Semi-supervised multi-instance learning with density peaks clustering

Mei Yang, Yu-Xuan Zhang, Fan Min, *Member, IEEE*

***Abstract*—The task of multi-instance learning (MIL) is to train a classifier to handle complicated data bags. A conspicuous feature of MIL is bag. Many methods have been developed based on different similarity measures between bags. In this paper, we propose semi-supervised multi-instance learning with density peaks clustering (SMDP) algorithm with three steps for this issue. First, we select the most n_c representative bags using a clustering based technique coupled with the Gaussian kernel and five distance measures. Three measures are off-the-shelf, while the other two are new. Second, each bag is transformed into the n_c-dimensional instance space using distances among bags. The value of the i-th feature of the new instance is the distance between the bag and the i-th representative bag. Third, the n_c-dimensional data table is employed to build a classifier. Result on sixteen data sets show that our algorithm is superior to state-of-the-art MIL algorithms especially on text and image data sets.**

***Index Terms*—Semi-supervised, Multi-instance learning, Density peaks clustering, Gaussian kernel.**

I. Introduction

Multi-instance learning (MIL) was introduced by Dietterich et al. [1] on their investigation into drug activity prediction. A MIL data set consists of multiple bags, each of which consists of multiple instances. Hence MIL is more general than traditional supervised learning, which is actually single-instance learning (SIL). The basic assumptions of MIL formulation are that bag and instances in it share the same label, and a bag receives a particular label if at least one of the instance in the bag possesses the label. For binary classification, a bag is positive if it contains at least one positive instance; otherwise it is negative. Moreover, in the most cases, we specify or predict the label of a bag instead of an instance. Due to its adaptability, MIL has been widely applied in diverse domains such as image classification [2–4], image annotation [5, 6], image retrieval [7–9], text classification [10–12], biological function annotation [4], medical diagnosis [13, 14], object tracking [15, 16] and sentiment analysis [17, 18].

This work is in part supported by the Natural Science Foundation of Sichuan Province under Grant number 2019YJ0314, and the Sichuan Province Youth Science and Technology Innovation Team under Grant number 2019JDTD0017. *(Corresponding author: Fan Min.)*

Mei Yang is with the School of Computer Science, Southwest Petroleum University, Chengdu 610500, China (e-mail: yangmei@swpu.edu.cn).

Yu-Xuan Zhang is with the School of Computer Science, Southwest Petroleum University, Chengdu 610500, China (e-mail: 201921000434@swpu.edu.com).

Fan Min is with the School of Computer Science, Southwest Petroleum University, Chengdu 610500, China (e-mail: minfan@swpu.edu.cn).

The authors would like to publish the paper on IJMLC or CAAI Transactions on Intelligence Technology.

The key issue of MIL is how to measure the similarity between two bags. Kernel based approaches [19–21] focus on designing a set-kernel based on the conventional kernel (e.g. Gaussian kernel) or using a existing set-kernel (e.g. fisher kernel). SVM based approach [11] modifies optimization objective function so that support vector machine can be employed to MIL. Clustering based approaches [22, 23] regard the bags as atomic data items and use some form of distance measures to calculate distance between two bags. All of these approaches explicitly or implicitly convert MIL problems into SIL problems. The simplest approach [11] is to treat all instances in a bag as the same label as the bag and solve it with SIL methods.

In this paper, we propose the semi-supervised multi-instance learning with density peaks clustering (SMDP) algorithm. Since the learning scenario is semi-supervised, the training set contains both labeled and unlabeled bags. We adopt the following algorithm framework with three stages. In the representative selection stage, n_c representative bags are selected from the training set. In the data conversion stage, each bag is converted to a n_c-dimensional instance according to its distance to each representative bag. To classify a bag, it is first converted to a n_c-dimensional instance, and its label is predicted by kNN according to labeled data. Naturally, this algorithm is able to classify both unlabeled training bags and new ones.

The first key issue of our algorithm is the selection of representatives. We design a technique inspired by the density peaks (DP) clustering algorithm [24]. The Gaussian kernel is employed to compute the density of each bag. The cutoff distance is set to a given ratio (e.g. 0.2) times the maximum distance between bags. The master of a bag is the closest bag with a higher density. The representativeness of each bag is then calculated as the product of its density and the distance to its master. The set of representatives is top-n_c bags with high representativeness for binary classification data sets. Especially for multi-class data sets, if we find representative bags based on all classes, some representative bags may come from the same class. This will disrupts the balance of the original class, and reduces final prediction accuracy. Our solution is to find a representative bags for each class and all unlabeled ones.

The second key issue is definition of the distance measure. We suppose two new strategies. For text data set, we exploit virtual Hausdorff distance measure which is inspired by the Simple-MI algorithm [25]. virtual center can represents a bag with the mean vector of all instances in the current bag. Therefore, the virtual Hausdorff distance between two bags is the distance between their mean vector. For image data sets,

we design the min-bias distance (d^{bia}). The bias $\mathbf{B}$ is the mean vector of all instances in the two bags. The distance between i-th bag and j-bag (d_{ij}^{bia}) is the minimum of the distances of all instances in the i-th bags to the bias. Since the distance is symmetric, only half of d_{ij}^{bia} should be computed. Compared with the existing distance [23, 26, 27], the advantages of two new distance measures include: 1) d^{vir} can get higher prediction accuracy on some benchmark data sets and text data sets; 2) d^{bia} can adapt well to image data sets; and 3) d^{vir} and d^{bia} both have lower time complexity.

Experiments are undertaken on five benchmark data sets, twenty text data sets and two image data sets. Results show that SMDP is highly competitive to the eight state-of-the-art MIL methods i.e., CCE [22], ISK [12], mi-FV [20], miGraph [10], MI-kernel [19], Mi-SVM & mi-SVM [11] and Simple-MI [25] in most MIL data sets, such as MUSK1, MUSK2, Fox, overwhelming majority of text data sets and image data sets.

The rest of the paper is organized as follows. We briefly review related work about MIL in Section II. Several crucial concepts will introduced in Section III, propose our algorithm in Section IV, report on our experiments in Section V, conclusions and further work finally in Section VI.

II. Related works

Many multi-instance learning (MIL) methods have been developed since the investigation of drug activity prediction [1]. To name a few, boosting algorithms [28–30], comparative study [25, 31], diverse density [8, 32, 33], decision tree [34], deep learning [35–37], fast bundle algorithm [38], k-nearest neighbor algorithms [27, 39, 40], logistic regression algorithms [41–43], neural network [14, 44, 45], rule learning algorithms [46, 47], and SVM algorithms [11, 48]. These MIL algorithms solve MIL problems in diverse ways. For example, Auer et al. [28] considered a boosting approach of MIL with balls as weak hypotheses in greater detail, and applied it to some multiple instance benchmark problems. Amores et al. [25] analyzed the performance of the fourteen methods across a variety of well-known databases, and also studied their behavior in synthetic scenarios in order to highlight their characteristics. Maron et al. [32] described a new general framework named Diverse Density. This framework is applied to learn a simple description of a person from a series of images containing that person, to a stock selection problem, and to the drug activity prediction problem.

A common approach for MIL is to map each bag into single vector which is handled by SIL methods. Gärtner et al. [19] proposed the MI-kernel algorithm which regarded each bag as a set of feature vectors and directly applied the set kernel . Andrews et al. [11] proposed the MI-SVM & mi-SVM algorithms. MI-SVM tries to generalized the notion of a margin to bags and aims at maximizing the bag margin directly; mi-SVM tries to maximized the usual pattern margin, or soft-margin, jointly over hidden label variables and a linear (or kernelized) discriminant function. Zhou et al. [22] proposed the CCE algorithm which collected all instances in the bags together and clustered them into n groups. Therefore Each bag was re-represented by n binary features and single-instance classifiers can be used to distinguish different classes of bags. Zhou et al. [10] proposed the miGraph algorithm which treated the instances in the bags with non-independent and non-identical distribution to obtained a better prediction accuracy. Amores [25] proposed the Simple-MI algorithm which showed that the extracted global bag-level information generally exhibits superior performance. It is the reason why some types of methods are more successful than others. Wei et al.[20] proposed mi-FV algorithms based on the fisher vector which can deal the large-scale MIL problems. Xu et al. [12] proposed the ISK algorithm which investigated data-dependent kernels that are derived directly from data. It speeded up the set-kernel computation significantly.

There are many MIL methods with semi-supervised scenario. For example, Zhou et al. [49] established a bridge between two branches by showing that multi-instance learning can be viewed as a special case of semi-supervised learning. Base on this, the experiments of MissSVM showed that solving multi-instance problems from the view of semi-supervised learning is feasible. Wang et al. [50] proposed a novel single object tracking-by-detection tracker which combined semi-supervised learning, multiple instance learning and the Bayesian theorem. Cheplygina et al. [51] given an overview of semi-supervised, multiple instance and transfer learning in medical imaging, both in diagnosis or segmentation tasks.

In the single-instance learning, DP clustering algorithm [24] is widely used. Du et al. [52] proposed a DP clustering based on k nearest neighbors, and provided another option for local density calculations. Wang et al. [53] developed the active learning through DP clustering, and got a higher classification accuracy using the same number of labeled data. Du et al. [54] integrated the speed of DP clustering algorithm with the robustness of fuzzy joint points algorithm, and identified the different values of the neighborhood membership degrees of the points.

III. Preliminaries

This section describes some basis of this work, including the data model in Section III-A, the problem statement in Section III-B, Density peaks (DP) clustering algorithm in Section III-C and existing distance measures in Section III-D. Table I lists some notations used throughout the paper.

A. The data model

Let $\mathcal{X} = R^p$ be the p-dimensional instances space, and $\mathcal{B} = \{(\mathbf{B}_1, y_1), \ldots, (\mathbf{B}_l, y_l), \mathbf{B}_{l+1}, \ldots, \mathbf{B}_N\}$ be a data set with l labeled bags and $(N - l)$ unlabeled bags. $\mathbf{B}_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{i,n_i}\}$ is a bag with n_i instances for $i \in [1..N]$, where $\mathbf{x}_{ij} = [x_{ij1}, \ldots, x_{ijp}] \in \mathcal{X}$. $y_i \in \mathcal{Y} = \{-1, +1\}$ is the label of $\mathbf{B}_i$ for $i \in [1..l]$. We also denote the labeled data set by $\mathcal{B}_l = \{(\mathbf{B}_1, y_1), \ldots, (\mathbf{B}_l, y_l)\}$, and the unlabeled data set by $\mathcal{B}_u = \{\mathbf{B}_{l+1}, \ldots, \mathbf{B}_N\}$.

In addition, $\mathbf{B}_i$ is positive if it contains at least one positive instance; otherwise it is negative.

TABLE I
NOTATIONS.

Notation	Meaning	Comments
$\mathcal{X}$	The instances space	Data model
$\mathcal{B}$	The data set	Input / Data model
N	The size of $\mathcal{B}$	Data model
c	The number of classes	Data model
$\mathcal{B}_i, i \in [1..c]$	The i-th class bags	Data model
N_i	The size of $\mathcal{B}_i$	Data model
$\mathcal{B}_l$	The labeled bags	Data model
l	The size of $\mathcal{B}_l$	Data model
$\mathcal{B}_u$	The unlabeled bags	Data model
$\mathbf{B}_i$	The i-th bag	Data model
$y_i, i \in [1..l]$	The label of $\mathbf{B}_i$	Data model
$\mathbf{x}_{ij}$	The j-th instance of $\mathbf{B}_i$	Data model
d_{ij}	The distance between $\mathbf{B}_i$ and $\mathbf{B}_j$	Algorithm variable
d_c	The cutoff distance	Algorithm variable
ρ_i	The local density of $\mathbf{B}_i$	Algorithm variable
δ_i	The distance to the master of $\mathbf{B}_i$	Algorithm variable
n_c	The number of clustering center	Algorithm variable
μ	The proportion of n_c to N	Algorithm settings
r	The ratio for d_c	Algorithm settings
$y'_i, i \in [l+1..N]$	The predicted label for $\mathbf{B}_i \in \mathcal{B}_u$	Output

B. Problem statement

In the close-world scenario, the learning task is to predict unlabeled bags. The problem is stated as follows.

Problem 1: **Semi-supervised multi-instance learning**
Input: $\mathcal{B} = \{(\mathbf{B}_1, y_1), \ldots, (\mathbf{B}_l, y_l), \mathbf{B}_{l+1}, \ldots, \mathbf{B}_N\}$.
Output: y'_i, $i \in [l+1..N]$.
Optimization objective: $\max \frac{|\{l \le i \le N | y'_i = y_i\}|}{N-l}$.

Here y_i and y'_i are the actual and the predicted labels, respectively. Hence the optimization objective is maximizing the prediction accuracy. Naturally, in the open-world scenario, the prediction accuracy can also be calculated as long as the testing set is given.

C. Density peaks (DP) clustering algorithm

DP clustering algorithm was widely applied in the field of SIL through the computation of the density for each instance to cluster. The cluster centers of instances are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities. The distances d^*_{ij} between two instances are firstly measured by some form of distance measures. For each instance $\mathbf{x}_i$, DP clustering computes both its local density ρ_i and its distance δ_i to the closest instance with a higher density according to d^*_{ij}.

There are two popular kernels for calculating ρ_i, i.e., cutoff kernel and Gaussian kernel. The cutoff kernel is defined as

$$\rho_i^c = \sum_{j \neq i} \mathcal{F}\left(d^*_{ij} - d_c\right), \tag{1}$$

where $\mathcal{F}(x)$ is

$$\mathcal{F}(x) = \begin{cases} 1, x < 0; \\ 0, x \geq 0. \end{cases} \tag{2}$$

The Gaussian kernel is defined as

$$\rho_i^g = \sum_{j \neq i} e^{-\left(\frac{d^*_{ij}}{d_c}\right)^2}. \tag{3}$$

The distance to master δ_i is defined as

$$\delta_i = \begin{cases} \max\left(d_{ij}\right), \rho_i = \max\limits_{j \in [1..N]}(\rho_j); \\ \min\limits_{j \in [1..N]: \rho_j > \rho_i}(d^*_{ij}), \text{otherwise}. \end{cases} \tag{4}$$

Especially, the δ_i of the instance with the highest density is the maximum one. Instances with both high ρ_i and high δ_i tend to be cluster centers. Therefore, λ_i is defined as

$$\lambda_i = \rho_i * \delta_i. \tag{5}$$

D. Existing distance measures

There are three off-the-shelf distance measures, i.e., average Hausdorff distance(d^{ave}) [23], maximal Hausdorff distance(d^{max}) [26] and minimal Hausdorff distance(d^{min}) [27]. Formally, for each bag-pair $\mathbf{B}_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{i,n_i}\}$ and $\mathbf{B}_j = \{\mathbf{x}_{j1}, \ldots, \mathbf{x}_{j,n_j}\}$, the distance d_{ij} in d^{ave}, d^{max} and d^{min} are

$$d_{ij}^{ave} = \frac{\sum\limits_{\mathbf{x}_{ia}} \min\limits_{\mathbf{x}_{jb}} d(\mathbf{x}_{ia}, \mathbf{x}_{jb}) + \sum\limits_{\mathbf{x}_{jb}} \min\limits_{\mathbf{x}_{ia}} d(\mathbf{x}_{ia}, \mathbf{x}_{jb})}{n_i + n_j}, \tag{6}$$

$$d_{ij}^{max} = \max\left\{\max_{\mathbf{x}_{ia}} \min_{\mathbf{x}_{jb}} d(\mathbf{x}_{ia}, \mathbf{x}_{jb}), \max_{\mathbf{x}_{jb}} \min_{\mathbf{x}_{ia}} d(\mathbf{x}_{ia}, \mathbf{x}_{jb})\right\}, \tag{7}$$

$$d_{ij}^{min} = \min_{\mathbf{x}_{ia}, \mathbf{x}_{jb}} d(\mathbf{x}_{ia}, \mathbf{x}_{jb}), \tag{8}$$

where $d(\mathbf{x}_{ia}, \mathbf{x}_{jb})$ is the distance (such as Euclidean distance) between $\mathbf{x}_{ia}$ and $\mathbf{x}_{jb}$. According to Eqs. (6) (7) (8), the time complexity is $O(pb^2N^2)$, where $b = \max n_i$, which is rather high.

IV. THE ALGORITHM

In this section, we first present the SMDP algorithm framework in Section IV-A. Two key issues of the algorithm are analyzed, including density peaks clustering for SMDP in Section IV-B and two new distance measures in Section IV-C.

A. Algorithm description

Algorithm 1 lists the SMDP framework. The initialization is implemented in Line 1.

Basing on the DP algorithm, finding the most representative n_c bags as cluster centers are implemented in Lines 2-19. Specifically, Lines 3-8 handle binary classification data sets. Line 4 calculates the number of clustering centers. Lines 5-8 find the index of n_c cluster centers. Concretely, the index of largest λ_j is searched in entire data set, and merged into C. Lines 10-18 is for multi-class data sets. Line 10 set the number of clustering to c plus one. Lines 11-15 traverse each class to get a representative bag. Lines 16-18 find a representative bag in all unlabeled ones.

Therefore, Lines 20-24 transform each bag into the n_c-dimensional instances space R^{n_c}. All bags are traversed again in the for loop. We explicitly map every bag to a feature vector,

Algorithm 1 Semi-supervised multi-instance learning with density peaks (SMDP)

Input: The data set $\mathcal{B}$, r for d_c ratio, the proportion μ of n_c to N.
Output: $y'_i, i \in [l+1..N]$.

1: $C = \emptyset$;//The set of representative bags
//Step1. Find the most representative n_c bags as cluster centers
2: **if** $(c == 2)$ **then**
3: $d_c = r \times \max\{d_{ij}\}$, where $\mathbf{B}_i, \mathbf{B}_j \in \mathcal{B}$;
4: $n_c = \lfloor \mu \times N \rfloor$;
5: **for** $(i \in [1..n_c])$ **do**
6: $c_i = \arg\max_{j \in [1..N] \setminus C} \lambda_j$, where $\mathbf{B}_j \in \mathcal{B}$;
7: $C = C \cup \{c_i\}$;
8: **end for**
9: **else**
10: $n_c = c + 1$;
11: **for** $(k \in [1..c])$ **do**
12: $d_c = r \times \max\{d_{ij}\}$, where $\mathbf{B}_i, \mathbf{B}_j \in \mathcal{B}_k$;
13: $c_k = \arg\max \lambda_{k'}$, where $\mathbf{B}_{k'} \in \mathcal{B}_k$;
14: $C = C \cup \{c_k\}$;
15: **end for**
16: $d_c = r \times \max\{d_{ij}\}$, where $\mathbf{B}_i, \mathbf{B}_j \in \mathcal{B}_u$;
17: $c_{n_c} = \arg\max \lambda_{k'}$, where $\mathbf{B}_{k'} \in \mathcal{B}_u$;
18: $C = C \cup \{c_{n_c}\}$;
19: **end if**
//Step2. Transform each bag into the n_c-dimensional instance space R^{n_c}
20: $\mathbf{B}^* = \emptyset$;
21: **for** $(i \in [1..N])$ **do**
22: $\mathbf{z}_i = [z_{i1}, \ldots, z_{i,n_c}]$ where $z_{ij} = d_{i,c_j}$; //The distance to each center
23: $\mathbf{B}^* = \mathbf{B}^* \cup \{\mathbf{z}_i\}$;
24: **end for**
//Step3. Classify
25: Build a classifier using $\mathbf{B}^*_l = \{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_l, y_l)\}$;
26: Classify instances in $\mathbf{B}^*_u = \{\mathbf{z}_{l+1}, \ldots, \mathbf{z}_N\}$ to obtain $y'_{l+1}, \ldots, y'_N$;

i.e., z_i, whose i-th feature corresponds to the distance between the bag ($\mathbf{B}_i$) and the i-th clustering centers ($\mathbf{B}_{c_i}$), as indicated in Line 22. Obviously, z_i is a n_c-dimensional feature vector, which is merged into $\mathbf{B}^*$, as indicated in Line 23. $\mathbf{B}^*$ will be employed to build a classifier in last step.

Classify strategy is same as [23], as indicated in Lines 25-26. The feature vectors corresponding to the labeled bags are used as training set. The others are used as testing set. Then we utilize the give learning algorithm to build a classifier from the transformed feature vectors, i.e., $\mathbf{B}^*_l$, which are associated with the corresponding bag's label. Finally, the instances in testing set $\mathbf{B}^*_u = \{\mathbf{z}_{l+1}, \ldots, \mathbf{z}_N\}$ will classified to obtain $y'_i \in \mathcal{Y} = \{-1, +1\}$, $i \in [l+1..N]$.

B. Density Peaks clustering for SMDP

As shown in Lines 2-19 of the algorithm, we design a technique inspired by the DP clustering to select the n_c representatives. By regarding bags as atomic data items, the distances d_{ij} between two bags are calculated in some form of distance measure. Both ρ_i and δ_i are calculated basing on d_{ij}. The cutoff kernel produces an integer value density. It may produce the same density for different bags and lead to different choices while sorting. This is rarely faced by the Gaussian kernel. Therefore, Eq. (3) was employed to calculate the ρ_i.

The master of a bag is its closest bag with a higher density. The representativeness of the bag is calculated by multiplying its local density ρ_i by the distance to its master δ_i, i.e., λ_i. For binary classification, the cutoff distance d_c is set to a given ratio times the maximum distance between bags in entire data set. The representatives are recognized as the top-n_c bags with high representativeness. For multi-class data sets, we find a representative bag for each class in which all bags have the same label. And we find a representative bag from all unlabeled ones. Therefore, the d_c of i-th class is set to a given ratio times the maximum distance between bags in the class or unlabeled ones. In this way, we can maintain the characteristics of each class as much as possible. And it can effectively avoid that some representative bags may come from a same class and maximize the accuracy of the final predictions.

C. Two new distance measures

It is noteworthy that different distance measures are applicable to different data sets because of the geometric distribution inequality of instances in the bag. For text data sets, inspired by the Simple-MI algorithm [25], the center instance in a bag could be the representative of the bag, which is defined as

$$\overline{\mathbf{x}_i} = \frac{\sum_{j=1}^{n_i} \mathbf{x}_{ij}}{n_i}. \tag{9}$$

It is essentially a virtual instance. Accordingly, a newly distance measure named virtual Hausdorff distance (d^{vir}) is proposed. It is calculated with $\overline{\mathbf{x}_i}$ instead of $\mathbf{B}_i$. That is

$$d^{vir}_{ij} = d(\overline{\mathbf{x}_i}, \overline{\mathbf{x}_j}), \tag{10}$$

i.e., the distance between $\mathbf{B}_i$ and $\mathbf{B}_j$ is replaced by the distance of $\overline{\mathbf{x}_i}$ and $\overline{\mathbf{x}_j}$.

For the image data sets, we propose the min-bias distance (d^{bia}), which is defined as

$$d^{bis}_{ij} = \min_{i=0}^{n_i} d(\mathbf{x}_{ia}, bias\ \mathbf{B}), \tag{11}$$

where $bias\ \mathbf{B} = \frac{\sum_{\mathbf{x}_{ia}} \mathbf{x}_{ia} + \sum_{\mathbf{x}_{jb}} \mathbf{x}_{jb}}{n_i + n_j}$, $d(\mathbf{x}_{ia}, bias\ \mathbf{B})$ is the distance between $\mathbf{x}_{ia}$ and $bias\ \mathbf{B}$. Note that, since the distance is symmetric, only half of d^{bia}_{ij} should be calculated.

Experimentally, d^{vir} and d^{bia} can adapt well to text data sets and image data sets, respectively. Furthermore, compared with Eqs. (6) (7) (8), in the same data set and assumption, the time complexity of which is reduced to $O(pbN^2)$, where $b =$

$\max n_i$. These issues will be discussed in detail in Sections V-C and V-D.

V. Experiments

In this section, we report the results of experiments to analyze the effectiveness for SMDP algorithm. Through the experiments, we aim to answer

1) Is the SMDP algorithm more efficient than some state-of-the-art MIL algorithms?
2) What is the benefits of multiple distance measures and new distance measures?

The experimental environment is a Windows 10 64-bit operating system, 4 GB memory, Intel (R) Core 2 Quad CPU Q9500@2.83 GHz. The source code with a graphical user interface (GUI) written in Java is available at github.com/InkiInki /Java/Multi-Instance.

A. Experimental setup

As shown in Algorithm 1, several parameters should be determined. For clustering, the default setting of d_c ratio is 0.2. For binary classification data sets, the proportion μ of n_c to N is ranges from 20% to 100% with an interval of 20%. For multi-class data sets the number of clustering center n_c is set to c plus one. The distance between two instances is measured by Euclidean distance. In the prediction phase, we learn from the transformed feature vectors (i.e., $\mathbf{B}_l^*$) by employing the kNN algorithm.

Experiments are undertaken on twenty-seven MIL data sets from different fields. These include five benchmark data sets [1, 11] (i.e., Musk1, Musk2, Elephant, Fox and Tiger), two image data sets [55, 56] (i.e., 2000-image and 1000-image) and twenty text data sets [10] commonly used in MIL. Since different distance measures are applicable to different data sets, d^{ave} [23] and d^{max} [26] are used in all data sets; d^{min} [27] is used in benchmark data sets and image data sets; d^{vir}(IV-C) is uesd in the benchmark data sets and text data sets; d^{bia} (IV-C) is used in the image data sets.

In benchmark data sets and image data sets, the compared algorithms are CCE [22], ISK [12], mi-FV [20], miGraph [10], MI-kernel [19], Mi-SVM & mi-SVM [11] and Simple-MI [25]. In text data set, the compared algorithms are ISK [12], miGraph [10], MI-kernel [19] and mi-FV [20]. The top-level properties of these methods are summarized in Table II. LinearSVM denotes that a standard SVM with linear kernel and maps instances are used. SVM denotes that a standard SVM with a designed kernel and original instances are used. SVM* denotes that a modified optimization objective function is used. kNN denote that k-nearest neighbors is used.

Experiments are repeated ten times with different data sets. We report the average prediction accuracy of 10-fold cross validation (10CV) and its standard deviation. The value following $'\pm'$ gives the standard deviation and the "best" results are shown in bold face.

TABLE II
Properties of MIL methods.

Algorithm	Category	Base Kernel	Classifier
CCE	Clustering Based	Linear Kernel	LinearSVM
ISK	Set-Kernel Based	Isolation Kernel	LinearSVM
mi-FV	Set-Kernel Based	Fisher Kernrl	LinearSVM
miGraph	Set-Kernel Based	Gaussian kernel	SVM
MI-Kernel	Set-Kernel Based	Gaussian kernel	SVM
Mi-SVM	SVM based	Gaussian kernel	SVM*
mi-SVM	SVM based	Gaussian kernel	SVM*
Simple-MI	Baseline	Gaussian kernel	SVM
SMDP	Clustering Based	Gaussian kernel	kNN

TABLE III
Characteristics of benchmark data sets.

Dataset	MUSK1	MUSK2	Elephant	Fox	Tiger
Dimensionality	166	166	230	230	230
No.bags	92	102	200	200	200
No. positive bags	47	39	100	100	100
No. negative bags	45	63	100	100	100
No. instances	476	6,598	1,391	1,320	1,220
Max No. instances per bag	40	1,044	13	13	13
Min No. instances per bag	2	1	2	2	1
Ave. No. instances per bag	5.17	64.69	13.91	13.20	12.20

B. Benchmark data sets

Table III lists the characteristics of five benchmark data sets. Detailed characteristics of these data sets are summarized in [1, 11].

Table IV lists the empirical results on benchmark data sets. As the empirical results shown, SMDP achieves good performance. For each data sets, the best classification performances of SMDP are obtained in different distance measures and proportion μ of n_c to N, respectively. In detail, d^{ave} performs best on MUSK1 and MUSK1, d^{vir} performs best on elephant and d^{min} performs best on fox and tiger.

TABLE IV
Experimental results(*10CV*) of SMDP on the benchmark data sets with different bag distance measures and different number of clustered groups($= \mu * |\mathcal{B}|$).

Data Set	Distance Measure	$\mu = 20\%$	$\mu = 40\%$	$\mu = 60\%$	$\mu = 80\%$	$\mu = 100\%$
MUSK1	d^{ave}	$87.0_{\pm 2.3}$	$\mathbf{92.1_{\pm 2.6}}$	$91.2_{\pm 1.1}$	$89.8_{\pm 2.1}$	$89.1_{\pm 1.5}$
	d^{max}	$85.5_{\pm 2.0}$	$87.0_{\pm 2.1}$	$86.0_{\pm 1.4}$	$88.0_{\pm 1.5}$	$86.1_{\pm 2.8}$
	d^{min}	$87.6_{\pm 1.3}$	$86.0_{\pm 1.6}$	$88.2_{\pm 1.0}$	$88.6_{\pm 1.3}$	$88.8_{\pm 2.3}$
	d^{vir}	$80.2_{\pm 1.4}$	$82.3_{\pm 2.6}$	$83.4_{\pm 2.2}$	$84.8_{\pm 2.7}$	$84.9_{\pm 2.1}$
MUSK2	d^{ave}	$89.8_{\pm 1.8}$	$\mathbf{90.4_{\pm 2.0}}$	$89.9_{\pm 1.8}$	$88.9_{\pm 1.6}$	$86.9_{\pm 1.9}$
	d^{max}	$83.6_{\pm 1.8}$	$88.7_{\pm 2.2}$	$86.7_{\pm 1.3}$	$84.0_{\pm 2.1}$	$82.7_{\pm 2.2}$
	d^{min}	$86.6_{\pm 1.7}$	$89.2_{\pm 1.0}$	$88.7_{\pm 1.2}$	$87.8_{\pm 1.9}$	$87.5_{\pm 1.2}$
	d^{vir}	$86.9_{\pm 2.4}$	$88.8_{\pm 2.0}$	$85.5_{\pm 1.9}$	$79.9_{\pm 1.7}$	$78.3_{\pm 1.9}$
Elephant	d^{ave}	$73.3_{\pm 1.3}$	$79.5_{\pm 1.5}$	$81.0_{\pm 1.8}$	$78.0_{\pm 1.3}$	$80.8_{\pm 2.2}$
	d^{max}	$66.3_{\pm 1.2}$	$70.0_{\pm 1.3}$	$72.3_{\pm 1.7}$	$73.8_{\pm 1.5}$	$76.3_{\pm 1.6}$
	d^{min}	$73.4_{\pm 1.4}$	$71.9_{\pm 1.1}$	$76.6_{\pm 1.0}$	$81.2_{\pm 1.3}$	$81.8_{\pm 1.6}$
	d^{vir}	$75.0_{\pm 1.2}$	$77.3_{\pm 1.3}$	$\mathbf{83.4_{\pm 1.0}}$	$82.9_{\pm 0.9}$	$82.8_{\pm 1.0}$
Fox	d^{ave}	$62.6_{\pm 2.4}$	$64.9_{\pm 1.9}$	$64.6_{\pm 1.7}$	$61.2_{\pm 2.0}$	$63.9_{\pm 1.0}$
	d^{max}	$59.9_{\pm 1.6}$	$56.6_{\pm 1.1}$	$60.4_{\pm 1.2}$	$60.9_{\pm 1.6}$	$58.6_{\pm 1.9}$
	d^{min}	$61.2_{\pm 1.8}$	$61.6_{\pm 1.6}$	$62.2_{\pm 1.6}$	$\mathbf{66.6_{\pm 1.6}}$	$64.5_{\pm 1.9}$
	d^{vir}	$61.3_{\pm 1.8}$	$58.0_{\pm 2.0}$	$63.2_{\pm 1.7}$	$61.1_{\pm 1.9}$	$64.6_{\pm 1.2}$
Tiger	d^{ave}	$68.6_{\pm 1.9}$	$72.1_{\pm 0.9}$	$71.5_{\pm 2.1}$	$73.4_{\pm 1.5}$	$73.2_{\pm 1.2}$
	d^{max}	$65.8_{\pm 1.6}$	$66.6_{\pm 1.3}$	$70.4_{\pm 1.2}$	$69.1_{\pm 1.5}$	$71.2_{\pm 1.8}$
	d^{min}	$72.6_{\pm 2.3}$	$74.3_{\pm 1.1}$	$76.2_{\pm 1.1}$	$75.4_{\pm 1.8}$	$\mathbf{76.7_{\pm 2.0}}$
	d^{vir}	$69.8_{\pm 2.1}$	$70.1_{\pm 1.5}$	$75.9_{\pm 1.2}$	$76.0_{\pm 2.0}$	$73.8_{\pm 1.4}$

Table V compares SMDP with 8 popular algorithms. SMDP

achieves the best accuracy in MUSK1, MUSK2 and fox data sets, and the relatively better accuracy than the remaining data sets.

TABLE V
ACCURACY(%) ON FIVE BENCHMARK DATA SETS.

Algorithm	MUSK1	MUSK2	Elephant	Fox	Tiger
CCE	$84.1_{\pm 1.4}$	$71.2_{\pm 1.7}$	$79.5_{\pm 1.2}$	$59.1_{\pm 1.3}$	$76.9_{\pm 1.1}$
ISK	$89.9_{\pm 0.9}$	$85.1_{\pm 1.1}$	$\mathbf{89.2_{\pm 0.6}}$	$61.5_{\pm 0.7}$	$\mathbf{82.6_{\pm 0.7}}$
mi-FV	$90.9_{\pm 4.2}$	$88.4_{\pm 3.9}$	$85.2_{\pm 2.7}$	$62.1_{\pm 3.1}$	$80.6_{\pm 5.4}$
miGraph	$88.7_{\pm 1.1}$	$87.8_{\pm 1.3}$	$85.3_{\pm 0.6}$	$61.3_{\pm 0.8}$	$81.6_{\pm 0.8}$
MI-Kernel	$88.1_{\pm 1.3}$	$85.0_{\pm 1.2}$	$84.2_{\pm 0.8}$	$60.6_{\pm 0.6}$	$80.9_{\pm 1.0}$
Mi-SVM	$77.8_{\pm 0.8}$	$84.1_{\pm 0.9}$	$81.3_{\pm 1.2}$	$59.4_{\pm 0.9}$	$79.8_{\pm 1.1}$
mi-SVM	$87.4_{\pm 0.7}$	$83.6_{\pm 1.0}$	$82.1_{\pm 0.6}$	$58.2_{\pm 0.5}$	$78.8_{\pm 0.9}$
Simple-MI	$85.7_{\pm 0.4}$	$83.2_{\pm 1.2}$	$81.8_{\pm 0.8}$	$57.7_{\pm 0.5}$	$79.2_{\pm 0.6}$
SMDP	$\mathbf{92.1_{\pm 2.6}}$	$\mathbf{90.4_{\pm 2.0}}$	$83.4_{\pm 1.0}$	$\mathbf{66.6_{\pm 1.6}}$	$76.7_{\pm 2.0}$

C. Text data sets

Twenty text data sets were derived from the 20 *Newsgroups corpus* [10]. For each data set, the size of bags is 100, the number of positive bags is the same as the number of negative bags.

Figure 1 shows the empirical results for twenty text data sets using SMDP with d^{ave} $(ave), d^{max}$ (max) and d^{vir} (vir). Each subgraph shows the performance in a text data set. The ordinate is average accuracy, and the abscissa is the proportion μ of n_c to N. Solid line, dashed line and dotted line are the performance of SMDP with d^{ave}, d^{max} and d^{vir}, respectively. The maximum accuracy and standard deviation corresponding each distance measure are shown in the legend. In particular, d^{ave} has a relatively large advantage in the data set of *comp.window.x*, d_{max} in the data set of *rec.autos*, d_{vir} in the data set of *comp.sys.ibm* and so on.

Table VI compares SMDP with four methods (i.e., ISK, miGraph, MI-Kernel and miFV). Concretely, SMDP achieves the best accuracy in most of these text data sets. SMDP is better than MI-Kernel and miFV on all 20 data sets, and better than ISK on 14 out of 20 data sets, and better than miGraph on 17 out of 20 data sets. In the end, the number of win/tie/lose of SMDP versus other methods is 11/0/9.

D. Image data sets

The data sets 2000-Image and 1000-Image [55, 56] contain twenty and ten categories of COREL images. There are 100 images in each category. Each image is regarded as a bag and the regions of interest in an image are regarded as instances described by nine features.

Table VII lists the empirical results on image data sets. The results show that d^{bia} can adapt well to image data sets and the others can not.

Table VIII compares SMDP with eight popular algorithms. SMDP achieves the best accuracy on 2000-image and 1000-image.

E. Runtime comparison of different distance measures

We compare the runtime of five distance measures on five benchmark data sets. As shown in Fig. 2, the abscissa of each subgraph is distance measures. The ordinate is the runtime (ms) of different distance measures. The results show that the time cost of both d^{vir} and d^{bia} is lower than the other distance measures.

TABLE VI
ACCURACY(%) ON TEXT DATA SETS

Dataset	ISK	MI-Kernel	miGragh	miFV	SMDP
alt.atheism	$83.2_{\pm 1.0}$	$56.5_{\pm 1.7}$	$82.1_{\pm 1.1}$	$71.4_{\pm 0.9}$	$\mathbf{88.1_{\pm 1.0}}$
comp.graphics	$\mathbf{85.3_{\pm 0.9}}$	$51.7_{\pm 1.3}$	$84.2_{\pm 0.9}$	$57.2_{\pm 0.6}$	$84.5_{\pm 1.4}$
comp.os.ms	$67.5_{\pm 1.6}$	$49.6_{\pm 2.1}$	$67.4_{\pm 1.3}$	$55.6_{\pm 0.4}$	$\mathbf{72.6_{\pm 2.1}}$
comp.sys.ibm	$76.7_{\pm 1.5}$	$53.1_{\pm 1.6}$	$78.3_{\pm 0.8}$	$57.7_{\pm 1.8}$	$\mathbf{88.1_{\pm 1.8}}$
comp.sys.mac	$80.8_{\pm 2.1}$	$50.8_{\pm 2.2}$	$\mathbf{83.0_{\pm 1.9}}$	$54.1_{\pm 0.6}$	$82.8_{\pm 1.5}$
comp.window.x	$82.4_{\pm 1.0}$	$51.7_{\pm 1.3}$	$80.6_{\pm 1.1}$	$66.0_{\pm 0.5}$	$\mathbf{84.6_{\pm 1.4}}$
misc.forsale	$70.1_{\pm 1.3}$	$55.3_{\pm 1.7}$	$69.3_{\pm 1.4}$	$62.8_{\pm 1.1}$	$\mathbf{71.4_{\pm 2.3}}$
rec.autos	$79.9_{\pm 1.7}$	$52.7_{\pm 1.9}$	$\mathbf{84.6_{\pm 1.7}}$	$59.7_{\pm 1.1}$	$81.0_{\pm 1.7}$
rec.motorcycles	$84.1_{\pm 1.4}$	$59.3_{\pm 1.3}$	$82.7_{\pm 1.3}$	$74.8_{\pm 1.4}$	$\mathbf{84.8_{\pm 1.1}}$
rec.sport.baseball	$86.3_{\pm 1.6}$	$54.0_{\pm 2.1}$	$\mathbf{88.8_{\pm 1.5}}$	$78.2_{\pm 1.8}$	$85.8_{\pm 1.3}$
rec.sport.hockey	$85.5_{\pm 1.7}$	$51.2_{\pm 1.8}$	$90.1_{\pm 1.2}$	$79.5_{\pm 1.0}$	$\mathbf{92.1_{\pm 1.3}}$
sci.crypt	$\mathbf{80.6_{\pm 1.3}}$	$54.6_{\pm 1.4}$	$76.8_{\pm 1.5}$	$61.3_{\pm 2.1}$	$78.8_{\pm 1.7}$
sci.electronics	$93.9_{\pm 0.8}$	$54.7_{\pm 1.7}$	$92.9_{\pm 0.7}$	$52.8_{\pm 0.9}$	$\mathbf{93.9_{\pm 0.5}}$
sci.med	$83.8_{\pm 1.8}$	$50.3_{\pm 1.8}$	$83.9_{\pm 1.6}$	$75.3_{\pm 1.9}$	$\mathbf{86.0_{\pm 0.9}}$
sci.religion	$83.7_{\pm 1.1}$	$52.7_{\pm 1.4}$	$82.5_{\pm 1.0}$	$65.2_{\pm 1.4}$	$83.1_{\pm 0.8}$
sci.space	$\mathbf{83.3_{\pm 1.7}}$	$54.4_{\pm 1.9}$	$83.1_{\pm 1.5}$	$74.5_{\pm 1.8}$	$\mathbf{84.1_{\pm 2.0}}$
talk.politics.guns	$\mathbf{79.7_{\pm 1.7}}$	$52.8_{\pm 1.5}$	$78.2_{\pm 1.9}$	$64.6_{\pm 0.7}$	$77.9_{\pm 1.5}$
talk.politics.mideast	$\mathbf{82.9_{\pm 1.6}}$	$53.6_{\pm 1.7}$	$81.6_{\pm 1.3}$	$73.7_{\pm 1.0}$	$82.5_{\pm 1.4}$
talk.politics.misc	$70.8_{\pm 1.5}$	$51.7_{\pm 1.8}$	$74.4_{\pm 1.4}$	$70.6_{\pm 1.6}$	$\mathbf{80.3_{\pm 0.8}}$
talk.religion.misc	$\mathbf{81.1_{\pm 1.3}}$	$53.8_{\pm 1.4}$	$76.2_{\pm 1.4}$	$73.4_{\pm 1.3}$	$74.9_{\pm 1.9}$
win/tie/loss	6/0/14	0/0/20	3/0/17	0/0/20	11/0/9

TABLE VII
EXPERIMENTAL RESULTS(*10CV*) OF SMDP ON THE IMAGE DATA SETS WITH DIFFERENT BAG DISTANCE MEASURES.

Datasets	d^{ave}	d^{max}	d^{min}	d^{bia}
2000-image	$49.1_{\pm 1.4}$	$38.3_{\pm 3.0}$	$44.3_{\pm 2.5}$	$\mathbf{81.2_{\pm 2.5}}$
1000-image	$64.6_{\pm 4.0}$	$50.8_{\pm 4.6}$	$57.7_{\pm 7.0}$	$\mathbf{89.5_{\pm 2.5}}$

TABLE VIII
ACCURACY (%) ON IMAGE DATA SETS.

Algorithm	2000-Image	1000-Image
CCE	$62.8_{\pm 1.3}$	$69.3_{\pm 1.5}$
ISK	$75.2_{\pm 0.6}$	$85.1_{\pm 1.1}$
mi-FV	$71.4_{\pm 0.6}$	$83.7_{\pm 1.2}$
miGraph	$72.1_{\pm 0.7}$	$84.5_{\pm 1.0}$
MI-Kernel	$73.2_{\pm 0.9}$	$83.7_{\pm 1.5}$
Mi-SVM	$66.9_{\pm 1.1}$	$76.4_{\pm 1.3}$
mi-SVM	$67.4_{\pm 1.1}$	$78.3_{\pm 1.3}$
Simple-MI	$63.7_{\pm 0.6}$	$70.5_{\pm 0.8}$
SMDP	$\mathbf{81.2_{\pm 2.5}}$	$\mathbf{89.5_{\pm 2.5}}$

F. Discussions

Now we can answer the questions proposed at the beginning of this section.

1) SMDP is more accurate than state-of-the-art methods on several MIL data sets, including three benchmark data sets, more than half text data sets and two image data sets. This is validated by Table V, VI and VIII. Unfortunately, it is significantly worse than some other algorithms such as ISK on the the elephant and tiger data sets. The reason may be that DP clustering techniques and five distance measures do not perform well on those data sets.
2) Different distance measures are appropriate for different data sets. This is validated by Table IV & VII and Fig.

Fig. 1. Experimental results of SMDP on the text data sets with different distance measures and different number of clustered groups($= \mu * |\mathcal{B}|$).

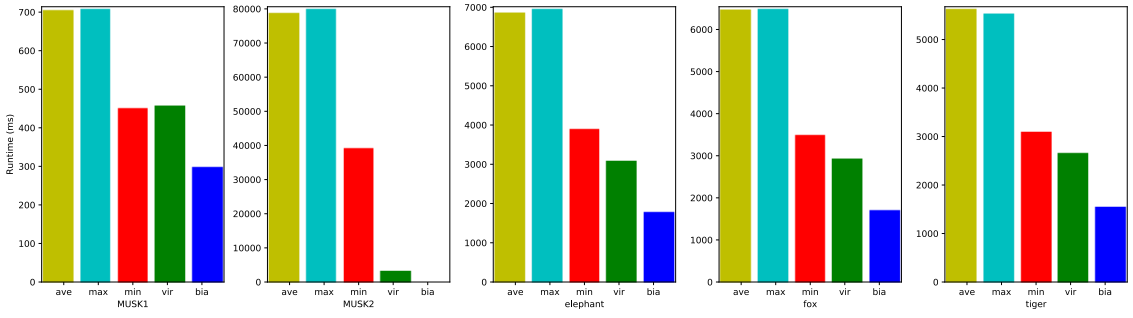

Fig. 2. Runtime comparison of different distance measures on the benchmark data sets (the number of clusters = $0.4N$).

1. Two new distance measures have the lower time cost than existing distance measures. This is validated by Fig. 2

VI. Conclusions and Further Work

This study has proposed the SMDP algorithm to handle the MIL classification problem. In the semi-supervised scenario, DP clustering algorithm and distance measure strategies were investigated. We also proposed two new distance measures, i.e., d^{vir} and d^{bia}. Experimental results verify the effectiveness of the algorithm.

The following research topics deserve further investigation:

1) More distance measures for the algorithm framework. Currently, SMDP only employs some distance measures. Other distance measures can be adopted or designed to accommodate data with different distribution structures between bag-level or instance-level. In addition, these measures should be fine-tuned or modified to fit the MIL problem.
2) Dynamic selection of appropriate distance measures in the process of finding clustering center. SMDP only employed the currently "best" distance measure basing on the empirical results.

In summary, SMDP is a comprehensive algorithm framework that can be enriched in the future.

References

[1] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1, pp. 31–71, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0004370296000343

[2] O. Maron and A. L. Ratan, "Multiple-instance learning for natural scene classification," in *Proceedings of the Fifteenth International Conference on Machine Learning*, ser. ICML 98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 341–349.

[3] X. F. Song, L. Jiao, S. Y. Yang, X. R. Zhang, and F. H. Shang, "Sparse coding and classifier ensemble based multi-instance learning for image categorization," *Signal Processing*, vol. 93, no. 1, pp. 1 – 11, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168412002599

[4] X.-S. Wei, H.-J. Ye, X. Mu, J. X. Wu, C. H. Shen, and Z.-H. Zhou, "Multiple instance learning with emerging novel class," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2019.

[5] J. Tang, H. J. Li, G.-J. Qi, and T.-S. Chua, "Image annotation by graph-based inference with integrated multiple/single instance representations," *IEEE Transactions on Multimedia*, vol. 12, no. 2, pp. 131–141, Feb 2010.

[6] S. L. Zhu and X. Q. Tan, "A novel automatic image annotation method based on multi-instance learning," *Procedia Engineering*, vol. 15, pp. 3439–3444, 2011, cEIS 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187770581102145X

[7] Q. Zhang, S. A. Goldman, W. Yu, and J. Fritts, "Content-based image retrieval using multiple-instance learning," in *Proceedings of the Nineteenth International Conference on Machine Learning*, ser. ICML 02. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 682–689.

[8] R. Rahmani, S. A. Goldman, H. Zhang, J. Krettek, and J. E. Fritts, "Localized content based image retrieval," in *Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval*, ser. MIR 05. New York, NY, USA: Association for Computing Machinery, 2005, p. 227236. [Online]. Available: https://doi.org/10.1145/1101826.1101863

[9] S. Conjeti, M. Paschali, A. Katouzian, and N. Navab, "Deep multiple instance hashing for scalable medical image retrieval," in *Medical Image Computing and Computer Assisted Intervention MICCAI 2017*, M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, and S. Duchesne, Eds. Cham: Springer International Publishing, 2017, pp. 550–558.

[10] Z.-H. Zhou, Y.-Y. Sun, and Y.-F. Li, "Multi-instance learning by treating instances as non-I.I.D. samples," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML 09. New York, NY, USA: Association for Computing Machinery, 2009, pp. 1249–1256. [Online]. Available: https://doi.org/10.1145/1553374.1553534

[11] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *NIPS*, 2002, pp. 561–568. [Online]. Available: http://papers.nips.cc/paper/2232-support-vector-machines-for-multiple-instance-learning

[12] B.-C. Xu, K. M. Ting, and Z.-H. Zhou, "Isolation set-kernel and its application to multi-instance learning," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD 19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 941–949. [Online]. Available: https://doi.org/10.1145/3292500.3330830

[13] L. Zhu, B. Zhao, and Y. Gao, "Multi-class multi-instance learning for lung cancer image classification based on bag feature selection," in *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 2, Oct 2008, pp. 487–492.

[14] Z. Y. Wang, J. Poon, S. D. Sun, and S. Poon, "Attention-based multi-instance neural network for medical diagnosis from incomplete and low quality data," in *2019 International Joint Conference on Neural Networks (IJCNN)*, July 2019, pp. 1–8.

[15] B. C. Zhang, Z. G. Li, A. Perina, A. D. Bue, and V. Murino, "Adaptive local movement modelling for object tracking," in *2015 IEEE Winter Conference on Applications of Computer Vision*, Jan 2015, pp. 25–32.

[16] M. Abdechiri, K. Faez, and H. Amindavar, "Visual object tracking with online weighted chaotic multiple instance learning," *Neurocomputing*, vol. 247, pp. 16–30, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231217305404

[17] D. Zhang, J. R. He, and R. Lawrence, "MI2LS: Multi-instance learning from multiple informationsources," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD 13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 149–157. [Online]. Available: https://doi.org/10.1145/2487575.2487651

[18] S. Angelidis and M. Lapata, "Multiple instance learning networks for fine-grained sentiment analysis," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 17–31, 2018.

[19] T. Grtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels," in *ICML*, 2002, pp. 179–186.

[20] X.-S. Wei, J. X. Wu, and Z.-H. Zhou, "Scalable algorithms for multi-instance learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 975–987, April 2017.

[21] K. M. Ting, Y. Zhu, and Z.-H. Zhou, "Isolation kernel and its effect on SVM," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD 18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 2329–2337. [Online]. Available: https://doi.org/10.1145/3219819.3219990

[22] Z.-H. Zhou and M.-L. Zhang, "Solving multi-instance problems with classifier ensemble based on constructive clustering," *Knowledge and Information Systems*, vol. 11, no. 2, pp. 155–170, 2007.

[23] M.-L. Zhang and Z.-H. Zhou, "Multi-instance clustering with applications to multi-instance prediction," *Applied Intelligence*, vol. 31, no. 1, pp. 47–68, 2009.

[24] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014. [Online]. Available: https://science.sciencemag.org/content/344/6191/1492

[25] J. Amores, "Multiple instance classification: review, taxonomy and comparative study," *Artificial Intelligence*, vol. 201, pp. 81–105, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0004370213000581

[26] G. Edgar, *Measure, topology, and fractal geometry, 3rd print*. Springer-Verlag, Berlin, 1995.

[27] J. Wang and J.-D. Zucker, "Solving multiple-instance problem: a lazy learning approach," pp. 1119–1125, 2000. [Online]. Available: http://cogprints.org/2124/

[28] P. Auer and R. Ortner, "A boosting approach to multiple instance learning," in *Machine Learning: ECML 2004*, J.-F. Boulicaut, F. Esposito, F. Giannotti, and D. Pedreschi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 63–74.

[29] T. Yao, Y. W. Pan, Y. H. Li, Z. F. Qiu, and T. Mei, "Boosting image captioning with attributes," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[30] Y. S. Xiao, X. Z. Yang, and B. Liu, "A new self-paced method for multiple instance boosting learning," *Information Sciences*, vol. 515, pp. 80 – 90, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025519311223

[31] J. Amores, "Vocabulary-based approaches for multiple-instance data: a comparative study," in *2010 20th International Conference on Pattern Recognition*, Aug 2010, pp. 4246–4250.

[32] O. Maron and T. Lozano-Pérez, "A framework for multiple-instance learning," in *Proceedings of the 10th International Conference on Neural Information Processing Systems*, ser. NIPS97. Cambridge, MA, USA: MIT Press, 1997, pp. 570–576.

[33] Q. Zhang and S. A. Goldman, "EM-DD: An improved multiple-instance learning technique," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, ser. NIPS01. Cambridge, MA, USA: MIT Press, 2001, pp. 1073–1080.

[34] H. Blockeel, D. Page, and A. Srinivasan, "Multi-instance tree learning," in *Proceedings of the 22nd International Conference on Machine Learning*, ser. ICML 05. New York, NY, USA: Association for Computing Machinery, 2005, pp. 57–64. [Online]. Available: https://doi.org/10.1145/1102351.1102359

[35] J. J. Wu, Y. N. Yu, C. Huang, and K. Yu, "Deep multiple instance learning for image classification and auto-annotation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3460–3469.

[36] Z. N. Yan, Y. Q. Zhan, Z. G. Peng, S. Liao, Y. Shinagawa, S. T. Zhang, D. N. Metaxas, and X. S. Zhou, "Multi-instance deep learning: discover discriminative local anatomies for bodypart recognition," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1332–1343, May 2016.

[37] M. X. Liu, J. Zhang, E. Adeli, and D. G. Shen, "Landmark-based deep multi-instance learning for brain disease diagnosis," *Medical Image Analysis*, vol. 43, pp. 157–168, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841517301524

[38] C. Bergeron, G. Moore, J. Zaretzki, C. M. Breneman, and K. P. Bennett, "Fast bundle algorithm for multiple-instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1068–1079, June 2012.

[39] L. X. Jiang, Z. H. Cai, D. H. Wang, and H. Zhang, "Bayesian citation-kNN with distance weighting," *International Journal of Machine Learning and Cybernetics*, vol. 5, no. 2, pp. 193–199, 2014.

[40] P. Villar, R. Montes, A. M. Sánchez, and F. Herrera, "Fuzzy-citation-KNN: a fuzzy nearest neighbor approach for multi-instance classification," in *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, July 2016, pp. 946–952.

[41] S. Ray and M. Craven, "Supervised versus multiple instance learning: an empirical comparison," in *Proceedings of the 22nd International Conference on Machine Learning*, ser. ICML 05. New York, NY, USA: Association for Computing Machinery, 2005, pp. 697–704. [Online]. Available:

https://doi.org/10.1145/1102351.1102439
[42] Z. Y. Fu and A. Robles-kelly, "Fast multiple instance learning via l1,2 logistic regression," pp. 1–4, 2008.
[43] R.-B. Chen, K.-H. Cheng, S.-M. Chang, S.-L. Jeng, P.-Y. Chen, C.-H. Yang, and C.-C. Hsia, "Multiple-instance logistic regression with lasso penalty," *arXiv preprint arXiv:1607.03615*, 2016.
[44] M.-L. Zhang and Z.-H. Zhou, "Adapting rbf neural networks to multi-instance learning," *Neural Processing Letters*, vol. 23, no. 1, pp. 1–26, 2006.
[45] P. G. Polishchuk, A. I. Rakhimbekova, and T. I. Madzhidov, "Multi-instance learning for structure-activity modeling for molecular properties," in *Analysis of Images, Social Networks and Texts: 8th International Conference, AIST 2019, Kazan, Russia, July 17–19, 2019, Revised Selected Papers*, vol. 1086. Springer Nature, 2020, p. 62.
[46] Y. Chevaleyre and J.-D. Zucker, "A framework for learning rules from multiple instance data," in *Machine Learning: ECML 2001*, L. D. Raedt and P. Flach, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 49–60.
[47] J. Luna, A. Cano, V. Sakalauskas, and S. Ventura, "Discovering useful patterns from multiple instance data," *Information Sciences*, vol. 357, pp. 23–38, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025516302365
[48] G. Melki, A. Cano, and S. Ventura, "MIRSVM: Multi-instance support vector machine with bag representatives," *Pattern Recognition*, vol. 79, pp. 228 – 241, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S003132031830061X
[49] Z.-H. Zhou and J.-M. Xu, "On the relation between multi-instance learning and semi-supervised learning," in *Proceedings of the 24th International Conference on Machine Learning*, ser. ICML 07. New York, NY, USA: Association for Computing Machinery, 2007, pp. 1167–1174. [Online]. Available: https://doi.org/10.1145/1273496.1273643
[50] Z. H. Wang, S. Yoon, S. J. Xie, Y. Lu, and D. S. Park, "Visual tracking with semi-supervised online weighted multiple instance learning," *The Visual Computer*, vol. 32, pp. 307–320, March 2016.
[51] V. Cheplygina, M. de Bruijne, and J. P. Pluim, "Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis," *Medical Image Analysis*, vol. 54, pp. 280–296, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841518307588
[52] M. J. Du, S. F. Ding, and H. J. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowledge-Based Systems*, vol. 99, pp. 135–145, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705116000794
[53] M. Wang, F. Min, Z.-H. Zhang, and Y.-X. Wu, "Active learning through density clustering," *Expert Systems with Applications*, vol. 85, pp. 305–317, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S095741741730369X
[54] M. J. Du, S. F. Ding, and Y. Xue, "A robust density peaks clustering algorithm using fuzzy neighborhood," *International Journal of Machine Learning and Cybernetics*, vol. 9, pp. 1131–1140, 2018.
[55] Y. X. Chen and J. Z. Wang, "Image categorization by learning and reasoning with regions," *J. Mach. Learn. Res.*, vol. 5, pp. 913–939, Dec. 2004.
[56] Y. X. Chen, J. B. Bi, and J. Z. Wang, "MILES: Multiple-instance learning via embedded instance selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1931–1947, Dec 2006.

Mei Yang is an associate professor with School of Computer Science, Southwest Petroleum University. Her current research interests include multi-instance learning and recommender systems.

Yu-xuan Zhang is a postgraduate student with with School of Computer Science, Southwest Petroleum University, Chengdu, China.

Fan Min (M'09) received the M.S. and Ph.D. degrees from the School of Computer Science and Engineering, University of Electronics Science and Technology of China, Chengdu, China, in 2000 and 2003, respectively. He visited the University of Vermont, Burlington, Vermont, from 2008 to 2009. He is currently a professor with Southwest Petroleum University, Chengdu. He has published more than 100 refereed papers in various journals and conferences, including the *Information Sciences*, *International Journal of Approximate Reasoning*, and *Knowledge-Based Systems*. His current research interests include data mining, recommender systems, active learning and granular computing.