University of Nebraska - Lincoln
DigitalCommons@University of Nebraska - Lincoln

CSE Conference and Workshop Papers | Computer Science and Engineering, Department of

2004

An Extended Kernel for Generalized Multiple-Instance Learning

Qingping Tao
University of Nebraska at Lincoln, qtao@cse.unl.edu

Stephen Scott
University of Nebraska at Lincoln, sscott2@unl.edu

N. V. Vinodchandran
University of Nebraska at Lincoln, vinod@cse.unl.edu

Thomas Takeo Osugi
University of Nebraska at Lincoln, tosugi@cse.unl.edu

Brandon Mueller
University of Nebraska at Lincoln, bmueller@cse.unl.edu

Follow this and additional works at: http://digitalcommons.unl.edu/cseconfwork

 Part of the Computer Sciences Commons

Tao, Qingping; Scott, Stephen; Vinodchandran, N. V.; Takeo Osugi, Thomas; and Mueller, Brandon, "An Extended Kernel for Generalized Multiple-Instance Learning" (2004). *CSE Conference and Workshop Papers*. 141.
http://digitalcommons.unl.edu/cseconfwork/141

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

16th IEEE International Conference on Tools with Artificial Intelligence, 2004. ICTAI 2004.
Digital Object Identifier: 10.1109/ICTAI.2004.29
Publication Year: 2004 , Page(s): 272 - 277

An Extended Kernel for Generalized Multiple-Instance Learning

Qingping Tao, Stephen Scott, N. V. Vinodchandran, Thomas Takeo Osugi, Brandon Mueller
Department of Computer Science & Engineering
University of Nebraska
Lincoln, NE 68588-0115
{qtao, sscott, vinod, tosugi, bmueller}@cse.unl.edu

Abstract

The multiple-instance learning (MIL) model has been successful in areas such as drug discovery and content-based image-retrieval. Recently, this model was generalized and a corresponding kernel was introduced to learn generalized MIL concepts with a support vector machine. While this kernel enjoyed empirical success, it has limitations in its representation. We extend this kernel by enriching its representation and empirically evaluate our new kernel on data from content-based image retrieval, biological sequence analysis, and drug discovery. We found that our new kernel generalized noticeably better than the old one in content-based image retrieval and biological sequence analysis and was slightly better or even with the old kernel in the other applications, showing that an SVM using this kernel does not overfit despite its richer representation.

1. Introduction

Dietterich et al. [3] introduced the multiple-instance learning (MIL) model motivated by the problem of predicting molecular binding affinity. They represented each molecule by a high-dimensional vector that describes its shape, and labeled molecules that bind at a particular site as positive examples and those that do not bind as negative. Then they learned an axis-parallel box that distinguishes the positives from the negatives. The motivation for the MIL model is the fact that a single molecule can have multiple conformations (shapes), and only one conformation need bind at the site for the molecule to be considered positive. Thus when an example is negative, all conformations in it are negative, but if an example is positive, then it may be the case that only one conformation of the set is positive, and the learner does not know which one. The MIL model has since been applied to content-based image retrieval [2, 9, 17, 19, 20], where each instance in a multi-instance example (*bag*) represents a feature of an image, and it is not known which feature corresponds to the content the user wants to retrieve. For both applications, it is typically assumed that the label of an example is a disjunction of the labels of the instances in the example.

Recently, Scott et al. [11] generalized the MIL model, allowing an example's label to be represented as an r-of-k threshold function rather than as a disjunction. They also presented an algorithm (GMIL-1) for learning in this new model. While GMIL-1 had advantages in terms of generalization error over conventional MIL algorithms, its time complexity was exponential in the dimension of the input space. This motivated Tao et al. [13] to define a kernel $k_\wedge$ that exactly corresponds to the feature mapping used by GMIL-1. They then showed that $k_\wedge$ was #P-complete to compute and presented a fully polynomial randomized approximation scheme (FPRAS) for it. Their results showed improvements in both generalization error and time complexity over GMIL-1 (and its faster version GMIL-2 [12]) on several data sets. Further, $k_\wedge$ outperformed conventional MIL algorithms on the high-dimensional Musk data sets.

However, the GMIL model Tao et al. used could be further generalized along the lines of Weidmann et al. [16]. One of our contributions is a new remapping that generalizes Weidmann et al.'s "count-based" MIL model and a kernel $k_{\min}$ that corresponds to that mapping. We then show that, as with $k_\wedge$, $k_{\min}$ is #P-complete to compute, so we give an FPRAS for it. Our final contribution is an empirical evaluation of our new kernel on several MIL data sets. We found that $k_{\min}$ can generalize better than $k_\wedge$ for a learning task in content-based image retrieval and in biological sequence analysis, but there is little room for improvement in the other learning tasks. However, we note that despite $k_{\min}$'s richer representation over $k_\wedge$, it does not overfit.

The rest of this paper is as follows. Section 2 introduces notation. In Section 3 we describe the MIL model, present Scott et al.'s generalization of it, and describe Tao et al.'s old kernel $k_\wedge$. In Section 4 we present our extension to $k_\wedge$ (which we call $k_{\min}$), show that computing $k_{\min}$ is #P-

IEEE COMPUTER SOCIETY

complete, and give an FPRAS for it. In Section 5 we describe experimental results of our new kernel on the applications content-based image retrieval, biological sequence analysis, and the Musk data sets. We conclude in Section 6.

2. Notation and Definitions

Let $\mathcal{X}$ denote $\{0, \ldots, s\}^d$ (though our results trivially generalize to $\mathcal{X} = \prod_{i=1}^{d}\{0, \ldots, s_i\}$). Let $B_{\mathcal{X}}$ denote the set of all axis-parallel boxes (including degenerate boxes) from $\mathcal{X}$. For multisets $P, Q \subseteq \mathcal{X}$, let $B(P \wedge Q)$ denote the set of boxes in $B_{\mathcal{X}}$ that contain a point from P and a point from Q. When P and Q contain single points then we will omit set notation. For example, $B(\{p\} \wedge \{q\})$ will be denoted $B(p \wedge q)$. The notion of approximation that we use is defined as follows.

Definition 1 *Let f be a counting problem. Then a randomized algorithm $\mathcal{A}$ is an* FPRAS (Fully Polynomial Randomized Approximation Scheme) *if for any instance x, and parameters $\epsilon, \delta > 0$, $\Pr[|\mathcal{A}(x) - f(x)| \leq \epsilon f(x)] \geq 1-\delta$ and A's running time is polynomial in $|x|$, $1/\epsilon$, and $1/\delta$. Further, we call $\mathcal{A}(x)$ an* ϵ-good approximation *of $f(x)$.*

To bound the sample sizes required to estimate quantities of interest, we will employ the Hoeffding bound.

Lemma 2 *(Hoeffding) Let X_i be independent random variables all with mean μ such that for all i, $a \leq X_i \leq b$. Then for any $\lambda > 0$, $\Pr\left[\left|\frac{1}{S}\sum_{i=1}^{S} X_i - \mu\right| \geq \lambda\right] \leq 2e^{-2\lambda^2 S/(b-a)^2}$.*

Since we're interested in ϵ-good approximations, we'll use $\lambda = \epsilon\mu$.

3. Multiple-Instance Learning

In the original MIL model [3], each example P is a *bag* (multiset) of n instances, and P is labeled positive if and only if at least one of the instances in P is labeled positive (it is unknown which instance(s) in P are labeled positive). Typically, the label of a point $p \in P$ is determined by its proximity to a target point c. The MIL model has since been extensively studied [15, 1, 8, 18] with applications focusing on molecular binding affinity (related to drug discovery) and content-based image retrieval [2, 9, 17, 19, 20].

Scott et al. [11] generalized the MIL model such that rather than P's label being a disjunction of the labels of P's instances, it is a threshold function. Each target concept is defined by a set of k "attraction" points $C = \{c_1, \ldots, c_k\}$ and a set of k' "repulsion" points $\bar{C} = \{\bar{c}_1, \ldots, \bar{c}_{k'}\}$. The label for a bag $P = \{p_1, \ldots, p_n\}$ is positive if and only if there is a subset of r points $C' \subseteq C \cup \bar{C}$ such that each attraction point $c_i \in C'$ is near some point in P (where "near" is defined as within a certain distance under some weighted norm) and each repulsion point $\bar{c}_j \in C'$ is not near any point in P. In other words, if one defines a boolean attribute a_i for each attraction point $c_i \in C$ that is 1 if there exists a point $p \in P$ near it and 0 otherwise and another boolean attribute $\bar{a}_i$ for each repulsion point $\bar{c}_j \in \bar{C}$ that is 1 if there is no point from P near it, then P's label is an r-of-$(k + k')$ threshold function over the attributes.

When they introduced their generalized MIL model, Scott et al. also gave an algorithm (GMIL-1) for it. GMIL-1 enumerates the set $B_{\mathcal{X}}$ of all possible boxes in the discretized space $\mathcal{X}$ and creates an attribute a_b for each box $b \in B_{\mathcal{X}}$. Given a bag $P \in \mathcal{X}^n$, the algorithm sets $a_b = 1$ if some point from P lies in b and $a_b = 0$ otherwise. To handle repulsion points, they defined complementary attributes $\bar{a}_b = 1 - a_b$. These $2|B_{\mathcal{X}}|$ attributes were given to Winnow [7], which learns a linear threshold unit.

Unfortunately, GMIL-1's time complexity is exponential in d. Later, Tao et al. [13] defined the kernel $k_\wedge$ that exactly corresponds to GMIL-1's remapping. When used with a support vector machine, this kernel allows one to simulate GMIL-1 in time polynomial in d and the bag size n.

Observation 3 *[13] Consider two bags $P, Q \subseteq \mathcal{X}$ and a mapping $\vec{\phi}_\wedge(P) = (a_1, \ldots, a_N)$ where $a_i = 1$ if the corresponding box $b_i \in B_{\mathcal{X}}$ contains a point from P and 0 otherwise. Then when using an SVM for learning, the remapping used by GMIL-1 corresponds to using the kernel*

$$k_\wedge(P, Q) = \vec{\phi}_\wedge(P) \cdot \vec{\phi}_\wedge(Q) = |B(P \wedge Q)|,$$

where $B(P \wedge Q)$ is the set of boxes that contain a point from P and a point from Q.

Tao et al. showed that exact computation of $k_\wedge$ is #P-complete, and then gave an FPRAS for it that with high probability computes an ϵ-good approximation of $k_\wedge$ in time[1] $O(n^2 d(\log s)\ln(1/\delta)/\epsilon^2)$. Their algorithm for estimating $|B(P \wedge Q)|$ is based on the general technique from Karp et al. [5] on the union of sets problem, where the goal is to take a description of m sets $B_1, \ldots, B_m$ and estimate the size of $B = \bigcup_{i=1}^{m} B_i$. Let $W = |B(P \wedge Q)| = |\bigcup_{p \in P, q \in Q} B(p \wedge q)|$. Given points $p, q \in \mathcal{X}$, let $\ell = (\ell_1, \ldots, \ell_d)$ be the lower corner of the bounding box of p and q, i.e. $\ell_i = \min\{p_i, q_i\}$ for all i. Similarly define $u = (u_1, \ldots, u_d)$ as the upper corner. Then $|B(p \wedge q)| = \left(\prod_{1 \leq i \leq d}(\ell_i + 1)\right)\left(\prod_{1 \leq i \leq d}(s_i - u_i + 1)\right)$. Being able to exactly compute this allows one to choose a set $B(p \wedge q)$ with probability $|B(p \wedge q)| / \left(\sum_{p \in P, q \in Q} |B(p \wedge q)|\right)$. Now since one can uniformly sample from $B(p \wedge q)$, it is possible to uniformly sample from $U = \{(p, q, b) : p \in P, q \in Q, b \in B(p \wedge q)\}$.

1 The algorithm as presented below has time complexity cubic in n. Tao et al. used a refined version of it, which has quadratic time complexity.

IEEE COMPUTER SOCIETY

Note that $|U| = \sum_{p\in P,q\in Q} |B(p \wedge q)|$. Consider all pairs $(p,q) \in P \times Q$. Define a total order $\prec$ on these pairs by sorting first by p's index in P, and then by q's index in Q. Now consider the set $G = \{(p,q,b) \in U :$ there are no pairs $(p',q') \prec (p,q)$ s.t. $b \in B(p' \wedge q')\}$. Then $|G| = |\bigcup_{p\in P,q\in Q} B(p \wedge q)| = W$. Checking whether $(p,q,b) \in G$ takes $O(dn)$ time. Tao et al.'s algorithm draws S samples (p,q,b) uniformly from U and increments a counter γ when $(p,q,b) \in G$.

Theorem 4 *[13] If $S \geq 4n^2 \ln(2/\delta)/\epsilon^2$, then*

$$\Pr\left[(1-\epsilon)W \leq \frac{|U|\gamma}{S} \leq (1+\epsilon)W\right] \geq 1-\delta.$$

Independently of Scott et al., Weidmann et al. [16] defined their own generalizations[2] of the MIL model. Their first (referred to as *presence-based* MIL in [16]) is the same as Scott et al.'s with $r = k$ and no repulsion points. Their second (*threshold-based* MIL) generalizes presence-based MIL by requiring each $c_i \in C$ to be near at least $t_i \geq 0$ distinct points from P for P to be labeled positive. Their third (*count-based* MIL) generalizes threshold-based by requiring the number of distinct points from P that are near c_i to be at least t_i and at most z_i, which cannot be represented by Scott et al.'s model.

4. Extending Tao et al.'s Kernel

We now extend $k_\wedge$ to work in a GMIL model that generalizes count-based MIL model of Weidmann et al. [16]. Recall that their count-based MIL model stipulates that a bag P is positive if and only if each concept point $c_i \in C$ is near at least t_i, and at most z_i, distinct points from P. We define a remapping and a kernel to capture the notion of count-based MIL, but using r-of-$(k+k')$ threshold concepts (expanding its representational ability beyond that of Weidmann et al.). Recall the old mapping, where $\vec{\phi}_\wedge(P)$ is a vector of $|B_\mathcal{X}|$ bits, and for each box $b \in B_\mathcal{X}$, attribute $a_b = 1$ if box b contains a point from bag P and 0 otherwise. In our new mapping $\vec{\phi}_{\min}(P)$, each box $b \in B_\mathcal{X}$ has n bits associated with it, and $a_{bi} = 1$ if box b contains at least i points from P and 0 otherwise. (Thus if b contains exactly j points from P, we have $a_{bi} = 1$ for $i \leq j$ and $a_{bi} = 0$ for $i > j$.) To see how this captures count-based MIL, imagine that there is exactly one target box b, and all positive bags have at least x and at most $y-1$ points in b. A weight vector capturing this target concept has $w_{bx} = +1$, $w_{by} = -1$, all other weights 0, and a bias term of $-1/2$. Adjusting the bias term and adding other nonzero components to the weight vector allows us to represent multiple target boxes in an r-of-k threshold function.

2 Note that Chen and Wang's algorithm DD-SVM [2] implies yet another generalization of the MIL model, though this generalization is not comparable to those of Scott et al. and Weidmann et al.

Let $P_b \subseteq P$ be the set of points from P that are contained in b. Then it is straightforward to see that the dot product $\vec{\phi}_{\min}(P) \cdot \vec{\phi}_{\min}(Q)$ is equivalent to the kernel

$$\begin{aligned} k_{\min}(P,Q) &= \sum_{b\in B_\mathcal{X}} \min(|P_b|,|Q_b|) \\ &= \sum_{b\in B(P\wedge Q)} \min(|P_b|,|Q_b|) \qquad (1) \\ &= \sum_{b\in B(P\wedge Q)} \frac{|P_b||Q_b|}{\max(|P_b|,|Q_b|)} \\ &= \sum_{b\in B(P\wedge Q)} \sum_{p\in P_b, q\in Q_b} \frac{1}{\max(|P_b|,|Q_b|)} \\ &= \sum_{p\in P,q\in Q} \sum_{b\in B(p\wedge q)} \frac{1}{\max(|P_b|,|Q_b|)} \ . \qquad (2) \end{aligned}$$

4.1. A Hardness Result for $k_{\min}$

Consider the corresponding counting problem #BOXMin which is defined as follows: Given a triple $\langle \mathcal{X}, P, Q\rangle$, compute $k_{\min}(P,Q)$. We need another related problem for showing the hardness of #BOXMin, which we now define. The problem #BOXAnd defined by Tao et al. is, given input the triple $\langle \mathcal{X}, P, Q\rangle$, compute $k_\wedge(P,Q) = |B(P \wedge Q)|$. In their proof showing that #BOXAnd is #P-complete, they also showed that a restricted version where $|P| = 1$ is #P-complete. We call this problem #RestrictedBOXAnd.

Theorem 5 *[13] #RestrictedBOXAnd is #P-complete.*

Theorem 6 *#BOXMin is #P-complete.*

Proof: #BOXMin is in #P: Given a triple $\langle \mathcal{X}, P, Q\rangle$, a nondeterministic machine first guesses a $b \in \mathcal{X}$ and then computes $\min(|P_b|,|Q_b|)$. If the minimum is 0, it rejects. Otherwise it branches into $\min(|P_b|,|Q_b|)$ paths and accepts. It is clear that the number of accepting paths = $k_{\min}(P,Q)$.

We now show that in fact computing $k_{\min}(P,Q)$ where P contains only one point is #P-complete by reducing #RestrictedBOXAnd to the restricted version of #BOXMin. The reduction is the identity map: an instance $\langle \mathcal{X}, \{p\}, Q\rangle$ of #RestrictedBOXAnd is mapped to the instance $\langle \mathcal{X}, \{p\}, Q\rangle$ of $k_{\min}(P,Q)$. Then we get

$$\begin{aligned} k_{\min}(\{p\},Q) &= \sum_{b\in B_\mathcal{X}} \min(|P_b|,|Q_b|) \\ &= \sum_{b\in B(p\wedge Q)} \min(|P_b|,|Q_b|) + \sum_{b\notin B(p\wedge Q)} \min(|P_b|,|Q_b|) \\ &= \sum_{b\in B(p\wedge Q)} 1 = |B(p \wedge Q)| = k_\wedge(\{p\},Q) \ . \ \square \end{aligned}$$

IEEE COMPUTER SOCIETY

4.2. Approximating $k_{\min}$

One way to approximate $k_{\min}$ is to approximate (1) via a simple change to Tao et al.'s algorithm for $k_\wedge$. When a sampled triple $(p, q, b) \in G$, we increment γ by $\min(|P_b|, |Q_b|)$ instead of by 1. Unfortunately, the best sample size bound we can get for this technique (via Lemma 2) is $S = n^6 \ln(2/\delta)/(2\epsilon^2)$, yielding a time complexity of $\Theta(n^7 d(\log s) \log(1/\delta)/\epsilon^2)$. Thus we instead approximate (2). We fix each (p, q) pair and approximate that term of the summation by uniformly sampling boxes from $B(p \wedge q)$ and taking the average of $1/\max(|P_b|, |Q_b|)$ for each box b in the sample. Multiplying this average by $|B(p \wedge q)|$ gives us an approximation of that term of the sum.

Theorem 7 *Let $\hat{k}_{\min}(P, Q)$ be our approximation of $k_{\min}(P, Q)$ via approximating each term of (2) individually as described above. Then after using $n^2(n-1)^2 \ln(2n^2/\delta)/(2\epsilon^2)$ total samples and $O(n^5 d \ln(n/\delta) \log s/\epsilon^2)$ total time, $\Pr\left[(1-\epsilon)k_{\min}(P, