Adaptive p-Posterior Mixture-Model Kernels for Multiple Instance Learning

Hua-Yan Wang WANGHY@CIS.PKU.EDU.CN
Key Laboratory of Machine Perception (Ministry of Education), Peking University

Qiang Yang QYANG@CSE.UST.HK
Department of Computer Science and Engineering, Hong Kong University of Science and Technology

Hongbin Zha ZHA@CIS.PKU.EDU.CN
Key Laboratory of Machine Perception (Ministry of Education), Peking University

Abstract

In multiple instance learning (MIL), how the instances determine the bag-labels is an essential issue, both algorithmically and intrinsically. In this paper, we show that the *mechanism* of how the instances determine the bag-labels is different for different application domains, and does not necessarily obey the traditional assumptions of MIL. We therefore propose an *adaptive* framework for MIL that adapts to different application domains by learning the domain-specific *mechanisms* merely from labeled bags. Our approach is especially attractive when we are encountered with novel application domains, for which the *mechanisms* may be different and unknown. Specifically, we exploit mixture models to represent the composition of each bag and an adaptable kernel function to represent the relationship between the bags. We validate on synthetic MIL datasets that the kernel function automatically adapts to different *mechanisms* of how the instances determine the bag-labels. We also compare our approach with state-of-the-art MIL techniques on real-world benchmark datasets.

1. Introduction

Multiple instance learning (MIL) has become an active area of investigation in machine learning since it was first put forward for drug activity prediction(Dietterich, 1997). In MIL, we consider "instance-bags", which are unordered sets of instances. Each instance is represented as a feature vector. According to the original definition, a bag of instances is labeled as positive if at least one of its instances is positive, and it is labeled as negative if all of its instances are negative. In real-world applications of MIL, the focus is on assigning labels to bags rather than instances. Many methods have been proposed to solve the MIL problem, including Axis-Parallel Rectangles (Dietterich, 1997), Diverse Density (Maron, 1998), EM-DD (Zhang, 2001), Citation k-NN (Wang, J., 2000), and variations of SVM (Andrews, 2003; Gartner, 2002; Kwok, 2007; Bunescu, 2007).

A major difficulty of MIL arises from the ambiguity caused by not knowing which instances determined the bag labels. According to the original definition of MIL (Dietterich, 1997), a bag can be labeled as positive based on just one positive instance in it. However, since the instance-labels are unknown in the outset, we need to leverage the available information conveyed by all instances to determine the label of a bag. This motivates us to carefully examine the underlying *mechanism* of how the bag labels are determined by the instances within the bag.

Firstly, examining the algorithmic aspect of the *mechanism* we could conclude that, even if there is an unambiguous intrinsic *mechanism* (*e.g.*, a bag is positive *iff* at least one instance is positive), it can hardly benefit a MIL algorithm deterministically due to the unknown instance labels. Instead, possible instance labels are usually leveraged in a probabilistic manner. For example, (Zhang, 2001) computes posteriors of instance labels in an EM-like algorithm; (Kwok, 2007) marginalizes a kernel function over possible instance

Appearing in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008.

labels. In other words, virtually *all* instances in a bag can contribute to a bag label.

Our second observation is that the intrinsic *mechanisms* of how the instances determine the bag-labels can vary in different application domains of MIL; these *mechanisms* do not necessarily obey the original definition of MIL. Instead, they must be relaxed to allow more flexibility. Recall that in the original definition of MIL, a bag is positive *iff* at least one instance is positive. This clearly defines MIL problems in some applications such as drug activity prediction, because the "positive" instances in these applications could serve as *strong* or even *definite* evidence for labeling a bag as positive. For example, if a molecule binds well to some target protein (positive instance), the molecule undoubtedly binds well and the associated bag is labeled positive. However, in other applications, this restriction is too limiting. In many real world applications, the bag-label determining *mechanism* can allow a bag label to be negative even when there exists a positive instance in it; such relationship between instances and bags should be probabilistic in nature. For example, in content-based image retrieval (Zhang, 2002), images are represented as "bags" of localized features (regions). The low-level instance representation, describing color, texture, and shape, may have no direct correspondence to high-level image-labels (*e.g.*, human faces, buildings, the sky *etc.*). Instead, they only serve as *weak* evidences that should be integrated together to determine the image-label. Intuitively, the localized image feature descriptors can be "a region appears like a human eye or a region appears like a human nose". The decision to label an image as a "human face" should leverage many such pieces of evidence, because a non-face images (negative bags) can also contain some other object that appears like a human nose (positive instance). In this example, a positive instance can be found in a negative bag, which violates the traditional definition of MIL. Therefore, our solution to the MIL problem should be flexible enough to allow for different bag-label determining *mechanisms*.

The *mechanisms* of how the instances determine the bag labels is an essential issue in MIL. However, to our best knowledge, none of existing MIL techniques has explicitly addressed the issue of the cross-domain differences of this *mechanism*. In this paper, we propose a new framework for MIL that includes the original definition of MIL as a special case, and yet allows for more flexible cases. Our solution is to automatically *adapt* the instance-to-bag-label *mechanism* to accommodate the differences in various formulations of the MIL problems. Our main contribution is to capture the *mechanism* by a simple model, embodied in a parameter p of a kernel function (Schölkopf, 2001) defined over the bags. This parameter is learned from labeled bags in the training data without *a priori* knowledge of that *mechanism*.

Our *adaptive* framework for MIL is supported by a number of motivations. First, explicitly describing the *mechanism* (as (Dietterich, 1997) did for drug activity prediction) for an application domain calls for strong domain knowledge. Second, a hand-crafted *mechanism* could be subjective and unreliable. Third, designing different MIL methods for different application domains is inefficient, given the large number of applications that has a potential to be formalized as MIL. Thus it is better to design an *adaptive* formalism for this task.

In our framework, a two-phase learning procedure is adopted to characterize a kernel function on the bags, which can be used as a distance function in classification via algorithms such as SVM, or as a similarity measure for information retrieval.

The first learning phase exploits the unlabeled instances with a mixture model to characterize the intrinsic structures of the feature space of instances. Each bag is represented by some *aggregate posteriors* on a mixture of components, which summarizes the bag as compositions of different "patterns".

While the first learning phase adapts to different characteristics of the instance space, the adaptive nature of our approach is shown mostly in the second learning phase. We define a kernel mapping by computing a power p of the aggregate posteriors. As we will show in the rest of the paper, the parameter p explicitly captures the domain-specific *mechanism* of how the instances determine the bag-labels, where the parameter p is learned by optimizing an objective function defined over the labeled bags. In this way, our framework can adapt MIL algorithms to different instance-to-bag-label *mechanisms* in many application domains, even if we have no *a priori* knowledge about them.

2. The p-Posterior Mixture Model Kernel

2.1. Aggregate posteriors

We use lowercase $\mathbf{x}$ to denote instances, and uppercase $\mathbf{X}$ to denote bags. In MIL, we are provided with a training set $\{(\mathbf{X}_i, y_i)\}_{i=1}^N$ consisting of labeled bags, where $y_i \in \{+1, -1\}$ are labels[1]. Let $\{\mathbf{x}_i\}_{i=1}^n$

[1] Although we address two-class classification in this paper, it is straightforward to generalize our approach to multi-class classification and continuous-output regression.

be all instances available for the learning algorithm, where each instance $\mathbf{x}_i$ resides in a feature space $\mathbb{R}^D$. The training set includes both the instances in labeled bags and possibly a large number of other unlabeled instances, because the unlabeled instances are often much easier to obtain than the labeled bags in many applications[2]. The distribution of instances in the sampling domain could demonstrate sophisticated patterns due to the underlying unknown *generative model* of instances. Previous approaches usually impose over simplified assumptions on the generative model; for example, APR (Dietterich, 1997) assumes that "positive" instances reside in an axis-parallel rectangle, and Diverse Density (Maron, 1998) assumes that the "positive" instances demonstrate a Gaussian-like pattern around some concept point. In our approach, a mixture model approximates the underlying generative model of instances, which is much more flexible and informative. We make no additional constraint on the instances used; the instances are chosen for training as long as they are from the same underlying generative model. Note that this is different from semi-supervised learning (Zhu, 2005), for which we usually require the unlabeled samples to come from the specific classes of labeled samples.

We approximate the underlying generative model of instances by several mixture models in $\mathbb{R}^D$. Fitting the mixture models to all unlabeled instances with a given number of mixture components K results in the optimal parameters and weights $\{(\mathbf{\Lambda}_i, w_i)\}_{i=1}^K$. For Gaussian mixture models (GMM) adopted in our experiments, we have $\{(\mu_i, \mathbf{\Sigma}_i, w_i)\}_{i=1}^K$.

Given the above, the likelihood of an instance $\mathbf{x}$ under the i-th mixture component is denoted as:

$$p_i(\mathbf{x}) := \Pr(x|\mathbf{\Lambda}_i) \tag{1}$$

For a bag of instances $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^M$ and a mixture model $\{(\mathbf{\Lambda}_i, w_i)\}_{i=1}^K$, we define the aggregation posteriors of a bag on the mixture components:

Definition 1 (Aggregate Posteriors) *The **aggregate posteriors** of a bag of instances $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^M$ with respect to the mixture model $\{(\mathbf{\Lambda}_i, w_i)\}_{i=1}^K$ is denoted as:*

$$\psi(\mathbf{X}) := \mathcal{C} \sum_{i=1}^{M} \left(\frac{w_1 p_1(\mathbf{x}_i)}{\sum_{j=1}^K w_j p_j(\mathbf{x}_i)}, \cdots, \frac{w_K p_K(\mathbf{x}_i)}{\sum_{j=1}^K w_j p_j(\mathbf{x}_i)} \right)$$

where $\mathcal{C}$ is a normalizing operator indicating dividing a vector by the sum of all its elements.

[2]For example, we can extract image regions (instances) in thousands of arbitrary unlabelled images collected from the Internet. This is much easier than manually labeling even a small number of these images which are bags.

It is straightforward to validate that $\frac{w_j p_j(\mathbf{x}_i)}{\sum_{j=1}^K w_j p_j(\mathbf{x}_i)}$ is the posterior probability that $\mathbf{x}_i$ is generated from the j-th mixture component. The normalizing operator $\mathcal{C}$ is induced such that the kernel function (defined later) would be unbiased towards sizes of bags; "large" bags and "small" bags are treated equally. The aggregate posteriors summarize frequencies of different "pattern" within the bag, which could be viewed as a "Bayesian" histogram because a frequentist would replace the K-component mixture model with K-means clustering, and replace the posteriors with a deterministic vote. Thus, the aggregate posteriors degenerate to a normalized histogram.

The first learning phase of our framework is itself endowed with much flexibility and can be customized for specific situations. For example, in some applications the number of available unlabeled instances may be small. We therefore have to reduce the degree of freedom in the mixture model accordingly. For example, we could add the restriction that the components of the Gaussian mixture model have diagonal covariance, or even isotropic covariance[3]. When the dimensionality of the instance space is too high to fit a Gaussian mixture model, we can also adopt the frequentist's point of view by representing the training bags as histograms obtained by K-means clustering of the instances.

2.2. The order-p kernel mapping

We have defined a mapping from bags of instances $\mathbf{X}$ to aggregate posteriors $\psi(\mathbf{X}) \in \mathbb{S}^K$, where $\mathbb{S}^K$ is the $(K-1)$-simplex that consists of all positive constant-sum real vectors. The aggregate posteriors summarize frequencies of different patterns[4] within the bag. For example, consider a toy case where $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{X}_3$ are three bags in some MIL problem, and we have:

$$\begin{aligned}
\psi(\mathbf{X}_1) &= (0, \quad 0.3, \quad 0.5, \quad 0.2), \\
\psi(\mathbf{X}_2) &= (0, \quad 0.2, \quad 0.6, \quad 0.2), \\
\psi(\mathbf{X}_3) &= (0.2, \quad 0.1, \quad 0.6, \quad 0.1).
\end{aligned}$$

We will carefully examine this toy case for an intuitive understanding of our approach. Aggregate posteriors of these bags all demonstrate relatively high values on the third mixture component, and low values on others. According to the definition of aggregate posteriors, the bags all have "major patterns" represented by the third mixture component, with a lot

[3]An isotropic covariance matrix is in the form $\lambda\mathbf{I}$.

[4]The "patterns" are represented by the components of the mixture model.

of instances contributing to that pattern, and "minor patterns" represented by other components, with fewer instances contributing to them.

