# Tri-perspective Embedding for Multi-instance Learning

Mei Yang[a], Wen-Xi Zeng[a], Wei-Zhi Wu[c,d], Fan Min[a,b]

[a] *School of Computer Science, Southwest Petroleum University, Chengdu, 610500, China*
[b] *Institute for Artificial Intelligence, Southwest Petroleum University, Chengdu, 610500, China*
[c] *School of Mathematics, School of Information Engineering, Zhejiang Ocean University, Zhoushan 316022, China*
[d] *Key Laboratory of Oceanographic Big Data Mining and Application of Zhejiang Province, Zhejiang Ocean University, Zhoushan 316022, China*

## Abstract

Multi-instance learning (MIL) handles the hierarchical data where each bag is composed of a set of instances. Most MIL algorithms mine overall bag information through representative instance selection. However, such information may be insufficient in the bag representation. In this paper, we propose a tri-perspective embedding for multi-instance learning (TEMI) algorithm, which exploits the bag information from the overall, positive, and negative perspectives. First, the bag perspective vector is used to extract the overall bag information. It is obtained using a model trained on the representative instance set. Second, we divide each bag into positive and negative sub-bags based on each instance's similarity to representative instances. The positive (negative) perspective vector is constructed with the positive (negative) sub-bag and all positive (negative) instances. Final, the bag embedding vector with strong discriminative is obtained by concatenating positive, bag, and negative perspective vectors. Experiments demonstrate that TEMI outperforms state-of-the-art MIL embedding algorithms on 38 data sets.

*Keywords:* discriminative, instance selection, multi-instance learning, tri-perspective embedding.

## 1. Introduction

Multi-instance learning (MIL) handles the complicated data where each bag is composed of a set of instances. Unlike traditional supervised learning, labels are provided at the bag level rather than the instance level. A standard assumption is that a bag is positive if it contains at least one positive instance, otherwise it is negative [1, 2]. MIL has been widely applied in many real-world scenarios, including drug activity prediction [3], bug localization [4], image retrieval [5, 6, 7, 8], text classification [9, 10, 11], web mining [12], etc.

[∗]Corresponding author. Tel.: +86 135 4068 5200.
Email address: minfan@swpu.edu.cn (F. Min).

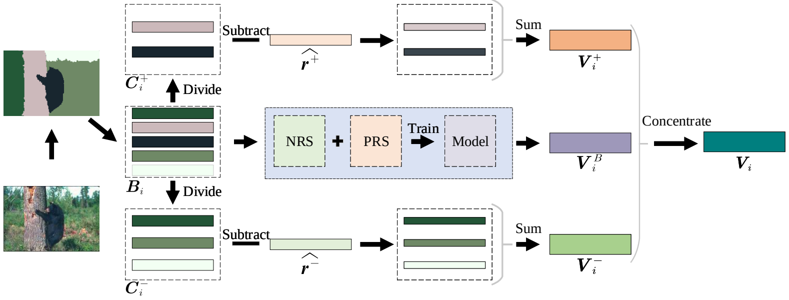


Figure 1: The overall framework of TEMI. Each bag is transformed into a new feature space through the learned three perspective vectors. $\boldsymbol{C}_i^+$ ($\boldsymbol{C}_i^-$) represents the positive (negative) sub-bag of the bag $\boldsymbol{B}_i$. $\widehat{\boldsymbol{r}^+}$ ($\widehat{\boldsymbol{r}^-}$) is the positive (negative) information vector of all bags. PRS and NRS represent the positive and negative representative instance set, respectively. $\boldsymbol{V}_i^+$, $\boldsymbol{V}_i^B$, and $\boldsymbol{V}_i^-$ represent the positive, bag, and negative perspective vector, respectively. $\boldsymbol{V}_i$ represents the bag embedding vector.

Existing MIL algorithms can be roughly divided into three categories. *Instance-based* algorithms [13, 14, 15, 16] indirectly predict the label of a bag by computing the labels of its instances. This is done through maximum margin [14], maximum likelihood criterion using Kernel Density Estimation [16], etc. *Bag-based* algorithms [17, 18, 19] treat each bag as a sample and cope with the bag structure for similarity calculation. This is done through bag-representative selector [19], graph kernel using inter-correlated components of the bag [18], etc. *Embedding-based* algorithms [20, 21, 22] convert each bag into a single vector with the help of representative samples. As a result, the bag label can be indirectly predicted with a SIL classifier.

Embedding-based algorithms can be further divided into the following two subcategories. *Distance-based* algorithms usually take advantage of the Hausdorff distance[8, 22, 23] and instance-bag distance [24, 25] to measure the similarity between the bag and the representative samples. This similarity result is also set as the attribute value of the bag's embedding vector. *Encoding-based* algorithms typically utilize the instance frequencies [26], the Gaussian mixture model [27], and the tree-based with Voronoi diagram [28] to embed each bag. As a result, the bag embedding vector has a large dimensionality. Unfortunately, most embedding-based algorithms mine the overall bag information from the single perspective. It cannot effectively preserve the distinguishability of the bag in the new feature space.

In this paper, we propose a tri-perspective embedding for multi-instance learning (TEMI) algorithm, which also makes use of the positive and negative information of the bag. Figure 1 illustrates the overall framework of TEMI. The mutual instance selection technique sequentially obtains a positive representative instance set (PRS) and a negative representative instance set (NRS) from the positive and negative bags. It effectively mines the positive and negative representativeness of the data set, respectively. The tri-perspective embedding technique embeds each bag into the new feature space from three different perspectives. First, the bag perspective vector extracts the overall bag information. It is the highest scoring instance in the bag computed by a model built on the PRS and NRS. Second, the positive perspective vector captures the positive information. It is computed from the positive sub-bag of the bag and the PRS. Third, the negative perspective vector mines negative information. The way is similar to obtaining positive information. Finally, the bag embedding vector is formed by concatenating the three perspective vectors. Experiments are undertaken on thirty-eight MIL classification data sets to quantify the performance of TEMI. Results show that TEMI is superior to rival algorithms in general and has higher stability.

The contribution of this paper is summarized as follows:

1) We propose a tri-perspective embedding technique. To the best of our knowledge, this is the first time to extract the overall bag, positive, and negative information of each bag from three perspectives. It explicitly embeds the bags into a feature space that preserves more important information.
2) We propose a mutual instance selection technique. It exploits the maximum feature difference between positive and negative instances to improve the discrimination between PRS and NRS.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces the proposed TEMI algorithm. Section 4 reports the experimental results. Section 5 summarizes this paper.

## 2. Related work

Multi-instance learning (MIL) was first proposed in drug activity prediction [3]. Since then, most MIL embedding algorithms use similarity information between bags and instances to embed bags into new feature spaces [20, 24, 29]. MILES [20] employs the minimum distance between the bag and the representative instances to measure the similarity. MILES has a high time consumption because all instances in the instance space are selected as representative instances. Further, since the number of positive instances is much lower than the number of negative instances, the bag embedding vector will have more noisy information. To handle this issue, MILIS [24] selects only one instance from each positive bag to trim the representative instances. MILFM [30] treats all instances in all positive bags and the cluster centers of instances in negative bags as representative instances. RSIS [31] projects instances into multiple random subspaces for clustering and computes the selection probabilities of instances in each cluster. These probabilities are used to select different instances to ensemble the final SVM classifier. MIKI [32] utilizes a weighted model to select instances with high positiveness as the representatives.

Using unsupervised learning or discriminant analysis to select the representative bag is an efficient method [8, 23]. BAMIC [23] treats a bag as an unlabeled sample. Then, each cluster center is used as a representative bag by the $k$-Medoids algorithm. Final, the Hausdorff distance is used to calculate the similarity between the bags and the representative bags. Another recent method, ELDB [8] selects discriminative bags as representatives through discriminant analysis and self-enhancement technique.

Additionally, encoding-based methods are also used to embed overall bag information [26, 33, 34, 35]. CCE [26] is the earliest method to perform bag embedding by encoding. All instances of the bags are first divided into $d$ clusters. Then each bag is embedded in a $d$-dimensional binary single vector. Specifically, if the instance in the bag belongs to the $i$-th cluster, the $i$-th attribute is set to 1; otherwise, it is 0. By repeating the above process with different values of $d$, many vectors are combined into the final single vector for prediction. However, the encoding method of CCE embeds the bags into a high-dimensional feature space. SALE [36] combines locality-sensitive hashing and random super histograms to reconstruct each bag. It retains the intrinsic characteristics of the instance, reduces the feature dimension, and drops the complexity. BSN [35] proposes a network-based encoding method. It uses MI-Net [37] to build a multi-instance neural network to learn instance-level representations. These representations are used to build bag similarity networks.

However, these algorithms still face the issue of insufficient bag information mining. As mentioned above, they only consider the information of the overall bag and ignore the positive (negative) information existing in the bag. Unlike previous methods, our proposed algorithm learns both the overall bag information and the two extreme information simultaneously. This maximizes the preservation of bag structure information.

## 3. Tri-perspective Embedding for Multi-instance Learning

In this section, we first introduce some basic notations, then present the mutual instance selection technique and tri-perspective embedding technique. Finally, we analyze the time complexity of TEMI.

### 3.1. Notations

Table 1: Notations.

| Notation | Meaning | Notation | Meaning |
|---|---|---|---|
| $\mathcal{X}$ | The instance space | $\boldsymbol{V}_i$ | The bag embedding vector of $\boldsymbol{B}_i$ |
| $\mathcal{Y}$ | The label space | $\boldsymbol{V}_i^B$ | The bag perspective vector of $\boldsymbol{B}_i$ |
| $\mathcal{T}$ | The data set | $\boldsymbol{V}_i^+$ | The positive perspective vector of $\boldsymbol{B}_i$ |
| $\mathcal{T}^+$ | The positive bag set | $\boldsymbol{V}_i^-$ | The negative perspective vector of $\boldsymbol{B}_i$ |
| $\mathcal{T}^-$ | The negative bag set | $\boldsymbol{x}_{ij}$ | The $j$-th instance in $\boldsymbol{B}_i$ |
| $\boldsymbol{B}_i$ | The $i$-th bag | $\boldsymbol{x}_{ij}^*$ | The $j$-th candidate instance in $\boldsymbol{X}_i$ |
| $\boldsymbol{X}_i$ | The set of candidate instances of $\boldsymbol{B}_i$ | $S_{ij}$ | The discriminative score of $\boldsymbol{x}_{ij}^*$ |
| $\boldsymbol{P}$ | The initial negative instance set | $N$ | The cardinality of $\mathcal{T}$ |
| $\boldsymbol{R}^{+0}$ | The initial positive representative instance set | $n_i$ | The cardinality of $\boldsymbol{B}_i$ |
| $\boldsymbol{R}^+$ | The positive representative instance set | $y_i$ | The label of $\boldsymbol{B}_i$ |
| $\boldsymbol{R}^-$ | The negative representative instance set | $N^+$ | The cardinality of $\mathcal{T}^+$ |
| $\boldsymbol{C}_i^+$ | The positive sub-bag of $\boldsymbol{B}_i$ | $N^-$ | The cardinality of $\mathcal{T}^-$ |
| $\boldsymbol{C}_i^-$ | The negative sub-bag of $\boldsymbol{B}_i$ | $p(x)$ | The proportion of candidate instances for each bag |

Let $\mathcal{X} = \mathbb{R}^d$ denote the instance space and $\mathcal{Y} = \{0, 1\}$ denote the label space. Let $\mathcal{T} = \{(\boldsymbol{B}_i, y_i)\}_{i=1}^N$ denote a data set, where $\boldsymbol{B}_i = \{\boldsymbol{x}_{ij}\}_{j=1}^{n_i}$ is a bag of instances and $y_i \in \mathcal{Y}$ is the label of $\boldsymbol{B}_i$. Here $\boldsymbol{x}_{ij} \in \mathcal{X}$ is a $d$-dimensional instance, $N$ and $n_i$ are the cardinality of $\mathcal{T}$ and $\boldsymbol{B}_i$, respectively. Let $\mathcal{T}^+ = \{(\boldsymbol{B}_i^+, y_i)\}_{i=1}^{N^+}$ and $\mathcal{T}^- = \{(\boldsymbol{B}_i^-, y_i)\}_{i=1}^{N^-}$ denote the positive and negative bag sets, respectively. $N^+$ and $N^-$

denote the cardinalities of the positive and negative bag sets. Table 1 shows some important notations in this paper.

Most multi-instance embedding learning algorithms perform bag embedding based on the similarity of the bag to the instance [24, 25, 26]. The similarity score is calculated between the instance $\boldsymbol{x}$ and the bag $\boldsymbol{B}$ using the following function:

$$s(\boldsymbol{x}, \boldsymbol{B}) = \max_{\boldsymbol{x}_k \in \boldsymbol{B}} \frac{1}{\|\boldsymbol{x} - \boldsymbol{x}_k\|_2 + \varepsilon}, \tag{1}$$

where $\varepsilon$ is a small number to avoid zero division. The intuition is that if the labels of instance $\boldsymbol{x}$ and bag $\boldsymbol{B}$ are consistent, the similarity is higher; otherwise, the similarity is lower. Next, we study how to transform each bag into a new feature space from two key points of instance selection and embedding methods.

### *3.2. Algorithm description*

Algorithm 1 presents the main steps of TEMI. Lines 1–4 show the main process of the mutual instance selection technique. Line 1 utilizes the label uniqueness of the negative instance space to select the initial negative instance set $\boldsymbol{P}$. Line 2 uses the high dissimilarity among heterogeneous instances to select an initial positive representative instance set $\boldsymbol{R}^{+0}$. Line 3 reduces noise instances in $\boldsymbol{R}^{+0}$ to obtain an optimized $\boldsymbol{R}^{+}$. Line 4 selects the negative representative instance set $\boldsymbol{R}^{-}$ according to the optimized $\boldsymbol{R}^{+}$ and each negative bag. Lines 5–7 show the process of obtaining the tri-perspective embedding vector. Line 5 selects the most discriminative instance in the bag as the bag perspective vector $\boldsymbol{V}_i^B$ according to the model trained on the representative instances. Lines 6 obtains the positive (negative) perspective vectors $\boldsymbol{V}_i^+$ ($\boldsymbol{V}_i^-$) for each bag based on $\boldsymbol{R}^+$ ($\boldsymbol{R}^-$). These three perspective vectors represent the overall bag, positive, and negative feature information in the bag, respectively. Line 7 converts each bag into new feature space based on three perspective vectors. This maintains the distinguishability of the bag in the new feature space. Final, the trained classifier $\mathcal{F}(\cdot)$ is utilized to predict labels for new bags.

### *3.3. The mutual instance selection technique*

Figure 2 shows the mutual instance selection technique consists of four steps.

1) Some instances are selected from the negative instance space as the initial negative instance set (INS).
2) According to the INS, the most discriminative instance in each positive bag forms the positive representative instance set (PRS).
3) Some instances in PRS that may be negative are deleted. The purpose is to improve the quality of PRS.
4) The most discriminative instance in each negative bag forms the negative representative instance set (NRS) according to the PRS.

**Algorithm 1** Tri-perspective Embedding for Multi-instance Learning

```
Input:
    The data set $\mathcal{T} = \{(\boldsymbol{B}_i, y_i)\}_{i=1}^{N}$;
Output:
    The representative instance sets $\boldsymbol{R}^+$ and $\boldsymbol{R}^-$;
    The SIL classifier $\mathcal{F}(\cdot)$;
$\boldsymbol{P} \leftarrow$ An initial negative instance set with high-density; //Section 3.3.1
$\boldsymbol{R}^{+0} \leftarrow$ The representative instance in each positive bag by computing instance discriminant scores based on $\boldsymbol{P}$; //Section 3.3.2, Eq. (2)
$\boldsymbol{R}^+ \leftarrow$ Optimize $\boldsymbol{R}^{+0}$;// Section 3.3.3
$\boldsymbol{R}^- \leftarrow$ The most discriminative score instance in each negative bag based on $\boldsymbol{R}^+$; //Section 3.3.4, Eq. (2)
$\boldsymbol{V}_i^B \leftarrow$ The bag perspective vector through computing the scores of instance based on $\boldsymbol{R}^+$ and $\boldsymbol{R}^-$; //Section (3.4.2), Eq. (5)
$\boldsymbol{V}_i^+(\boldsymbol{V}_i^-) \leftarrow$ The positive (negative) perspective vector is obtained based on $\boldsymbol{R}^+$ ($\boldsymbol{R}^-$); //Section (3.4.2), Eq. (7)
$\boldsymbol{V}_i \leftarrow$ Concatenate $\boldsymbol{V}_i^+$, $\boldsymbol{V}_i^-$, and $\boldsymbol{V}_i^B$ of $\boldsymbol{B}_i$ to the bag embedding vector $\boldsymbol{V}_i$; //Section (3.4.2), Eq. (9)
$\mathcal{F}(\cdot) \leftarrow$ Train a classifier with the vectors $\{(\boldsymbol{V}_i, y_i)\}_{i=1}^{N}$.
Output $\boldsymbol{R}^+$, $\boldsymbol{R}^-$ and $\mathcal{F}(\cdot)$.
```

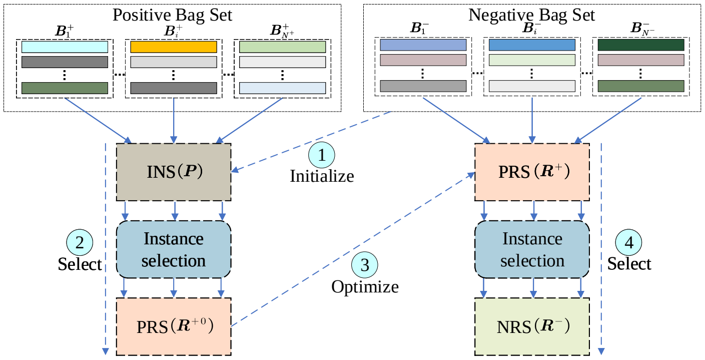


Figure 2: The mutual instance selection technique consisting of four steps.

### *3.3.1. Negative instance initialization*

The initial negative instance set (INS) consists of some instances in the negative bags. To ensure the representativeness of INS, the peak density is adopted to select instances. First, let $\boldsymbol{I}^- = \bigcup \boldsymbol{B}_i^{-1}$ denote the set of all the instances from the negative bags. The density of each negative instance $\boldsymbol{x}_i^- \in \boldsymbol{I}^-$ is the same as the Density Peaking (DP) algorithm [38]. Finally, the INS is generated by combining the instances with the top-$t$ highest density, i.e. $\boldsymbol{P} = \{\boldsymbol{p}_i\}_{i=1}^{t}$, where $t = \lceil 0.2N^- \rceil$, $N^-$ is the cardinality of negative bags and $\boldsymbol{p}_i$ is the $i$-th high-density instance.

### *3.3.2. Positive instance selection*

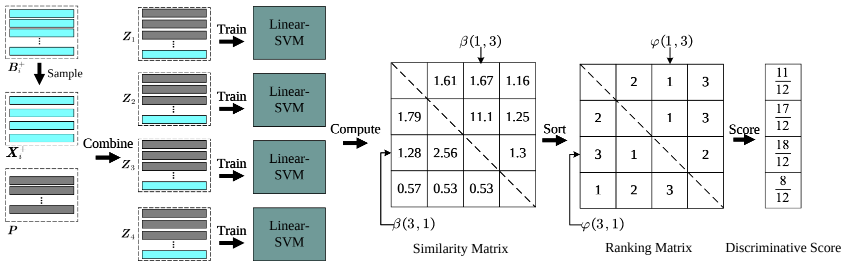


Figure 3: An example of calculating the discriminative score. This shows the computation of the discriminative scores for the 4 instances with the highest peak density in $\boldsymbol{B}_i^+$.

Due to the high dissimilarity among heterogeneous instances [39], INS is used for discriminative evaluation of instances in the positive bag. The exemplar-SVM [40] is employed to select the instance with the highest discriminative similarity as the positive representative instance. To reduce the runtime overhead, $\lfloor p(x)n_i \rfloor$ instances from $\boldsymbol{B}_i^+$ are selected as candidate instances $\boldsymbol{X}_i^+$, where $p(x)$ is the proportion of candidate instances and $n_i$ the cardinality of $\boldsymbol{B}_i^+$. Figure 3 shows the calculation process of the discriminative similarity of the four instances in $\boldsymbol{X}_i^+$, which consists of five steps.

1) Each candidate instance $\boldsymbol{x}_{ij}^* \in \boldsymbol{X}_i^+$ is separately combined with the instances of the $\boldsymbol{P}$ to form a new set $\boldsymbol{Z}_j = \{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_t, \boldsymbol{x}_{ij}^*\}$. In particular, $\boldsymbol{x}_{ij}^*$ from the positive bag is labeled as positive. The instances from $\boldsymbol{P}$ are labeled as negative.
2) Each $\boldsymbol{Z}_j$ is used to train a linear SVM model. The purpose is to learn the weight vector $\boldsymbol{w}_{ij}$ of $\boldsymbol{x}_{ij}^*$.
3) The similarity matrix $\boldsymbol{\beta}$ of $\boldsymbol{x}_{ij}^*$ is computed. $\beta(1,3) = 1/|T_{11} - T_{13}| = 1.67$ represents the similarity between instance $\boldsymbol{x}_{i1}^*$ and $\boldsymbol{x}_{i3}^*$, which $T_{13} = \boldsymbol{x}_{i3}^* \cdot \boldsymbol{w}_{i1}$ represents the confidence of instance $\boldsymbol{x}_{i3}^*$ to $\boldsymbol{x}_{i1}^*$.
4) The similarity matrix is sorted to get the ranking matrix $\boldsymbol{\varphi}$. $\varphi(1,3) = 1$ means it ranks first among $\varphi(1,2)$, $\varphi(1,3)$, and $\varphi(1,4)$. Similarly, $\varphi(3,1) = 3$ means it ranks third among $\varphi(3,1)$, $\varphi(3,2)$, and $\varphi(3,4)$.
5) The discriminative score of $\boldsymbol{x}_{ij}^*$ is calculated by the sorting matrix as follows [41]:

$$S_{ij} = \sum_{\boldsymbol{x}_k \in \boldsymbol{X}_i^+ \setminus \boldsymbol{x}_j} \frac{1}{\varphi(j,k) \cdot \varphi(k,j)}. \tag{2}$$

Through the above 5 steps, the discriminative scores of four instances are $\frac{11}{12}$, $\frac{17}{12}$, $\frac{18}{12}$, and $\frac{8}{12}$, respectively. The instance $\boldsymbol{x}_{i3}^*$ with the highest score is selected as the positive representative instance $\boldsymbol{r}_i^{+0}$ of $\boldsymbol{B}_i^+$.

Similarly, the instances with the highest discriminative scores in all positive bags are selected as representatives to form the positive representative instance set (PRS) $\boldsymbol{R}^{+0} = \{\boldsymbol{r}_i^{+0}\}_{i=1}^{N^+}$.

### *3.3.3. Positive instance optimization*

Due to the small proportion of positive instances in the positive bags, there is no guarantee that the all instances in the PRS are positive. Therefore, we use the difference between positive and negative instances to optimize $\boldsymbol{R}^{+0}$. The average distance $d_i^+$ ($d_i^-$) of $\boldsymbol{r}_i^{+0}$ to $\boldsymbol{R}^{+0}$ ($\boldsymbol{P}$) is calculated. If $d_i^+ > d_i^-$, indicating that the instance $\boldsymbol{r}_i^{+0}$ is closer to positive, keep it; otherwise, it is closer to negative, delete it. Until the all positive representative instances are traversed, the optimized positive representative instance set $\boldsymbol{R}^+ = \{\boldsymbol{r}_i^+\}_{i=1}^{N_r^+}$ is obtained. $N_r^+$ is the cardinality of the PRS after optimization.

### *3.3.4. Negative instance selection*

Similarly, based on Eq. (2), we can select a representative from each negative bag to form the negative representative set $\boldsymbol{R}^- = \{\boldsymbol{r}_i^-\}_{i=1}^{N^-}$. As with getting $\boldsymbol{Z}_j$, a new set $\boldsymbol{Z}_j^- = \{\boldsymbol{r}_1^+, \cdots, \boldsymbol{r}_{N_r^+}^+, \boldsymbol{x}_{ij}^{*-}\}$ is obtained based on $\boldsymbol{R}^+$ and $\boldsymbol{B}_i^-$. It is used to train an SVM model to select negative representatives. The difference is that instances from $\boldsymbol{R}^+$ are marked as positive, and an instance from $\boldsymbol{B}_i^-$ is marked as negative. Notably, the negative representative instance set is not optimized. This is because the instance selected from the negative bag is definitely a negative instance.

## *3.4. Bag embedding methods*

In this section, we detail two bag embedding methods. Section 3.4.1 describes a distance-based benchmark embedding method. Section 3.4.2 introduces the three-perspective embedding method proposed in this paper.

### *3.4.1. A distance-based benchmark embedding method*

The success of algorithms using distance-based embedding has been illustrated in several applications [8, 23, 25]. As a benchmark technique, it is considered for comparison purposes. In the conventional distance-based embedding, the data set is clustered into $k$ clusters by treating the bag as a sample [23]. Each bag is represented by a feature vector whose $i$-th eigenvalue is the Hausdorff distance from the center of the $i$-th cluster to the bag.

With the distance-based embedding method, the representative samples are chosen using the mutual selection technique proposed in this paper. A sample is an instance rather than a bag, and has positive and negative distinctions. The similarity between a representative instance $\boldsymbol{r}$ and a bag $\boldsymbol{B}$ can be calculated by

$$d(\boldsymbol{r}, \boldsymbol{B}) = \max_{\boldsymbol{x}_i \in \boldsymbol{B}} \exp\left(-\|\boldsymbol{x}_i - \boldsymbol{r}\|_2\right). \tag{3}$$

As a result, the embedding vector $\boldsymbol{V}_i$ for bag $\boldsymbol{B}_i$ is a $(N_r^+ + N^-)$-dimensional vector:

$$\boldsymbol{V}_i = [d(\boldsymbol{r}_1^+, \boldsymbol{B}_i), \cdots, d(\boldsymbol{r}_{N_r^+}^+, \boldsymbol{B}_i), d(\boldsymbol{r}_1^-, \boldsymbol{B}_i), \cdots, d(\boldsymbol{r}_{N^-}^-, \boldsymbol{B}_i)]. \tag{4}$$

The advantage of $\boldsymbol{V}_i$ is reducing the disturbing information of the final embedding vector, since the selected representative instances are not disturbed by the instances in the negative bag.

*3.4.2. Tri-perspective embedding method*

According to the mutual instance selection technique given in Section 3.3, two sets of discriminative representative instances $\boldsymbol{R}^{+}$ and $\boldsymbol{R}^{-}$ are obtained. Figure 1 shows the main process of tri-perspective embedding method.

First, a linear SVM model is trained based on $\boldsymbol{R}^{+}$ and $\boldsymbol{R}^{-}$. Since the PRS and NRS are composed of the most discriminative instance in each bag, the model can effectively mine the overall information of the bag. Specifically, $T_{ij} = \boldsymbol{w}_{pn} \cdot \boldsymbol{x}_{ij}$ is utilized to calculate the representative score of each instance $\boldsymbol{x}_{ij} \in \boldsymbol{B}_i$, where the $\boldsymbol{w}_{pn}$ is the weight vector obtained by the model. The instance with the highest score will be regarded as the bag perspective vector:

$$\boldsymbol{V}_i^{B} = \boldsymbol{x}_{i \mathop{\arg\max}\limits_{1 \leq j \leq n_i} T_{ij}}. \tag{5}$$

Second, the two extreme information of each bag is mined. Each bag $\boldsymbol{B}_i$ is divided into two sub-bags by PRS and NRS. The positive sub-bag $\boldsymbol{C}_i^{+}$ represents the most positive instances of the bag; conversely, the negative sub-bag $\boldsymbol{C}_i^{-}$ represents the most negative instances. The positive extreme information is extracted from $\boldsymbol{C}_i^{+}$, whereas the negative extreme information is extracted from $\boldsymbol{C}_i^{-}$. Specifically, the similarity scores $s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{+})$ and $s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{-})$ of each instance $\boldsymbol{x}_{ij} \in \boldsymbol{B}_i$ to $\boldsymbol{R}^{+}$ and $\boldsymbol{R}^{-}$, respectively, are calculated according to Eq. (2). If an instance has a higher positive score, it belongs to a positive sub-bag.

Next, the instances are separated into positive or negative sub-bag according to the scores:

$$\begin{aligned} \boldsymbol{C}_i^{+} &= \{\boldsymbol{x}_{ij} \in \boldsymbol{B}_i \mid s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{+}) \geq s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{-})\}, \\ \boldsymbol{C}_i^{-} &= \{\boldsymbol{x}_{ij} \in \boldsymbol{B}_i \mid s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{+}) < s(\boldsymbol{x}_{ij}, \boldsymbol{R}^{-})\}. \end{aligned} \tag{6}$$

Here, $\boldsymbol{C}_i^{+}$ ($\boldsymbol{C}_i^{-}$) is those instances with higher positive (negative) scores, so they are more likely to show the positive (negative) feature information of the bag $\boldsymbol{B}_i$. Accordingly, the positive and negative perspective vectors of the bag $\boldsymbol{B}_i$ are obtained:

$$\begin{aligned} \boldsymbol{V}_i^{+} &= \sum_{\boldsymbol{x}_{ij} \in \boldsymbol{C}_i^{+}} \boldsymbol{x}_{ij} - \widehat{\boldsymbol{r}^{+}}, \\ \boldsymbol{V}_i^{-} &= \sum_{\boldsymbol{x}_{ij} \in \boldsymbol{C}_i^{-}} \boldsymbol{x}_{ij} - \widehat{\boldsymbol{r}^{-}}, \end{aligned} \tag{7}$$

where $\widehat{\boldsymbol{r}^{+}}$ ($\widehat{\boldsymbol{r}^{-}}$) is the positive (negative) information shared by all bags. The $\widehat{\boldsymbol{r}^{+}}$ is calculated as follows:

$$\widehat{\boldsymbol{r}^{+}} = \frac{1}{N^{+}} \sum_{i=1}^{N^{+}} \boldsymbol{r}_{\tau}, \tag{8}$$

where $\tau = \mathop{\arg\min}\limits_{1 \leq u \leq N_r^{+}} \min\limits_{1 \leq j \leq n_i} d(\boldsymbol{r}_u^{+}, \boldsymbol{x}_{ij})$ and $d(\boldsymbol{r}_u^{+}, \boldsymbol{x}_{ij})$ is the distance between $\boldsymbol{r}_u^{+} \in \boldsymbol{R}^{+}$ and $\boldsymbol{x}_{ij} \in \boldsymbol{B}_i^{+}$. Obviously, $\boldsymbol{r}_{\tau}$ is the representative instance that results in the closest distance from $\boldsymbol{R}^{+}$ to $\boldsymbol{B}_i^{+}$. Similarly, based on $\boldsymbol{R}^{-}$ and $\boldsymbol{B}_i^{-}$, the $\widehat{\boldsymbol{r}^{-}}$ is also obtained through Eq. (8).

Finally, the bag embedding vector of $\boldsymbol{B}_i$ is a $3 \times d$-dimensional vector composed of three perspective vectors:

$$\boldsymbol{V}'_i = \boldsymbol{V}^{+}_i \| \boldsymbol{V}^{B}_i \| \boldsymbol{V}^{-}_i, \tag{9}$$

where $\|$ represents the concatenation between vectors. Next, the bag embedding vector $\boldsymbol{V}_i$ is $\ell$-2 normalized by $\boldsymbol{V}_i = \boldsymbol{V}'_i / \|\boldsymbol{V}'_i\|_2$, where each attribute of $\boldsymbol{V}'_i$ is sign squared by $V'_{il} \leftarrow sign(V'_{il})\sqrt{|V'_{il}|}$ [42]. Inspired by the standard deviation of the data, this concatenated vector $\boldsymbol{V}_i$ includes the overall bag information and the information of the two extreme instances. This ensures that the bag is strongly discriminative in the new feature space. Ins the end, the bag embedding vectors are used to learn the classification model.

### *3.5. Time complexity analysis*

The TEMI time overhead mainly consists of representative instance construction and bag embedding. For the simplicity of analysis, we assume that each bag has the same number of instances, and the number of positive and negative bags is balanced. TEMI first selects $0.2N$ initial negative instances (INS) from the negative instance space. The Density Peaks is selected as a selectors, which has a complexity of $O((n/2)^2 d)$. Then the INS and each instance in each bag are utilized to build a linear SVM model for scoring, which has a complexity of $O((0.2N + 1)nd)$. In the bag embedding phase, the PRS and NRS are used to train the model to score each instance with complexity of $O(Nd)$. To sum up, the worst time complexity of TEMI is $O(n^2 d)$.

## 4. Experiments

We conduct experiments on 38 real-world data sets to evaluate the effectiveness of TEMI. To ensure the validity of the experiment, we use the average accuracy of 5 times 10-fold cross-validation (10CV) as metric. The experimental environment is the Windows 11 64-bit operating system, 16 GB memory, AMD Ryzen 7 4800U CPU 1.8GHz, Python 3.9.2.

### *4.1. Data sets*

Table 2 shows the details of seven domain data sets. 38 data sets are used to validate TEMI in our experiment. In the following parts, the domain knowledge of these data sets will briefly introduced. **Drug prediction:** This type of data set has two sub data sets: Musk1 and Musk2 [3]. A molecule is described as having multiple isomers. Each bag corresponds to a molecule and each instance corresponds to an isomer. **Image retrieval:** There are three data sets: Elephant, Fox and Tiger [14]. Each bag is an image, and the instance is a fragment of the image. For example, if the positive bag contains the image of Fox, the negative bag contains the images of other animals. **Messidor:** Messidor is an image classification problem [43, 44]. The data includes 1,200 fundus images from 654 diabetic patients and 546 healthy patients. **UCSB Breast:** UCSB Breast is an image classification problem [45]. The original data set included 58 excerpts of TMA images from 32 benign and 26 malignant breast cancer patients.

Table 2: Data sets information.

| Name | Bags | Instances | Attributes | Max / Min | Name | Bags | Instances | Attributes | Max / Min |
|---|---|---|---|---|---|---|---|---|---|
| Musk1 | 92 | 476 | 166 | 40/2 | News.rsh | 100 | 1,982 | 200 | 38/8 |
| Musk2 | 102 | 6,598 | 166 | 1,044/1 | News.sc | 100 | 4,284 | 200 | 71/20 |
| Elephant | 200 | 1,391 | 230 | 13/2 | News.se | 100 | 3,192 | 200 | 58/12 |
| Fox | 200 | 1,320 | 230 | 13/2 | News.sm | 100 | 3,045 | 200 | 54/11 |
| Tiger | 200 | 1,220 | 230 | 13/1 | News.sr | 100 | 4,677 | 200 | 71/21 |
| Mutagenesis1 | 188 | 10,486 | 7 | 88/28 | News.ss | 100 | 3,655 | 200 | 59/16 |
| Mutagenesis2 | 42 | 2,132 | 7 | 86/26 | News.tpg | 100 | 3,588 | 200 | 59/13 |
| Messidor | 1,200 | 12,352 | 687 | 12/8 | News.tpmd | 100 | 3,376 | 200 | 55/15 |
| UCSB Breast | 58 | 2,002 | 708 | 40/21 | News.tpmc | 100 | 4,788 | 200 | 75/21 |
| News.aa | 100 | 5,443 | 200 | 76/22 | News.trm | 100 | 4,606 | 200 | 79/25 |
| News.cg | 100 | 3,094 | 200 | 58/12 | Web1 | 113 | 3,423 | 5,863 | 200/4 |
| News.co | 100 | 5,175 | 200 | 82/25 | Web2 | 113 | 3,423 | 6,519 | 200/4 |
| News.csi | 100 | 4,827 | 200 | 74/19 | Web3 | 113 | 3,423 | 6,306 | 200/4 |
| News.csm | 100 | 4,473 | 200 | 71/17 | Web4 | 113 | 3,423 | 6,059 | 200/4 |
| News.cw | 100 | 3,110 | 200 | 54/12 | Web5 | 113 | 3,423 | 6,407 | 200/4 |
| News.mf | 100 | 5,306 | 200 | 84/29 | Web6 | 113 | 3,423 | 6,417 | 200/4 |
| News.ra | 100 | 3,458 | 200 | 59/15 | Web7 | 113 | 3,423 | 6,450 | 200/4 |
| News.rm | 100 | 4,730 | 200 | 73/22 | Web8 | 113 | 3,423 | 5,999 | 200/4 |
| News.rsb | 100 | 3,358 | 200 | 58/15 | Web9 | 113 | 3,423 | 6,279 | 200/4 |

**Mutagenesis:** Mutagenesis is a drug activity prediction problem [46]. There are two versions, easy (1) and hard (2), of the data set. **Newsgroups:** Newsgroup is a text classification data set [18]. There are 20 categories in total. The positive bags contain 3% of posts from the target category and 97% of posts randomly sampled from other categories. Negative bags are all other types of posts. **Web recommendation:** This is a web page classification problem [47]. Nine users rate the web page, so there are nine different data sets. A web page is considered as a bag. The links in the web page are viewed as instances.

### *4.2. Comparison Algorithms*

Table 3: Representative samples selection (RSS) type, bag embedding method (BEM), and parameter settings (PS) for the eight algorithms.

| Algorithms | RSS type | BEM | PS |
|---|---|---|---|
| BAMIC | Bag selection-based | Distance-based | Number of representative bags (Number of bags $N$)<br>Distance metric (Average Hausdorff distance) |
| MILFM | Instance selection-based | Distance-based | Number of cluster centers (50)<br>Distance metric (Instance-bag maximum distance) |
| Simple-MI | Statistics-based | Encoding-based | No Parameter |
| miFV | Statistics-based | Encoding-based | Components of Gaussian mixture model (2) |
| miVLAD | Instance selection-based | Encoding-based | Size of Code book (2) |
| MILDM | Instance selection-based | Distance-based | Instance selection mode (Global)<br>Distance metric (Same as MILFM)<br>Number of discriminative instances ($N$) |
| StableMIL | Instance selection-based | Distance-based | Number of causal instances ($0.2N$)<br>Distance metric (Same as MILFM) |
| ELDB | Bag selection-based | Distance-based | Bag selection mode (Global)<br>Distance metric (Same as BAMIC)<br>Number of discriminative bags ($N$) |

Table 3 summarizes the representative sample selection type, bag embedding method, and parameter setting for the eight algorithms. Eight state-of-the-art MIL algorithms are compared. **Bamic** [23] treats each bag as a sample and employs the $k$-Means algorithm to select the bag cluster centers as representative samples. The average Hausdorff distance is used to calculate the similarity between the bags. **MILFM** [30] utilizes AdaBoost to select the bag features embedded by instance prototypes. The instance prototypes include all the instances from the positive bag and the clustering centers of the instances from the negative bag. **Simple-MI** [48] seems the arithmetic mean vector of the instances for each bag as the bag embed vector. **miFV** [27] trains the Gaussian mixture model through all instances. The mixture weight, mean vector and covariance matrix of $k$-component Gaussian models are used to embed each bag into a single vector. **miVLAD** [34] first clusters all instances into $k$ clusters through the $k$-Means clustering algorithm. The cluster center of each cluster is selected as the representative instance. **MILDM** [22] selects the representative instance via discriminative instance pool evaluation criteria. **StabelMIL** [25] considers the positive instance that can change the negative bag label as a causal instance. Therefore, they select a certain scale of positive instances from the positive bag as causal instances. **ELDB** [8] selects more distinguishable bags with the discriminative analysis and reinforcement technique as representative samples. Similar to Bamic's algorithm, the similarity between bags is calculated by the average Hausdorff distance.

### *4.3. Parameter Analysis*

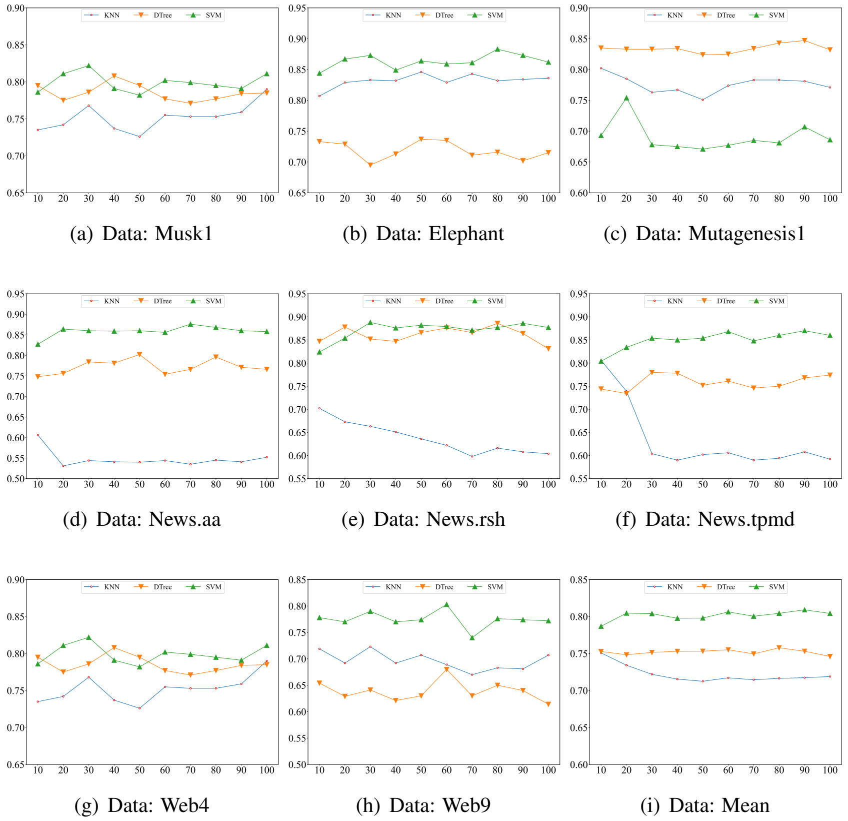
Figure 4 shows the experimental results of parametric analysis on eight data, including Drug prediction, Image retrieval, Newsgroups, and Web recommendation. Specially, Figure 4(i) represents the mean results of eight data sets on different classifiers. Lines with different colors indicate the results on different classifiers, including K-Nearest Neighbors (KNN), Decision Tree (DTree), and Support Vector Machine (SVM). The horizontal axis is the proportion of candidate instances, and the vertical axis represents the classification accuracy.

Figures 4(a)–4(h) show that as the proportion of candidate instances increases, the experimental results on the eight data for different classifiers do not differ considerably. In particular, the results of the "News.tpmd" on the KNN classifier vary greatly. This may be due to too few candidate instances, resulting in the bag not being effectively embedded. The experimental results on SVM for most of the data are significantly higher than the other two classifiers. In particular, the results of Mutagenesis1 in DTree are higher than other classifiers. This may be due to the lower dimensionality of Mutagenesis1, and DTree is more suitable for low-dimensional data. Figure 4(i) shows that the number of candidate instances has little effect on the final classification result, and the SVM outperforms the other two classifiers.

### *4.4. Pairwise accuracy comparison*

Both distance-based benchmark embedding (DBE) and tri-perspective embedding (TEMI) technique aggregate instance-level information to obtain new bag embedding vectors. The vector size of distance-based benchmark embedding method depends on the number of representative instances. Whereas, the vector size of tri-perspective embedding method depends on the size of the input data dimension.

Figure 5 statistics the pairwise accuracy comparison of the two embedding methods on 38 data sets. We conducted an experiment to compare the classification performance



(a) Data: Musk1

(b) Data: Elephant

(c) Data: Mutagenesis1

(d) Data: News.aa

(e) Data: News.rsh

(f) Data: News.tpmd

(g) Data: Web4

(h) Data: Web9

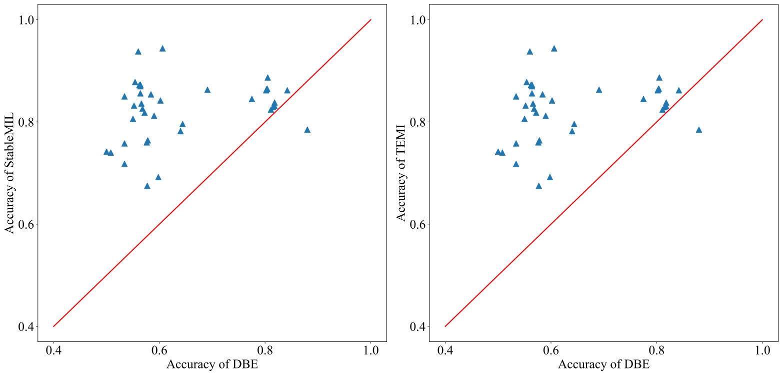
(i) Data: Mean

Figure 4: Parameter analysis of TEMI with the proportion of candidate instances in each bag and three classifiers. (a)–(h) show the accuracy on 8 data sets, respectively. (i) shows the mean accuracy on 8 data sets.

of distance-based benchmark and tri-perspective embedding methods. StableMIL is used to compare the classification performance of distance-based method. The reason is that they use the same distance function. To ensure a fair comparison, we used the same parameter settings: 50% candidate instances and the SVM classifier. Figure 5(a) illustrates that the distance-based embedding (DBE) provides better accuracy values than StableMIL on 29 data sets. This demonstrates the effectiveness of the mutual instance selection method proposed in this paper. Figure 5(b) illustrates that TEMI outperforms DBE in most data sets. This shows that, based on the same representative instance, the TEMI can improve the distinguishability of the bag in the new feature space more than DBE.

### *4.5. Ablation experiments*

Figure 6 shows the ablation experiments of the encoding-based embedding method. The experimental results are divided into 3 levels (the higher the level, the higher the



(a) StableMIL vs DBE

(b) TEMI vs DBE

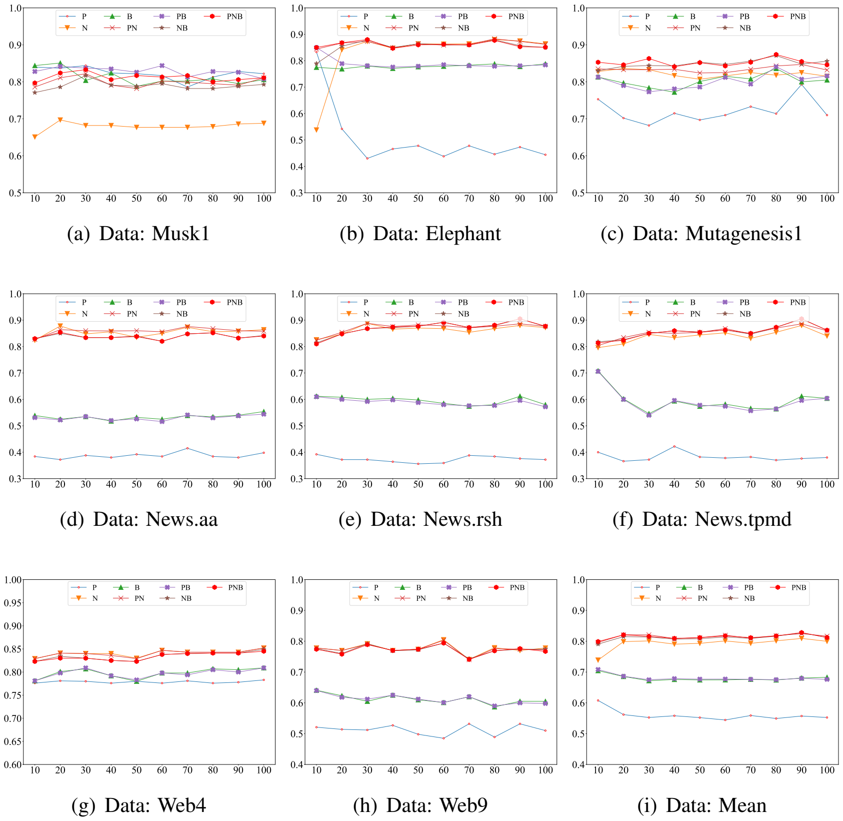
Figure 5: Pairwise accuracy of bag embedding methods on 38 data sets. Figure 5(a) shows the accuracy of StableMIL versus DBE. Figure 5(b) shows the accuracy of TEMI versus DBE. Each axis corresponds to the accuracy of a method. Each dot represents the accuracy for a particular data set. The dots on the *x*=*y* line indicates that the performance of the two methods is roughly the same. The dots above the line indicate that on a particular data set, the *y*-axis method has better accuracy than the *x*-axis method.

accuracy): Level-One: “PNB”, “PN”, “NB” and “N”; Level-Two: “B” and “PB”; Level-Three: “P”. In terms of performance on most data sets, embedding only from a single perspective is inferior to combining perspectives. Specifically, the results of embedding from “P” are the worst, which may be because the positive representative instances cannot effectively divide the bag. The positive representative instances would incorrectly treat an instance in the negative bag as a positive instance. Therefore, the distinguishability of embedded bags is reduced. However, the results of embedding from “N” perform best in a single perspective, because the negative representative instances can effectively divide the negative instances in the bag. Instances in the negative bag are all negative, and only some of the instances in the positive bag are negative. This highlights the distinguishability of the bag in the new feature space. It is not difficult to see that the results of “PN” and “NB” are second only to “PNB” and sometimes slightly higher than “PNB”. But it can be seen from Figure 6(i) that the mean accuracy of “PNB” is higher than that of “PN” and “NB”. Therefore, to reduce the redundancy of parameters, we adopt the “PNB” with the highest classification performance for bag embedding.

### *4.6. Performance Comparison*

Table 4 compares the performance of TEMI with eight comparison algorithms. $d$ is the dimension of the data set. The small black dots emphasize the best performance value for each data set. The mean ranking represents the average of the classification performance rankings of the current algorithm on all data sets.

Experimental results show that TEMI has superior classification performance on 60.5% of the data sets. The average classification score is 5% higher than the second



(a) Data: Musk1

(b) Data: Elephant

(c) Data: Mutagenesis1

(d) Data: News.aa

(e) Data: News.rsh

(f) Data: News.tpmd

(g) Data: Web4

(h) Data: Web9

(i) Data: Mean

Figure 6: Accuracy of ablation experiments under 8 data. “P”, “N”, and “B” mean to keep only the positive, negative, and bag perspective embedding method. “PN” represents the concatenated “P” and “N” perspective embedding method. “PB” represents the concatenated “P” and “B” perspective embedding method. “NB” represents the concatenated “N” and “B” perspective embedding method. “PNB” represents bag embedding from three perspectives.

place and 20.7% higher than the penultimate place. Specifically, the 7 and 12 data sets on Web recommendation and Newsgroups have the highest classification performance, respectively. It may be because: a) The framework of instance selection can effectively select the instance with the largest amount of information; and b) Embedding the bag into the new feature space from three perspectives can retain as much information as possible.

It is worth noting that: a) Even if other algorithms do not get the best classification performance, their performance on some Newsgroup data sets is better than TEMI. For example, miFV has considerable advantages on the Messidor and News.rm data sets. The reason may be that the Gaussian mixture model proposed by miFV can effectively dig out key information. The performance of StableMIL on the Mutagenesis2 data set is also better than TEMI. This may be because it is suitable for low-dimensional data; and b) The poor classification performance of MILFM, MILDM, StableMIL, and ELDB on Newsgroups and Web recommendation data sets may be caused by the distribution of data instances. Taking Newsgroup data as an example, the positive bag only contains 3% of posts in the target category and 97% of posts randomly selected from other categories. Therefore, the representative instance or representative bag they choose may not be a representative.

Table 4: Accuracy ($mean \pm std$) with standard deviations on 38 MIL data sets. The highest average accuracy is marked with •.

| Datasets | $(d)$ | BAMIC | MILFM | Simple-MI | miFV | miVLAD | MILDM | StableMIL | ELDB | **TEMI** |
|---|---|---|---|---|---|---|---|---|---|---|
| Musk1 | (166) | $.853_{\pm.019}$ | $.871_{\pm.005}$ | $.799_{\pm.018}$ | •$.898_{\pm.023}$ | $.802_{\pm.028}$ | $.824_{\pm.019}$ | $.848_{\pm.004}$ | $.880_{\pm.018}$ | $.862_{\pm.011}$ |
| Musk2 | (166) | •$.866_{\pm.010}$ | $.822_{\pm.035}$ | $.756_{\pm.027}$ | $.850_{\pm.017}$ | $.812_{\pm.017}$ | $.826_{\pm.006}$ | $.822_{\pm.007}$ | $.861_{\pm.017}$ | $.824_{\pm.026}$ |
| Elephant | (230) | $.756_{\pm.020}$ | $.817_{\pm.013}$ | $.823_{\pm.013}$ | $.845_{\pm.010}$ | $.852_{\pm.007}$ | $.761_{\pm.010}$ | $.765_{\pm.023}$ | $.758_{\pm.032}$ | •$.887_{\pm.019}$ |
| Fox | (230) | $.602_{\pm.030}$ | $.454_{\pm.022}$ | $.620_{\pm.014}$ | $.619_{\pm.011}$ | $.635_{\pm.025}$ | $.588_{\pm.022}$ | $.536_{\pm.014}$ | $.607_{\pm.020}$ | •$.675_{\pm.019}$ |
| Tiger | (230) | $.693_{\pm.009}$ | $.733_{\pm.011}$ | $.804_{\pm.009}$ | $.770_{\pm.012}$ | $.855_{\pm.003}$ | $.643_{\pm.016}$ | $.710_{\pm.030}$ | $.722_{\pm.020}$ | •$.863_{\pm.012}$ |
| Mutagenesis1 | (7) | $.848_{\pm.009}$ | $.848_{\pm.002}$ | $.837_{\pm.011}$ | $.823_{\pm.019}$ | $.818_{\pm.016}$ | $.807_{\pm.017}$ | $.838_{\pm.021}$ | $.849_{\pm.017}$ | •$.864_{\pm.014}$ |
| Mutagenesis2 | (7) | $.830_{\pm.010}$ | $.835_{\pm.011}$ | $.795_{\pm.010}$ | $.834_{\pm.015}$ | $.780_{\pm.037}$ | $.755_{\pm.029}$ | •$.840_{\pm.037}$ | $.828_{\pm.008}$ | $.785_{\pm.041}$ |
| Messidor | (687) | $.633_{\pm.006}$ | $.545_{\pm.000}$ | $.625_{\pm.019}$ | •$.712_{\pm.006}$ | $.686_{\pm.004}$ | $.545_{\pm.024}$ | $.545_{\pm.000}$ | $.638_{\pm.004}$ | $.692_{\pm.010}$ |
| Ucsb_breast | (708) | $.704_{\pm.050}$ | $.540_{\pm.028}$ | $.804_{\pm.015}$ | $.852_{\pm.020}$ | $.776_{\pm.015}$ | $.556_{\pm.008}$ | $.544_{\pm.020}$ | $.704_{\pm.036}$ | •$.856_{\pm.029}$ |
| News.aa | (200) | $.842_{\pm.012}$ | $.528_{\pm.010}$ | $.824_{\pm.010}$ | $.828_{\pm.013}$ | $.832_{\pm.007}$ | $.552_{\pm.056}$ | $.520_{\pm.042}$ | $.856_{\pm.020}$ | •$.878_{\pm.007}$ |
| News.cg | (200) | $.812_{\pm.010}$ | $.540_{\pm.011}$ | $.776_{\pm.008}$ | $.796_{\pm.010}$ | $.814_{\pm.010}$ | $.496_{\pm.051}$ | $.512_{\pm.033}$ | $.811_{\pm.010}$ | •$.818_{\pm.007}$ |
| News.co | (200) | $.724_{\pm.008}$ | $.473_{\pm.024}$ | $.552_{\pm.044}$ | $.730_{\pm.017}$ | $.714_{\pm.027}$ | $.468_{\pm.050}$ | $.498_{\pm.058}$ | •$.737_{\pm.013}$ | $.718_{\pm.028}$ |
| News.csi | (200) | $.806_{\pm.020}$ | $.531_{\pm.021}$ | $.752_{\pm.007}$ | $.788_{\pm.013}$ | $.768_{\pm.023}$ | $.562_{\pm.045}$ | $.518_{\pm.073}$ | $.797_{\pm.013}$ | •$.836_{\pm.016}$ |
| News.csm | (200) | $.796_{\pm.010}$ | $.500_{\pm.011}$ | $.774_{\pm.008}$ | $.782_{\pm.015}$ | $.796_{\pm.028}$ | $.474_{\pm.036}$ | $.542_{\pm.031}$ | $.811_{\pm.017}$ | •$.826_{\pm.022}$ |
| News.cw | (200) | $.790_{\pm.017}$ | $.532_{\pm.056}$ | $.680_{\pm.036}$ | •$.852_{\pm.016}$ | $.816_{\pm.022}$ | $.594_{\pm.045}$ | $.524_{\pm.036}$ | $.797_{\pm.014}$ | $.842_{\pm.039}$ |
| News.mf | (200) | $.684_{\pm.012}$ | $.522_{\pm.034}$ | $.566_{\pm.033}$ | •$.748_{\pm.019}$ | $.706_{\pm.029}$ | $.486_{\pm.043}$ | $.514_{\pm.037}$ | $.702_{\pm.006}$ | $.740_{\pm.019}$ |
| News.ra | (200) | $.770_{\pm.017}$ | $.516_{\pm.008}$ | $.750_{\pm.000}$ | $.778_{\pm.029}$ | $.794_{\pm.014}$ | $.544_{\pm.038}$ | $.540_{\pm.037}$ | $.768_{\pm.008}$ | •$.806_{\pm.027}$ |
| News.rm | (200) | $.816_{\pm.029}$ | $.550_{\pm.026}$ | $.774_{\pm.019}$ | •$.874_{\pm.008}$ | $.834_{\pm.031}$ | $.548_{\pm.058}$ | $.560_{\pm.028}$ | $.797_{\pm.024}$ | $.850_{\pm.009}$ |
| News.rsb | (200) | •$.834_{\pm.010}$ | $.570_{\pm.021}$ | $.752_{\pm.004}$ | $.830_{\pm.014}$ | $.800_{\pm.014}$ | $.490_{\pm.040}$ | $.526_{\pm.029}$ | $.821_{\pm.011}$ | $.832_{\pm.013}$ |
| News.rsh | (200) | $.820_{\pm.006}$ | $.500_{\pm.000}$ | $.798_{\pm.013}$ | $.884_{\pm.016}$ | $.860_{\pm.021}$ | $.482_{\pm.071}$ | $.500_{\pm.039}$ | $.822_{\pm.010}$ | •$.938_{\pm.020}$ |
| News.sc | (200) | $.762_{\pm.004}$ | $.512_{\pm.007}$ | $.734_{\pm.005}$ | $.772_{\pm.007}$ | $.822_{\pm.028}$ | $.476_{\pm.057}$ | $.534_{\pm.034}$ | $.771_{\pm.010}$ | •$.854_{\pm.016}$ |
| News.se | (200) | $.940_{\pm.000}$ | $.530_{\pm.000}$ | $.920_{\pm.000}$ | $.926_{\pm.012}$ | $.920_{\pm.009}$ | $.568_{\pm.029}$ | $.530_{\pm.000}$ | $.940_{\pm.000}$ | •$.944_{\pm.185}$ |
| News.sm | (200) | $.824_{\pm.008}$ | $.520_{\pm.000}$ | $.716_{\pm.010}$ | $.850_{\pm.013}$ | $.812_{\pm.017}$ | $.534_{\pm.051}$ | $.520_{\pm.041}$ | $.827_{\pm.008}$ | •$.870_{\pm.017}$ |
| News.sr | (200) | •$.842_{\pm.010}$ | $.490_{\pm.049}$ | $.780_{\pm.006}$ | $.802_{\pm.010}$ | $.766_{\pm.024}$ | $.518_{\pm.032}$ | $.552_{\pm.045}$ | $.839_{\pm.010}$ | $.764_{\pm.010}$ |
| News.ss | (200) | $.820_{\pm.000}$ | $.508_{\pm.007}$ | $.828_{\pm.010}$ | $.868_{\pm.013}$ | $.826_{\pm.028}$ | $.486_{\pm.029}$ | $.524_{\pm.048}$ | $.818_{\pm.013}$ | •$.872_{\pm.007}$ |
| News.tpg | (200) | $.796_{\pm.008}$ | $.556_{\pm.008}$ | $.776_{\pm.008}$ | $.774_{\pm.015}$ | $.792_{\pm.012}$ | $.462_{\pm.078}$ | $.486_{\pm.041}$ | $.801_{\pm.006}$ | •$.812_{\pm.012}$ |
| News.tpmd | (200) | $.832_{\pm.004}$ | $.522_{\pm.035}$ | $.842_{\pm.012}$ | $.784_{\pm.010}$ | $.832_{\pm.012}$ | $.588_{\pm.046}$ | $.550_{\pm.033}$ | $.829_{\pm.003}$ | •$.873_{\pm.008}$ |
| News.tpmc | (200) | $.682_{\pm.004}$ | $.608_{\pm.024}$ | $.690_{\pm.028}$ | $.750_{\pm.020}$ | •$.768_{\pm.039}$ | $.520_{\pm.041}$ | $.564_{\pm.037}$ | $.679_{\pm.011}$ | $.758_{\pm.016}$ |
| News.trm | (200) | $.722_{\pm.012}$ | $.518_{\pm.013}$ | $.614_{\pm.008}$ | •$.750_{\pm.013}$ | $.738_{\pm.019}$ | $.462_{\pm.049}$ | $.536_{\pm.034}$ | $.724_{\pm.017}$ | $.742_{\pm.020}$ |
| Web1 | $(5,863)$ | •$.851_{\pm.019}$ | $.825_{\pm.032}$ | $.800_{\pm.006}$ | $.838_{\pm.009}$ | $.811_{\pm.019}$ | $.820_{\pm.019}$ | $.822_{\pm.019}$ | $.825_{\pm.020}$ | $.831_{\pm.012}$ |
| Web2 | $(6,519)$ | $.796_{\pm.004}$ | $.831_{\pm.016}$ | •$.840_{\pm.014}$ | $.824_{\pm.004}$ | $.827_{\pm.017}$ | $.825_{\pm.004}$ | $.805_{\pm.020}$ | $.829_{\pm.022}$ | $.838_{\pm.017}$ |
| Web3 | $(6,306)$ | $.813_{\pm.004}$ | $.825_{\pm.033}$ | $.796_{\pm.016}$ | $.818_{\pm.013}$ | $.820_{\pm.013}$ | $.822_{\pm.009}$ | $.807_{\pm.009}$ | $.814_{\pm.006}$ | •$.829_{\pm.015}$ |
| Web4 | $(6,059)$ | $.778_{\pm.011}$ | $.778_{\pm.032}$ | $.809_{\pm.006}$ | $.805_{\pm.007}$ | $.856_{\pm.016}$ | $.807_{\pm.021}$ | $.722_{\pm.031}$ | $.798_{\pm.013}$ | •$.865_{\pm.015}$ |
| Web5 | $(6,407)$ | $.791_{\pm.006}$ | $.795_{\pm.012}$ | $.785_{\pm.016}$ | $.771_{\pm.009}$ | $.831_{\pm.009}$ | $.822_{\pm.020}$ | $.781_{\pm.061}$ | $.781_{\pm.012}$ | •$.845_{\pm.006}$ |
| Web6 | $(6,417)$ | $.795_{\pm.012}$ | $.845_{\pm.035}$ | $.793_{\pm.007}$ | $.778_{\pm.011}$ | $.845_{\pm.013}$ | $.842_{\pm.009}$ | $.733_{\pm.034}$ | $.809_{\pm.023}$ | •$.862_{\pm.011}$ |
| Web7 | $(6,450)$ | $.547_{\pm.035}$ | $.611_{\pm.009}$ | $.627_{\pm.022}$ | $.684_{\pm.031}$ | $.731_{\pm.029}$ | $.624_{\pm.051}$ | $.620_{\pm.275}$ | $.528_{\pm.045}$ | •$.760_{\pm.014}$ |
| Web8 | $(5,999)$ | $.491_{\pm.021}$ | $.624_{\pm.014}$ | $.640_{\pm.023}$ | $.709_{\pm.029}$ | $.698_{\pm.025}$ | $.569_{\pm.040}$ | $.590_{\pm.319}$ | $.505_{\pm.035}$ | •$.782_{\pm.016}$ |
| Web9 | $(6,279)$ | $.509_{\pm.045}$ | $.585_{\pm.025}$ | $.671_{\pm.035}$ | $.729_{\pm.023}$ | $.780_{\pm.025}$ | $.551_{\pm.019}$ | $.549_{\pm.037}$ | $.493_{\pm.044}$ | •$.796_{\pm.027}$ |
| Average | | $.765_{\pm.014}$ | $.624_{\pm.018}$ | $.751_{\pm.014}$ | $.798_{\pm.015}$ | $.795_{\pm.019}$ | $.613_{\pm.034}$ | $.616_{\pm.045}$ | $.768_{\pm.016}$ | •$.823_{\pm.016}$ |
| Mean rank | | 4.79 | 6.68 | 5.37 | 3.55 | 3.76 | 7.29 | 7.26 | 4.45 | •1.84 |

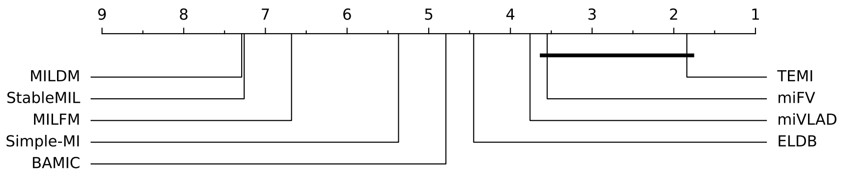


Figure 7: Comparison of TEMI with 8 comparison algorithms with Bonferroni-Dunn test. Algorithms not connected to TEMI in the CD plot were considered to have significant performance of the control algorithm (CD = 1.71, significance level 0.05).

Figure 7 reports the post hoc Bonferroni-Dunn test [49] on 9 algorithms. The critical difference (CD) plot at the 0.05 significance level. The mean ranks for each algorithm are marked along the axis (lower grades on the left). In addition, algorithms with a mean ranking within one CD of TEMI are connected by thick lines. Otherwise, any TEMI-independent algorithm is considered significantly different.

### *4.7. Efficiency comparison*

Table 5: The CPU runtime (in seconds) of one time 10CV of the comparison algorithm on the 5 MIL classification data sets.

| Data sets | $(d/n/N)$ | BAMIC | MILFM | Simple-MI | miFV | miVLAD | MILDM | StableMIL | ELDB | **TEMI** |
|---|---|---|---|---|---|---|---|---|---|---|
| Musk1 | $(230/1,320/200)$ | 0.234 | 4.947 | •0.156 | 1.462 | 0.659 | 1.386 | 8.124 | 1.025 | 0.942 |
| Elephant | $(230/1,320/200)$ | 1.373 | 11.356 | •0.194 | 3.825 | 1.106 | 7.454 | 14.747 | 5.056 | 3.621 |
| News.aa | $(200/5,443/100)$ | 19.417 | 75.589 | •0.172 | 4.277 | 1.461 | 66.069 | 56.242 | 54.389 | 17.779 |
| News.cg | $(200/3,094/100)$ | 6.686 | 32.179 | •0.169 | 3.002 | 1.084 | 22.543 | 35.584 | 20.288 | 6.823 |
| Web4 | $(6,059/3,423/113)$ | 24.678 | 103.770 | •0.257 | 386.411 | 16.823 | 57.940 | 621.167 | 62.513 | 71.996 |
| Mean rank | | 3.40 | 8.00 | •1.00 | 5.20 | 2.20 | 6.40 | 8.60 | 5.60 | 4.60 |

Table 5 shows the runtime of TEMI compared with 8 competing algorithms on 5 data sets. Section 3.5 analyzes the worst time complexity of TEMI as $O(n^2d)$. By contrast, BAMIC costs $O(N^2d)$, MILFM costs $O(n^2d)$, Simple-MI costs $O(Nd)$, miFV costs $O(nd)$, miVLAD costs $O(nd)$, MILDM costs $O(n^2d)$, StableMIL costs $O(n^2d)$, and ELDB costs $O(n^2d)$. The results show that the speed of TEMI is slightly lower than that of Simple-MI, miVLAD, and BAMIC. This may be because Simple-MI does not need to consume a lot of time to calculate the distance of instances. The $k$-Means algorithm used by miVLAD has low time complexity. BAMIC only needs to calculate the distance between bags. The main time overhead of TEMI is in the instance selection phase, and its time complexity is $O(n^2d)$. However, the classification performance of these 3 algorithms is worse than TEMI.

## 5. Conclusion

In this paper, we propose TEMI to embed bags from three perspectives: positive, negative, and bag. According to Figure 5, TEMI outperforms distance-based benchmark embedding algorithms. This demonstrates its effectiveness in mining the overall and extreme information of the bag. According to Figure 6, "PNB" embedding outperforms other joint embedding methods. This shows its ability to enhance the distinguishability of bags. According to Table 4, TEMI outperforms various state-of-the-art algorithms especially on Web data sets.

There are still some topics that require further investigation. First, the representative instance scoring strategy is rather complex, making the algorithm time-intensive. Second, experiments are not performed on more image data sets. Therefore, we will explore better instance scoring mechanisms to optimize our algorithm and try to conduct experiments on more authoritative image data sets.

## Acknowledgments

This work was supported by Central Government Funds of Guiding Local Scientific and Technological Development (2021ZYD0003), the National Natural Science Foundation of China (61976194, 62076221), and the Open Project of Key Laboratory of Oceanographic Big Data Mining and Application of Zhejiang Province (OBDMA202102).

## References

[1] J. Foulds, E. Frank, A review of multi-instance learning assumptions, The Knowledge Engineering Review 25 (1) (2010) 1–25.

[2] M.-A. Carbonneau, V. Cheplygina, E. Granger, G. Gagnon, Multiple instance learning: A survey of problem characteristics and applications, Pattern Recognition 77 (2018) 329–353.

[3] T. G. Dietterich, R. H. Lathrop, T. Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, Artificial Intelligence 89 (1-2) (1997) 31–71.

[4] X. Huo, M. Li, Z.-H. Zhou, Control flow graph embedding based on multi-instance decomposition for bug localization, in: AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 4223–4230.

[5] G.-H. Liu, J.-Y. Yang, Z. Li, Content-based image retrieval using computational visual attention model, Pattern Recognition 48 (8) (2015) 2554–2566.

[6] B. Li, W. Xiong, O. Wu, W. Hu, S. Maybank, S. Yan, Horror image recognition based on context-aware multi-instance learning (2015).

[7] S. Conjeti, M. Paschali, A. Katouzian, N. Navab, Deep multiple instance hashing for scalable medical image retrieval, in: Medical Image Computing and Computer Assisted Intervention, 2017, pp. 550–558.

[8] M. Yang, Y.-X. Zhang, X. Z. Wang, F. Min, Multi-instance ensemble learning with discriminative bags, IEEE Transactions on Systems, Man, and Cybernetics: Systems (2021) 1–12.

[9] B. Liu, Y. Xiao, Z. Hao, A selective multiple instance transfer learning method for text categorization problems, Knowledge-Based Systems 141 (2018) 178–187.

[10] X.-S. Wei, H.-J. Ye, X. Mu, J. Wu, C. Shen, Z.-H. Zhou, Multi-instance learning with emerging novel class, IEEE Transactions on Knowledge and Data Engineering 33 (5) (2019) 2109–2120.

[11] B.-C. Xu, K. M. Ting, 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, 2019, pp. 941–949.

[12] D. S. Tarragó, C. Cornelis, R. Bello, F. Herrera, A multi-instance learning wrapper based on the Rocchio classifier for web index recommendation, Knowledge-Based Systems 59 (0950-7051) (2014) 173–181.

[13] O. Maron, T. Lozano-Pérez, A framework for multiple-instance learning, Advances in Neural Information Processing Systems (1998) 570–576.

[14] S. Andrews, I. Tsochantaridis, T. Hofmann, Support vector machines for multiple-instance learning, in: Conference and Workshop on Neural Information Processing Systems, Vol. 2, 2002, p. 7.

[15] Y. Xiao, B. Liu, Z. Hao, L. Cao, A similarity-based classification framework for multiple-instance learning, IEEE Transactions on Cybernetics 44 (4) (2014) 500–515.

[16] A. W. Faria, F. G. F. Coelho, A. Silva, H. P. Rocha, G. Almeida, A. P. Lemos, A. P. Braga, MILKDE: A new approach for multiple instance learning based on positive instance selection and kernel density estimation, Engineering Applications of Artificial Intelligence 59 (2017) 196–204.

[17] Y. Chen, J. Z. Wang, Image categorization by learning and reasoning with regions, The Journal of Machine Learning Research 5 (2004) 913–939.

[18] Z.-H. Zhou, Y.-Y. Sun, 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, 2009, pp. 1249–1256.

[19] Gabriella, A. Cano, S. Ventura, Mirsvm: Multi-instance support vector machine with bag representatives, Pattern Recognition 79 (2018) 228–241.

[20] Y. X. Chen, J. B. Bi, J. Z. Wan, MILES: Multiple-instance learning via embedded instance selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (12) (2006) 1931–1947.

[21] X. Wang, D. Wei, H. Cheng, J. Fang, Multi-instance learning based on representative instance and feature mapping, Neurocomputing 216 (2016) 790–796.

[22] J. Wu, S. R. Pan, X. Q. Zhu, C. Q. Z. abd Xin Dong Wu, Multi-instance learning with discriminative bag mapping, IEEE Transactions on Knowledge and Data Engineering 30 (6) (2018) 1065–1080.

[23] Z.-H. Zhou, M.-L. Zhang, Multi-instance clustering with applications to multi-instance prediction, Applied Intelligence 31 (1) (2009) 47–68.

[24] Y. F. Zhou, R.-K. Antonio, J. Zhou, MILIS: Multiple instance learning with instance selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (5) (2011) 958–977.

[25] W. Zhang, L. Liu, J. Li, Robust multi-instance learning with stable instances, in: European Conference on Artificial Intelligence, Vol. 325, 2020, pp. 1682–1689.

[26] Z.-H. Zhou, M.-L. Zhang, Solving multi-instance problems with classifier ensemble based on constructive clustering, Knowledge and Information Systems 11 (2) (2007) 155–170.

[27] X.-S. Wei, J. Wu, Z.-H. Zhou, Scalable multi-instance learning, in: IEEE International Conference on Data Mining, 2014, pp. 1037–1042.

[28] E. Ş. Küçükaşcı, M. G. Baydoğan, Bag encoding strategies in multiple instance learning problems, Information Sciences 467 (2018) 559–578.

[29] W.-J. Li, D. Y. Yeung, MILD: Multiple-instance learning via disambiguation, IEEE Transactions on Knowledge and Data Engineering 22 (1) (2010) 76–89.

[30] R. C. Hong, M. Wang, Y. Gao, D. C. Tao, X. L. Li, X. D. Wu, Image annotation by multiple-instance learning with discriminative feature mapping and selection, IEEE Transactions on Cybernetics 44 (5) (2014) 669–680.

[31] M.-A. Carbonneau, E. Granger, A. J. Raymond, G. Gagnon, Robust multiple-instance learning ensembles using random subspace instance selection, Pattern Recognition 58 (2016) 83–99.

[32] Y.-L. Zhang, Z.-H. Zhou, Multi-instance learning with key instance shift, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, 2017, pp. 3441–3447.

[33] M. Yang, W.-X. Zeng, F. Min, Multi-instance embedding learning through high-level instance selection, in: Advances in Knowledge Discovery and Data Mining, 2022, pp. 122–133.

[34] X.-S. Wei, J. X. Wu, Z.-H. Zhou, Scalable algorithms for multi-instance learning, IEEE Transactions on Neural Networks and Learning Systems 28 (4) (2017) 975–987.

[35] X. Wang, Y. Yan, P. Tang, W. Liu, X. Guo, Bag similarity network for deep multi-instance learning, Information Sciences 504 (2019) 578–588.

[36] D. Xu, J. Wu, D. Li, Y. Tian, X. Zhu, X. Wu, Sale: Self-adaptive LSH encoding for multi-instance learning, Pattern Recognition 71 (2017) 460–482.

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

[38] A. Rodriguez, A. Laio, Clustering by fast search and find of density peaks, science 344 (6191) (2014) 1492–1496.

[39] K.-T. Lai, D. Liu, M.-S. Chen, S.-F. Chang, Recognizing complex events in videos by learning key static-dynamic evidences, in: Computer Vision – ECCV 2014, 2014, pp. 675–688.

[40] T. Malisiewicz, A. Gupta, A. A. Efros, Ensemble of exemplar-svms for object detection and beyond, in: International Conference on Computer Vision, 2011, pp. 89–96.

[41] M. Rastegari, H. Hajishirzi, A. Farhadi, Discriminative and consistent similarities in instance-level multiple instance learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 740–748.

[42] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the fisher vector: Theory and practice, International journal of computer vision 105 (3) (2013) 222–245.

[43] E. Decencière, X. W. Zhang, G. Cazuguel, B. Lay, et al., Feedback on a publicly distributed image database: The messidor database, Image Analysis & Stereology 33 (3) (2014) 231–234.

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

[45] M. Kandemir, C. Zhang, F. A. Hamprecht, Empowering multiple instance histopathology cancer diagnosis by cell graphs, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2014, pp. 228–235.

[46] A. Srinivasan, S. Muggleton, R. King, Comparing the use of background knowledge by inductive logic programming systems, in: Proceedings of the 5th International Workshop on Inductive Logic Programming, 1995, pp. 199–230.

[47] Z.-H. Zhou, K. Jiang, M. Li, Multi-instance learning based web mining, Applied Intelligence 22 (2) (2005) 135–147.

[48] J. Amores, Multiple instance classification: Review, taxonomy and comparative study, Artificial Intelligence 201 (4) (2013) 81–105.

[49] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.