Multiple-Instance Learning from Triplet Comparison Bags

SENLIN SHU, Chongqing University, China
DENG-BAO WANG, Southeast University, China
SUQIN YUAN, The University of Sydney, Australia
HONGXIN WEI, Southern University of Science and Technology, China
JIUCHUAN JIANG, Nanjing University of Finance and Economics, China
LEI FENG*, Nanyang Technological University, Singapore
MIN-LING ZHANG, Southeast University, China

Multiple-instance learning (MIL) solves the problem where training instances are grouped in bags, and a binary (positive or negative) label is provided for each bag. Most of the existing MIL studies need fully labeled bags for training an effective classifier, while it could be quite hard to collect such data in many real-world scenarios, due to the high cost of data labeling process. Fortunately, unlike fully labeled data, *triplet comparison data* can be collected in a more accurate and human-friendly way. Therefore, in this paper, we for the first time investigate MIL from *only triplet comparison bags*, where a triplet (X_a, X_b, X_c) contains the weak supervision information that bag X_a is more similar to X_b than to X_c. To solve this problem, we propose to train a bag-level classifier by the *empirical risk minimization* framework and theoretically provide a generalization error bound. We also show that a convex formulation can be obtained only when specific convex binary losses such as the square loss and the double hinge loss are used. Extensive experiments validate that our proposed method significantly outperforms other baselines.

CCS Concepts: • **Computing methodologies** → **Learning settings**; **Machine learning algorithms**.

Additional Key Words and Phrases: Multi-Instance Learning, Triplet Comparison, Empirical Risk Minimization

ACM Reference Format:
Senlin Shu, Deng-Bao Wang, Suqin Yuan, Hongxin Wei, Jiuchuan Jiang, Lei Feng, and Min-Ling Zhang. 2022. Multiple-Instance Learning from Triplet Comparison Bags. 1, 1 (December 2022), 17 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

In supervised learning, a large number of fully labeled examples are required for training an effective model. However, collecting such high-quality data would be laborious and expensive in many real-world scenarios. To alleviate this issue, various weakly supervised learning problems [50] have been widely studied, including semi-supervised learning [34, 52], noisy-label learning [18, 40], partial-label learning [45], positive-unlabeled classification [31], positive-confidence

[*] Corresponding author: Lei Feng <lfengqaq@gmail.com>.

Authors' addresses: Senlin Shu, shusenlin@126.com, Chongqing University, Chongqing, China; Deng-Bao Wang, wangdb@seu.edu.cn, Southeast University, Nanjing, China; Suqin Yuan, suqinyuan.cs@gmail.com, The University of Sydney, Sydney, Australia; Hongxin Wei, weihx@sustech.edu.cn, Southern University of Science and Technology, Shenzhen, China; Jiuchuan Jiang, jcjiang@nufe.edu.cn, Nanjing University of Finance and Economics, Nanjing, China; Lei Feng, lfengqaq@gmail.com, Nanyang Technological University, Singapore; Min-Ling Zhang, zhangml@seu.edu.cn, Southeast University, Nanjing, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

Manuscript submitted to ACM

classification [21], similar-unlabeled classification [5], similarity-confidence classification [8], and unlabeled-unlabeled classification [26].

This paper focuses on an important weakly supervised learning problem called *multiple-instance learning* (MIL) [1, 9, 13, 16]. MIL also has other variant settings, such as multiple-instance regression [25, 39] and multi-instance multi-label learning [19, 49]. In this paper, we consider the conventional MIL setting for binary classification where training instances are grouped in bags, and a binary (positive or negative) label is provided for each bag. A positive bag means that at least one training instance in the bag is positive and a negative bag means that all the training instances in the bag are negative. MIL aims to learn an effective classifier from labeled bags for accurately predicting the label of any unseen test bag. Intuitively, MIL is more difficult than ordinary binary classification because the labels of the instances in each bag are unavailable. So far, MIL has been successfully applied to various real-world problems such as drug activity prediction [13], visual tracking [4], text categorization [3], and face detection [44].

Up to now, many effective MIL methods have been developed, such as EM-DD [46], MI-SVM [3], MIBoosting [43], MILES [10], miGraph [51], MIForests [23], and MI-ODM [47]. Although these methods work well, all of them require fully labeled bags for learning an effective classifier. Unfortunately, it could be quite hard for us to collect such perfect MIL datasets consisted of fully labeled bags in real-world scenarios. For example, a molecule (bag) may contain many low-energy shapes (instances), and whether the molecule has some special shapes decides the label of the molecule that indicates whether the molecule can be used to make the drug. In this example, it could be laborious and expensive for human experts to accurately figure out all the correct bag labels of all the molecules. On the other hand, humans generally perform quite well in assessing the similarity on a relative scale (i.e., instance A is more similar to instance B than to instance C). In image annotation, it was also shown [42] that keeping only the relative comparison information can help an algorithm be resilient against measurement errors and achieve high accuracy. Another example is text categorization [49], where a document usually contains multiple sections each of which can be represented as an instance, and the document can be regarded as belonging to different categories if it is viewed from different aspects. In this example, it could also be laborious and expensive for annotators to accurately figure out all the correct bag labels of all the documents. Fortunately, annotators could easily distinguish which two documents in a triplet of documents are more similar from multiple perspectives. Additionally, the task of identifying proteins [37] is a further example, where the primary sequence of each protein could be represented as a bag that contains multiple amino acid sequences. It could be difficult for medical scientists to accurately distinguish identify whether a protein belongs to a certain protein super-family due to complex and diverse amino acid sequences. Fortunately, it could be easy to distinguish which two proteins in a triplet of proteins are more similar since the similarity of protein structure could be measured by comparing different proteins. Inspired by the above examples, one may ask

> *Whether it is possible for us to successfully conduct multiple-instance learning using only triplet comparison information?*

This paper for the first time provides an affirmative answer to this question. Specifically, we focus on learning an effective bag-level binary classifier from only *triplet comparison bags*, where a triplet (X_a, X_b, X_c) contains the weak supervision information that bag X_a is more similar to X_b than to X_c. This learning problem can be encountered in many real-world scenarios where the relative similarity information of bags could be obtained. For example, in image annotation, an image represents a bag that contains a set of objects (instances). It could be difficult for us to obtain the label of each object in the image so that we cannot easily collect the bag label of a specific image. However, it would be much easier for us to obtain the relative similarity information of bags, thanks to the emergence of image search engines. Specifically, when we enquiry an image X_a from the image search engine, we can obtain an ordered list of images (including image X_b and image X_c) from the image search engine where the order is determined by the relative similarity. In this way, we can collect a triplet (X_a, X_b, X_c), which has the meaning that bag X_a is more similar to X_b than to X_c. Therefore, as justified by the above example, our studied learning problem is practically significant in reality.

Our main contributions can be summarized as follows:

- We for the first time investigate MIL from only *triplet comparison bags.* To solve this new MIL problem, we propose to learn a bag-level classifier by the *empirical risk minimization* framework and theoretically provide a generalization error bound.
- We also show that a convex formulation can be obtained only when specific convex binary losses such as the square loss and the double hinge loss are used.
- Extensive experiments demonstrate that our proposed method significantly outperforms other baselines.

The rest of this paper is organized as follows. Section 2 introduces related studies and preliminary knowledge. Section 3 presents the technical details and theoretical analysis of our proposed method. Section 4 reports the experimental results of comparative studies. Section 5 concludes this paper.

2 BACKGROUND

In this section, we introduce related studies and preliminary knowledge. As this paper focuses on learning a bag-level binary classifier from triplet comparison bags, two of the existing weakly supervised learning problems are highly related to our work, i.e., *multiple-instance learning* [9] and *triplet comparison classification* [12], where triplet comparison classification aims to learn an instance-level binary classifier from only triplet comparison data. In what follows, we will introduce ordinary binary classification, triplet comparison classification, and multiple-instance learning.

2.1 Ordinary Binary Classification

Let the feature space be $\mathcal{X} \in \mathbb{R}^d$ (with d dimensions) and the label space be $\mathcal{Y} = \{-1, +1\}$. Let $(\boldsymbol{x}, y)$ be an example composed of an instance $\boldsymbol{x}$ and a label y, and it is generally assumed that each training example $(\boldsymbol{x}, y)$ is independently sampled from an unknown data distribution with probability density $p(\boldsymbol{x}, y)$. The goal of ordinary binary classification is to construct an instance-level binary classifier by minimizing the (expected) classification risk

$$R(f) = \mathbb{E}_{p(\boldsymbol{x},y)}\left[\ell(f(\boldsymbol{x}), y)\right],$$

where $\mathbb{E}_{p(\boldsymbol{x},y)}[\cdot]$ denotes the expectation over $p(\boldsymbol{x}, y)$ and $\ell : \mathbb{R} \times \mathcal{Y} \mapsto \mathbb{R}_+$ denotes a binary loss. As $p(\boldsymbol{x}, y)$ is unknown and we only have a limited number of training examples $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ that are independently drawn from $p(\boldsymbol{x}, y)$, a common strategy is to minimize the empirical risk $\widehat{R}(f) := \frac{1}{n}\sum_{i=1}^n \ell(f(\boldsymbol{x}_i), y_i)$, which is called *empirical risk minimization.* As can be easily verified, $\mathbb{E}_{p(\boldsymbol{x},y)}[\widehat{R}(f)] = R(f)$. In this case, we call $\widehat{R}(f)$ an unbiased estimator of the classification risk $R(f)$ (*unbiased risk estimator* in short).

2.2 Triplet Comparison Classification

Triplet comparison classification [12] is an interesting weakly supervised binary classification problem proposed recently, which aims to train an instance-level binary classifier from triplet comparison examples. A given triplet comparison example $(\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{x}_c)$ indicates that instance $\boldsymbol{x}_a$ is more similar to $\boldsymbol{x}_b$ than to $\boldsymbol{x}_c$. To solve this triplet comparison classification problem, the pioneering work [12] assumes that the generation process of three examples in a triplet is independent and derives an unbiased risk estimator accordingly. The given triplet comparison datasets were assumed to be decomposed into three pointwise sets $\{(\boldsymbol{x}_i^1)\}_{i=1}^{n_1}$, $\{(\boldsymbol{x}_i^2)\}_{i=1}^{n_2}$, and $\{(\boldsymbol{x}_i^3)\}_{i=1}^{n_3}$. The marginal densities of the three sets can be expressed as $p_1(\boldsymbol{x})$, $p_2(\boldsymbol{x})$, and $p_3(\boldsymbol{x})$ respectively. In this way, we have the following relationships among these densities:

$$\begin{bmatrix} p_1(\boldsymbol{x}) \\ p_2(\boldsymbol{x}) \\ p_3(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} \pi_+ & \pi_- \\ A & B \\ \pi_- & \pi_+ \end{bmatrix} \begin{bmatrix} p_+(\boldsymbol{x}) \\ p_-(\boldsymbol{x}) \end{bmatrix}, \tag{1}$$

where $\pi_+ = p(y = +1)$, $\pi_- = p(y = -1)$, $p_+(\boldsymbol{x}) = p(\boldsymbol{x}|y = +1)$, $p_-(\boldsymbol{x}) = p(\boldsymbol{x}|y = -1)$, $A = (\pi_+^3 + 2\pi_+^2\pi_-)/(1 - \pi_+\pi_-)$, and $B = (2\pi_+\pi_-^2 + \pi_-^3)/(1 - \pi_+\pi_-)$. Based on the generation process of triplet comparison data in Eq. (1), Cui et al. [12] showed that the following proposition holds.

Proposition 1 (Theorem 2 in Cui et al. [12]). *The classification risk $R(f)$ can be equivalently represented as*

$$R(f) = \mathbb{E}_{\boldsymbol{x}^1 \sim p_1(\boldsymbol{x})}\left[\lambda_1 \ell_+(f(\boldsymbol{x}^1)) + \lambda_2 \ell_-(f(\boldsymbol{x}^1))\right] + \mathbb{E}_{\boldsymbol{x}^2 \sim p_2(\boldsymbol{x})}\left[\lambda_3 \ell_+(f(\boldsymbol{x}^2)) + \lambda_4 \ell_-(f(\boldsymbol{x}^2))\right] + \mathbb{E}_{\boldsymbol{x}^3 \sim p_3(\boldsymbol{x})}\left[\lambda_5 \ell_+(f(\boldsymbol{x}^3)) + \lambda_6 \ell_-(f(\boldsymbol{x}^3))\right],$$

where

$$\lambda_1 = \frac{\pi_+(c\pi_+ - b\pi_-)}{ac - b^2}, \quad \lambda_2 = \frac{\pi_-(a\pi_- - b\pi_+)}{ac - b^2}, \quad \lambda_3 = \frac{\pi_+(cA - bB)}{ac - b^2},$$
$$\lambda_4 = \frac{\pi_-(aB - bA)}{ac - b^2}, \quad \lambda_5 = \frac{\pi_+(c\pi_- - b\pi_+)}{ac - b^2}, \quad \lambda_6 = \frac{\pi_-(a\pi_+ - b\pi_-)}{ac - b^2},$$

and $a = \pi_+^2 + A^2 + \pi_-^2$, $b = 2\pi_+\pi_- + AB$, $c = \pi_-^2 + B^2 + \pi_+^2$, $\ell_+(f(\boldsymbol{x})) = \ell(f(\boldsymbol{x}), y = +1)$, *and* $\ell_-(f(\boldsymbol{x})) = \ell(f(\boldsymbol{x}), y = -1)$.

As this proposition indicates, we can recover the classification risk $R(f)$ using only triplet comparison data. This implies that we can learn an instance-level binary classifier by minimizing the empirical approximation of the above unbiased risk estimator.

2.3 Multiple-Instance Learning

In this paper, we focus on the conventional MIL with binary labels, let $\{(X_i, Y_i)\}_{i=1}^n$ be the MIL training set with n bags, where $X_i = \{\boldsymbol{x}_{i1}, \ldots, \boldsymbol{x}_{ij}, \ldots, \boldsymbol{x}_{ib_i}\}$ is a bag with $\boldsymbol{x}_{ij} \in \mathcal{X}$ representing the j-th instance in the i-th bag and b_i denotes the number of instances in the bag X_i. If X_i contains at least one positive instance, then X_i is a positive bag (i.e., $Y_i = +1$), otherwise X_i is a negative bag (i.e., $Y_i = -1$). In this way, MIL aims to learn a bag-level binary classifier for accurately predicting the label of any test bag. To achieve this goal, we need to design a function that inputs a bag (a set of instances) and outputs a real value. Note that we can also achieve this goal by using instance-level methods that predict the label of each instance in a bag to determine the label of the bag. However, the instance-level methods are usually worse than the bag-level methods as shown in previous work [9]. Thus, some researchers improve the performance of the instance-level methods by using various techniques such as instance selection [24] and attention mechanism [20]. In this paper, we aim to learn a bag-level linear-in-parameter classifier with a specially designed kernel for MIL: $g(X) = \boldsymbol{w}^\top \boldsymbol{\phi}(X)$, where $\boldsymbol{w} \in \mathbb{R}^d$ is a learning parameter and $\boldsymbol{\phi}(X) \in \mathbb{R}^d$ is a vector of basis functions that transforms a bag into a feature vector. It is noteworthy that if we set $\widetilde{w} = [\boldsymbol{w}\ b]^\top$ and $\widetilde{\boldsymbol{\phi}}(X) = [\boldsymbol{\phi}(X)\ 1]^\top$, we can recover $g(X) = \boldsymbol{w}^\top \boldsymbol{\phi}(X) + b$. In this paper, we construct $\boldsymbol{\phi}(X)$ by the statistical kernel [17] associated with the minimax statistic and polynomial kernel, following the previous studies [6, 15].

3 MIL FROM TRIPLET COMPARISON BAGS

In this section, we first introduce the generation process of triplet comparison bags and then propose an empirical risk minimization formulation for MIL from triple comparison bags, finally provide a theoretical guarantee by a generalization error bound.

3.1 Generation of Triplet Comparison Bags

Following Cui et al. [12], we adopt an analogous generation process of triplet comparison bags. We assume that three bags in a triplet are sampling independently and the collected training set composed of triplet comparison bags can be decomposed into three pointwise sets $\{(X_i^1)\}_{i=1}^{n_1}$, $\{(X_i^2)\}_{i=1}^{n_2}$, and $\{(X_i^3)\}_{i=1}^{n_3}$. The marginal densities of the three sets can be expressed as $p_1(X)$, $p_2(X)$, and $p_3(X)$ respectively. In this way, we have the following relationships among these densities:

$$\begin{bmatrix} p_1(X) \\ p_2(X) \\ p_3(X) \end{bmatrix} = \begin{bmatrix} \theta_+ & \theta_- \\ C & D \\ \theta_- & \theta_+ \end{bmatrix} \begin{bmatrix} p_+(X) \\ p_-(X) \end{bmatrix}, \tag{2}$$

where $\theta_+ = p(Y = +1)$, $\theta_- = p(Y = -1)$, $p_+(X) = p(X|Y = +1)$, $p_-(X) = p(X|Y = -1)$, $C = (\theta_+^3 + 2\theta_+^2\theta_-)/(1 - \theta_+\theta_-)$, and $D = (2\theta_+\theta_-^2 + \theta_-^3)/(1 - \theta_+\theta_-)$. As the generation process is quite similar to that of Cui et al. [12], we provided all the details of this generation process in Appendix A. Given the generation process of triplet comparison bags, we have the following learning method.

3.2 Formulation

Motivated by Proposition 1, we propose to learn a bag-level classifier by minimizing the following empirical risk:

$$\begin{aligned} \widehat{R}_{\mathrm{Trip}}(g) = \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\left(\lambda_1\ell_+(g(X_i^1)) + \lambda_2\ell_-(g(X_i^1))\right) + \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\left(\lambda_3\ell_+(g(X_i^2)) + \lambda_4\ell_-(g(X_i^2))\right) \\ + \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\left(\lambda_5\ell_+(g(X_i^3)) + \lambda_6\ell_-(g(X_i^3))\right), \end{aligned} \tag{3}$$

where

$$\lambda_1 = \frac{\theta_+(c\theta_+ - b\theta_-)}{ac - b^2}, \quad \lambda_2 = \frac{\theta_-(a\theta_- - b\theta_+)}{ac - b^2}, \quad \lambda_3 = \frac{\theta_+(cC - bD)}{ac - b^2},$$
$$\lambda_4 = \frac{\theta_-(aD - bC)}{ac - b^2}, \quad \lambda_5 = \frac{\theta_+(c\theta_- - b\theta_+)}{ac - b^2}, \quad \lambda_6 = \frac{\theta_-(a\theta_+ - b\theta_-)}{ac - b^2},$$

and $a = \theta_+^2 + C^2 + \theta_-^2$, $b = 2\theta_+\theta_- + CD$, $c = \theta_-^2 + D^2 + \theta_+^2$. It is worth noting that when $\theta_+ > 0.5$, we have $\lambda_2 < 0$, $\lambda_4 < 0$, and $\lambda_5 < 0$. This implies that minimizing Eq. (3) may not be a convex problem even if a convex binary loss (e.g., the hinge loss) is used, which could make the problem difficult to solve. Therefore, we need to think about how we can make the problem convex so that we can easily solve this problem. Fortunately, as showed by previous studies [5, 15, 36], if the used binary loss satisfies the condition $\ell_+(g(X)) - \ell_-(g(X)) = -g(X)$, Eq. (3) becomes a convex objective function, and thus minimizing Eq. (3) is a convex problem. When we choose ℓ that satisfies the above condition, $\widehat{R}_{\mathrm{Trip}}(g)$ can be equivalently expressed as

$$\widehat{R}_{\mathrm{Trip}}(g) = \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\big((\lambda_1+\lambda_2)\ell_{+}(g(X_i^1)) + \lambda_2 g(X_i^1)\big) + \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\big((\lambda_3+\lambda_4)\ell_{+}(g(X_i^2)) + \lambda_4 g(X_i^2)\big) + \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\big((\lambda_5+\lambda_6)\ell_{-}(g(X_i^3)) - \lambda_5 g(X_i^3)\big). \tag{4}$$

Here, because only triplet comparison bags are available, for the vector of basis function $\boldsymbol{\phi}$ (from $g(X) = \boldsymbol{w}^\top\boldsymbol{\phi}(X)$), we have $\boldsymbol{\phi}(X) \in \mathbb{R}^d$ where $d = n_1 + n_2 + n_3$.

Now we need to consider a convex binary loss ℓ used in Eq. (4), which satisfies the condition $\ell_{+}(g(X)) - \ell_{-}(g(X)) = -g(X)$ for practical implementation. In this paper, we consider the square loss and the double hinge loss [14].

3.3 Practical Implementation

Let us first introduce the following symbols for convenience:

$$\boldsymbol{X}^1 = [\tilde{\boldsymbol{x}}_1^1, \ldots, \tilde{\boldsymbol{x}}_i^1, \ldots, \tilde{\boldsymbol{x}}_{n_1}^1]^\top \in \mathbb{R}^{n_1\times d},$$
$$\boldsymbol{X}^2 = [\tilde{\boldsymbol{x}}_1^2, \ldots, \tilde{\boldsymbol{x}}_i^2, \ldots, \tilde{\boldsymbol{x}}_{n_2}^2]^\top \in \mathbb{R}^{n_2\times d},$$
$$\boldsymbol{X}^3 = [\tilde{\boldsymbol{x}}_1^3, \ldots, \tilde{\boldsymbol{x}}_i^3, \ldots, \tilde{\boldsymbol{x}}_{n_3}^2]^\top \in \mathbb{R}^{n_3\times d},$$

where $\tilde{\boldsymbol{x}}_i^1 = \phi(X_i^1)$, $\tilde{\boldsymbol{x}}_i^2 = \phi(X_i^2)$, and $\tilde{\boldsymbol{x}}_i^3 = \phi(X_i^3)$. Then, we can insert the square loss and the double hinge loss into Eq. (4) for practical implementation. We also adopt the widely used L_2 regularization to restore stability and ensure generalization. In what follows, we present the technical details of the solution when we use the square loss and the double hinge loss in Eq. (4).

Square Loss. By inserting the square loss $\ell_{\mathrm{SQ}}(z,t) = \frac{1}{4}(tz-1)^2$ into Eq. (4), we have the following objective function:

$$\begin{aligned} J_{\mathrm{SQ}}(\boldsymbol{w}) &= \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\left[\frac{\lambda_1+\lambda_2}{4}(\tilde{\boldsymbol{x}}_i^{1\top}\boldsymbol{w}-1)^2 + \lambda_2\tilde{\boldsymbol{x}}_i^{1\top}\boldsymbol{w}\right] + \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\left[\frac{\lambda_3+\lambda_4}{4}(\tilde{\boldsymbol{x}}_i^{2\top}\boldsymbol{w}-1)^2 + \lambda_4\tilde{\boldsymbol{x}}_i^{2\top}\boldsymbol{w}\right] \\ &\quad + \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\left[\frac{\lambda_5+\lambda_6}{4}(-\tilde{\boldsymbol{x}}_i^{3\top}\boldsymbol{w}-1)^2 - \lambda_5\tilde{\boldsymbol{x}}_i^{3\top}\boldsymbol{w}\right] + \frac{\gamma}{2}\|\boldsymbol{w}\|_2^2 \\ &= \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\left[\frac{\lambda_1+\lambda_2}{4}(\boldsymbol{w}^\top\tilde{\boldsymbol{x}}_i^1\tilde{\boldsymbol{x}}_i^{1\top}\boldsymbol{w}+1) - \frac{\lambda_1-\lambda_2}{2}\tilde{\boldsymbol{x}}_i^{1\top}\boldsymbol{w}\right] + \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\left[\frac{\lambda_3+\lambda_4}{4}(\boldsymbol{w}^\top\tilde{\boldsymbol{x}}_i^2\tilde{\boldsymbol{x}}_i^{2\top}\boldsymbol{w}+1) - \frac{\lambda_3-\lambda_4}{2}\tilde{\boldsymbol{x}}_i^{2\top}\boldsymbol{w}\right] \\ &\quad + \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\left[\frac{\lambda_5+\lambda_6}{4}(\boldsymbol{w}^\top\tilde{\boldsymbol{x}}_i^3\tilde{\boldsymbol{x}}_i^{3\top}\boldsymbol{w}+1) - \frac{\lambda_5-\lambda_6}{2}\tilde{\boldsymbol{x}}_i^{3\top}\boldsymbol{w}\right] + \frac{\gamma}{2}\|\boldsymbol{w}\|_2^2 \\ &= \boldsymbol{w}^\top(\frac{\gamma}{2}I_{d\times d} + \frac{\lambda_1+\lambda_2}{4n_1}\boldsymbol{X}^{1\top}\boldsymbol{X}^1 + \frac{\lambda_3+\lambda_4}{4n_2}\boldsymbol{X}^{2\top}\boldsymbol{X}^2 + \frac{\lambda_5+\lambda_6}{4n_3}\boldsymbol{X}^{3\top}\boldsymbol{X}^3)\boldsymbol{w} - (\frac{\lambda_1-\lambda_2}{2n_1}\boldsymbol{1}_{n_1}^\top\boldsymbol{X}^1 + \frac{\lambda_3-\lambda_4}{2n_2}\boldsymbol{1}_{n_2}^\top\boldsymbol{X}^2 \\ &\quad + \frac{\lambda_5-\lambda_6}{2n_3}\boldsymbol{1}_{n_3}^\top\boldsymbol{X}^3)\boldsymbol{w} + \text{constant}. \end{aligned}$$

By setting the derivative with respect to $\boldsymbol{w}$ to zero, we can obtain the following analytical solution:

$$\boldsymbol{w} = \Big(\gamma\boldsymbol{I}_{d\times d} + \frac{\lambda_1+\lambda_2}{2n_1}\boldsymbol{X}^{1\top}\boldsymbol{X}^1 + \frac{\lambda_3+\lambda_4}{2n_2}\boldsymbol{X}^{2\top}\boldsymbol{X}^2 + \frac{\lambda_5+\lambda_6}{2n_3}\boldsymbol{X}^{3\top}\boldsymbol{X}^3\Big)^{-1}\Big(\frac{\lambda_1-\lambda_2}{2n_3}\boldsymbol{X}^{1\top}\boldsymbol{1}_{n_1} + \frac{\lambda_3-\lambda_4}{2n_2}\boldsymbol{X}^{2\top}\boldsymbol{1}_{n_2} + \frac{\lambda_5-\lambda_6}{2n_3}\boldsymbol{X}^{3\top}1_{n_3}\Big), \tag{5}$$

where $\boldsymbol{I}_{d\times d}$ denotes the $d\times d$ identity matrix and $\boldsymbol{1}_{n_1}$ denotes the $n_1 \times 1$ vector whose elements are all ones.

Double Hinge Loss. By inserting the double hinge loss $\ell_{\mathrm{DH}}(z,t) = \max(-tz, \max(0, \frac{1}{2} - \frac{1}{2}tz))$ into Eq. (4), we have the following objective function:

$$
\begin{aligned}
J_{\mathrm{DH}}(\boldsymbol{w}) = & \frac{\gamma}{2}\|\boldsymbol{w}\|_2^2 + \left[\frac{\lambda_1+\lambda_2}{n_1}\mathbf{1}_{n_1}^\top \boldsymbol{\xi} + \frac{\lambda_2}{n_1}\mathbf{1}_{n_1}^\top \boldsymbol{X}^1\boldsymbol{w}\right] + \left[\frac{\lambda_3+\lambda_4}{n_2}\mathbf{1}_{n_2}^\top \boldsymbol{\eta} + \frac{\lambda_4}{n_2}\mathbf{1}_{n_2}^\top \boldsymbol{X}^2\boldsymbol{w}\right] + \left[\frac{\lambda_5+\lambda_6}{n_3}\mathbf{1}_{n_3}^\top \boldsymbol{\zeta} - \frac{\lambda_5}{n_3}\mathbf{1}_{n_3}^\top \boldsymbol{X}^3\boldsymbol{w}\right] \\
\text{s.t.} \quad & \boldsymbol{\xi} \geq \mathbf{0}_{n_1}, \quad \boldsymbol{\xi} \geq \frac{1}{2}(\mathbf{1}_{n_1} - \boldsymbol{X}^1\boldsymbol{w}), \quad \boldsymbol{\xi} \geq -\boldsymbol{X}^1\boldsymbol{w}, \\
& \boldsymbol{\eta} \geq \mathbf{0}_{n_2}, \quad \boldsymbol{\eta} \geq \frac{1}{2}(\mathbf{1}_{n_2} - \boldsymbol{X}^2\boldsymbol{w}), \quad \boldsymbol{\eta} \geq -\boldsymbol{X}^2\boldsymbol{w}, \\
& \boldsymbol{\zeta} \geq \mathbf{0}_{n_3}, \quad \boldsymbol{\zeta} \geq \frac{1}{2}(\mathbf{1}_{n_3} + \boldsymbol{X}^3\boldsymbol{w}), \quad \boldsymbol{\zeta} \geq \boldsymbol{X}^3\boldsymbol{w},
\end{aligned}
$$

where $\geq$ for vectors means the element-wise inequality. Below, we rewrite the above optimization problem into a standard quadratic programming form. We denote $\boldsymbol{\alpha} = [\boldsymbol{w}^\top \boldsymbol{\xi}^\top \boldsymbol{\eta}^\top \boldsymbol{\zeta}^\top] \in \mathbb{R}^{(d+n_1+n_2+n_3)}$ as a new variable and we also introduce the following notations:

$$
P = \begin{bmatrix} \gamma I_{d\times d} & \mathbf{0}_{d\times n_1} & \mathbf{0}_{d\times n_2} & \mathbf{0}_{d\times n_3} \\ \mathbf{0}_{n_1\times d} & \mathbf{0}_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \mathbf{0}_{n_1\times n_3} \\ \mathbf{0}_{n_2\times d} & \mathbf{0}_{n_2\times n_1} & \mathbf{0}_{n_2\times n_2} & \mathbf{0}_{n_2\times n_3} \\ \mathbf{0}_{n_3\times d} & \mathbf{0}_{n_3\times n_1} & \mathbf{0}_{n_3\times n_2} & \mathbf{0}_{n_3\times n_3} \end{bmatrix}, \quad \boldsymbol{q} = \begin{bmatrix} \frac{\lambda_2}{n_1}\boldsymbol{X}^{1\top}\mathbf{1}_{n_1} + \frac{\lambda_4}{n_2}\boldsymbol{X}^{2\top}\mathbf{1}_{n_2} - \frac{\lambda_5}{n_3}\boldsymbol{X}^{3\top}\mathbf{1}_{n_3} \\ \frac{\lambda_1+\lambda_2}{n_1}\mathbf{1}_{n_1} \\ \frac{\lambda_3+\lambda_4}{n_2}\mathbf{1}_{n_2} \\ \frac{\lambda_5+\lambda_6}{n_3}\mathbf{1}_{n_3} \end{bmatrix},
$$

$$
G = \begin{bmatrix} \mathbf{0}_{n_1\times d} & -I_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \mathbf{0}_{n_1\times n_3} \\ -\frac{1}{2}\boldsymbol{X}^1 & -I_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \mathbf{0}_{n_1\times n_3} \\ -\boldsymbol{X}^1 & -I_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \mathbf{0}_{n_1\times n_3} \\ \mathbf{0}_{n_2\times d} & \mathbf{0}_{n_2\times n_1} & -I_{n_2\times n_2} & \mathbf{0}_{n_2\times n_3} \\ -\frac{1}{2}\boldsymbol{X}^2 & \mathbf{0}_{n_2\times n_1} & -I_{n_2\times n_2} & \mathbf{0}_{n_1\times n_3} \\ -\boldsymbol{X}^2 & \mathbf{0}_{n_2\times n_1} & -I_{n_2\times n_2} & \mathbf{0}_{n_1\times n_3} \\ \mathbf{0}_{n_3\times d} & \mathbf{0}_{n_3\times n_1} & \mathbf{0}_{n_3\times n_2} & -I_{n_3\times n_3} \\ \frac{1}{2}\boldsymbol{X}^3 & \mathbf{0}_{n_3\times n_1} & \mathbf{0}_{n_3\times n_2} & -I_{n_3\times n_3} \\ \boldsymbol{X}^3 & \mathbf{0}_{n_3\times n_1} & \mathbf{0}_{n_3\times n_2} & -I_{n_3\times n_3} \end{bmatrix}, \quad \boldsymbol{h} = \begin{bmatrix} \mathbf{0}_{n_1} \\ -\frac{1}{2}\mathbf{1}_{n_1} \\ \mathbf{0}_{n_1} \\ \mathbf{0}_{n_2} \\ -\frac{1}{2}\mathbf{1}_{n_2} \\ \mathbf{0}_{n_2} \\ \mathbf{0}_{n_3} \\ -\frac{1}{2}\mathbf{1}_{n_3} \\ \mathbf{0}_{n_3} \end{bmatrix},
$$

Then, the optimization objective becomes:

$$
\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top P\boldsymbol{\alpha} + \boldsymbol{q}^\top\boldsymbol{\alpha} \quad \text{s.t.} \quad G\boldsymbol{\alpha} \leq \boldsymbol{h}, \tag{6}
$$

which is the standard quadratic programming form and can be easily solved by any off-the-shelf quadratic programming toolbox.

3.4 Generalization Error Bound

Here, we theoretically provide a generalization error bound for our proposed formulation. Let $\mathscr{X}$ be the bag-level domain set and $\mathcal{G} = \{g(X) = \boldsymbol{w}^\top\boldsymbol{\phi}(X)$ be a function class with $\|\boldsymbol{w}\| \leq C_{\boldsymbol{w}}$ and $\sup_{X\in\mathscr{X}} \|\boldsymbol{\phi}(X)\| \leq C_{\boldsymbol{\phi}}\}$. In our analysis, we simply adopt the double hinge loss as the used loss function ℓ because it is 1-Lipschitz, and this loss function is also used in our experiments. In contrast to the empirical risk $\widehat{R}_{\mathrm{Trip}}(g)$, we denote the expected risk of a bag-level classifier

g (with triplet comparison bags) as

$$R_{\mathrm{Trip}}(g) = \mathbb{E}_{X^1 \sim p_1(X)}\left[\lambda_1 \ell_+(g(X^1)) + \lambda_2 \ell_-(g(X_i^1))\right] + \mathbb{E}_{X^2 \sim p_2(X)}\left[\lambda_3 \ell_+(g(X^2)) + \lambda_4 \ell_-(g(X^2))\right] + \mathbb{E}_{X^3 \sim p_3(X)}\left[\lambda_5 \ell_+(g(X^3)) + \lambda_6 \ell_-(g(X^3))\right].$$

Then we analyze the generalization error bound based on the widely used *Rademacher complexity* [7].

Definition 1. *Let n be a positive integer, $X_1, \ldots, X_n$ be independent and identically distributed random variables drawn from a probability distribution with density μ, $\mathcal{G} = g : \mathcal{X} \mapsto \mathbb{R}$ be a class of measurable functions, and $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ be Rademacher variables that take value only from $\{+1, -1\}$ with even probabilities. Then, the (expected) Rademacher complexity of $\mathcal{G}$ is defined as*

$$\mathfrak{R}_n(\mathcal{G}) := \mathbb{E}_{X_1,\ldots,X_n \overset{\text{i.i.d.}}{\sim} \mu} \mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup\nolimits_{g \in \mathcal{G}} \frac{1}{n} \sum\nolimits_{i=1}^{n} \sigma_i h(X_i)\Big].$$

For the function class $\mathcal{G}$ and any probability density μ, $\mathfrak{R}_n(\mathcal{G})$ can be normally bounded by $\mathfrak{R}_n(\mathcal{G}) \leq C_{\mathcal{G}}/\sqrt{n}$, where $C_{\mathcal{G}}$ is a positive constant. This condition holds for many model classes including the used model class $\mathcal{G} = \{g(X) = \boldsymbol{w}^\top \boldsymbol{\phi}(X)\}$.

Theorem 1. *With the introduced definitions and conditions above, for any $\delta > 0$, with probability at least $1 - \delta$, we have the following generalization error bound:*

$$\sup\nolimits_{g \in \mathcal{G}} \left| R_{\mathrm{Trip}}(g) - \widehat{R}_{\mathrm{Trip}}(g) \right| \leq C(2C_{\mathcal{G}} + C_{\boldsymbol{w}} C_{\boldsymbol{\phi}} \sqrt{\frac{\log \frac{6}{\delta}}{2}}),$$

where

$$C = \frac{|\lambda_1| + |\lambda_2|}{\sqrt{n_1}} + \frac{|\lambda_3| + |\lambda_4|}{\sqrt{n_2}} + \frac{|\lambda_5| + |\lambda_6|}{\sqrt{n_3}}.$$

The proof is provided in Appendix B. This theorem shows that our proposed formulation is consistent, i.e., $\widehat{R}_{\mathrm{Trip}}(g) \longrightarrow R_{\mathrm{Trip}}(g)$ as $n_1 \longrightarrow \infty$, $n_2 \longrightarrow \infty$, and $n_3 \longrightarrow \infty$. Therefore, it is clear that increasing the number of triplet bags can decrease the generalization error. In addition, the convergence rate of our proposed formulation is $\mathcal{O}(1/\sqrt{n_1} + 1/\sqrt{n_2} + 1/\sqrt{n_3})$, where $\mathcal{O}$ denotes the order in probability. It is also noteworthy that this order is the optimal parametric rate for empirical risk minimization without additional assumptions [29].

4 EXPERIMENTS

In this section, we evaluate our proposed method by extensive experiments on both benchmark datasets and text categorization datasets. The compared methods in the experiments are listed as follows:

- **TC-SQ** (ours): An ERM-based method proposed in this paper using the square loss for MIL from triplet comparison bags. The closed-form solution is reported in Eq. (5).
- **TC-DH** (ours): Another ERM-based method proposed in this paper using the double hinge loss for MIL from triplet comparison bags. For TC-DH, we solve the standard quadratic programming problem in Eq. (6) using the off-the-shelf optimization toolbox CVXOPT [2].
- **SD-SQ** & **SD-DH** [15]: Two convex learning formulations for MIL from similar and dissimilar bags. SD-SQ employs the square loss and SD-DH employs the double hinge loss. For a triplet comparison bag (X_a, X_b, X_c), we can regard (X_a, X_b) as a similar pair and (X_a, X_c) as a dissimilar pair, since X_a is more similar to X_b than to X_c.

- **KM** [27]: The k-means clustering method with $k = 2$ on all the triplet comparison bags while ignoring all the comparison information.
- **CKM** [38]: The constrained k-means clustering method with $k = 2$. CKM uses pairwise similar (dissimilar) information as must-link (cannot-link) constraints. We extract the pairwise relationship in the same manner as adopted by SD-SQ and SD-DH.
- **TL** [35]: This is a *triplet loss* proposed to learn a metric directly from triplet comparison bags. Using the learned metric, we conduct k-means clustering on test bags.

For all the above methods, we employ $\boldsymbol{\phi}(X)$ to transform each bag into a feature vector so that we can directly employ KM, CKM, and TL as baselines for MIL from triplet comparison bags. Note that the degree of the polynomial kernel is simply fixed at 1. For TC-SQ, TC-DH, SD-SQ, and SD-DH, the regularization parameter γ is selected from $\{10^0, 10^1, \ldots, 10^6\}$. For TL, the number of training epochs is set to 200 with full batch size, the learning rate is set to 10^{-3}, the weight decay is selected from $\{10^{-3}, 10^{-2}, 10^{-1}\}$, the embedding dimension is set to 128, and the parameter α is selected from $\{10^0, 10^1, 10^2\}$. For KM and CKM, performances are measured by the clustering accuracy $1 - \min(r, 1 - r)$ where r is the error rate.

In the training process, we do not need to know the value of the bag-level class prior θ_+ in advance, since we are able to empirically estimate θ_+ according to our introduced data generation process of triplet bags. Specifically, we can exactly obtain the equation $n_2/(n_2 + n_3) = 1 - \theta_+(1 - \theta_+)$ following Cui et al. [12]. Then, we also assume that the positive class prior should be larger than the negative class prior, i.e. $\theta_+ > 0.5$. Note that if $\theta_+ < 0.5$, we can switch the role of positive class and negative class. Thus, the bag-level positive class prior can be estimated as

$$\theta_+ = \frac{1 + \sqrt{1 - 4(1 - \frac{n_2}{n_2+n_3})}}{2}.$$

Since θ_+ is assumed to be larger than 0.5, we select θ_+ only from $\{0.6, 0.7, 0.8\}$ for performance evaluation under different bag-level class priors. For the used benchmark datasets and text categorization datasets, we sample 600 bags for $\theta_+ = 0.6$, 540 bags for $\theta_+ = 0.7$, and 480 bags for $\theta_+ = 0.8$, following the introduced generation process of triplet comparison bags. For image datasets, we sample 1500 bags for training for all different θ_+ in $\{0.6, 0.7, 0.8\}$. We repeat the sampling-and-training process 10 times and record mean classification accuracy with standard deviation.

4.1 Experiments on Benchmark Datasets

We first conduct performance evaluation on five benchmark datasets, including `Musk1`, `Musk2`, `Elephant`, `Fox`, and `Tiger`[1]. For these datasets, `Musk1` contains 47 positive bags and 45 negative bags. `Musk2` contains 39 positive bags and 63 negative bags. The other three datasets contain 100 positive bags and 100 negative bags. Obviously, these datasets are not large enough to be used for generating the triplet comparison bags, since a triplet comparison bag contains three common bags. Therefore, it is needed to augment datasets and we follow Bao et al. [6] to do data augmentation for increasing the number of training bags. Specifically, we first make copies of bags chosen randomly from the original datasets and then add Gaussian noise with mean zero and variance 0.01 to each dimension of instances in the copied bags. In this way, the number of bags in `Musk1` and `Musk2` is expanded to 10 times and the number of bags in `Fox`, `Elephant`, and `Tiger` is expanded to 5 times. Table 1 reports the characteristics of these benchmark datasets after preprocessing. Table 2 shows the classification accuracy of each method on these benchmark datasets. As shown in

[1] http://www.cs.columbia.edu/~andrews/mil/datasets.html

Table 1. The characteristics of the used benchmark datasets. The last two columns indicate the average number (mean±std) of positive and negative instances per bag.

Dataset	# Features	# Positive bags	# Negative bags	# Avg. Pos. Ins. per bag	# Avg. Neg. Ins. per bag
Musk1	166	475	445	2.2±2.5	2.9±7.0
Musk2	166	413	607	8.9±22.7	49.9±169.7
Elephat	230	504	496	3.9±4.2	3.2±3.6
Fox	230	498	502	3.2±3.6	3.4±3.8
Tiger	230	506	494	2.8±3.1	3.4±3.9

Table 2. Classification accuracy on the benchmark datasets. The best performance is highlighted in bold.

Datasets	θ_+	Our Proposed		Baselines				
		TC-SQ	TC-DH	SD-SQ	SD-DH	TL	KM	CKM
Musk1	0.6	**0.710** **(0.095)**	0.696 (0.078)	0.627 (0.087)	0.626 (0.091)	0.545 (0.027)	0.528 (0.034)	0.533 (0.033)
	0.7	**0.844** **(0.032)**	0.817 (0.047)	0.751 (0.053)	0.746 (0.034)	0.600 (0.041)	0.563 (0.049)	0.560 (0.037)
	0.8	**0.898** **(0.033)**	0.881 (0.035)	0.816 (0.012)	0.819 (0.011)	0.622 (0.036)	0.603 (0.036)	0.613 (0.045)
Musk2	0.6	**0.755** **(0.080)**	0.753 (0.086)	0.659 (0.058)	0.633 (0.033)	0.584 (0.073)	0.593 (0.061)	0.518 (0.086)
	0.7	**0.828** **(0.047)**	0.805 (0.050)	0.737 (0.045)	0.735 (0.064)	0.584 (0.072)	0.561 (0.076)	0.491 (0.076)
	0.8	**0.881** **(0.037)**	**0.881** **(0.035)**	0.824 (0.036)	0.817 (0.016)	0.610 (0.127)	0.593 (0.081)	0.538 (0.119)
Elephant	0.6	**0.695** **(0.057)**	0.636 (0.073)	0.599 (0.050)	0.594 (0.011)	0.540 (0.041)	0.591 (0.009)	0.494 (0.016)
	0.7	**0.799** **(0.061)**	0.776 (0.078)	0.698 (0.005)	0.701 (0.000)	0.630 (0.053)	0.690 (0.007)	0.591 (0.027)
	0.8	**0.871** **(0.019)**	0.850 (0.033)	0.801 (0.000)	0.801 (0.000)	0.739 (0.027)	0.794 (0.007)	0.684 (0.042)
Fox	0.6	**0.606** **(0.024)**	0.601 (0.012)	0.603 (0.012)	0.603 (0.004)	0.558 (0.038)	0.604 (0.003)	0.584 (0.033)
	0.7	**0.711** **(0.018)**	0.707 (0.010)	0.706 (0.007)	0.705 (0.005)	0.570 (0.053)	0.680 (0.041)	0.645 (0.053)
	0.8	**0.807** **(0.007)**	**0.807** **(0.007)**	0.805 (0.011)	0.806 (0.009)	0.628 (0.055)	0.770 (0.068)	0.666 (0.096)
Tiger	0.6	0.682 (0.071)	**0.688** **(0.068)**	0.544 (0.052)	0.600 (0.009)	0.448 (0.069)	0.597 (0.006)	0.439 (0.081)
	0.7	**0.763** **(0.044)**	0.744 (0.067)	0.699 (0.002)	0.699 (0.002)	0.578 (0.092)	0.652 (0.087)	0.538 (0.087)
	0.8	**0.841** **(0.035)**	0.832 (0.020)	0.804 (0.003)	0.804 (0.002)	0.637 (0.090)	0.786 (0.029)	0.606 (0.096)

Table 2, the baseline achieves decent performance, while our proposed methods TC-SQ and TC-DH are even better. In addition, TC-SQ achieves the best performance in most cases.

4.2 Experiments on Text Categorization

We also conduct experiments on three datasets for the task of biocreative text categorization [32, 33]. In this task, we aim to decide whether a given <protein, document> pair should be annotated with some Gene Ontology (GO) codes. The given inputs are some documents (bags) composed of paragraphs (instances), and each paragraph is represented by

Table 3. The characteristics of the used datasets for the biocreative text categorization task.

Dataset	# Features	# Positive bags	# Negative bags	# Avg. Pos. Ins. per bag	# Avg. Neg. Ins. per bag
Component	200	423	2707	2.9±8.7	8.9±7.6
Function	200	443	4799	1.8±6.8	8.8±7.0
Process	200	757	10961	1.4±6.0	8.7±6.9

Table 4. Classification accuracy on the biocreative text categorization datasets. The best performance is highlighted in bold.

Datasets	θ_+	Our Proposed		Baselines				
		TC-SQ	TC-DH	SD-SQ	SD-DH	TL	KM	CKM
Component	0.6	**0.784** **(0.053)**	0.766 (0.089)	0.610 (0.137)	0.602 (0.000)	0.771 (0.028)	0.776 (0.011)	0.782 (0.024)
	0.7	**0.835** **(0.044)**	0.831 (0.044)	0.721 (0.030)	0.711 (0.018)	0.790 (0.040)	0.787 (0.039)	0.797 (0.047)
	0.8	0.862 (0.032)	**0.875** **(0.047)**	0.800 (0.000)	0.800 (0.000)	0.830 (0.040)	0.833 (0.027)	0.807 (0.039)
Function	0.6	**0.846** **(0.032)**	0.840 (0.068)	0.678 (0.075)	0.657 (0.074)	0.802 (0.033)	0.798 (0.022)	0.819 (0.026)
	0.7	**0.854** **(0.041)**	0.852 (0.040)	0.736 (0.028)	0.715 (0.039)	0.813 (0.044)	0.818 (0.037)	0.827 (0.033)
	0.8	**0.890** **(0.029)**	**0.890** **(0.036)**	0.800 (0.000)	0.800 (0.000)	0.859 (0.032)	0.858 (0.023)	0.853 (0.037)
Process	0.6	0.877 (0.012)	**0.879** **(0.014)**	0.707 (0.058)	0.694 (0.074)	0.802 (0.014)	0.788 (0.022)	0.834 (0.015)
	0.7	**0.888** **(0.014)**	0.881 (0.025)	0.739 (0.030)	0.725 (0.045)	0.847 (0.021)	0.832 (0.023)	0.855 (0.015)
	0.8	0.900 (0.015)	**0.901** **(0.014)**	0.802 (0.003)	0.801 (0.000)	0.888 (0.017)	0.878 (0.017)	0.857 (0.017)

a feature vector. The used features are word occurrence frequencies and some statistics about the nature of the protein-GO code interaction for each paragraph. The GO consists of three hierarchical domains of standardized biological terms referring to cellular components, biological processes, and molecular functions. A <protein, document> pair is labeled with a GO code if the document contains some paragraphs that link the protein to the component, process, or function described by the GO code. Thus, there are three datasets in this biocreative text categorization task: Component, Function, and Process[2]. Table 3 reports the detailed information of the three datasets. Table 4 shows the classification accuracy of each method on the three datasets. From Table 4, we can also observe that our proposed methods TC-SQ and TC-DH are clearly superior to other compared baselines. In addition, TC-DH achieves similar performance as TC-SQ in the task of biocreative text categorization.

4.3 Experiments on Image Datasets

We further conduct experiments on MNIST[3] [22], Fashion[4] [41] and KMNIST[5] [11]. Since the three datasets are not multi-instance datasets, we follow [48] to do the bag construction procedure on the three datasets. Specifically, the multi-instance MNIST-bags, in which each bag contains several images and its bag size is drawn from a Gaussian distribution with fixed mean and variance. The bag is positive if it contains a target digit, otherwise negative. Using 10

[2] https://figshare.com/articles/dataset/MIProblems_A_repository_of_multiple_instance_learning_datasets/6633983?file=12144479

[3] http://yann.lecun.com/exdb/mnist/

[4] https://github.com/zalandoresearch/fashion-mnist

[5] http://codh.rois.ac.jp/kmnist/

Table 5. The characteristics of the used image datasets.

Dataset	# Features	# Positive bags	# Negative bags	# Avg. Pos. Ins. per bag	# Avg. Neg. Ins. per bag
MNIST Bags	784	1,750	1,750	10±2	10±2
Fashion Bags	784	1,750	1,750	10±2	10±2
KMNIST Bags	784	1,750	1,750	10±2	10±2

Table 6. Classification accuracy on the image datasets. The best performance is highlighted in bold.

Datasets	θ_+	Our Proposed		Baselines				
		TC-SQ	TC-DH	SD-SQ	SD-DH	TL	KM	CKM
MNIST Bags	0.6	0.624 (0.018)	**0.633** **(0.024)**	0.570 (0.009)	0.567 (0.013)	0.537 (0.013)	0.555 (0.012)	0.558 (0.013)
	0.7	0.734 (0.018)	**0.747** **(0.021)**	0.700 (0.008)	0.690 (0.005)	0.569 (0.011)	0.555 (0.013)	0.569 (0.012)
	0.8	**0.822** **(0.016)**	0.816 (0.018)	0.801 (0.001)	0.800 (0.000)	0.587 (0.015)	0.555 (0.008)	0.587 (0.018)
Fashion Bags	0.6	**0.670** **(0.024)**	0.667 (0.017)	0.633 (0.017)	0.587 (0.007)	0.531 (0.006)	0.548 (0.007)	0.553 (0.007)
	0.7	**0.762** **(0.019)**	0.753 (0.024)	0.708 (0.006)	0.708 (0.006)	0.555 (0.012)	0.562 (0.012)	0.551 (0.008)
	0.8	**0.828** **(0.013)**	0.815 (0.021)	0.800 (0.000)	0.800 (0.000)	0.577 (0.008)	0.578 (0.011)	0.551 (0.007)
KMNIST Bags	0.6	**0.641** **(0.014)**	0.640 (0.013)	0.552 (0.014)	0.560 (0.013)	0.528 (0.006)	0.545 (0.007)	0.540 (0.008)
	0.7	**0.733** **(0.008)**	0.725 (0.007)	0.682 (0.007)	0.700 (0.002)	0.556 (0.011)	0.555 (0.007)	0.543 (0.007)
	0.8	**0.815** **(0.005)**	0.809 (0.006)	0.801 (0.002)	0.800 (0.000)	0.571 (0.011)	0.571 (0.008)	0.539 (0.009)

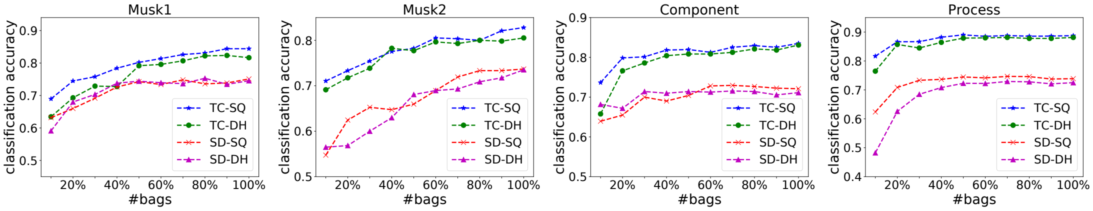

Fig. 1. Classification accuracy of each method when the number of triplet bags increases.

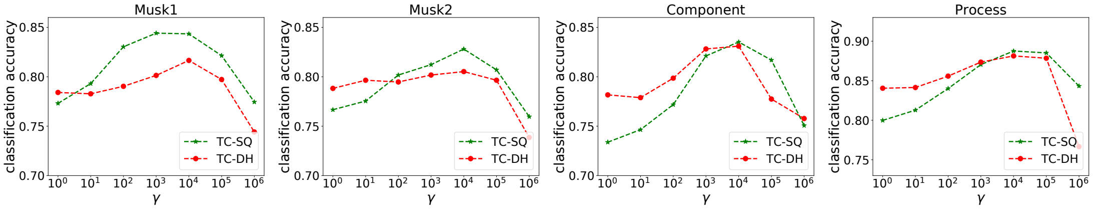

Fig. 2. Classification accuracy of TC-SQ and TC-DH by varying γ.

different digits as targets, we obtain 10 different multi-instance MNIST-bag datasets. In the same way, we can obtain 10 Fashion-bag datasets and 10 KMNIST-bag datasets. Table 5 reports the detailed information of those datasets. Table 6 reports the macro averaged classification accuracy over 10 one-vs-rest datasets, respectively. From Table 6, we can also observe that our proposed methods TC-SQ and TC-DH are clearly superior to other compared baselines.

4.4 Further Analysis

Performance of Increasing Triplet Bags. As shown in Theorem 1, the performance of our methods is expected to be improved if more triplet comparison bags are provided. To empirically validate this theoretical finding, we conduct experiments on two benchmark datasets (`Musk1` and `Musk2`) and two biocreative text classification datasets (`Component` and `Process`) with class prior $\theta_{+} = 0.7$ by varying the number of triplet comparison bags (100% means that all the bags are used for training). As shown in Figure 1, the classification accuracy of our proposed methods generally increases given more triplet comparison bags. This observation is clearly in accordance with our derived generalization error bound in Theorem 1, since the generalization error decreases as the number of triplet bags increases. In addition, our proposed methods (TC-SQ and TC-DH) generally outperform SD-SQ and SD-DH when different numbers of training bags are provided.

Influence of regularization. To show the effect of the regularization on the proposed methods, we perform MIL from triplet comparison bags using our proposed methods on the above four datasets (`Musk1`, `Musk2`, `Component`, and `Process`) with $\theta_{+} = 0.7$ and γ is selected from $\{10^0, \ldots, 10^6\}$. We show the classification accuracy in Figure 2. From Figure 2, we can find that the best performance is achieved at some intermediate value of γ, which suggests that the regularization term plays an important role.

5 CONCLUSION

Most of the existing multiple-instance learning studies need fully labeled bags for training an effective classifier, while it could be hard to collect such data in real-world scenarios. Therefore, we investigated a novel weakly supervised learning problem called multiple-instance learning from triplet comparison bags, where we aim to train a bag-level binary classifier from only triplet comparison bags. A triplet (X_a, X_b, X_c) contains the weak supervision information that bag X_a is more similar to X_b than to X_c. To the best of our knowledge, this paper provided the first attempt to study this problem. To solve this new MIL problem, we proposed to train a bag-level classifier by the empirical risk minimization framework and theoretically provided a generalization error bound. Extensive experiments clearly demonstrated that our proposed method significantly outperforms other baselines. In future work, we will investigate multiple-instance learning with other types of weak supervision.

A GENERATION PROCESS OF TRIPLET COMPARISON BAGS

Recall the assumption that three bags in a triplet are sampling independently. Therefore, for a triplet (X_a, X_b, X_c), the bag labels (Y_a, Y_b, Y_c) can only appear to be one of the following cases:

$$\mathcal{Y}_1 = \{(+1,+1,+1), (+1,+1,-1), (+1,-1,-1), (-1,+1,+1), (-1,-1,+1), (-1,-1,-1)\}.$$

Otherwise, the first bag is more similar to the third bag than to the second bag, and in this case, (Y_a, Y_b, Y_c) appears to be one of the following cases:

$$\mathcal{Y}_2 = \{(+1,-1,+1), (-1,+1,-1)\}.$$

According to the above distributions $\mathcal{Y}_1$ and $\mathcal{Y}_2$, we can actually collect two distinct types of datasets as follows:

$$\mathcal{D}_1 = \{(X_a, X_b, X_c) | (Y_a, Y_b, Y_c) \in \mathcal{Y}_1\}, \quad \mathcal{D}_2 = \{(X_a, X_b, X_c) | (Y_a, Y_b, Y_c) \in \mathcal{Y}_2\}.$$

The two types of datasets $\mathcal{D}_1$ and $\mathcal{D}_2$ can be considered to be generated from the following underlying distributions:

$$p_1(X_a, X_b, X_c) = \frac{p(X_a, X_b, X_c, (Y_a, Y_b, Y_c) \in \mathcal{Y}_1)}{\theta_T},$$
$$p_2(X_a, X_b, X_c) = \theta_+ p_+(X) p_+(X) p_-(X) + \theta_- p_-(X) p_+(X) p_-(X),$$

where $\theta_T = 1 - \theta_+\theta_-$, $\theta_+ = p(y = +1)$ and $\theta_- = p(y = -1)$ and $p_+(X) = p(X|y = +1)$ and $p_-(X) = p(X|y = -1)$. Then we have

$$\mathcal{D}_1 = \{(X_{1,a}, X_{1,b}, X_{1,c})\}^{m_1} \sim p_1(X_a, X_b, X_c), \quad \mathcal{D}_2 = \{(X_{2,a}, X_{2,b}, X_{2,c})\}^{m_2} \sim p_2(X_a, X_b, X_c).$$

Furthermore, we denote the pointwise data collected from $\mathcal{D}_1$ and $\mathcal{D}_2$ by ignoring the triplet comparison relation as $\mathcal{D}_{1,a} = \{X_{1,a}\}^{m_1}$, $\mathcal{D}_{1,b} = \{X_{1,b}\}^{m_1}$, $\mathcal{D}_{1,c} = \{X_{1,c}\}^{m_1}$, $\mathcal{D}_{2,a} = \{X_{2,a}\}^{m_2}$, $\mathcal{D}_{2,b} = \{X_{2,b}\}^{m_2}$ and $\mathcal{D}_{2,c} = \{X_{2,c}\}^{m_2}$. From Theorem 1 in Cui et al. [12], samples in $\mathcal{D}_{1,a}$, $\mathcal{D}_{1,c}$, $\mathcal{D}_{2,a}$ and $\mathcal{D}_{2,c}$ are independently drawn from

$$\tilde{p}_1(X) = \theta_+ p_+(X) + \theta_- p_-(X),$$

samples in $\mathcal{D}_{1,b}$ are independently drawn from

$$\tilde{p}_2(X) = \frac{(\theta_+^3 + 2\theta_+^2\theta_-)p_+(X) + (2\theta_+\theta_-^2 + \theta_-^3)p_-(X)}{\theta_T},$$

and samples in $\mathcal{D}_{2,b}$ are independently drawn from

$$\tilde{p}_3(X) = \theta_- p_+(X) + \theta_+ p_-(X).$$

Those indicate that from triplet comparison data, we can essentially obtain samples that can be drawn independently from three different distributions. Then we denote the three aggregated datasets (from respective distributions) as

$$\tilde{\mathcal{D}}_1 = \{\tilde{X}_i^1\}_{i=1}^{n_1} = \mathcal{D}_{1,a} \cup \mathcal{D}_{1,c} \cup \mathcal{D}_{2,a} \cup \mathcal{D}_{2,c}, \quad \tilde{\mathcal{D}}_2 = \{\tilde{X}_i^2\}_{i=1}^{n_2} = \mathcal{D}_{1,b}, \quad \tilde{\mathcal{D}}_3 = \{\tilde{X}_i^3\}_{i=1}^{n_3} = \mathcal{D}_{2,b},$$

where

$$\tilde{\mathcal{D}}_1 \sim \tilde{p}_1(X), \quad \tilde{\mathcal{D}}_2 \sim \tilde{p}_2(X), \quad \tilde{\mathcal{D}}_3 \sim \tilde{p}_3(X).$$

Let $C = \frac{\theta_+^3 + 2\theta_+^2\theta_-}{\theta_T}$ and $D = \frac{2\theta_+\theta_-^2 + \theta_-^3}{\theta_T}$, we can express the relationship between these densities as

$$\begin{bmatrix} \tilde{p}_1(X) \\ \tilde{p}_2(X) \\ \tilde{p}_3(X) \end{bmatrix} = \begin{bmatrix} \theta_+ & \theta_- \\ C & D \\ \theta_- & \theta_+ \end{bmatrix} \begin{bmatrix} p_+(X) \\ p_-(X) \end{bmatrix}.$$

B PROOF OF THEOREM 1

Recall that by using the loss function that satisfies the linear-odd condition, $\widehat{R}_{\text{Trip}}(g)$ can be also represented as:

$$\widehat{R}_{\text{Trip}}(g) = \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\big((\lambda_1 + \lambda_2)\ell_+(g(X_i^1)) + \lambda_2 g(X_i^1)\big) + \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\big((\lambda_3 + \lambda_4)\ell_+(g(X_i^2)) + \lambda_4 g(X_i^2)\big)$$
$$+ \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\big((\lambda_5 + \lambda_6)\ell_-(g(X_i^3)) - \lambda_5 g(X_i^3)\big).$$

In this way, we can represent $R_{\mathrm{Trip}}(g)$ as

$$R_{\mathrm{Trip}}(g) = \mathbb{E}_{\widetilde{p}_1(X)}\Big[(\lambda_1+\lambda_2)\ell_+(g(X^1)) + \lambda_2 g(X^1)\Big] + \mathbb{E}_{\widetilde{p}_2(X)}\Big[(\lambda_3+\lambda_4)\ell_+(g(X^2)) + \lambda_4 g(X^2)\Big] + \mathbb{E}_{\widetilde{p}_3(X)}\Big[(\lambda_5+\lambda_6)\ell_-(g(X^3)) - \lambda_5 g(X^3)\Big],$$

where we assumed that the collected data $\{X_i^1\}_{i=1}^{n_1}$ are independently sampled from $\widetilde{p}_1(X)$, the collected data $\{X_i^2\}_{i=1}^{n_2}$ are independently sampled from $\widetilde{p}_2(X)$ and the collected data $\{X_i^3\}_{i=1}^{n_3}$ are independently sampled from $\widetilde{p}_3(X)$. Let us further introduce

$$\widehat{R}_1(g) = \frac{1}{n_1}\sum\nolimits_{i=1}^{n_1}\big((\lambda_1+\lambda_2)\ell_+(g(X_i^1)) + \lambda_2 g(X_i^1)\big), \quad R_1(g) = \mathbb{E}_{\widetilde{p}_1(X)}\Big[(\lambda_1+\lambda_2)\ell_+(g(X^1)) + \lambda_2 g(X^1)\Big],$$
$$\widehat{R}_2(g) = \frac{1}{n_2}\sum\nolimits_{i=1}^{n_2}\big((\lambda_3+\lambda_4)\ell_+(g(X_i^2)) + \lambda_4 g(X_i^2)\big), \quad R_2(g) = \mathbb{E}_{\widetilde{p}_2(X)}\Big[(\lambda_3+\lambda_4)\ell_+(g(X^2)) + \lambda_4 g(X^2)\Big],$$
$$\widehat{R}_3(g) = \frac{1}{n_3}\sum\nolimits_{i=1}^{n_3}\big((\lambda_5+\lambda_6)\ell_-(g(X_i^3)) - \lambda_5 g(X_i^3)\big), \quad R_3(g) = \mathbb{E}_{\widetilde{p}_3(X)}\Big[(\lambda_5+\lambda_6)\ell_-(g(X^3)) - \lambda_5 g(X^3)\Big].$$

In this way, we have:

$$\widehat{R}_{\mathrm{Trip}}(g) = \widehat{R}_1(g) + \widehat{R}_2(g) + \widehat{R}_3(g), \quad R_{\mathrm{Trip}}(g) = R_1(g) + R_2(g) + R_3(g).$$

Thus,

$$\sup_{g\in\mathcal{G}}\Big|R_{\mathrm{Trip}}(g) - \widehat{R}_{\mathrm{Trip}}(g)\Big| \le \sup_{g\in\mathcal{G}}\Big|R_1(g) - \widehat{R}_1(g)\Big| + \sup_{g\in\mathcal{G}}\Big|R_2(g) - \widehat{R}_2(g)\Big| + \sup_{g\in\mathcal{G}}\Big|R_3(g) - \widehat{R}_3(g)\Big|.$$

Hence the problem becomes how to find an upper bound of each term in the right hand size of the inequality.

Lemma 1. *With the introduced definitions and conditions in Theorem 1, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\sup_{g\in\mathcal{G}}\Big|R_1(g) - \widehat{R}_1(g)\Big| \le (|\lambda_1| + |\lambda_2|)\Big(\frac{2C_{\mathcal{G}}}{\sqrt{n_1}} + C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_1}}\Big).$$

Proof. Firstly, it is easy to verify that the double hinge loss ℓ_{DH} is 1-Lipschitz. Suppose an example in $\widehat{R}_1(g)$ is replaced by another arbitrary example, then the change of $\sup_{g\in\mathcal{G}}\big(R_1(g) - \widehat{R}_1(g)\big)$ is no greater than $(|\lambda_1|+|\lambda_2|)C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}/n_1$. Then, by applying McDiarmid's inequality [28], for any $\delta > 0$, with probability at least $1-\frac{\delta}{2}$,

$$\sup_{g\in\mathcal{G}}\big(R_1(g) - \widehat{R}_1(g)\big) \le \mathbb{E}\Big[\sup_{g\in\mathcal{G}}\big(R_1(g) - \widehat{R}_1(g)\big)\Big] + (|\lambda_1| + |\lambda_2|)C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_1}}.$$

Besides, it is routine [30] to show

$$\mathbb{E}\Big[\sup_{g\in\mathcal{G}}\big(R_1(g) - \widehat{R}_1(g)\big)\Big] \le 2(|\lambda_1| + |\lambda_2|)\mathfrak{R}_{n_1}(\mathcal{G}),$$

where we have used the Talagrand's lemma (Lemma 4.2 in Mohri et al. [30]), i.e., $\mathfrak{R}_n(\ell\circ\mathcal{G}) \le \rho\mathfrak{R}_n(\mathcal{G})$ if ℓ is a ρ-Lipschitz loss function. By considering $\mathfrak{R}_n(\mathcal{G}) \le C_{\mathcal{G}}/\sqrt{n}$, we have

$$\sup_{g\in\mathcal{G}}\big(R_1(g) - \widehat{R}_1(g)\big) \le (|\lambda_1| + |\lambda_2|)\Big(\frac{2C_{\mathcal{G}}}{\sqrt{n_1}} + C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_1}}\Big).$$

By further taking into account the other side $\sup_{g\in\mathcal{G}}\big(\widehat{R}_1(g)-R_1(g)\big)$, we have for any $\delta > 0$, with probability at least $1-\delta$,

$$\sup_{g\in\mathcal{G}}\left|R_1(g)-\widehat{R}_1(g)\right| \leq (|\lambda_1|+|\lambda_2|)\Big(\frac{2C_{\mathcal{G}}}{\sqrt{n_1}}+C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_1}}\Big),$$

which completes the proof of Lemma 1. □

Lemma 2. *With the introduced definitions and conditions in Theorem 1, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\sup_{g\in\mathcal{G}}\left|R_2(g)-\widehat{R}_2(g)\right| \leq (|\lambda_3|+|\lambda_4|)\Big(\frac{2C_{\mathcal{G}}}{\sqrt{n_2}}+C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_2}}\Big).$$

Lemma 3. *With the introduced definitions and conditions in Theorem 1, for any $\delta > 0$, with probability at least $1-\delta$, we have*

$$\sup_{g\in\mathcal{G}}\left|R_2(g)-\widehat{R}_2(g)\right| \leq (|\lambda_5|+|\lambda_6|)\Big(\frac{2C_{\mathcal{G}}}{\sqrt{n_3}}+C_{\boldsymbol{w}}C_{\boldsymbol{\phi}}\sqrt{\frac{\log\frac{2}{\delta}}{2n_3}}\Big).$$

Lemma 2 and Lemma 3 can be proved similarly as Lemma 1, hence we omit the proof. By combining Lemma 1, Lemma 2 and Lemma 3 together, Theorem 1 is immediately proved. □

REFERENCES

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

[2] Martin S Andersen, Joachim Dahl, and Lieven Vandenberghe. 2013. CVXOPT: Python software for convex optimization. *URL https://cvxopt. org* 64 (2013).

[3] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. 2002. Support vector machines for multiple-instance learning. In *NeurIPS*. 577–584.

[4] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. 2009. Visual tracking with online multiple instance learning. In *CVPR*. 983–990.

[5] Han Bao, Gang Niu, and Masashi Sugiyama. 2018. Classification from Pairwise Similarity and Unlabeled Data. In *ICML*. 452–461.

[6] Han Bao, Tomoya Sakai, Issei Sato, and Masashi Sugiyama. 2018. Convex formulation of multiple instance learning from positive and unlabeled bags. *Neural Networks* 105 (2018), 132–141.

[7] Peter L Bartlett and Shahar Mendelson. 2002. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research* 3, 11 (2002), 463–482.

[8] Yuzhou Cao, Lei Feng, Yitian Xu, Bo An, Gang Niu, and Masashi Sugiyama. 2021. Larning from Similarity-Confidence Data. In *ICML*. 1272–1282.

[9] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. 2018. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition* 77 (2018), 329–353.

[10] Yixin Chen, Jinbo Bi, and James Ze Wang. 2006. MILES: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 12 (2006), 1931–1947.

[11] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. 2018. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718* (2018).

[12] Zhenghang Cui, Nontawat Charoenphakdee, Issei Sato, and Masashi Sugiyama. 2020. Classification from triplet comparison data. *Neural Computation* 32, 3 (2020), 659–681.

[13] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence* 89, 1-2 (1997), 31–71.

[14] M. C. du Plessis, G. Niu, and M. Sugiyama. 2015. Convex formulation for learning from positive and unlabeled data. In *ICML*. 1386–1394.

[15] Lei Feng, Senlin Shu, Yuzhou Cao, Lue Tao, Hongxin Wei, Tao Xiang, Bo An, and Gang Niu. 2021. Multiple-Instance Learning from Similar and Dissimilar Bags. In *KDD*. 374–382.

[16] James Richard Foulds and Eibe Frank. 2010. A review of multi-instance learning assumptions. *The Knowledge Engineering Review* 25 (2010), 1–25.

[17] Thomas Gärtner, Peter A Flach, Adam Kowalczyk, and Alexander J Smola. 2002. Multi-instance kernels. In *ICML*. 179–186.

[18] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*.

[19] Sheng-Jun Huang, Wei Gao, and Zhi-Hua Zhou. 2018. Fast multi-instance multi-label learning. *IEEE transactions on pattern analysis and machine intelligence* 41, 11 (2018), 2614–2627.

[20] Maximilian Ilse, Jakub Tomczak, and Max Welling. 2018. Attention-based deep multiple instance learning. In *ICML*. PMLR, 2127–2136.

[21] T. Ishida, G. Niu, and M. Sugiyama. 2018. Binary classification for positive-confidence data.. In *NeurIPS*. 5917–5928.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[23] Christian Leistner, Amir Saffari, and Horst Bischof. 2010. MIForests: Multiple-instance learning with randomized trees. In *ECCV*. 29–42.

[24] Xin-Chun Li, De-Chuan Zhan, Jia-Qi Yang, and Yi Shi. 2021. Deep multiple instance selection. *Science China Information Sciences* 64 (2021), 1–15.

[25] Dong Liang, Xinbo Gao, Wen Lu, and Jie Li. 2021. Deep blind image quality assessment based on multiple instance regression. *Neurocomputing* 431 (2021), 78–89.

[26] Nan Lu, Gang Niu, Aditya K. Menon, and Masashi Sugiyama. 2019. On the Minimal Supervision for Training Any Binary Classifier from Only Unlabeled Data. In *ICLR*.

[27] James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *BSMSP*. 281–297.

[28] Colin McDiarmid. 1989. On the method of bounded differences. *Surveys in Combinatorics* 141, 1 (1989), 148–188.

[29] Shahar Mendelson. 2008. Lower bounds for the empirical minimization algorithm. *IEEE Transactions on Information Theory* 54, 8 (2008), 3797–3803.

[30] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2012. *Foundations of Machine Learning*. MIT Press.

[31] Gang Niu, Marthinus Christoffel du Plessis, Tomoya Sakai, Yao Ma, and Masashi Sugiyama. 2016. Theoretical comparisons of positive-unlabeled learning against positive-negative learning. In *NeurIPS*. 1199–1207.

[32] Soumya Ray and Mark Craven. 2005. Learning statistical models for annotating proteins with function information using biomedical text. *BMC bioinformatics* 6, Suppl 1 (2005), S18.

[33] Soumya Ray and Mark Craven. 2005. Supervised versus multiple instance learning: An empirical comparison. In *ICML*. ACM, 697–704.

[34] Tomoya Sakai, Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. 2017. Semi-supervised classification based on classification from positive and unlabeled data. In *ICML*. 2998–3006.

[35] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *CVPR*. 815–823.

[36] Takuya Shimada, Han Bao, Issei Sato, and Masashi Sugiyama. 2020. Classification from pairwise similarities/dissimilarities and unlabeled data via empirical risk minimization. *Neural Computation* (2020).

[37] Qingping Tao, Stephen Scott, NV Vinodchandran, and Thomas Takeo Osugi. 2004. SVM-based generalized multiple-instance learning via approximate box counting. In *ICML*. 101.

[38] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. 2001. Constrained k-means clustering with background knowledge. In *ICML*. 577–584.

[39] Zhuang Wang, Vladan Radosavljevic, Bo Han, Zoran Obradovic, and Slobodan Vucetic. 2008. Aerosol optical depth prediction from satellite observations by multiple instance regression. In *ICDM*. SIAM, 165–176.

[40] Hong-Xin Wei, Lei Feng, Xiang-Yu Chen, and Bo An. 2020. Combating noisy labels by agreement: A joint training method with co-regularization. In *CVPR*. 13726–13735.

[41] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017).

[42] Ling Xiao, Renfa Li, Juan Luo, et al. 2006. Sensor localization based on nonmetric multidimensional scaling. *STRESS* 2, 1 (2006).

[43] Xin Xu and Eibe Frank. 2004. Logistic regression and boosting for labeled bags of instances. In *PAKDD*. Springer, 272–281.

[44] Cha Zhang and Paul Viola. 2007. Multiple-instance pruning for learning efficient cascade detectors. *NeurIPS*, 1681–1688.

[45] Min-Ling Zhang, Fei Yu, and Cai-Zhi Tang. 2017. Disambiguation-free partial label learning. *IEEE Transactions on Knowledge and Data Engineering* 29, 10 (2017), 2155–2167.

[46] Qi Zhang and Sally A Goldman. 2001. EM-DD: An improved multiple-instance learning technique. In *NeurIPS*. 1073–1080.

[47] Teng Zhang and Hai Jin. 2020. Optimal Margin Distribution Machine for Multi-Instance Learning. In *IJCAI*. 2383–2389.

[48] Weijia Zhang, Xuanhui Zhang, Min-Ling Zhang, et al. 2022. Multi-instance causal representation learning for instance label prediction and out-of-distribution generalization. *NeurIPS* 35 (2022), 34940–34953.

[49] Zhi-Li Zhang and Min-Ling Zhang. 2006. Multi-instance multi-label learning with application to scene classification. *NeurIPS* 19 (2006).

[50] Zhi-Hua Zhou. 2018. A brief introduction to weakly supervised learning. *National Science Review* 5, 1 (2018), 44–53.

[51] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. 2009. Multi-instance learning by treating instances as non-iid samples. In *ICML*. 1249–1256.

[52] Xiaojin Zhu and Andrew B Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 3, 1 (2009), 1–130.