by interior point method. Interior point method transforms constrained optimization problem into unconstrained optimization problem by adding an obstacle function to the original target function, the original constraints will be replaced. Then the unconstrained optimization problem would be solved by Newton's method. By optimizing Eq.(16), we get the final classifier g. The main steps of MILUD are shown in Algorithm 1.

2.3 Analysis of Generalization Error Bounds

Similar to [16], we analyze the upper bound of generalization error for g. Denote $\mathcal{H}$ as the domain set, $C_{\boldsymbol{\omega}}$ and C_{φ} are certain positive constants. Define $\mathcal{G} = \left\{ g(\boldsymbol{\varphi}) = \boldsymbol{\omega}^\top \boldsymbol{\varphi} \mid \|\boldsymbol{\omega}\| \leq C_{\boldsymbol{\omega}}, \sup_{\boldsymbol{\varphi} \in \mathcal{H}} \|\boldsymbol{\varphi}\| \leq C_{\varphi} \right\}$ as function class. The expected

risk $\mathcal{R}(g)$ and empirical risk $\widehat{\mathcal{R}}(g)$ of classifier can be written as:

$$\begin{aligned}\mathcal{R}(g) &= \theta_N \mathbb{E}_{p(\boldsymbol{\varphi}|y=-1)}[g(\boldsymbol{\varphi})] + \mathbb{E}_{p(\boldsymbol{\varphi})}[\ell_{DH}(g(\boldsymbol{\varphi}))], \\ \widehat{\mathcal{R}}(g) &= \frac{\theta_N}{N_n}\sum_{u=1}^{N_n} g\left(\boldsymbol{\varphi}_u^N\right) + \frac{1}{N_u}\sum_{v=1}^{N_u}\ell\left(g\left(\boldsymbol{\varphi}_v^U\right)\right).\end{aligned} \tag{17}$$

Theorem 1. *For any fixed g and any $\delta \in (0,1)$, the difference between $\mathcal{R}(g)$ and $\widehat{\mathcal{R}}$ satisfies:*

$$\mathcal{R}(g) - \widehat{\mathcal{R}}(g) \leq \sqrt{C_\omega^2 C_\varphi^2 \log\frac{2}{\delta}/2}\left(\frac{2\theta_N}{\sqrt{N_n}} + \frac{1}{\sqrt{N_u}}\right). \tag{18}$$

The details of this proof is presented in the supplemental file. The results of this theorem indicates that the generalization error of our model decreases with the increase of $\sqrt{N_n}$ and $\sqrt{N_u}$. In other words, increasing the number of negative and unlabeled bags can reduce the error and improve the performance of our method. The conclusion of this proof is also verified by experimental results. both contributes to reducing the error.

3 Experiments

3.1 Experiment Settings

To verify the superiority of our approach, we compare MILUD with four classical methods, including C-kNN [17], aMILGDM [18], KISVM [15] and PU-SKC [3]. The first three are proposed to solve the problem with complete data and balanced labels. PU-SKC is one representative of PU-MIL, which train the classifier via concise statistical mapping and PU empirical risk loss. We take the experiments on eight public datasets, i.e. Corel_bm, Corel_hd, Sival_ab, Sival_bc, Atoms, Bonds, Elephant and Tiger. The specific information of these datasets is shown in Table 1.

Since these datasets in Table 1 are too small to evaluate NU or PU methods, similar with [3], we augment the information of datasets by increasing the number of bags. Specifically, we randomly select bags from original dataset and duplicate them with the Gaussian noise of mean zero and variance 0.01. In this way, we increase the number of negative and unlabeled bags to 20 and 180, respectively. The remaining 100 positive bags and 100 negative bags are test set. In addition, we take comparative experiments to verify the effectiveness of MILUD under different unlabeled bags composition. The ratio of negative bags in unlabeled bags are set as {0.1,0.3,0.5,0.7}. Finally, we repeat 30 times experiments and report the mean value of each method under different classification metrics.

3.2 Results

Classification Performance We compared the performance of MILUD and other four state-of-the-art approaches on eight datasets. The classification accuracy, area under the curve (AUC) and F-measure are adopted to evaluated their

Table 1. The main information of eight public dataset.

Name	# of Pos Bags	# of Neg Bags	Features	Avg # of Insts
Corel_bm	100	100	9	3.46
Corel_hd	100	100	9	3.85
Sival_ab	60	60	30	31.68
Sival_bc	60	60	30	31.78
Atoms	125	63	10	8.61
Bonds	125	63	16	21.25
Elephant	100	100	230	6.96
Tiger	100	100	230	6.10

performance. Under the four different class-prior probability θ_N of {0.1,0.3,0.5,0.7}, the average classification accuracy and standard deviation of 30 independent trials of different methods are presented in Table 2.

Table 2. Average classification accuracy (standard deviation) of different compared methods on eight public datasets. The first highest score is in bold.

Dataset	θ_N	Accuracy				
		MILUD	PU-SKC	C-kNN	aMILGDM	KI-SVM
Corel_bm	0.1	**0.694(0.048)**	0.571(0.075)	0.665(0.032)	0.647(0.031)	0.630(0.041)
	0.3	**0.662(0.040)**	0.608(0.053)	0.592(0.024)	0.600(0.032)	0.556(0.035)
	0.5	**0.602(0.067)**	0.589(0.060)	0.560(0.025)	0.567(0.029)	0.528(0.035)
	0.7	**0.562(0.059)**	0.524(0.044)	0.542(0.023)	0.546(0.016)	0.538(0.050)
Corel_hd	0.1	**0.839(0.034)**	0.74210.064)	0.776(0.041)	0.701(0.034)	0.602(0.070)
	0.3	**0.866(0.031)**	0.821(0.031)	0.662(0.036)	0.617(0.032)	0.588(0.103)
	0.5	**0.845(0.036)**	0.794(0.061)	0.611(0.025)	0.583(0.025)	0.572(0.053)
	0.7	**0.774(0.082)**	0.689(0.081)	0.560(0.023)	0.555(0.029)	0.703(0.049)
Sival_ab	0.1	**0.773(0.048)**	0.740(0.043)	0.699(0.034)	0.689(0.032)	0.763(0.046)
	0.3	**0.733(0.050)**	0.711(0.050)	0.631(0.031)	0.612(0.028)	0.523(0.059)
	0.5	**0.719(0.057)**	0.705(0.054)	0.579(0.028)	0.575(0.020)	0.524(0.076)
	0.7	**0.654(0.054)**	0.607(0.070)	0.565(0.026)	0.562(0.029)	0.515(0.072)
Sival_bc	0.1	**0.875(0.038)**	0.808(0.041)	0.747(0.035)	0.754(0.040)	0.860(0.036)
	0.3	**0.883(0.035)**	0.825(0.033)	0.643(0.030)	0.626(0.026)	0.680(0.038)
	0.5	**0.823(0.068)**	0.797(0.068)	0.599(0.035)	0.585(0.025)	0.683(0.076)
	0.7	**0.742(0.090)**	0.728(0.077)	0.569(0.037)	0.567(0.027)	0.659(0.036)
Atoms	0.1	0.6017(0.051)	0.525(0.029)	0.517(0.022)	0.605(0.036)	**0.623(0.047)**
	0.3	**0.671(0.047)**	0.576(0.043)	0.508(0.027)	0.572(0.029)	0.564(0.040)
	0.5	**0.660(0.055)**	0.556(0.052)	0.502(0.022)	0.549(0.028)	0.599(0.036)
	0.7	**0.574(0.097)**	0.524(0.054)	0.506(0.023)	0.523(0.023)	0.555(0.023)
Bonds	0.1	**0.616(0.063)**	0.520(0.058)	0.527(0.024)	0.605(0.039)	0.559(0.074)
	0.3	0.620(0.048)	0.601(0.057)	0.523(0.021)	0.551(0.023)	**0.642(0.040)**
	0.5	**0.572(0.076)**	0.572(0.063)	0.523(0.018)	0.543(0.026)	0.530(0.024)
	0.7	**0.608(0.074)**	0.564(0.063)	0.511(0.020)	0.523(0.023)	0.539(0.026)
Elephant	0.1	0.785(0.041)	**0.788(0.043)**	0.691(0.034)	0.658(0.034)	0.677(0.064)
	0.3	**0.779(0.057)**	0.752(0.069)	0.634(0.000)	0.601(0.025)	0.682(0.053)
	0.5	**0.749(0.060)**	0.730(0.049)	0.586(0.032)	0.566(0.019)	0.676(0.065)
	0.7	**0.647(0.059)**	0.643(0.060)	0.563(0.026)	0.552(0.024)	0.653(0.062)
Tiger	0.1	**0.714(0.050)**	0.712(0.056)	0.699(0.033)	0.665(0.044)	0.579(0.035)
	0.3	**0.739(0.056)**	0.728(0.058)	0.630(0.034)	0.595(0.023)	0.558(0.041)
	0.5	**0.688(0.042)**	0.676(0.046)	0.569(0.028)	0.565(0.024)	0.531(0.025)
	0.7	**0.591(0.063)**	0.587(0.064)	0.547(0.021)	0.551(0.021)	0.499(0.086)

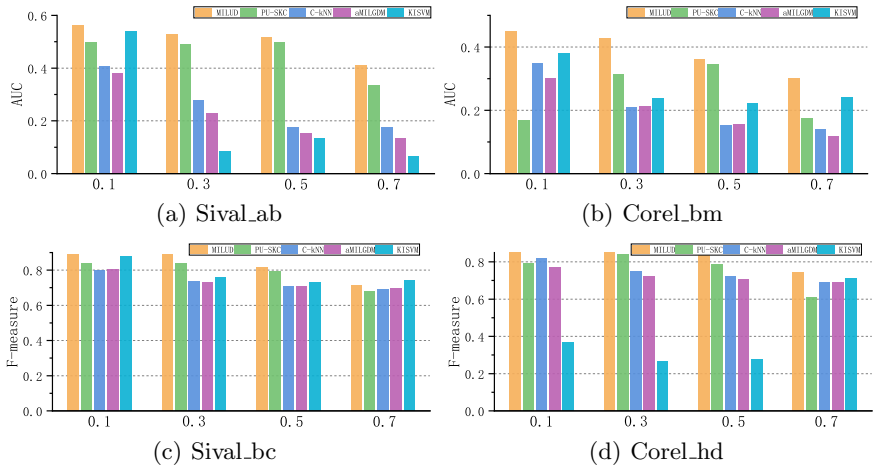
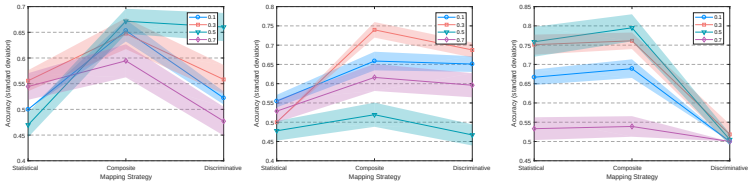


Fig. 1. The results of AUC and F-measure on four datasets. The first line are the AUC of Sival_ab and Corel_bm, the second line are the F-measure of Sival_bc and Corel_hd.

We also provide the AUC and F-measure results on Sival_ab, Corel_bm, Sival_bc and Corel_hd in Fig.1. The results in Table 2 and Fig.1 present that the performance of our MILUD is better than other methods in most of cases and simultaneously verify the effectiveness of our proposed method.

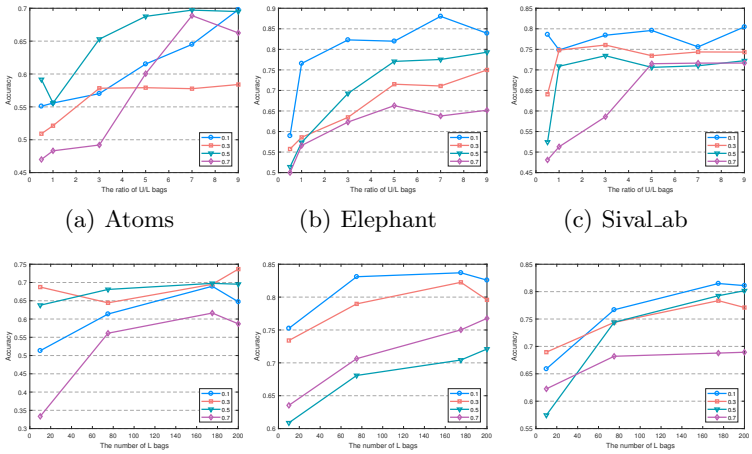
Effect of Mapping Combination To verify the validity of our mapping combination, we conduct experiments on three datasets Atoms, Bonds and Sival_ab with different class-prior probability θ_N. We compare the NU empirical risk minimization classifier over three mapping rules, included the composite mapping $\boldsymbol{\Phi}_{cps}$ employed in MILUDM, statistical feature mapping $\boldsymbol{\Phi}_{3m}$ and discriminative feature mapping $\boldsymbol{\Phi}_{DIP}$. It should be noted that the $\boldsymbol{\Phi}_{DIP}$ makes negative and unlabeled samples far away directly without fake-label strategy. Experiment results in Fig.(2) are the accuracy and standard deviation on three mapping rules, which illustrates that the mapping combination improves the classifier performance by making up the shortcomings of extracting information unilaterally.

Impact of Negative & Unlabeled Data Increasing To explore the impact of unlabeled and negative data size on MILUD, we take two experiments with increase of unlabeled or negative bags on Atoms, Elephant and Sival_ab. Specificlly, we randomly select 20 negative and 10 unlabeled samples as initial training set. Then increase the number of unlabeled or negative samples of training set and observe the tendency of classification performance. As seen from the accuracy results in Fig. 3, performance of MILUD becomes better with the increase of two types of bags. The results verify that our method can be effective to utilize the information of unlabeled data to improve the performance of our model.



(a) Atoms (b) Bonds (c) Sival_ab

Fig. 2. The result of accuracy and standard deviation on three mapping strategy on three datasets Atoms, Bonds and Sival_ab. Each figure contains the results on four θ_N.



(a) Atoms (b) Elephant (c) Sival_ab

(d) Atoms (e) Elephant (f) Sival_ab

Fig. 3. The change trend of accuracy when the number of U/L bags increases at four class prior. The first line are the accuracy of increasing unlabeled bags, the second line are the accuracy of increasing L bags.

4 Conclusion

In this paper, we focus on an important but unheeded NU-MIL problem, and propose a two-stages method named MILUD to solve it. The composite mapping utilizes both data based statistical features and label based discriminative mapping information, which ensure that the information of bags can be fully preserved. Then, a convex learning model is derived by minimizing the empirical loss to solve the NU problem. Experimental results on eight public datasets indicate our method outperforms other compared methods in most of cases.

References

1. Amores, J.: Multiple instance classification: Review, taxonomy and comparative study. Artificial Intelligence **201**, 81–105 (2013)

2. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. In: Advances in Neural Information Processing Systems. pp. 561–568. MIT Press (2002)
3. Bao, H., Sakai, T., Sato, I., Sugiyama, M.: Convex formulation of multiple instance learning from positive and unlabeled bags. Neural Networks **105**, 132–141 (2018)
4. Carbonneau, M., Cheplygina, V., Granger, E., Gagnon, G.: Multiple instance learning: A survey of problem characteristics and applications. Pattern Recognition **77**, 329–353 (2018)
5. Carbonneau, M., Granger, E., Raymond, A.J., Gagnon, G.: Robust multiple-instance learning ensembles using random subspace instance selection. Pattern Recognition **58**, 83–99 (2016)
6. Chen, Y., Bi, J., Wang, J.Z.: MILES: multiple-instance learning via embedded instance selection. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(12), 1931–1947 (2006)
7. Chen, Y., Wang, J.Z.: Image categorization by learning and reasoning with regions. Journal of Machine Learning Research **5**, 913–939 (2004)
8. Cheplygina, V., Tax, D.M.J., Loog, M.: Multiple instance learning with bag dissimilarities. Pattern Recognition **48**(1), 264–275 (2015)
9. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence **89**(1-2), 31–71 (1997)
10. Fu, Z., Robles-Kelly, A., Zhou, J.: MILIS: multiple instance learning with instance selection. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5), 958–977 (2011)
11. Gärtner, T., Flach, P.A., Kowalczyk, A., Smola, A.J.: Multi-instance kernels. In: Proceedings of the Nineteenth International Conference. pp. 179–186. Morgan Kaufmann (2002)
12. Leistner, C., Saffari, A., Bischof, H.: Miforests: Multiple-instance learning with randomized trees. In: Proceedings of 11th European Conference on Computer Vision. Lecture Notes in Computer Science, vol. 6316, pp. 29–42. Springer (2010)
13. Li, F., Sminchisescu, C.: Convex multiple-instance learning by estimating likelihood ratio. In: Advances in Neural Information Processing Systems. pp. 1360–1368. Curran Associates, Inc. (2010)
14. Li, W., Yeung, D.: MILD: multiple-instance learning via disambiguation. IEEE Transactions on Knowledge and Data Engineering **22**(1), 76–89 (2010)
15. Li, Y., Kwok, J.T., Tsang, I.W., Zhou, Z.: A convex method for locating regions of interest with multi-instance learning. In: European Conference of Machine Learning. Lecture Notes in Computer Science, vol. 5782, pp. 15–30. Springer (2009)
16. Sakai, T., du Plessis, M.C., Niu, G., Sugiyama, M.: Semi-supervised classification based on classification from positive and unlabeled data. In: Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 2998–3006. PMLR (2017)
17. Wang, J., Zucker, J.: Solving the multiple-instance problem: A lazy learning approach. In: Proceedings of the Seventeenth International Conference on Machine Learning. pp. 1119–1126. Morgan Kaufmann (2000)
18. Wu, J., Pan, S., Zhu, X., Zhang, C., Wu, X.: Multi-instance learning with discriminative bag mapping. IEEE Transactions on Knowledge and Data Engineering **30**(6), 1065–1080 (2018)
19. Xiao, Y., Liu, B., Hao, Z., Cao, L.: A similarity-based classification framework for multiple-instance learning. IEEE Transactions on Cybernetics **44**(4), 500–515 (2014)