Bag Similarity Network for Deep Multi-instance Learning

Xinggang Wang[a], Yongluan Yan[a], Peng Tang[a], Wenyu Liu[a,*], Xiaojie Guo[b]

[a]*School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China*
[b]*School of Computer Software, Tianjin University, Tianjin, China*

Abstract

The effectiveness of multi-instance learning (MIL) has been demonstrated by its wide spectrum of applications in computer vision, biometrics, and natural language processing. Recently, solving MIL problems using deep neural networks has proven to be highly effective. However, in current multi-instance neural networks, the feature representation of each bag is learned individually, and the relations between bags are not considered. In this study, we propose a novel neural network for MIL that emphasizes modeling the affinities between bags. It achieves a more effective bag representation than previous methods. Specifically, a bag with multiple instances is modeled by its similarity to other bags, and the similarity calculation is carried out in a novel neural network, termed the bag similarity network (BSN). Training the BSN involves two representation learning problems: instance feature learning and bag similarity learning. To avoid the complex interdependence of these problems, we decouple the BSN training process by first

*Corresponding author
Email addresses: xgwang@hust.edu.cn (Xinggang Wang), yongluanyan@hust.edu.cn (Yongluan Yan), pengtang@hust.edu.cn (Peng Tang), liuwy@hust.edu.cn (Wenyu Liu), xj.max.guo@gmail.com (Xiaojie Guo)

training an instance feature learning network, and then construct a bag similarity network, each of which is optimized end-to-end by back-propagation. Experiments are conducted to demonstrate clearly the advantage of the proposed method over other state-of-the-art methods on various MIL datasets.
Keywords: Multi-instance learning, neural networks, similarity learning

1. Introduction

Multimedia data, such as text and images, are ubiquitous and are expected to benefit society. However, such data are often loosely controlled (e.g., images are rarely given with precise annotations), and thus they are difficult to employ directly. Moreover, in practice, it is impossible to label all the data manually. Weakly supervised learning (WSL), which requires only little supervision, has been developed to mitigate the cost of data annotation. As a representative of WSL, multi-instance learning (MIL) was originally proposed for drug activity prediction [7], and its applicability has since been broadened to a variety of computer vision and machine learning tasks, such as text classification, medical image classification, object detection [5, 30, 27], and semantic segmentation [20, 19, 12], with promising performance.

In MIL, each sample is in the form of a labeled bag that contains a set of instances associated with input features. The goal of MIL in a binary task is to train a classifier so that the labels of testing bags may be predicted based on the assumption that a positive bag contains at least one positive instance, whereas a negative bag contains only negative instances. A variety of algorithms have been proposed for MIL problems; they can be roughly divided into three groups according to the underlying principle: the instance-space paradigm, which aims at

learning instance models (such as mi-SVM [2], EM-DD [38], and MIBoosting [34]), the bag-space paradigm, which treats the bags collectively, and the discriminant learning process is performed in the space of bags (such as MInd [4] and mi-Graph [39]), and the embedded-space paradigm, which learns bag statistics as bag representations with or without a vocabulary (such as miFV [31] and MI-Net [29]). A thorough review of classical MIL methods can be found in [1]. In this study, we propose bag embedding learning by the bag-space approach in a neural network.

Neural networks [21, 40, 37, 36] have been effectively used in MIL problems. Ramon and De Raedt [21] designed a multi-instance neural network that takes a bag as input, uses hidden nodes to infer instance probabilities, and calculates bag probabilities from the related instance probabilities using a log-sum-exp function over the instances. The log-sum-exp function is a convex max function that relaxes the bag-instance constraints in MIL. Zhang et al. [37] improved multi-instance neural networks by feature selection using diverse density and principal component analysis (PCA). Zhang and Zhou [36] demonstrated that ensemble methods can be integrated with multi-instance neural networks and improve them. Wang et al. [29] revisited traditional multi-instance neural networks and introduced new networks that employ different recently proposed deep learning techniques, such as deep supervision and residual connections (MI-Nets). In MI-Nets, a bag representation is generated by aggregating the related instance representations by using a max or average pooling layer in the neural network. As demonstrated by the multi-instance dissimilarity (MInD) method [4], it is more effective to represent each bag by a vector containing the bag's dissimilarities to other bags in the training set and treat these dissimilarities as a bag representation. Based on

this representation, various classifiers can be adopted for bag classification and impressive performance can be obtained.

However, instance similarity in MInD is measured using certain fixed metrics, such as the min–min Hausdorff distance and mean–mean Hausdorff distance, which may not be optimal. Hence, it is natural to ask whether *it is possible to learn some bag similarity metrics for boosting the performance of MIL.* We consider this a bag similarity learning problem.

To answer the question, we propose a novel neural network structure, namely the bag similarity network (BSN), owing to the highly effective representation learning of these networks. We notice that the input of a BSN is quite different from that of traditional neural networks. In conventional neural networks, only one instance is taken as the input, such as an image or a sentence. By contrast, the proposed BSN takes a target bag and all the training bags as input. Each bag contains multiple instances. If there are a target bag and N training bags, each containing M instances, then there are $(N+1) \times M$ instances as input to the network. BSN is an $(N+1) \times M$-stream neural network that has received little attention.

When designing the BSN, we let the instance features be learnable. Then, the similarity between two instances can be easily obtained through an inner product operation. As a result, in the network, we have $N \times M$ similarities between the target bag and the training bags. The $N \times M$ similarities are represented by hidden nodes in the BSN. Then, we propose a differentiable Hausdorff pooling layer to obtain a bag similarity representation as in the set-to-set distance in MInD [4]. A comparison between the bag similarity obtained by MInD and the proposed method is shown in Fig. 1, where it can be seen that the diagonal-blockness of the

similarity matrix is quite satisfactory and is significantly better than that by MInD. That is, the learned similarity representation is more discriminative; furthermore, it is easy to classify using a linear classifier.

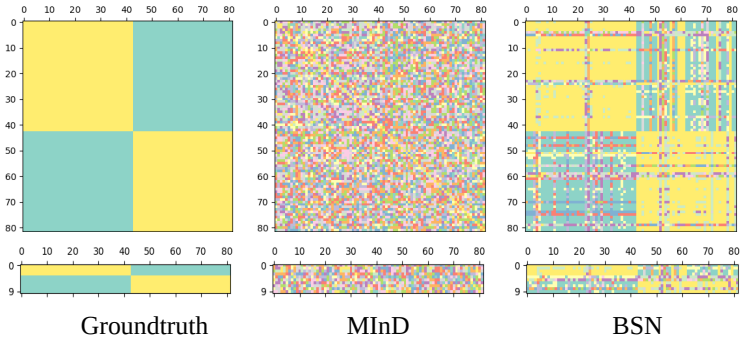

Figure 1: Comparison between the fixed similarity metric (MInD) [4] and the learned similarity matrix by BSN. The **top** row shows the similarity matrices of training bags, and the **bottom** row shows the similarity matrices of testing bags. Even though the diagonal-blockness of the similarity matrix by BSN is not perfect, it is quite satisfactory, whereas MInD lacks this desired property.