The kernel function for the bags serves as a similarity measure that affects the decisions in label prediction. Therefore how to define the kernel function depends on the intrinsical *mechanism* that the bag-labels are determined. Since the *mechanism* varies in different application domains, the kernel function should vary accordingly. On one hand, in some applications such as drug activity prediction, positive bags are determined by a few (at least one, actually) positive instances serving as *strong* evidences, and there can be many negative instances in positive bags. Hence the "minor patterns" in the aggregate posteriors are endowed with considerable significance, given that the "major patterns" could be dominated by overwhelming negative instances. On the other hand, in other applications such as image classification, the positive bags are determined by integrating a lot of low-level *weak* evidences from instances. Hence we should focus on the "major patterns" in accordance with the voting-like *mechanism*. The "minor patterns", however, should be underrated because they are attributable to random noise and outliers. For example, in an image classification task, the "minor patterns" could be generated by the image background clutter.

For the toy case, the similarity in minor patterns between $\mathbf{X}_1$ and $\mathbf{X}_2$ is greater than that between $\mathbf{X}_2$ and $\mathbf{X}_3$, but the similarity in major patterns between $\mathbf{X}_2$ and $\mathbf{X}_3$ is greater than that between $\mathbf{X}_1$ and $\mathbf{X}_2$. According to our previous analysis, whether $\mathbf{X}_2$ should be considered more similar to $\mathbf{X}_1$ or $\mathbf{X}_3$ depends on whether we should place more emphasis on major or minor patterns; the latter in turn depends on the domain-specific instance-to-bag-label *mechanism*. To endow the kernel function with such flexibility, we define the p-posterior-mixture-model (*ppmm*) kernel:

Definition 2 (p-Posterior-Mixture-Model Kernel) *The p-posterior-mixture-model (ppmm) kernel function on a pair of bags $\mathbf{X}_1$ and $\mathbf{X}_2$ is defined as*

$$\kappa_p(\mathbf{X}_1, \mathbf{X}_2) := < \psi(\mathbf{X}_1)^p, \psi(\mathbf{X}_2)^p >$$

where $p \in (0, \infty)$, and $< \bullet, \bullet >$ denotes the standard inner-product in $\mathbb{R}^K$.

For the toy case, it is easy to validate that:

$$\begin{aligned} \kappa_p(\mathbf{X}_1, \mathbf{X}_2) &> \kappa_p(\mathbf{X}_2, \mathbf{X}_3), \quad \text{if} \quad p < 1; \\ \kappa_p(\mathbf{X}_1, \mathbf{X}_2) &= \kappa_p(\mathbf{X}_2, \mathbf{X}_3), \quad \text{if} \quad p = 1; \\ \kappa_p(\mathbf{X}_1, \mathbf{X}_2) &< \kappa_p(\mathbf{X}_2, \mathbf{X}_3), \quad \text{if} \quad p > 1. \end{aligned}$$

The parameter p tunes the kernel in a way that a larger p makes it put more emphasis on major patterns, and a smaller p draws more attention to the minor patterns. According to our previous analysis, we can predict that a larger p is preferred in applications such as image classification, and a smaller p is preferred in applications such as drug activity prediction. However these judgements are based on the fact that we already have sufficient *a priori* knowledge about these two application domains. If we encounter a novel application domain of MIL, for which we have no *a priori* knowledge, the p-posterior-mixture-model kernel can be adapted to that novel domain by learning the domain-specific instance-to-bag-label *mechanism*. Learning the *mechanism* is implemented by optimizing an objective function of p defined on labeled bags.

Given labeled bags $\{(\mathbf{X}_i, y_i)\}_{i=1}^N$, $y_i \in \{+1, -1\}$, we learn the parameter p via maximizing the *alignment* (Cristianini, 2002) between the p-posterior-mixture-model kernel and the ideal kernel, which measures the kernel's degree of agreement with the bag-labels:

$$\arg\max_p \frac{< \mathbf{K}_p \, , \, \mathbf{y}\mathbf{y}^T >_F}{\sqrt{< \mathbf{K}_p \, , \, \mathbf{K}_p >_F < \mathbf{y}\mathbf{y}^T \, , \, \mathbf{y}\mathbf{y}^T >_F}} \qquad (2)$$

where $< \bullet, \bullet >_F$ denotes the Frobenius inner-product between matrices. $\mathbf{K}_p$ is the p-posterior-mixture-model kernel matrix.

The optimization problem in (2) is easily resolved by exhaustive search within a certain interval of p (*e.g.* $p \in (0, 3]$ in our later experiments). Because the target function is extremely easy to evaluate and empirically quite smooth, and the search space is only one dimensional, even the exhaustive search is fast and scales linearly with respect to the interval considered.

3. Experiments

3.1. Synthetic data

To empirically validate our analysis in previous sections, we simulate three different multiple instance learning datasets endowed with different instance-to-bag-label *mechanisms*.

MIL dataset 1 is synthesized as follows: 1) randomly generate isotropic-covariance Gaussian mixture models in $\mathbb{R}^D$ with K equally weighted components, from which $N \times S$ instances are sampled; 2) one mixture component is randomly chosen, and the instances generated by that component are labeled as positive; 3) all $N \times S$ instances are randomly put into N bags, with S instances in each; 4) each bag is labeled as positive

iff there is at least one positive instance in it.

MIL dataset 2 and 3 are synthesized similarly. But the instances generated by $\frac{K}{5}$ mixture components are labeled as positive in MIL dataset 2, and each bag is labeled as positive *iff* positive instances in the bag exceed 20%. The instances generated by $\frac{K}{2}$ mixture components are labeled as positive in MIL dataset 3, and each bag is labeled as positive *iff* positive instances in the bag exceed 50%.

Although the synthetic datasets are endowed with different instance-to-bag-label *mechanisms*, all other aspects of these datasets are the same, which can *not* be exploited by the algorithm to distinguish these datasets. They all have approximately the same ratio of positive and negative bags if K and S are properly chosen. Although these tasks have different ratios of positive and negative instances, the instance labels are kept from the learning algorithm, which only observe 50%-50% bag-labels. Our approach is expected to discover the *mechanism* difference among these datasets in such a challenging setting. Moreover, in the first learning phase, the number of mixture components is deliberately set to be different from K, in order to simulate the fact that the characteristics of the underlying true generative model are usually unknown.

We repeated this experiment for many different choices of the bag size S, mixture model size K, instance space dimensionality D, and we observed that the optimal p value is almost always the smallest for MIL dataset 1, intermediate for MIL dataset 2, and the largest for MIL dataset 3. In Figure 1 we plotted the kernel alignment as a function of p in a typical run of the experiment with $K = 20$, $D = 5$, $S = 13$, and total number of bags $N = 200$. Note that this specific setting results in approximately the same number of positive bags and negative bags in all datasets.

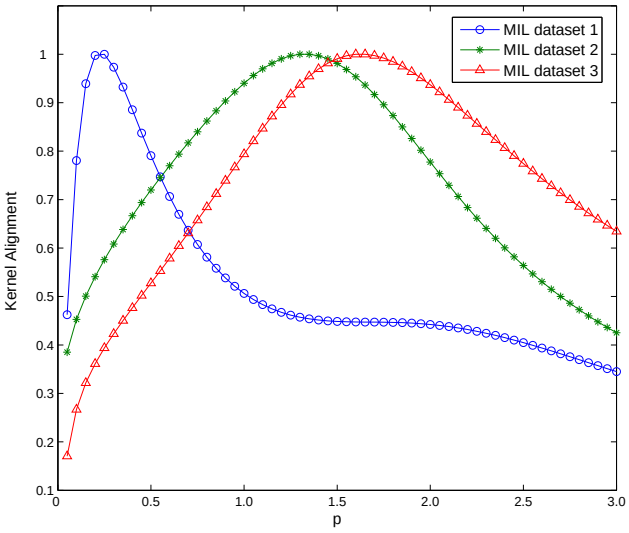

Figure 1. p versus kernel alignment in synthetic MIL dataset 1, 2, and 3. Kernel alignment values are normalized by its maximum value in either dataset. The optimal p values in these datasets justified our previous analysis.

3.2. MIL benchmark datasets

We tested our method on standard MIL benchmark datasets[5] (Andrews, 2003), which consist of MIL tasks in various application domains including drug activity prediction, image classification, and text classification.

3.2.1. Drug activity prediction

The concept of multiple instance learning had been originated from the application of drug activity prediction. In this application, the molecules are regarded as bags, and various shapes a molecule can adopt constitute instances within the bag. A molecule is considered a "positive" bag if it binds well to some target protein, which is true if at least one of its shapes (instances) binds well. The instances are represented as vectors describing that shape. In bio-chemical experiments, we can only observe whether a molecule binds well or not, but if the molecule binds well, we cannot further identify which shape(s) binds well and contributes to the positive bag-label.

Datasets of drug activity prediction for MIL are MUSK1 and MUSK2. The MUSK1 dataset consists of 47 positive bags, 45 negative bags, and totally 476 instances, each represented as a 166 dimensional vector. The MUSK2 dataset consists of 39 positive bags, 63 negative bags, and totally 6598 instances.

3.2.2. Image classification

Content-based image classification/retrieval is another application domain of multi-instance learning. Its major difficulty arises from the fact that an image consists of not only the object-of-interest, which determines its category label, but also background clutter, which may take up even a larger portion of the image. To segment the object-of-interest from background clutter is yet a challenging open problem in computer vision. A common strategy to perform classification without identifying the object-of-interest beforehand is to represent an image by many localized feature vectors instead of a single global feature description. Each localized feature is computed based on a small region of the image,

[5]The datasets used in this section are available online at http://www.cs.columbia.edu/ andrews/mil/datasets.html

so we could expect that the object-of-interest is captured by a number of local features, even if there are also other irrelevant local features arisen from background clutter. It is therefore quite natural to formalized content-based image classification/retrieval as a multi-instance learning problem, where images are the bags and local features are the instances.

The MIL benchmark dataset includes three image classification tasks—to discriminate images that contain *elephant*, *tiger*, and *fox* from irrelevant images, respectively. Each image (bag) is segmented into a set of regions (instances), and each region is represented as a 230 dimensional vector describing its color, texture, and shape characteristics. Each classification tasks has 100 positive bags and 100 negative bags.

3.2.3. Text classification

Another application domain of multiple instance learning is text classification/retrieval. A document can be divided into a number of segments, which could have different topic focuses. And the category label of the whole document (bag) should be decided by taking into account all these segments (instance), which constitutes a multiple instance learning problem.

The MIL benchmark dataset contains text data chosen from the OHSUMED (Hersh, 1994) dataset on medical issues. We perform two text classification tasks: TREC1 and TREC2. Each consists of 200 positive documents (bags) and 200 negative documents. Each document is segmented into overlapping 50-word-passages, which results in over 3000 passages (instances) in either of the tasks. Each passage is indexed by a sparse high-dimensional (over 60,000 index terms) feature vector.

3.2.4. Results

In order to make our results comparable to previous published results on these datasets, our experiments are conducted in the same way as in most previous works. For each classification task, we use 10-fold cross-validation. Classification accuracies are measured on the 10% hold-out data. Our method is compared with a number of existing multiple instance learning techniques. We replicated the results reported in their original papers for comparison if their results are measured similarly (using 10-fold cross-validation). Some results not available in their papers are marked as N/A (see Table 1).

For our approach (The PPMM Kernel), the only parameter that has to be manually set is the dimensionality K of aggregate posteriors (*i.e.* number of mixture components). We set $K = 30$ for drug activity prediction data and image data, and $K = 40$ for text data, since the instances in the text data have higher dimensionality and there are more labeled bags for training. Note that K is chosen subjectively but not carefully tuned for each task—tuning the parameter for each task could results in higher classification performance but may be impractical in real-world scenarios. Other implementation details of our approach in these experiments are the same as in Section 3.1, except that we adopt K-means clustering and histogram representations for these tasks, because they generally have high dimensional instance representation, and relatively small number of instances. Due to the local-optimal nature of K-means clustering, we tried multiple randomly seeded runs of algorithm, and chose the best one based on their performances on the training set.

One major advantage of our approach is the capacity to utilize a large number of unlabeled instances, but no extra unlabeled instance is available for the benchmark datasets, which implicates that the potential performance of our approach could possibly be underestimated in these experiments. Nevertheless, our approach performs generally better than or comparable with other MIL techniques (see Table 1). Since the benchmark datasets for MIL are rather small (number of bags ranges from tens to hundreds), slight differences in classification accuracy should not be overly emphasized. Instead, the most encouraging result we obtained is the optimal p values learned in these datasets. Note that the p values for drug activity prediction tasks (MUSK1 and MUSK2) are generally smaller than that for image classification tasks (ELEPHANT, TIGER, and FOX). Although the p value learned in the FOX dataset is smaller than other image datasets, we can further observe that all methods perform unsatisfactorily on the FOX dataset, which may indicate that this classification task itself could be impractical, hence the learned p value may be unreliable. Interestingly, comparing Table 1 and Figure 1 we could observe that the p values learned in real-world drug activity prediction tasks are close to that learned in synthetic task 1, and the p values learned in real-world image classification tasks are close to that learned in task 2 and task 3. We can also observe that the p values for text classification tasks (TREC1 and TREC2) are also small; possibly this is because the instance representation in the text domain are also high-level, and serving as *strong* evidences for bag-labels. In contrast, instance representation in the image domain are generally low-level (*e.g.* color, texture, shape), and they can only be considered as *weak* evidences for the

Table 1. Empirical results of multiple instance learning methods, the last row shows the optimal p value learned in each task. MUSK1 and MUSK2 are drug activity prediction datasets. ELEPHANT, TIGER, FOX are image classification datasets. TREC1 and TREC2 are text classification datasets. Best performance in each task is in bold. The average performance over all tasks is shown in the last column.

DATESETS:	MUSK1	MUSK2	ELEPHANT	TIGER	FOX	TREC1	TREC2	**Average**
APR (DIETTERICH, 1997)	92.4%	89.2%	N/A	N/A	N/A	N/A	N/A	N/A
DD (MARON, 1998)	88.0%	84.0%	N/A	N/A	N/A	N/A	N/A	N/A
EM-DD (ZHANG, 2001)	84.8%	84.9%	78.3%	72.1%	56.1%	85.8%	84.0%	78.0%
CITATION k-NN (WANG, J., 2000)	91.3%	86.0%	80.5%	78.0%	60.0%	87.0%	81.0%	80.5%
mi-SVM (ANDREWS, 2003)	87.4%	83.6%	82.0%	78.9%	58.2%	93.6%	78.2%	80.3%
MI-SVM (ANDREWS, 2003)	77.9%	84.3%	81.4%	**84.0%**	59.4%	**93.9%**	**84.5%**	80.8%
MISS-SVM (ZHOU, 2007)	87.6%	80.0%	N/A	N/A	N/A	N/A	N/A	N/A
MG-ACC KERNEL (KWOK, 2007)	90.1%	**90.4%**	N/A	N/A	N/A	N/A	N/A	N/A
PPMM KERNEL (this paper)	**95.6%**	81.2%	**82.4%**	80.2%	**60.3%**	93.3%	79.5%	**81.8%**
OPTIMAL VALUE OF p	0.7	0.15	2.1	1.3	0.8	0.75	0.4	

high-level bag-labels (*i.e.* elephant, tiger, fox).

4. Related Work

The concept of multiple instance learning was originally proposed in (Dietterich, 1997) for the application of drug activity prediction. The author assumes that positive instances all reside in an axis-parallel rectangle (APR), which implicates specific constraints that the shape should satisfy in order to bind well to some target protein. Although this assumption can be appropriate for this specific application, it is not clear how to adapt it to other applications, which may have more complex intrinsic structures in the instance space, and different instance-to-bag-label *mechanisms*.

Diverse Density (DD) (Maron, 1998) is another general framework for MIL. The author assumes that positive instances form a Gaussian-like pattern around some "concept point" in the instance space, which is expected to be close to at least one point in each positive bag and far away from all instances in negative bags. This assumption on the structure of instance space could also be over-simplified for some applications. And the algorithm, by definition, relies on the instance-to-bag-label *mechanism* in the original definition of multiple instance learning.

Citation k-NN adapts the memory based classification method k-NN to MIL, which considers not only the *references*, but also the *citers* as neighbors of a bag in determine its label, in order to be less affected by the negative instances in positive bags. It had been empirically proved to be more robust than standard k-NN. Nevertheless, the role of instance-to-bag-label *mechanism* is not clear in this framework.

Support vector machines (SVM) and the kernel trick (Schölkopf, 2001) have been very successful in traditional supervised learning. There also have been many attempts to apply them to MIL. These works falls into two major categories as summarized in (Kwok, 2007). The first family of methods try to modify the optimization problem of SVM, such as MI-SVM and mi-SVM (Andrews, 2003), which may result in non-convex optimization problems and suffer from local minima. The second family of methods design kernel functions on the bags, including (Gartner, 2002) and (Kwok, 2007). Our approach also falls into the second category, but it possesses a unique characteristic as *adapts* to various application domains with different instance-to-bag-label *mechanisms*.

The aggregate posteriors are essentially positive constant-sum real vectors, which reside in a simplex. Data in a simplex had been addressed from the metric learning perspective (Lebanon, 2003; Wang, H.-Y., 2007), which are related to our approach because the kernel defined for aggregate posteriors also gives rise to a distance metric on the simplex.

The idea of defining a kernel function based on posterior probabilities on mixture models had also been exploited in (Hertz, 2006). The author proposed the *KernelBoost* algorithm for learning with a large number of unlabeled data and few labeled data, in which the weak kernel mappings are defined as posterior probabilities on mixture models.

The two-phase learning scheme in our approach makes use of both unlabeled instances and labeled bags. It

is therefore conceptually related to semi-supervised learning (Zhu, 2005) and self-taught learning (Raina, 2007).

5. Conclusion

In this paper, we proposed a novel framework for *adapting* multiple instance learning to different *mechanisms* of how the instances determine the bag-labels. We showed that this *mechanism* is different in different application domains of multiple instance learning, and our approach well captures this domain-specific *mechanism* through learning with unlabeled instances and labeled bags.

To the best of our knowledge, this paper is the first work that addresses the problem of *adapting* multiple instance learning to different application domains with different instance-to-bag-label *mechanisms*. The major advantage of such a self-adaptive framework lies in that, if we are encountered with some novel application domain, which could be well formalized as multiple instance learning, but we have no *a priori* knowledge about the instance-to-bag-label *mechanisms* in that domain, we can learn the *mechanisms* from labeled bags, and design a kernel function *adapted* to this *mechanism*.

Acknowledgement

This work was supported in part by NKBRPC No. 2004CB318000, NHTRDP 863 Grant No. 2006AA01Z302, and No. 2007AA01Z336. Qiang Yang thanks Hong Kong CERG grants 621307 and and CAG grant HKBU1/05C.

Hua-Yan Wang would like to thank Haiyan Sun for her encouragements and insightful comments.

References

Andrews, S., Tsochantaridis, I., and Hofmann, T. (2003). Support vector machines for multiple-instance learning. In *NIPS* 15

Bunescu, R.C., Mooney, R.J. (2007) Multiple Instance Learning for Sparse Positive Bags. In *Proceedings of the 24th ICML*

Cristianini, N., Shawe-Taylor, J., Elisseeff, A., and Kandola, J. (2002). On kernel-target alignment. In *NIPS* 14

Dietterich, T.G., Lathrop, R.H. and Lozano-Perez, T. (1997). Solving the multiple instance problem with axisparallel rectangles. *Artificial Intelligence*, 89, 31-71.

Gartner, T., Flach, P., Kowalczyk, A., and Smola, A. (2002). Multi-instance kernels. IIn *Proceedings of the 19th ICML*

Hersh, W., Buckley, C., Leone, T.J., Hickam, D. (1994). OHSUMED: an interactive retrieval evaluation and new large test collection for research. In *SIGIR'94*

Hertz, T., Hillel, A.B., Weinshall, D. (2006). Learning a Kernel Function for Classification with Small Training Samples. In *Proceedings of the 23rd ICML*

Kwok, J.T. and Cheung, Pak-Ming. (2007). Marginalized Multi-Instance Kernels. In *IJCAI'07*

Lebanon, G. (2003). "Learning Riemannian metrics", In *Proceedings of the 19th UAI*

Maron, O., Lozano-Pérez, T. (1998). A Framework for Multiple-Instance Learning. In *NIPS* 10

Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y. (2007). Self-taught learning: Transfer Learning from Unlabeled Data. In *Proceedings of the 24st ICML*

Schölkopf, B., Smola, A.J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press

Wang, H.-Y., Zha, H., Qin, H. (2007). "Dirichlet aggregation: unsupervised learning towards an optimal metric for proportional data", In *Proceedings of the 24th ICML*

Wang, J. and Zucker, J.-D. (2000). Solving the multiple-instance problem: a lazy learning approach. *Proceedings of the 17th ICML*

Zhang, Q., Goldman, S.A. (2001). EM-DD: An improved multiple instance learning technique. In *NIPS* 13

Zhang, Q., Goldman, S.A., Yu, W. and Fritts, J. (2002). Content-based image retrieval using multiple-instance learning. In *Proceedings of the 19th ICML*

Zhou, Z.-H., and Xu, J.-M. (2007). On the Relation Between Multi-Instance Learning and Semi-Supervised Learning. In *Proceedings of the 24th ICML*

Zhu, X. (2005). Semi-supervised learning literature survey. Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison