Hierarchical Sampling for Multi-Instance Ensemble Learning

Hanning Yuan, Meng Fang, and Xingquan Zhu, *Senior Member*, *IEEE*

Abstract—In this paper, we propose a Hierarchical Sampling-based Multi-Instance ensemble LEarning (HSMILE) method. Due to the unique multi-instance learning nature, a positive bag contains at least one positive instance whereas samples (instance and sample are interchangeable terms in this paper) in a negative bag are all negative, simply applying bootstrap sampling to individual bags may severely damage a positive bag because a sampled positive bag may not contain any positive sample at all. To solve the problem, we propose to calculate probable positive sample distributions in each positive bag and use the distributions to preserve at least one positive instance in a sampled bag. The hierarchical sampling involves inter- and intrabag sampling to adequately perturb bootstrap sample sets for multi-instance ensemble learning. Theoretical analysis and experiments confirm that HSMILE outperforms existing multi-instance ensemble learning methods.

Index Terms—Multi-instance learning, ensemble learning, hierarchical sampling

1 Introduction

MULTI-INSTANCE learning (MIL), originated from drug activity predictions [4], represents a special type of machine learning task where a group of instances (i.e., a bag) shares one label, but no label is available for individual instances inside the bag. A bag is positive if it contains at least one positive instance; otherwise, it is labeled as a negative bag. Given a number of labeled bags, the goal of MIL is to construct a learner to predict a previously unseen bag to be either positive or negative. MIL represents a large body of real-world applications. Examples include content-based image retrieval [1], visual tracking [15], and gene annotation [7].

The main challenge of the MIL lies in the fact that genuine labels of individual instances in a positive bag remain unknown. Simply propagating bag labels to instances inside each bag may introduce a significant amount of label errors [13]. A number of MIL algorithms exist to either

1. build bag-level discriminate machines, such as 1-norm SVM [3], MI kernel [7],
2. identify most probably positive samples in each bag and convert MIL into single instance learning problems [14];
3. synthesize rules from positive and negative bags [16], or
4. simply propagate bag labels to the instances such that generic instance-based learner can apply.

Similar to other machine learning algorithms, most MI learners are data driven with unstable performances. As ensemble learning (such as Bagging predictor [2]) is a general treatment to boost unstable learners, combining ensemble and MIL has also been studied [12] for multi-instance learners.

Intuitively, by taking each bag as a single observation, one can form bootstrap sets using random bag sampling to build an MI ensemble [12]. In this paper, we refer to this approach as traditional MI ensemble (TMIE). An inherent disadvantage of TMIE is the low diversity of the ensemble learners [6], because once a bag is sampled, all instances in the bag are forwarded to the bootstrap set, which violates the random nature of bootstrap sampling. On the other hand, simply applying bootstrap sampling to all instances without considering bag constraint can achieve maximum diversity, but may end up forming a positive bag containing negative samples only.

In this paper, we propose a hierarchical sampling method, at instance and bag levels, for multi-instance ensemble learning. At interbag level, a bootstrap set is constructed by treating each bag as an observation. At intrabag (i.e., instance) level, sampling is employed to perturb instances inside the bag to construct diverse base learners. The main technical challenge of the hierarchical inter- and intrabag sampling stems from the reality that genuine labels of instances in a positive bag are unknown. In the paper, we propose a modified sampling method to ensure that at least one probable positive instance is preserved in each sampled positive bag.

2 HSMILE Algorithm Details

In this section, we will describe the HSMILE algorithm, where Fig. 1 shows the conceptual view of the hierarchical sampling process. HSMILE, in Algorithm 1, takes four parameters, T, L, I, and J as the input, and the output is an ensemble for classifying a bag B_x with unknown label.

Algorithm 1. The HSMILE algorithm.

Require: A training set with M bags $T = \{B_1, \ldots, B_M\}$;
Multi-instance learner L; # of Inter-bag sampling times I; # of Intra-bag sampling times J; a test bag with unknown class label B_x
for $i \leftarrow 1$ to I **do**
  $T_i \leftarrow$ Inter-bag sampling from T
  for $j \leftarrow 1$ to J **do**
    $T_i^j \leftarrow$ Intra-bag sampling (T_i) // Algorithm 2
    $L_i^j \leftarrow$ Training an MI base learner from T_i^j
  end for
  $L_i(B_x) \leftarrow \arg\max_{y \in \mathcal{Y}} \Sigma_{j=1, L_i^j(B_x)=y}^{J} 1$
end for
$y_x \leftarrow \arg\max_{y \in \mathcal{Y}} \Sigma_{i=1, L_i(B_x)=y}^{I} 1$
return y_x the label of B_x

At the first step, the interbag sampling repeats I times on T by treating each bag as an observation and builds a set of bootstrap sets $T_1, \ldots, T_I$, each of which has the same number of bags as T. In the second stage, the intrabag sampling, as shown in Algorithm 2, is applied J times to each set T_i. At the jth ($j \leq J$) time of the intrabag sampling (Algorithm 2), depending on whether a bag B_τ of T_i is a positive or a negative bag, instances in B_τ are sampled using different approaches to form a new bag. All new bags of T_i form the T_i^j, from which a base MI classifier L_i^j is learned and is used as an ensemble member to predict a test bag's label.

Algorithm 2. Intra-bag sampling (T).

Require: A training set with M bags $T = \{B_1, \ldots, B_M\}$
for $i \leftarrow 1$ to M **do**
  $m_i \leftarrow$ the number of instance in B_i;
  if the bag label of B_i is ***positive*** **then**

- *H. Yuan is with the School of Software, Beijing Institute of Technology, 5 South Zhongguancun Street, Haidian District, Beijing 100081, P.R. China. E-mail: yhn@whut.edu.cn.*
- *M. Fang is with the Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, 235 Jones Street, NSW, Australia. E-mail: Meng.Fang@student.uts.edu.au.*
- *X. Zhu is with the Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL. E-mail: xzhu3@fau.edu.*

Manuscript received 20 May 2011; revised 28 Dec. 2011; accepted 28 Oct. 2012; published online 14 Dec. 2012.
Recommended for acceptance by L. Khan.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-05-0280.
Digital Object Identifier no. 10.1109/TKDE.2012.245.

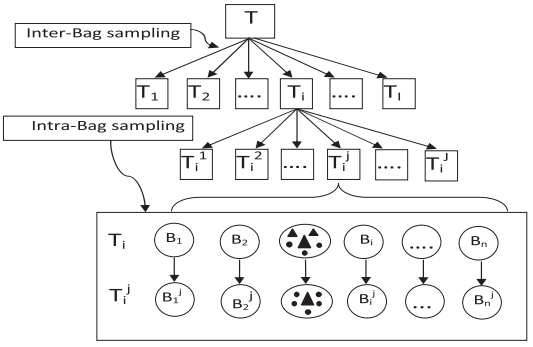


Fig. 1. The conceptual view of the HSMILE. Dots denote negative instances and triangles represent positive samples. The size of the triangle implies the likelihood of an instance to be genuinely positive.

$[P[b_{i,1} \mid B^-], \ldots, P[b_{i,|B_i|} \mid B^-]] \leftarrow$ calculate probable positive instance distributions in B_i;
$x^*_{B_i} \leftarrow$ Rejection sampling to select a probable positive instance from B_i using $[P[b_{i,1} \mid B^-], \ldots, P[b_{i,|B_i|} \mid B^-]]$;
$B_i^{'} \leftarrow x^*_{B_i}$;
$B_i^{'} \leftarrow$ randomly sample $m_i - 1$ instances from B_i;
else if the bag label of B_i is ***negative*** **then**
$B_i^{'} \leftarrow$ Bootstrap sampling selecting m_i instances from B_i;
end if
end for
return $T^{'} = \{B_1^{'}, \ldots, B_M^{'}\}$

The ensemble of HSMILE has two tiers. At the first tier, an ensemble L_i is achieved by combining a set of base learners, each of which is trained from T_i^j. The outputs of L_i are combined to form the second tier ensemble for final prediction.

2.1 Inter- and Intrabag Hierarchical Sampling

One basic requirement for intrabag sampling is to preserve bag label for each new bag after the sampling. For a positive bag, preserving correct label for the new bag (after sampling) is difficult because the sampling process cannot guarantee that a sampled bag contains at least one positive instance. In the worst scenario, if all positive instances in the original positive bag are missed during the sampling process, the bag label will conflict to the bag content. To solve the problem, we propose an intrabag sampling method to maximize the possibility of preserving at least one positive instance in each sampled positive bag. Following this intuition, the fundamental challenge is to find probable positive instances in each positive bag. We solve this problem by finding probable positive sample distributions in each bag (Section 2.2), and then use the distributions to guide the intrabag sampling (Section 2.3).

2.2 Probable Positive Sample Distributions

Denoting B_i an MI bag, the kth instance in B_i is denoted by $b_{i,k}$, and B_i^+ and B_j^- each represents the ith positive bag and the jth negative bag, respectively. Given a training set T constituting of P positive bags and N negative bags $T = \{B_1^+, \ldots, B_P^+, B_1^-, \ldots, B_N^-\}$, for all negative bags, the probability of an instance x in a positive bag being positive can be viewed as $\mathrm{P}(x \mid B_1^-, \ldots, B_N^-)$ (or a shorthand notation $\mathrm{P}(x \mid B^-)$). Because each positive bag contains at least one positive sample, the summation of the above probability over all instances in B_i should be greater or equal to 1. Accordingly, we introduce probable positive sample distributions for all instances in a positive bag B_i as given by

$$\begin{aligned} \mathbf{Pr}(B_i \mid B^-) &= [\Pr(b_{i,1} \mid B^-), \ldots, \Pr(b_{i,|B_i|} \mid B^-)], \\ \Pr(b_{i,k} \mid B^-) &= \frac{\mathrm{P}(b_{i,k} \mid B^-)}{\sum_{l=1, b_{i,l} \in B_i}^{|B_i|} \mathrm{P}(b_{i,l} \mid B^-)}. \end{aligned} \tag{1}$$

According to Bayes' rule, the probability $\mathrm{P}(x \mid B_1^-, \ldots, B_N^-)$ can be written as

$$\mathrm{P}(x \mid B_1^-, \ldots, B_N^-) = \frac{\mathrm{P}(B_1^-, \ldots, B_N^- \mid x)\mathrm{P}(x)}{\mathrm{P}(B_1^-, \ldots, B_N^-)}. \tag{2}$$

Without loss of generality, we can regard $\mathrm{P}(x)$ as a uniform prior (i.e., a constant), which is a common assumption in most existing work [8]. $\mathrm{P}(B_1^-, \ldots, B_N^-)$ is the probability of observing all N negative bags, which is calculated as (3) under assumption that negative bags are independent and identically distributed

$$\mathrm{P}(B_1^-, \ldots, B_N^-) = \mathrm{P}(B_1^-)\mathrm{P}(B_2^-) \ldots \mathrm{P}(B_N^-). \tag{3}$$

Rearrange (2) using (3) with assumption that negative bags $B_1^-, \ldots, B_N^-$ are conditionally independent, given instance x, we have

$$\mathrm{P}(x \mid B_1^-, \ldots, B_N^-) = \frac{\mathrm{P}(B_1^- \mid x) \ldots \mathrm{P}(B_N^- \mid x)\mathrm{P}(x)}{\mathrm{P}(B_1^-)\mathrm{P}(B_2^-) \ldots \mathrm{P}(B_N^-)} \tag{4}$$

$$\mathrm{P}(B_j^- \mid x) = \mathrm{P}(b_{j,1}^- \mid x)\mathrm{P}(b_{j,2}^- \mid x) \ldots \mathrm{P}(b_{j,|B_j|}^- \mid x). \tag{5}$$

According to Bayes' rule, we have

$$\mathrm{P}(B_j^- \mid x) = \mathrm{P}(x \mid B_j^-)\mathrm{P}(B_j^-)/\mathrm{P}(x). \tag{6}$$

Then, (4) becomes

$$\mathrm{P}(x \mid B_1^-, \ldots, B_N^-) = \frac{\prod_{j=1}^N \mathrm{P}(x \mid B_j^-)}{(\mathrm{P}(x))^{|B_1^-|+\cdots+|B_N^-|-1}}. \tag{7}$$

Under the uniform prior assumption for $\mathrm{P}(x)$ and the formula given in (1), the probable positive sample distribution for instance $b_{i,k}$ in B_i can be calculated as follows:

$$\Pr(b_{i,k} \mid B^-) = \frac{\prod_{j=1}^N \mathrm{P}(b_{i,k} \mid B_j^-)}{\sum_{l=1, b_{i,l} \in B_i}^{|B_i|} \prod_{j=1}^N \mathrm{P}(b_{i,l} \mid B_j^-)}. \tag{8}$$

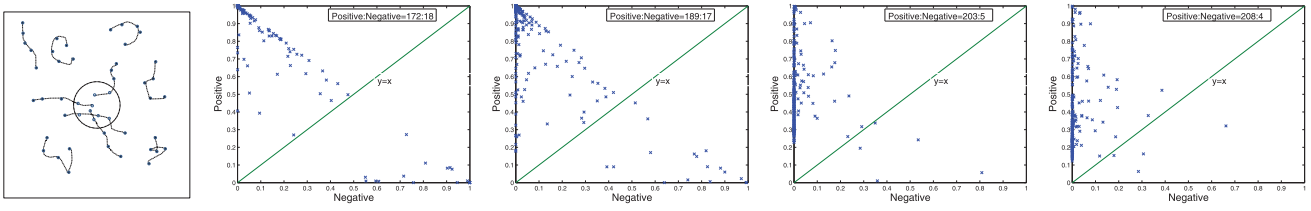
In (8), $\mathrm{P}(b_{i,k} \mid B_j^-)$ defines the conditional probability of instance $b_{i,k}$ given negative bag B_j^-. In [3], [15], the authors proposed a Gaussian-like distribution based *most-likely-cause* estimator to estimate $P(x \mid B_i)$ by looking only at the instance in the bag B_i which is mostly likely relevant to x. We employ the similar estimator in (9) so the conditional probability $\mathrm{P}(b_{i,k} \mid B_j^-)$ is determined by the distance between $b_{i,k}$ and its most similar peers in the negative bag B_j^-.

To calculate the distance between two instances, we use simple euclidean distance as defined in (10), where $b_{j,k,f}$ is the fth feature value of instance $b_{j,k}$:

$$\Pr(b_{i,k} \mid B_j^-) \propto 1 - \max_{\tau, b_{j,\tau}^- \in B_j^-} \exp\left(-\frac{\left\|b_{i,k} - b_{j,\tau}^-\right\|^2}{\sigma^2}\right) \tag{9}$$

$$\left\|b_{i,k} - b_{j,\tau}^-\right\|^2 = \sum_f \left(b_{i,k,f} - b_{j,\tau,f}^-\right)^2. \tag{10}$$

According to the distribution given in (8), one possible way for intrabag sampling is to select the most probable positive sample, using the maximum a posteriori in (11) and include the selected sample into each sampled bags. There are two disadvantages with this approach: 1) the estimated probability value in (8) might be inaccurate, and 2) including the same sample into all bags reduces the diversity. In the next section, we combine rejection sampling and random sampling, by using probable sample distributions, for intrabag sampling:



(a) Dataset (b) low density (5 inst./bag) (c) low density (10 inst./bag) (d) high density (5 inst./bag) (e) high density (10 inst./bag)

Fig. 2. Validation of the probable positive sample distribution on synthetic multi-instance data sets. Each point in (b)-(e) denotes the largest distribution value using (9) of the negative instance (x-axis) and positive instance (y-axis) in each positive bag, respectively. Each synthetic data set contains about 200 positive and 200 negative bags. Density specifies the percentage of positive instances in a positive bag. (a) A conceptual view of the synthetic data set; (b) low density (23.47 percent positive instances in each positive bag) with five instances/bag; (c) low density (19.66 percent) with 10 instances/bag; (d) high density (83.65 percent) with five instances/bag; (e) high density (72.64 percent) with 10 instances/bag.

$$x^*_{B_i} = \arg\max_{x \in B_i} \Pr\left(x \mid B_1^-, \ldots, B_N^-\right). \tag{11}$$

In (9), the estimation of the most probable positive samples is calculated without taking positive bags into consideration. This is mainly because genuine labels of the instances in positive bags are unknown. In our study, we have tried different metrics considering positive bags, for example, distance to positive bags. The results, however, do not show significant improvement, compared to the proposed method that only focuses on negative bags.

2.2.1 Validation on Synthetic Set

In Fig. 2, we report the probabilities calculated by (9) on synthetic multi-instance data sets as shown in Fig. 2a. In the synthetic data set, each instance represents a point in a two-dimensional space. An instance is positive if its euclidian distance to the origin, which is the center of the square, is less than a predefined value r (i.e., a small circle region). To generate each bag, we randomly select one starting point and follow random walk (along x- and y-axis using a random step size $[-1, 1]$) to the next point, and so on. By varying the radius size r, we can control the density of positive instances in positive bags. Figs. 2b, 2c, 2d, and 2e report two sets of results with sparse versus dense densities.

Each point in Figs. 2b, 2c, 2d, and 2e represents one positive bag. The x-axis and the y-axis denote the largest distribution value (9) of the negative instance and positive instance in each bag, respectively. A point above the $y = x$ line indicates that, for this specific bag, (9) indeed capture the positive instance. The number of points below the $y = x$ line corresponds to the number of positive bags to be incorrectly classified as negative bags.

The results in Fig. 2 show that (8) can indeed help capture majority positive instances in each positive bag. In sparse density scenario, each positive bag only contains one (Fig. 2b) or two (Fig. 2c) positive instances on average. Our method can accurately identify 90.5 and 91.7 percent of positive bags, and the results for dense bags are actually much better.

In addition to the above results, we also check the ranking loss of all positive bags as follows: For each positive bag, we sort all instances according to their distribution values in a descending order. We check the number of negative instances that are misplaced in each bag (a negative instance is misplaced if it has a higher probability value than any positive instance in the bag), and divide the total number of misplaced instances by the total number of instances in all positive bags. The ranking loss is 5.37, 3.73, 2.31, and 2.59 percent for four data sets corresponding to Figs. 2b, 2c, 2d, and 2e, respectively. This further demonstrate that our method is effective to capture positive instances in positive bags.

2.3 Rejection and Random Intrabag Sampling

Given a positive bag B_i with m_i instances, intrabag sampling aims to generate J bags, under conditions that 1) each sampled bag has the same number of instances as B_i; 2) a sampled bag preserves the same bag label as B_i with maximum possibility; and 3) sampled bags have the maximum diversity. To achieve the goal, we combine the strength of rejection sampling [9] and random sampling to first select a probable positive sample from B_i (using rejection sampling), and then apply random sampling to select remaining samples.

Rejection sampling is a statistical solution to independently select samples from a given distribution. To employ rejection sampling to select a probable positive sample, we randomly select a number from 1 to $|B_i|$, say j, and then reject instance $b_{i,j}$ with probability $1 - P(b_{i,j} \mid B^-)$. In other words, a sample with a larger probable positive distribution value will have a better chance of being selected in a sampled bag. After that, we employ pure random sampling to select $m_i - 1$ samples from B_i to form a sampled bag $B_i^{'}$. Due to page limitations, we omit the details on rejection sampling, interested readers can refer to external sources [9] for theoretical aspects of the rejection sampling.

2.4 Computational Complexity

Given a training set with M bags and each bag containing m instances on average, assume that an MIL algorithm scales linearly to the number of bags and quadratic to the number of instances in each bag $O(Mm^2)$ (which is a reasonable assumption for most MIL algorithms). For interbag sampling, HSMILE requires $O(I)$ time complexity. For each interbag sampling set T_i, it takes $O(Mm^2)$ to calculate the probable positive sample distribution (for positive bags) and intrabag sampling (including rejection sampling) costs $O(JMm)$. In addition, it takes $O(JMm^2)$ to train J MI learners.

In summary, the time complexity of HSMILE is $O(I[Mm^2 + JMm + JMm^2]) = O(IJMm^2)$, which is asymptotically $I \times J$ times more expensive than a single MI leaner. This is equivalent to training $I \times J$ MI learners from the original training set.

3 HSMILE RATIONALE

3.1 Diversity Enhancement

Existing study [6] has shown that high diversity of base classifiers is essential to ensure the performance gain of an ensemble predictor. In this section, we prove that bootstrap sample sets of HSMILE have a higher diversity than the ones from TMIE.

Definition 1. *Denote $\overline{P}_{\{x \in T\}}(T_i)$ the average probability of a sample in T appearing in a sample set T_i, the Diversity of T_i, denoted by $\mathcal{D}(T_i)$ is $1 - \overline{P}_{\{x \in T\}}(T_i)$.*

Lemma 1. *Given an MI training set T, the diversity of a bootstrap sample set T_i from HSMILE is greater than the diversity of the same size sample set from traditional MI ensemble but less than the diversity of the same size sample set from pure instance-level bootstrap sampling.*

Proof. Given an MI training set T with M bags and r instances ($r \gg M$), without loss generality, let us assume that all bags in

T have the same size m. For pure instance-level bootstrap sampling, the probability of an instance in T to appear in the same size bootstrap sample set T_{pure}, $\overline{P}_{\{x\in T\}}(T_{pure})$ is $1-(1-\frac{1}{r})^r$ and $\mathcal{D}(T_{pure})=1-(1-(1-\frac{1}{r})^r)$. For interbag sampling, the probability of one instance is chosen is equal to the probability of the bag containing the instance being chosen, which is $\frac{1}{M}$. Because we have to repeat interbag sampling M times to select M bags from T to build T_i, the probability of an instance in T to appear in T_i, $\overline{P}_{\{x\in T\}}(T_i)$ is $1-(1-\frac{1}{M})^M$ and $\mathcal{D}(T_i)=1-(1-(1-\frac{1}{M})^M)$.

Intrabag sampling is built upon the interbag sampling. Supposing there are m instances in a bag, the probability of one instance in T to appear in T_i^j, $\overline{P}_{\{x\in T\}}(T_i^j)$, after both interbag sampling and intrabag sampling, is $(1-(1-\frac{1}{M})^M)(1-(1-\frac{1}{m})^m)$ and $\mathcal{D}(T_i^j)=1-((1-(1-\frac{1}{M})^M)(1-(1-\frac{1}{m})^m))$. Because $(1-(1-\frac{1}{m})^m)\leq 1$, the diversity of T_i^j is greater than the diversity of T_i, i.e., $\mathcal{D}(T_{pure})\geq\mathcal{D}(T_i^j)\geq\mathcal{D}(T_i)$. □

For TMIE and HSMILE, each of the base learners is trained from T_i and T_i^j, respectively. When using the same type of learning algorithm to train base learners, HSMILE has a higher diversity than TMIE.

3.2 Variance Reduction

It has been shown in theory that the error rate reduction for an ensemble predictor mainly attributes to the reduction of the variance of the base learner's error rate [2], [10]. In this section, we study HSMILE and TMIE and conclude that HSMILE achieves better variance reduction than its peers.

According to Tumer [10], the classification error rate is linearly proportional to the boundary errors (i.e., the errors corresponding to the difference between the actual decision boundary and the Bayes decision boundary), so our study will, therefore, focus on the classifier boundary error. Denote c_i the label of the ith class and $\mathrm{P}(c_i \mid x)$ is the posteriori probability of c_i given instance x (we use $\mathrm{P}_i(x)$ as a shorthand of $\mathrm{P}(c_i \mid x)$). For a two-class problem, Bayes optimal decision boundary is the loci of all point x^* where $\mathrm{P}_n(x^*)=\mathrm{P}_p(x^*)$ [10], n and p denote the label of negative and positive class, respectively. Due to factors, such as data errors and limitations of the classification algorithm, the actual decision boundary may vary from the Bayes optimal boundary. Denote $d,(d=x_d-x^*)$ the bias of the actual decision boundary varying from the Bayes optimal boundary and $F_i(.)$ the output of the actual classifier, the actual decision boundary is the loci of all x_d, where

$$F_n(x^*+d)=F_p(x^*+d). \tag{12}$$

According to Tumer [10], the output of a classifier *w.r.t.* an instance x can be expressed as

$$F_i(x)=\mathrm{P}(c_i \mid x)+\varepsilon_i(x)=\mathrm{P}_i(x)+\eta_i(x)+\beta_i, \tag{13}$$

where $\varepsilon_i(x)$ is the error associated with the class c_i given sample x, which can be decomposed into two components: $\eta_i(x)$ the noise of the classifier in predicting sample x, and β_i, the bias of the underlying predictor.

3.2.1 Single MI Predictor Variance

According to (12), the actual decision boundary of a single MI predictor consists of points $x_d, x_d=x^*+d$, where $F_n(x^*+d)=F_p(x^*+d)$. Following decomposition in (13), we have

$$\mathrm{P}_n(x^*+d)+\varepsilon_n(x_d)=\mathrm{P}_p(x^*+d)+\varepsilon_p(x_d). \tag{14}$$

A linear approximation of $\mathrm{P}_i(x)$ around x^* can be expressed as (15) under assumption that the posteriors are locally monotonic function [10],

$$\mathrm{P}_i(x^*+d)\cong\mathrm{P}_i(x^*)+d\mathrm{P}_i^{'}(x^*), i=n,p, \tag{15}$$

where $\mathrm{P}_i^{'}(.)$ denotes the derivation of $\mathrm{P}_k(.)$. Since x^* is a point on the Bayes optimal boundary where $\mathrm{P}_n(x^*)=\mathrm{P}_p(x^*)$, the bias d can be calculated in (16), with $s=\mathrm{P}_n^{'}(x^*)-\mathrm{P}_p^{'}(x^*)$:

$$d=\frac{\varepsilon_p(x_d)-\varepsilon_n(x_d)}{s}=\frac{\eta_p(x_d)-\eta_n(x_d)}{s}+\frac{\beta_p-\beta_n}{s}. \tag{16}$$

Because learner bias β_i is a constant with respect to the underlying data set and learning algorithms, assume that noise $\eta_i(x)$ is independent and variance $\sigma_{\eta_i}^2$ follows Gaussian distributions, the variance of the SMI is, therefore, denoted by

$$\sigma_d^2=\frac{\sigma_{\eta_n}^2+\sigma_{\eta_p}^2}{s^2}. \tag{17}$$

3.2.2 Traditional MI Ensemble Variance

For TMIE, the final prediction of the ensemble predictor is the average probabilities of the I base classifiers, as shown in

$$F_i^{TMIE}(x)=\frac{1}{I}\Sigma_{t=1}^{I}F_i^t(x), \tag{18}$$

where $F_i^t(x)$ is the output of each base classifier. Using bias and variance decomposition in (13), we have

$$F_i^{TMIE}(x)=\mathrm{P}_i(x)+\frac{1}{I}\Sigma_{t=1}^{I}\left(\eta_i^t(x)+\beta_i^t\right). \tag{19}$$

According to (12), we have

$$F_p^{TMIE}(x^*+d^{TMIE})=F_n^{TMIE}(x^*+d^{TMIE}). \tag{20}$$

Similar to SMI, the offset of the TMIE boundary from the Bayes optimal boundary can be calculated as

$$d^{TMIE}=\frac{\eta_p^{TMIE}(x_d)-\eta_n^{TMIE}(x_d)}{s}+\frac{\beta_p^{TMIE}-\beta_n^{TMIE}}{s}. \tag{21}$$

Because noise of $\eta_i(x)$ are assumed to be independent of each other and learner bias β_i is a constant with respect to the underlying data set and learning algorithms, the variance of a TMIE predictor is given by

$$\sigma_{d^{TMIE}}^2=\frac{\sigma_{\eta_p^{TMIE}}^2+\sigma_{\eta_n^{TMIE}}^2}{s^2}=\frac{1}{I}\frac{\sigma_{\eta_n}^2+\sigma_{\eta_p}^2}{s^2}=\frac{1}{I}\sigma_d^2. \tag{22}$$

3.2.3 Hierarchical Sampling MI Ensemble Variance

For HSMILE, suppose we apply inter- and intrabag sampling I and J times, respectively, to each T_i. F_i^{HSMILE} denotes the output of HSMILE given instance x, where

$$F_i^{HSMILE}(x)=\frac{1}{I*J}\sum_{\tau=1}^{I}\sum_{j=1}^{J}F_i^{t,j}(x). \tag{23}$$

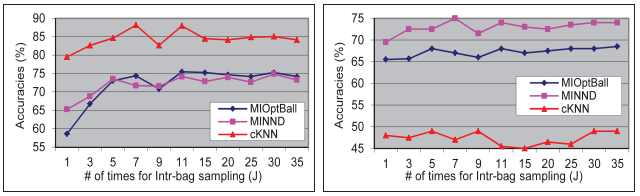
Following the above induction process for TMIE, the variance of HSMILE can be denoted by

$$\sigma_{d^{HSMILE}}^2=\frac{1}{I*J}\sigma_d^2. \tag{24}$$

Because $I\geq 1$ and $J\geq 1$, we have $\sigma_{d^{HSMILE}}^2\leq\sigma_{d^{TMIE}}^2\leq\sigma_d^2$, which means that HSMILE receives higher variance reductions than TMIE, and both of them have higher variance reduction than a single MI predictor.

3.2.4 Diversity, Variance, and Independence Assumption

When ensemble members are dependent on each other, the diversity enhancement of HSMILE is still valid, whereas the variance reduction of TMIE and HSMILE, compared to SMI, will



(a) Musk$_1$ (b) Tiger

Fig. 3. Prediction accuracies (y-axis) with respect to different number of times for intra-bag sampling (x-axis).

deteriorate. Assume that ensemble members are α-dependent on each other, the variance of a member τ is given as $\sigma^2_{\eta_i^\tau} = \alpha_\tau \sigma^2_{\eta_i}$, where α_τ is a random value in the range [0, α]. The variances of TMIE and HSMILE are

$$\sigma^2_{d^{TMIE}} = \frac{\sum_{\tau=1}^{I} \alpha_\tau}{I} \sigma^2_d; \quad \sigma^2_{d^{HSMILE}} = \frac{\sum_{\tau}^{I} \sum_{j}^{J} \alpha_{\tau,j}}{I * J} \sigma^2_d. \tag{25}$$

If ensemble members are identical (i.e., $\alpha_1 = \alpha_2 = \cdots = \alpha$), the variance of TMIE and HSMILE is equal to $\alpha \times \sigma^2_d$. If $\alpha = 1$, both TMIE and HSMILE degenerate as a single SMI learner.

4 Experiments

We implement the proposed sampling method using Java and WEKA machine learning tool [11]. Two baseline methods, including SMI, TMIE, and two HSMILE variants denoted by HSMILE$_t$ and HSMILE$_n$, are implemented for comparison purposes. HSMILE$_t$ and HSMILE$_n$ are almost identical to HSMILE except their instance-level sampling module. For HSMILE$_t$ (t means "total"), instances are randomly sampled inside each bag. For HSMILE$_n$ (n means "negative"), instances in a negative bag are randomly sampled, and no sampling for a positive bag. Five real-world MI data sets are collected as our benchmark testbed.[1]

We use 10-fold cross validations and select three MI algorithms, Citation KNN (cKNN), MI Optimal Ball (MIOptimalBall), and MI Nearest Neighbor with Distribution Learner (MINND), with default parameter settings as the base learners. For fair comparisons, in each fold, the training and test data sets remain the same for all methods. In our experiments, the number of interbag sampling for HSMILE is set to 10.

4.1 Intrabag Sampling Results

To study the impact of the intrasampling times (J) on the algorithm performance, we vary the J values, from 1 to 35, and report the performance of HSMILE in Fig. 3. Because the J value determines the ensemble size for each individual bag, increasing sampling times for each bag can bring positive impact on the algorithm performance. Such improvement, in practice, can be observed across all MI learners. On the other hand, because each bag is reproduced J times through intra-bag sampling, increasing J value directly increases the training set size and requires significant extra computational costs. According to the impact of the sampling times J with respect to the algorithm performance, as reported in Fig. 3, and the computational cost concerns, we use $J = 3$ for all experiments in the remaining sections.

4.2 Accuracy Comparisons and Analysis

Table 1 reports results across different learning methods and different benchmark data sets. For each row (i.e., one learning algorithm *w.r.t.* one benchmark data set), the method with the highest mean accuracy is bold faced. A † indicates that HSMILE is statistically significantly better, t-test at 95 percent level, than TMIE for the particular data set and MI learner.

1. The source code and data sets can be downloaded from http://www.cse.fau.edu/~xqzhu/hsmile.html.

TABLE 1
Classification Accuracy Comparisions

dataset	MI Method	SMI	TMIE	HSMILE$_t$	HSMILE$_n$	HSMILE
Musk$_1$	MIOptBall	66.16	66.16	72.82	71.31	**73.53** †
	MINND	71.51	68.18	70.20	65.05	**75.95** †
	cKNN	82.42	**84.64**	83.53	83.53	83.53
Musk$_2$	MIOptBall	**79.99**	67.66	69.33	76.16	78.00
	MINND	61.66	65.83	63.66	64.83	**68.50** †
	cKNN	73.50	73.66	71.50	74.50	**75.33** †
Tiger	MIOptBall	64.99	65.50	65.50	**69.50**	67.00
	MINND	70.49	74.50	72.50	**76.49**	73.50
	cKNN	42.00	45.00	43.00	45.00	**47.00** †
Fox	MIOptBall	59.00	58.50	58.50	57.50	**59.50** †
	MINND	51.50	53.50	55.50	53.50	**57.50** †
	cKNN	39.00	40.00	42.00	40.00	**43.00** †
Elephant	MIOptBall	63.50	64.50	58.50	61.00	**65.50** †
	MINND	67.50	77.00	83.00	**84.50**	84.00
	cKNN	42.00	43.00	40.00	**45.00**	**45.00**
Average Accuracy		62.55	63.28	63.30	64.39	**66.36**

A † indicates that HSMILE is statistically significantly better, t-test at 95 percent confidence level, than TMIE.

For all five methods, ensemble learning indeed achieves performance gain for single multi-instance learner. For some data sets we observed, MI ensemble learning can receive more than 10 percent performance gain, in comparison with a single MI learner. For example, the accuracy of SMI on the "Elephant" data set is 67.5 percent, and the absolute ensembling accuracy gains for TMIE, HSMILE$_t$, HSMILE$_n$, and HSMILE are 9.5, 15.5, 17.0, and 16.5 percent, respectively. On the other hand, although SMI occasionally outperforms ensemble predictors, it actually has the lowest average accuracy across all methods, which asserts that similar to generic supervised learning, simple ensemble learning is an effective way for boosting multi-instance learners.

Among three base learners, both cKNN and MINND are KNN related. Existing research has concluded that kNN is a stable learner and is not an ideal candidate for ensemble learning [2]. Our results indicate that there is no clear evidence to support this hypothesis, and both MIOptimalBall and KNN related MI ensemble learners can receive good performance gain for multi-instance learning. This may be because that the bag labeling constraint alleviates the stable learning condition and makes a typical stable learner relatively unstable.

Among all ensembling approaches, HSMILE has won TMIE, HSMILE$_t$, and HSMILE$_n$ 13, 14, and 10 times, respectively, on the total 15 tests. Among all 15 tests, HSMILE is statistically significantly better than traditional multi-instance learning (TMIE) on nine tests. Recall that HSMILE employs inter- and intrabag sampling to 1) retain the most probably positive sample in each positive bag, and 2) generate bootstrap MI sample sets with maximum diversities; it is clear that generic sampling approaches that do not take bag labeling constraints into consideration for ensemble learning is inferior to the HSMILE for multi-instance ensemble learning.

5 Conclusions

In this paper, we proposed a new hierarchical sampling method for multi-instance ensemble learning (HSMILE). We argued that due to the unique multi-instance bag labeling constraint, traditional MI ensemble approaches, which carry out sampling at interbag level, are insufficient to generate diverse MI bootstrap sets. Pure instance-level bootstrap sampling without considering the bag labeling constraint, on the other hand, may result in inconsistent positive bags and deteriorate the learning performance. To solve the problem (i.e., increasing diversity and maintaining bag label consistency), HSMILE employs hierarchical

sampling, at both bag and instance levels, to ensure that each bag is sufficiently randomly perturbed but still complies with the bag labeling constraints. Theoretical studies analyzed the rationality of the HSMILE from both diversity enhancement and variance reduction perspectives. Experimental comparisons using three MI learning methods and five benchmark data sets confirmed that HSMILE outperforms its peers for multi-instance ensemble learning.

ACKNOWLEDGMENTS

This work was supported by Australian Research Council Future Fellowship (FT100100971) and National Science Foundation of China (NSFC 71201120).

REFERENCES

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support Vector Machines for Multiple-Instance Learning," *Proc. Advances in Neural Information Processing Systems (NIPS),* vol. 15, pp. 561-568, 2003.
[2] L. Breiman, "Bagging Predictors," *Machine Learning,* vol. 24, no. 2, pp. 123-140, 1996.
[3] Y. Chen, J. Bi, and J. Wang, "Miles: Multiple-Instance Learning via Embedded Instance Selection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 12, pp. 1931-1947, Dec. 2006.
[4] T. Dietterich, R. Lathrop, and T. Lozano-Pérez, "Solving the Multiple-Instance Problem with Axis-Parallel Rectangles," *Artificial Intelligence,* vol. 99, no. 7, pp. 31-71, 1997.
[5] J. Han and M. Kamber, *Data Mining: Concepts and Techniques.* Morgan Kaufmann, 2006.
[6] L. Kuncheva and C. Whitaker, "Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy," *Machine Learning,* vol. 51, no. 2, pp. 181-207, 2003.
[7] Y. Li, S. Ji, S. Kumar, J. Ye, and Z. Zhou, "Drosophila Gene Expression Pattern Annotation through Multi-Instance Multi-Label Learning," *IEEE/ACM Trans. Computational Biology and Bioinformatics,* vol. 9, no. 1, pp. 98-112, Jan./Feb. 2012.
[8] O. Maron and T. Lozano-Pérez, "A Framework For Multiple-Instance Learning," *Proc. Neural Information Processing Systems (NIPS),* vol. 10, pp. 570-576, 1998.
[9] C. Robert and G. Casella, *Monte Carlo Statistical Methods.* Springer-Verlag, 2004.
[10] K. Tumer and J. Ghosh, "Analysis of Decision Boundaries in Linearly Combined Neural Classifiers," *Pattern Recognition,* vol. 29, no. 2, pp. 341-348, 1996.
[11] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, 2005.
[12] Z. Zhou and M. Zhang, "Ensembles of Multi-Instance Learners," *Proc. 14th European Conf. Machine Learning,* pp. 492-502, 2003.
[13] X. Zhu and X. Wu, "Class Noise vs. Attribute Noise: A Quantitative Study of Their Impacts," *Artificial Intelligence Rev.,* vol. 21, no. 3, pp. 177-210, 2004.
[14] W. Li and D. Yeung, "MILD: Multiple-Instance Learning via Disambiguation," *IEEE Trans. Knowledge and Data Eng.,* vol. 22, no. 1, pp. 76-89, Jan. 2010.
[15] B. Babenko, M. Yang, and S. Belongie, "Robust Object Tracking with Online Multiple Instance Learning," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 33, no. 8, pp. 1619-1632, Aug. 2011.
[16] D. Nguyen, C. Nguyen, R. Hargraves, L. Kurgan, and K. Cios, "mi-DS: Multiple-Instance Learning Algorithm," *IEEE Trans. Cybernetics,* vol. 43, no. 1, pp. 143-154, Feb. 2013.