Bag similarity is a reference-based representation, and all training bags are used as reference bags. By fixing the similarity function as an inner product operator, the principle of this similarity learning problem is to learn suitable instance features. For a target bag, the similarity representation depends on both the target and all the reference bags. Representation learning for both the target and the reference bags is a joint optimization problem. In each iteration of BSN training, all instance features should be updated, which is time-consuming. Thus, we propose a decoupled training scheme: We first train a MI-Net to obtain all instance features; then, we fix the neural features of reference instances and update the features of target instances.

In summary, the contributions of this study are as follows:

- *We propose a learnable bag similarity representation for MIL. To the best of our knowledge, this is the first study that integrates similarity learning with multi-instance neural networks.*

- *To solve bag similarity learning problems, we propose a novel bag similarity network that takes $(N+1) \times M$ streams as input. For effective training, we propose a decoupled training scheme.*

- *The proposed BSN method has achieved state-of-the-art performance on several different MIL tasks.*

2. Related Work

2.1. Multi-instance Learning

MIL has long been an active research topic owing to its ability to handle weakly labeled data. Utilizing weakly labeled data is highly important, because labeling for big data is costly. MIL has been applied in various computer vision [17, 41, 33, 28, 18] and medical image analysis problems [35, 16]. For example, in object detection, Wang et al. [30] formulated the problem of weakly supervised object detection as a MIL problem and proposed a relaxed MIL solution that uses deep learning features as instance representation. Cinbis et al. [5] proposed a multi-fold MIL to avoid poor local optimal solutions. Tang et al. [26] proposed a bag-in-bag formulation for modeling contextual information around objects. Investigating new MIL methods is essential for understanding weakly labeled data.

In MIL, we are given a set of bags $X = \{X_1, X_2, ..., X_N\}$. Each bag X_i can be represented by distinct instances $X_i = \{x_{i1}, x_{i2}, ..., x_{im_i}\}$, where x_{ij} denotes the j^{th} instance in bag X_i and m_i denotes the number of instances in this bag.

We assume that $Y_i \in \{0, 1\}$ and $y_{ij} \in \{0, 1\}$ represent the label of bag X_i and the label of instance x_{ij}, respectively. During the training phase, only bag labels are available, whereas instance labels are unknown. There are two standard MIL constraints regarding bag and instance labels: if $Y_i = 0$, then all instances in the corresponding bag X_i are negative; otherwise, at least one instance $x_{ij} \in X_i$ is positive.

2.2. Multi-instance Neural Network

In the recent years, neural networks have become the most effective method for addressing MIL problems. Ilse et al. [13] added an attention module in multi-instance neural networks for instance selection and obtained impressive results for cancer detection in histopathology images. Even in the multi-label setting, Feng et al. [8] confirmed that deep neural networks are effective.

MI-Net [29] is a typical multi-instance neural network that focuses on MIL problems. MI-Net contains L fully connected (FC) layers and one MIL pooling layer (generally, L is equal to 4). The first $L-1$ FC layers are followed by a non-linear transformation such as the rectified linear unit (ReLU) [10], which learns the representations of all instances in the corresponding bag. Here, x_{ij}^{ℓ} denotes the ℓ^{th} layer output of j^{th} instance x_{ij} in bag X_i. The MIL pooling layer is used to map all instance-level features to obtain bag-level representations. Three widely used pooling schemes $M(x^{L-1}_{ij|j=1\ldots m_i})$ are mentioned in [29]: 1) max pooling $M(x^{L-1}_{ij|j=1\ldots m_i}) = \max_j x_{ij}^{L-1}$, 2) mean pooling $M(x^{L-1}_{ij|j=1\ldots m_i}) = \frac{1}{m_i}\sum_{j=1}^{m_i} x_{ij}^{L-1}$, and 3) log–sum–exp (LSE) pooling $M(x^{L-1}_{ij|j=1\ldots m_i}) = \frac{\log[\frac{1}{m_i}\sum_{j=1}^{m_i}\exp(r\cdot x_{ij}^{L-1})]}{r}$, where r is a parameter controlling the smoothness of approximation to the max function. Thus, regardless of the number of input instances, the MIL pooling layer

aggregates them into a bag-level representation. Finally, the probability of a bag being positive can be calculated by an *FC* layer with only one neuron and sigmoid activation, and then the bag label is predicted.

The proposed BSN is also based on neural networks. However, unlike previous multi-instance networks that learn a bag embedding without considering the bag's relation to other bags, BSN learns a bag embedding by comparing the bag with the other bags. Furthermore, BSN is different from traditional bag similarity methods that use fixed bag similarity metrics, as it learns bag similarity using neural networks.

In addition, BSN can be regarded as a special instantiation of memory-augmented neural networks [23], which are widely used in meta-learning. Here, memory refers to external memory and is different from the internal memory in long short-term memory (LSTM) networks [11]. The reference training bags with their feature extraction networks can be considered external memory in BSN.

3. Bag Similarity Network for MIL

Unlike traditional methods, the proposed method addresses MIL problems from the new perspective of bag similarity learning. In the proposed design, each bag is represented by a vector of its similarities to other bags in the training set, and these similarities are treated as a bag-level representation–hence the term bag similarity network. Figure 2 shows the overall architecture of BSN, where it can be seen that to avoid the complications of updating parameters and reduce computational load, two key steps are required for training. The first is to learn instance-level representations of reference bags by MI-Net. In the second step, we train the neural network by computing the similarity of each training bag to the reference

bags. Furthermore, we propose a Hausdorff pooling layer to generate a vector bag representation; based on this, bag classification can be easily carried out by an FC layer.

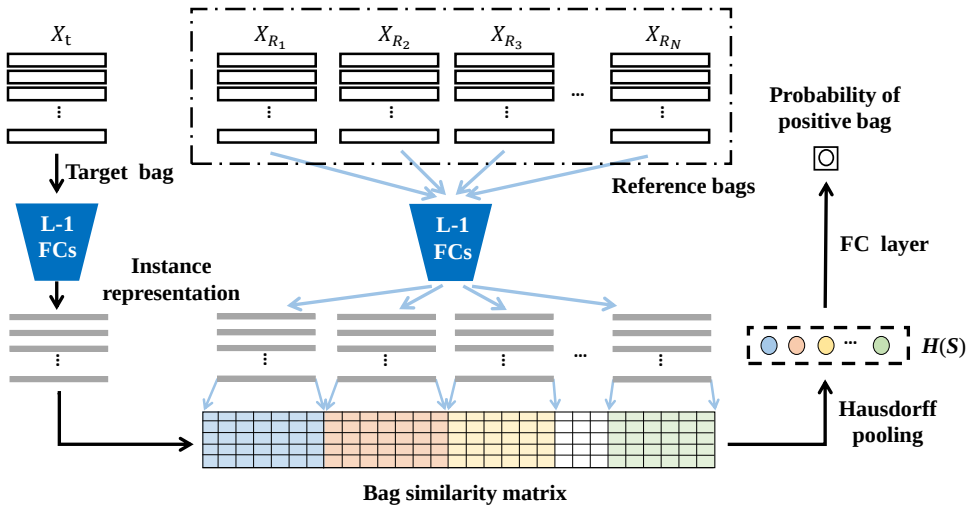

Figure 2: Architecture of Bag Similarity Network

3.1. Instance-level Representations

In Fig. 2, bag-level representations are calculated based on the similarity to all reference bags. For simplicity, we regard all training bags $X = \{X_1, X_2, ..., X_N\}$ as reference bags. To distinguish reference bags from training bags, we denote the former by $X_R = \{X_{R_1}, X_{R_2}, ..., X_{R_N}\}$, where N is the number of reference bags. Then, we follow the MI-Net [29] method to construct a multi-instance neural network, which contains L *FC* layers and one MIL pooling layer, as the base network. We employ the same training strategy in the base network. After training, instance-level representations of all reference bags are learned. We feed one reference bag $X_{R_i} = \{x_{R_i1}, x_{R_i2}, ..., x_{R_im_i}\}$ into the base network, and collect the

outputs $X_{R_i}^{L-1}$ of the $(L-1)$-th layer for all instances in bag X_{R_i}. We follow the same process for all reference bags and obtain suitable instance-level representations, leading to a reference vector as follows:

$$X_R^{L-1} = \{X_{R_1}^{L-1}, X_{R_2}^{L-1}, ..., X_{R_N}^{L-1}\}, \tag{1}$$

where $X_{R_i}^{L-1} = \{x_{R_i1}^{L-1}, x_{R_i2}^{L-1}, ..., x_{R_im_i}^{L-1}\}$ denotes the learned instance-level representations of reference bag X_{R_i}.

3.2. Bag Similarity Matrix

To compute the similarity between bag X_t and the reference bags, we construct the BSN, which also contains L *FC* layers and one pooling layer. The first $L-1$ layers are used for learning instance-level representations for bag X_t. We denote the outputs of the $(L-1)^{th}$ layer by $X_t^{L-1} = \{x_{t1}^{L-1}, x_{t2}^{L-1}, ..., x_{tm_t}^{L-1}\}$. Then, we calculate the inner product $f(x,y) = x^T y$ of the instances of X_t and each reference bag to form the bag similarity matrix:

$$S = [f(X_t^{L-1}, X_{R_1}^{L-1}), ..., f(X_t^{L-1}, X_{R_N}^{L-1})], \tag{2}$$

where the similarity between X_t and X_{R_i} is represented by $f(X_t^{L-1}, X_{R_i}^{L-1})$, which is defined as

$$\begin{pmatrix} f(x_{t1}^{L-1}, x_{R_i1}^{L-1}) & f(x_{t1}^{L-1}, x_{R_i2}^{L-1}) & ... & f(x_{t1}^{L-1}, x_{R_im_i}^{L-1}) \\ f(x_{t2}^{L-1}, x_{R_i1}^{L-1}) & f(x_{t2}^{L-1}, x_{R_i2}^{L-1}) & ... & f(x_{t2}^{L-1}, x_{R_im_i}^{L-1}) \\ ... & ... & ... & ... \\ f(x_{tm_t}^{L-1}, x_{R_i1}^{L-1}) & f(x_{tm_t}^{L-1}, x_{R_i2}^{L-1}) & ... & f(x_{tm_t}^{L-1}, x_{R_im_i}^{L-1}) \end{pmatrix}, \tag{3}$$

where m_t and m_i denote the number of instances in X_t and X_{R_i}, respectively. Therefore, X_t can be represented by a bag similarity matrix, which describes the similarity of X_t to reference bags $X_R = \{X_{R_1}, X_{R_2}, ..., X_{R_N}\}$.

3.3. Hausdorff Pooling

After calculating the bag similarity matrix S between bag X_t and reference bags X_R, we propose new pooling methods to describe bag similarity. Inspired by the Hausdorff method [4], we convert the similarity matrix $S \in \mathbb{R}^{m_t \times \sum_i m_i}$ into a bag-level representation $H(S) \in \mathbb{R}^{1 \times N}$ that represents the similarity of X_t to all reference bags $X_R = \{X_{R_1}, X_{R_2}, .., X_{R_N}\}$. Specifically, we can formulate this bag-level representation as

$$\begin{aligned} H(S) = \quad & [h(f(X_t^{L-1}, X_{R_1}^{L-1})), h(f(X_t^{L-1}, X_{R_2}^{L-1})), \\ & ..., h(f(X_t^{L-1}, X_{R_N}^{L-1}))], \end{aligned} \tag{4}$$

where $h(\cdot)$ can be one of the following:

$$\begin{cases} \text{max-max pooling :} & h(A) = \max\limits_i \max\limits_j A_{ij}; \\ \text{mean-max pooling :} & h(A) = \frac{1}{N} \sum_{i=1}^{N} \max\limits_j A_{ij}; \\ \text{min-max pooling :} & h(A) = \min\limits_i \max\limits_j A_{ij}; \\ \text{mean-mean pooling :} & h(A) = \frac{1}{N} \frac{1}{M} \sum_{i=1}^{N} \sum_{j=1}^{M} Aij, \end{cases} \tag{5}$$

where $A \in \mathbb{R}^{N \times M}$ stands for the similarity matrix between two bags. It should be noted that we simply choose mean pooling as MIL pooling in the base network if mean–mean pooling is adopted in BSN; otherwise, max pooling is used in the base network. Subsequently, we obtain the bag-level representation $H(S)$ between bag X_t and all reference bags. Finally, the score of the positive bag $\hat{Y}_t$ is calculated by an *FC* layer with one neuron and sigmoid activation.

3.4. Training Loss and Optimization

Another point that should be considered is the loss function for training. To predict bag labels, it is natural to choose the standard cross entropy loss function,

which is the same as the loss function in MI-Net [29]:

$$L(\hat{Y}_t, Y_t) = -\left((1 - Y_t)\log(1 - \hat{Y}_t) + Y_t \log \hat{Y}_t\right), \tag{6}$$

where $\hat{Y}_t$ is the probability that bag B_t is positive, and Y_t is the label of X_t. To make the BSN algorithm more explicit, we have outlined the training and testing procedures in Algorithm 1. The reader is referred to the complete algorithm for details that cannot be covered in the text.

4. Experiments

In this section, we conduct MIL experiments with BSN on various tasks, including drug activation prediction, automatic image annotation, text categorization, and medical image diagnosis including colon cancer detection in histopathology images. Moreover, we compare BSN with the following state-of-the-art MIL methods: mi-SVM and MI-SVM [2], MI-Kernel [9], EM-DD [38], mi-Graph [39], miVLAD and miFV [31], MInD [4], MI-Net [29], and Attention-based MI-Net [13].

4.1. Datasets

MUSK1 and **MUSK2** [7] are used to predict the molecular activity of drugs, where bags are molecules and instances are different conformations of these molecules. A molecule always exhibits multiple shapes, and each shape is described by a 166-dimensional feature vector. In this case, a good molecule will bind well to the target protein if at least one of the shapes is appropriate, whereas a poor molecule will not bind well if none of its shapes can bind. Thus, we can formulate this drug activation prediction as a MIL problem. In MUSK1, there are 476 instances

Algorithm 1 BSN for MIL

Input: Training bags $X = \{X_1, X_2, ..., X_N\}$ and corresponding bag labels $Y = \{Y_1, Y_2, ..., Y_N\}$; testing bag X_t

Output: Label $\hat{Y}_t$ of the testing bag X_t

1: **Training:**

2: Set all training bags X as reference bags $X_R = \{X_{R_1}, X_{R_2}, ..., X_{R_N}\}$.

3: Initialize MI-Net and train it on all reference bags

4: **for** $i = 1$ **to** N **do**

5: Extract instance-level representations of X_{R_i}

6: **end for**

7: Initialize BSN

8: **for** $i = 1$ **to** N **do**

9: Extract instance-level representations of each instance in bag X_i.

10: Compute bag similarity matrix S of bag X_i and reference bags X_R.

11: Perform Hausdorff pooling and obtain the bag-level representation $H(S)$ of bag X_i.

12: Predict the label $\hat{Y}_i$ of bag X_i

13: Compute the loss function $L(\hat{Y}_i, Y_i)$ and update the weights of BSN.

14: **end for**

15: **Testing:**

16: Follow steps 9–12 to predict $\hat{Y}_t$ for the testing bag X_t.

divided into 47 positive bags and 45 negative bags. In MUSK2, 6598 instances are included, which are divided into 39 positive bags and 63 negative bags.

Fox, **Tiger**, and **Elephant** [2] are widely adopted MIL datasets for solving localized content-based image retrieval problems. The bags are images and instances are image segments. For each category, positive bags are from the target class of animal images, and negative bags are randomly chosen from other classes of animal images. They all consist of 100 positive bags and 100 negative bags, and each bag contains 2 to 13 instances. Moreover, an instance is represented as a 230-dimensional feature vector to describe color, texture, and shape information in an image region.

20 Newsgroups [39] contains posts from newsgroups on 20 subjects for text categorization—another typical application of MIL. For each category, there are 50 positive bags and 50 negative bags. The positive bags contain an average of 3.7% positive instances. The instances of negative bags are all randomly drawn from other categories. Each instance is represented by 200 term frequency-inverse document frequency features.

Messidor [6, 14] is a public diabetic retinopathy screening dataset that contains $1,200$ eye fundus images from 654 diabetic and 546 healthy patients. Each image is resized to 700×700 pixels and split up into patches of 135×135 pixels. We regard these images and patches as bags and instances, respectively. Each bag contains 8 to 12 instances. Each instance is represented by a 687-dimensional feature vector describing intensity histogram and texture. For data preprocessing, we reduce the dimensionality to 100 by PCA [32].

Colon Cancer [13] contains 100 H&E images generated from normal or malignant tissue appearance. The majority of the nuclei of each cell are marked in each

image. In this dataset, we regard H&E images and image patches as bags and instances, respectively. As nuclei are associated with multi-class labels, we focus on epithelial cells and determine whether an H&E image contains one or more nuclei from the epithelial class.

4.2. Experiment Setup

The BSN method requires two steps to obtain bag labels during the training phase. The first is to use MI-Net [29] as the base network for generating instance-level representations of all reference bags. In the second step, we optimize BSN by computing the similarity between reference bags. As mentioned in Sec. 3, the base network and BSN both contain four *FC* layers, and the number of neurons in these layers is 256, 128, 64, and 1. The first three *FC* layers of the base network are followed by a MIL-pooling layer to aggregate instance-level representations into bag-level representations. After learning instance-level representations, BSN computes the similarity between the target bag and reference bags. In the next step, we use the proposed Hausdorff pooling layer to form a feature vector that provides the bag-level representation from the bag similarity matrix. Finally, in the last *FC* layer, we estimate the probability of the input bag being positive.

Regarding the weights of the *FC* layers, they are initialized by the truncated normal distribution [3], where the mean and standard deviation are set to 0 and 0.05, respectively (for the Newsgroups dataset, the standard deviation is set to 0.1). Biases are all initialized to be 0, and the momentum is set to 0.9. The initial learning rate and weight decay vary from dataset to dataset; details are shown in Table 1. We divide the learning rate by 2 at 1500 and 3000 iterations and terminate the training at 5000 iterations in all experiments, except for Messidor. For Messidor, the learning rate is divided by 2 at $5,000$, $10,000$, $20,000$, and $30,000$

Table 1: Parameter details for training BSN with MI-Net, including initial learning rate (LR), weight decay (WD), and the standard deviation of initial weights (W-Std). The parameters for Colon Cancer dataset are the same as in Attention-based MI-Net [13].

Dataset	LR	WD	W-Std
MUSK1	0.001	0.01	0.05
MUSK2	0.001	0.01	0.05
Fox	0.0005	0.01	0.05
Tiger	0.001	0.01	0.05
Elephant	0.0005	0.01	0.05
20 Newsgroups	0.001	0.001	0.1
Messidor	0.0005	0.001	0.05

iterations, and the training is terminated at $40,000$ iterations. All networks are optimized by stochastic gradient descent techniques, and the batch input consists of only one bag for both training and testing. In the Colon Cancer dataset, the experimental setting and network architecture follow those in [13]. The hyper-parameters are selected using cross-validation on training sets. The source code of the experiments will be provided on publication.

4.3. Experimental Results

We provide results of the BSN for drug activation prediction, content-based image retrieval, text categorization, and medical image diagnosis. Following the standard experimental settings in other related studies, we ran 10-fold cross validations 10 times independently and report the averages over 10 trials for all the experiments in this part, except Messidor and Colon Cancer dataset, where we ran five 10-fold cross-validations. Except for MI-Net [29], the results of previous

Table 2: Bag classification results ($mean \pm std$) of different methods on MUSK1 and MUSK2 (task: drag activation prediction), as well as Fox, Tiger, and Elephant (task: content-based image retrieval).

Dataset	MUSK1	MUSK2	Fox	Tiger	Elephant
mi-SVM	0.874	0.836	0.582	0.784	0.822
MI-SVM	0.779	0.843	0.578	0.840	0.814
MI-Kernel	0.880	0.893	0.603	0.842	0.843
EM-DD	0.849 ± 0.098	0.869 ± 0.108	0.609 ± 0.101	0.730 ± 0.096	0.771 ± 0.098
mi-Graph	0.889 ± 0.073	*0.903 ± 0.086*	0.616 ± 0.079	*0.860 ± 0.083*	0.869 ± 0.078
miVLAD	0.871 ± 0.097	0.872 ± 0.095	0.620 ± 0.098	0.811 ± 0.087	0.850 ± 0.080
miFV	*0.909 ± 0.089*	0.884 ± 0.094	0.621 ± 0.109	0.813 ± 0.083	0.852 ± 0.081
MInD	0.893 ± 0.019	0.888 ± 0.034	$\mathbf{0.651 \pm 0.011}$	0.819 ± 0.021	0.857 ± 0.018
MI-Net	0.893 ± 0.099	0.872 ± 0.096	0.627 ± 0.080	0.832 ± 0.087	*0.891 ± 0.074*
Att. Net	0.892 ± 0.040	0.858 ± 0.048	0.615 ± 0.043	0.839 ± 0.022	0.868 ± 0.022
Gated Att. Net	0.900 ± 0.050	0.863 ± 0.042	0.603 ± 0.029	0.845 ± 0.018	0.857 ± 0.027
BSN	$\mathbf{0.931 \pm 0.094}$	$\mathbf{0.906 \pm 0.109}$	*0.640 ± 0.111*	$\mathbf{0.878 \pm 0.092}$	$\mathbf{0.907 \pm 0.073}$

MIL methods are from the reference papers. For MI-Net, we re-implemented it and obtained better results than in [29]. For fair comparison, we used these new results as higher baselines. All the experimental results were obtained under the same experimental conditions.

Drug Activation Prediction. Table 2 (second and third columns) provides the average accuracy and the corresponding standard deviation of methods under comparison. We note that the standard deviation for mi-SVM, MI-SVM, and MI-Kernel is not available in the original papers and are therefore omitted. The best performance on each dataset is highlighted in bold, and the second best in italics. It can be seen that miFV has the second best accuracy (90.9%), and both MI-Net and Attention Net reach approximately 89.3%, slightly behind miFV on

Table 3: Bag classification results of different methods on 20 Newsgroups (task: text categorization).

Dataset	MI-Kernel	mi-Graph	miFV	MInD	MI-Net	Att. Net	Gated Att. Net	BSN
alt.atheism	0.602 ± 0.039	0.655 ± 0.040	0.848 ± 0.119	$\mathit{0.861 \pm 0.089}$	0.846 ± 0.101	0.784 ± 0.084	0.780 ± 0.074	$\mathbf{0.903 \pm 0.101}$
comp.graphics	0.470 ± 0.033	0.778 ± 0.016	0.594 ± 0.120	0.825 ± 0.118	$\mathit{0.831 \pm 0.123}$	0.774 ± 0.081	0.764 ± 0.073	$\mathbf{0.861 \pm 0.134}$
comp.windows.misc	0.510 ± 0.052	0.631 ± 0.015	0.615 ± 0.172	$\mathit{0.730 \pm 0.094}$	$\mathit{0.730 \pm 0.112}$	0.686 ± 0.088	0.700 ± 0.080	$\mathbf{0.769 \pm 0.101}$
comp.ibm.pc.hardware	0.469 ± 0.036	0.595 ± 0.027	0.665 ± 0.147	0.780 ± 0.127	$\mathit{0.803 \pm 0.155}$	0.632 ± 0.087	0.640 ± 0.080	$\mathbf{0.813 \pm 0.29}$
comp.sys.mac.hardware	0.445 ± 0.032	0.617 ± 0.048	0.660 ± 0.157	$\mathit{0.835 \pm 0.098}$	0.811 ± 0.138	0.744 ± 0.084	0.754 ± 0.082	$\mathbf{0.865 \pm 0.113}$
comp.window.x	0.508 ± 0.043	0.698 ± 0.021	0.768 ± 0.155	0.785 ± 0.111	$\mathit{0.836 \pm 0.135}$	0.766 ± 0.093	0.780 ± 0.075	$\mathbf{0.869 \pm 0.111}$
misc.forsale	0.518 ± 0.025	0.552 ± 0.027	0.565 ± 0.146	$\mathit{0.729 \pm 0.102}$	0.707 ± 0.119	0.706 ± 0.076	0.674 ± 0.072	$\mathbf{0.768 \pm 0.128}$
rec.autos	0.529 ± 0.033	0.720 ± 0.037	0.667 ± 0.166	0.775 ± 0.088	$\mathit{0.812 \pm 0.129}$	0.762 ± 0.081	0.724 ± 0.091	$\mathbf{0.830 \pm 0.121}$
rec.motorcycles	0.506 ± 0.035	0.640 ± 0.028	0.802 ± 0.144	0.577 ± 0.102	$\mathit{0.853 \pm 0.119}$	0.750 ± 0.097	0.814 ± 0.066	$\mathbf{0.868 \pm 0.116}$
rec.sport.baseball	0.517 ± 0.028	0.647 ± 0.031	0.779 ± 0.148	0.837 ± 0.078	$\mathit{0.871 \pm 0.113}$	0.774 ± 0.080	0.790 ± 0.078	$\mathbf{0.887 \pm 0.113}$
rec.sport.hockey	0.513 ± 0.034	0.850 ± 0.025	0.823 ± 0.137	0.833 ± 0.096	0.918 ± 0.111	$\mathit{0.936 \pm 0.041}$	0.932 ± 0.045	$\mathbf{0.947 \pm 0.085}$
sci.crypt	0.563 ± 0.036	0.696 ± 0.021	0.760 ± 0.146	0.768 ± 0.122	$\mathit{0.808 \pm 0.154}$	0.804 ± 0.063	0.748 ± 0.088	$\mathbf{0.826 \pm 0.142}$
sci.electronics	0.506 ± 0.019	0.871 ± 0.017	0.555 ± 0.156	$\mathbf{0.940 \pm 0.078}$	$\mathit{0.928 \pm 0.090}$	0.854 ± 0.053	0.828 ± 0.064	0.927 ± 0.088
sci.med	0.506 ± 0.019	0.621 ± 0.039	0.783 ± 0.125	0.832 ± 0.091	$\mathit{0.862 \pm 0.111}$	0.772 ± 0.090	0.742 ± 0.101	$\mathbf{0.885 \pm 0.107}$
sci.space	0.547 ± 0.025	0.757 ± 0.034	0.818 ± 0.131	0.796 ± 0.110	0.820 ± 0.087	$\mathit{0.888 \pm 0.062}$	$\mathbf{0.894 \pm 0.063}$	0.869 ± 0.112
soc.religion.christian	0.492 ± 0.034	0.590 ± 0.047	0.814 ± 0.138	$\mathit{0.841 \pm 0.140}$	0.829 ± 0.122	0.726 ± 0.088	0.708 ± 0.100	$\mathbf{0.878 \pm 0.113}$
talk.politics.guns	0.477 ± 0.038	0.585 ± 0.060	0.747 ± 0.150	$\mathit{0.806 \pm 0.094}$	0.782 ± 0.095	0.714 ± 0.074	0.708 ± 0.100	$\mathbf{0.814 \pm 0.117}$
talk.politics.mideast	0.559 ± 0.028	0.736 ± 0.026	0.793 ± 0.135	$\mathit{0.830 \pm 0.108}$	0.825 ± 0.119	0.750 ± 0.084	0.784 ± 0.064	$\mathbf{0.867 \pm 0.127}$
talk.politics.misc	0.515 ± 0.037	0.704 ± 0.036	0.697 ± 0.152	0.720 ± 0.119	0.748 ± 0.140	0.788 ± 0.091	$\mathit{0.806 \pm 0.078}$	$\mathbf{0.829 \pm 0.135}$
talk.religion.misc	0.554 ± 0.043	0.633 ± 0.035	0.739 ± 0.151	0.725 ± 0.104	$\mathit{0.778 \pm 0.114}$	0.738 ± 0.074	0.746 ± 0.082	$\mathbf{0.821 \pm 0.115}$
average	0.515	0.679	0.726	0.791	*0.820*	0.767	0.766	**0.855**

MUSK1. The proposed method achieves the best result, namely, 93.1%, outperforming miFV and MI-Net/Gated Att. Net by 2.2% and more than 3%, respectively. The efficacy of BSN is further confirmed on MUSK2, although the improvement over the second best performance (90.3%) obtained by mi-Graph is not as pronounced as that over miFV on MUSK2. It is worth mentioning that, compared with miFV and mi-Graph, BSN is considerably more stable. Moreover, compared with the performance of MI-Net, the performance of BSN is significantly and consistently enhanced by computing the similarity between bags and treating it as a bag-level reference to predict bag labels. That is, the effectiveness of the proposed strategy is corroborated.

Content-based Image Retrieval. Content-based image retrieval can be formulated as a MIL problem, namely, to identify the target objects in images. To solve

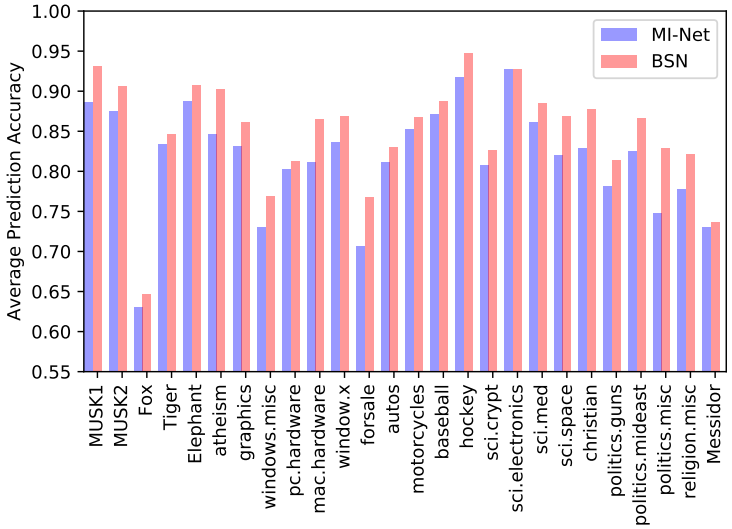

Figure 3: Effectiveness of the proposed bag similarity representation on MUSK1, MUSK2, Fox, Tiger, Elephant, 20 Newsgroups, and Messidor datasets

this problem, we conduct experiments on three animal image datasets (Fox, Tiger, and Elephant). The numerical results are given in Table 2 (last three columns) and indicate that BSN achieves superior performance on Tiger and Elephant, and competitive performance on Fox. The average prediction accuracy on these datasets was improved by 1.5% compared with that of MI-Net. By describing bag-level representations based on bag similarity, we obtain more robust bag label predictions and avoid being trapped into local optima.

Text Categorization. There are several studies concerned with the application of MIL to text categorization. Comparisons are made between MI-Kernel, mi-Graph, miFV, MInD, MI-Net, and BSN. Table 3 provides the average accuracy over 10 runs. It can be seen that BSN outperforms all competitors in all cases except for

sci.electronics, where the accuracy of MInD (best) and MI-Net (second best) is higher by 1.3% and 0.1%, respectively. The average accuracy on all 20 datasets indicates that both MI-Net and BSN outperform the others, including MI-Kernel, mi-Graph, and miFV, by approximately 10%. Furthermore, the average accuracy of BSN is higher than that of MI-Net and MInD by 3.5% and 8.1%, respectively.

Table 4: Comparison of different methods ($mean \pm std$) for bag classification on Messidor dataset

mi-SVM	MI-SVM	miVLAD	miFV	MInD	MI-Net	Att.Net	Gated Att. Net	BSN
0.620	0.640	0.691	0.715	0.665	*0.730*	0.703	0.698	**0.737**
$\pm$0.039	$\pm$0.050	$\pm$0.037	$\pm$0.047	$\pm$0.071	$\pm$0.051	$\pm$0.041	$\pm$0.048	$\pm$0.050

Medical Image Diagnosis. In addition to drug activation prediction, localized content-based image retrieval, and text categorization, medical image diagnosis is another typical application of MIL. Thus, Messidor, a public diabetic retinopathy screening dataset, is also included in our experiments. We used 10-fold cross-validation. In each fold, there are $1,080$ bags (90% bags of this dataset) for training. Unlike in the previous datasets, where the entire training set was used as reference bags, we select the 150 most positive bags and 150 most negative bags by MI-Net as reference bags to reduce computational load. Table 4 shows that BSN achieves the best result on Messidor, outperforming miVLAD, miFV, and MI-Net by 6.7%, 3.1%, and 1.0%, respectively. Furthermore, it outperforms mi-SVM, MI-SVM, and MInD by a large margin.

Colon Cancer Detection. There are more MIL studies [25, 13] focusing on automatic detection of cancerous regions in H&E images to facilitate medical diagnosis. We choose the Colon Cancer dataset [13] on this task to compare the proposed method with other deep learning MIL methods. Table 5 presents the re-

Table 5: Comparison of different methods ($mean \pm std$) for bag classification on Colon Cancer dataset

Method	Accuracy	precision	Recall	F-score
Instance+max	0.842 ± 0.021	0.866 ± 0.017	0.816 ± 0.031	0.839 ± 0.023
Embedding+max	0.824 ± 0.015	0.884 ± 0.014	0.753 ± 0.020	0.813 ± 0.017
Att. Net	$\mathbf{0.904 \pm 0.011}$	$\mathbf{0.953 \pm 0.014}$	$\mathit{0.855 \pm 0.017}$	$\mathbf{0.901 \pm 0.011}$
Gated Att. Net	$\mathit{0.898 \pm 0.020}$	$\mathit{0.944 \pm 0.016}$	0.851 ± 0.035	$\mathit{0.893 \pm 0.022}$
BSN	0.869 ± 0.008	0.820 ± 0.019	$\mathbf{0.983 \pm 0.019}$	0.886 ± 0.030

sults of Attention-based MI-Net (Att. Net) [13], Gated Attention MI-Net (Gated Att. Net) [13], MI-Nets (Instance+max and Embedding+max), and BSN.

The results demonstrate that Att. Net obtains the best precision ($95.3 \pm 1.4\%$), whereas BSN obtains the best recall ($98.3 \pm 1.9\%$). The overall performance is measured by accuracy and the F-score. It is concluded that BSN outperforms MI-Nets (Instance+max and Embedding+max) but is worse than Att. Nets. This is because the reference H&E images are highly similar and the similarity representations in BSN are not sufficiently discriminative. Therefore, for this task, we suggest combining BSN with Attention-based MI-Nets.

5. Ablation Study and Discussion

Effectiveness of Bag Similarity Representation. We have already argued that the proposed bag similarity representation are superior to other methods such as MI-Net. To justify the argument and indicate the improvement, we offer a detailed comparison between the proposed BSN and MI-Net in Fig. 3. The average accuracy of BSN on the 20 Newsgroups dataset was improved by 4.3% compared

with that of MI-Net, whereas the average accuracy on the MUSK, animal, and Messidor datasets was improved by 3.2%. As both BSN and MI-Net are based on the same instance feature learning network, the results clearly demonstrate that bag similarity representation is useful for MIL problems. We recall the comparison in Fig. 1, where it is seen that BSN tends to obtain a block diagonal similarity matrix. This is a highly desired property for both clustering and classification in machine learning, as it is related to a group itself and simultaneously assists in discriminating among different groups. For further performance improvement, it would be interesting to design the training loss function so that diagonal-blockness may be achieved as much as possible.

Influence of Different Hausdorff Pooling Functions As mentioned previously, we introduce four Hausdorff pooling methods to convert the bag similarity matrix into a bag-level feature vector: max–max, mean–max, min–max, and mean–mean pooling functions, which are all differentiable. Usually, the second operator in Hausdorff polling is preferably a max or mean function. We compare the influence of these four pooling methods on MUSK1, MUSK2, Fox, Tiger, and Elephant datasets in Fig. 4. It can be seen that the four pooling methods yield comparable results on the MUSK1, MUSK2, and Fox datasets. Mean–max and mean–mean pooling are slightly better than max–max and min–max pooling on the Tiger dataset, whereas max–max and mean–max pooling are better than min–max and mean–mean pooling on the Elephant dataset. In spite of the slight difference in performance, if we consider the high non-linearity of the neural network and the comparisons provided in Sec. 4, we can still appreciate the robustness and stability of the bag similarity learning framework.

Necessity of Decoupled Training. In most learning tasks, we favor end-to-

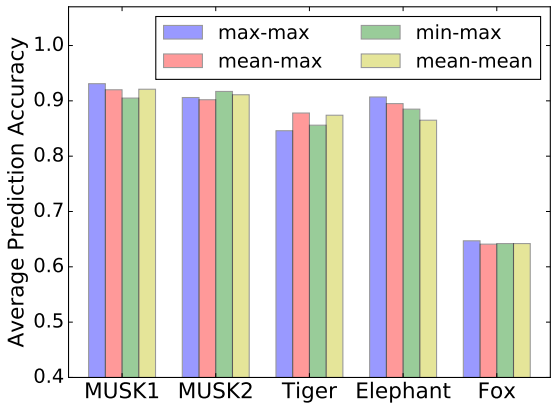

Figure 4: Comparison of different MIL pooling methods on MUSK1, MUSK2, Fox, Tiger, and Elephant datasets

end optimization. However, for our target task, it is impractical to train a fully end-to-end BSN, as it requires updating the instance-level representations of all reference bags at every iteration. The time complexity of training a fully end-to-end BSN is proportional to the number of reference/training bags. If there are n training bags, a fully end-to-end learning scheme will be n times slower than the proposed decoupled scheme. In addition, the interdependence of instance feature learning and bag similarity learning increases model complexity. Hence, to resolve complex dependencies and reduce training cost, a trade-off is necessary, such as the proposed decoupling.

Ineffectiveness of Finetuning from Pre-trained Models. A large number of neural network studies adopt a pre-trained model, such as AlexNet [15] and GoogLeNet [24], and then finetune the network according to this model. The underlying principle of the finetuning strategy is to transfer the knowledge in pre-trained models for a new task. Those pre-trained models are most likely trained

on large-scale image datasets, such as ImageNet [22], and are able to capture high-level (such as category-level) image information to obtain an effective representation. However, in MIL problems, finetuning is ineffective. This is largely due to the small scale of most MIL datasets, which makes learning a suitable pre-trained model difficult. In our experiments, we also attempted to initialize the $(N+1)$-th stream (for the target bag) (Fig. 2) with the well-trained MI-Net (for the reference bag). We also notice a decrease in performance. These results hint that the similarity representation is quite different from the embedded bag representation because similarity contains rich contextual/global information, whereas the embedded bag representation contains only individual information.

6. Conclusion

We proposed BSN, a novel bag similarity network for MIL problems. BSN achieved state-of-the-art performance on widely used MIL benchmarks. Its main advantage is that it can learn a discriminative bag similarity representation with rich contextual information. Compared with previous bag similarity techniques, it is a learnable metric, and thus it is more effective in various types of data in various domains. For effective learning, a decoupled training scheme was designed by considering the characteristics of MIL and the complexity of the model. BSN demonstrated its ability to achieve the desired diagonal-blockness of the similarity matrix, which is critical for clustering and classification in machine learning theory. This ability additionally brings robustness and stability to BSN.

Acknowledgement

We thank the anonymous reviewers for the helpful comments. This work was supported by NSFC (No. 61876212, No. 61733007 and No. 6157220) and Hubei Scientific and Technical Innovation Key Project.

References

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

[2] Andrews, S., Tsochantaridis, I., and Hofmann, T. (2002). Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems*, pages 561–568.

[3] Barr, D. R. and Sherrill, E. T. (1999). Mean and variance of truncated normal distributions. *The American Statistician*, 53(4):357–361.

[4] Cheplygina, V., Tax, D. M., and Loog, M. (2015). Multiple instance learning with bag dissimilarities. *Pattern Recognition*, 48(1):264–275.

[5] Cinbis, R. G., Verbeek, J., and Schmid, C. (2017). Weakly supervised object localization with multi-fold multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1):189–203.

[6] Decencière, E., Zhang, X., Cazuguel, G., Lay, B., Cochener, B., Trone, C., Gain, P., Ordonez, R., Massin, P., Erginay, A., Charton, B., and Klein, J.-C. (2014). Feedback on a publicly distributed image database: the Messidor database. *Image Analysis and Stereology*, pages 231–234.

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

[8] Feng, J. and Zhou, Z.-H. (2017). Deep miml network. In *Thirty-First AAAI Conference on Artificial Intelligence*, pages 1884–1890.

[9] Gärtner, T., Flach, P. A., Kowalczyk, A., and Smola, A. J. (2002). Multi-instance kernels. In *International Conference on Machine Learning*, volume 2, pages 179–186.

[10] Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323.

[11] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[12] Huang, Z., Wang, X., Wang, J., Liu, W., and Wang, J. (2018). Weakly-supervised semantic segmentation network with deep seeded region growing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7014–7023.

[13] Ilse, M., Tomczak, J. M., and Welling, M. (2018). Attention-based deep multiple instance learning. In *International Conference on Machine Learning*, pages 2132–2141.

[14] Kandemir, M. and Hamprecht, F. A. (2015). Computer-aided diagnosis from weak supervision: A benchmarking study. *Computerized Medical Imaging and Graphics*, 42:44–50.

[15] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105.

[16] Li, C., Wang, X., Liu, W., Latecki, L. J., Wang, B., and Huang, J. (2019). Weakly supervised mitosis detection in breast histopathology images using concentric loss. *Medical Image Analysis*, 53:165–178.

[17] Li, Q., Wu, J., and Tu, Z. (2013). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 851–858.

[18] Li, Y., Wang, X., Liu, W., and Feng, B. (2018). Deep attention network for joint hand gesture localization and recognition using static rgb-d images. *Information Sciences*, 441:66–78.

[19] Pathak, D., Krahenbuhl, P., and Darrell, T. (2015a). Constrained convolutional neural networks for weakly supervised segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1796–1804.

[20] Pathak, D., Shelhamer, E., Long, J., and Darrell, T. (2015b). Fully convolutional multi-class multiple instance learning. In *3rd International Conference on Learning Representations, Workshop Track Proceedings*.

[21] Ramon, J. and De Raedt, L. (2000). Multi instance neural networks. In *Proceedings of the ICML-2000 workshop on attribute-value and relational learning*, pages 53–60.

[22] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large

scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252.

[23] Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., and Lillicrap, T. (2016). Meta-learning with memory-augmented neural networks. In *International Conference on Machine Learning*, pages 1842–1850.

[24] Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations*.

[25] Sirinukunwattana, K., e Ahmed Raza, S., Tsang, Y.-W., Snead, D. R., Cree, I. A., and Rajpoot, N. M. (2016). Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Transactions on Medical Imaging*, 35(5):1196–1206.

[26] Tang, P., Wang, X., Bai, S., Shen, W., Bai, X., Liu, W., and Yuille, A. L. (2018). Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–10.

[27] Tang, P., Wang, X., Bai, X., and Liu, W. (2017). Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2843–2851.

[28] Wang, M., Luo, C., Hong, R., Tang, J., and Feng, J. (2016). Beyond object proposals: Random crop pooling for multi-label image recognition. *IEEE Transactions on Image Processing*, 25(12):5678–5688.

[29] Wang, X., Yan, Y., Tang, P., Bai, X., and Liu, W. (2018). Revisiting multiple instance neural networks. *Pattern Recognition*, 74:15–24.

[30] Wang, X., Zhu, Z., Yao, C., and Bai, X. (2015). Relaxed multiple-instance svm with application to object discovery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1224–1232.

[31] Wei, X.-S., Wu, J., and Zhou, Z.-H. (2016). Scalable algorithms for multi-instance learning. *IEEE Transactions on Neural Networks and Learning Systems*, 28(4):975–987.

[32] Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52.

[33] Wu, J., Yu, Y., Huang, C., and Yu, K. (2015). Deep multiple instance learning for image classification and auto-annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3460–3469.

[34] Xu, X. and Frank, E. (2004). Logistic regression and boosting for labeled bags of instances. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 272–281. Springer.

[35] Xu, Y., Zhu, J.-Y., Eric, I., Chang, C., Lai, M., and Tu, Z. (2014). Weakly supervised histopathology cancer image segmentation and classification. *Medical Image Analysis*, 18(3):591–604.

[36] Zhang, M.-L. and Zhou, Z.-H. (2004a). Ensembles of multi-instance neural networks. In *International Conference on Intelligent Information Processing*, pages 471–474. Springer.

[37] Zhang, M.-L. and Zhou, Z.-H. (2004b). Improve multi-instance neural networks through feature selection. *Neural Processing Letters*, 19(1):1–10.

[38] Zhang, Q. and Goldman, S. A. (2001). Em-dd: An improved multiple-instance learning technique. In *Advances in Neural Information Processing Systems*, pages 1073–1080.

[39] Zhou, Z.-H., Sun, Y.-Y., and Li, Y.-F. (2009). Multi-instance learning by treating instances as non-iid samples. In *International Conference on Machine Learning*, pages 1249–1256.

[40] Zhou, Z.-H. and Zhang, M.-L. (2002). Neural networks for multi-instance learning. In *Proceedings of the International Conference on Intelligent Information Technology*, pages 455–459.

[41] Zhu, F. and Shao, L. (2014). Weakly-supervised cross-domain dictionary learning for visual recognition. *International Journal of Computer Vision*, 109(1-2):42–59.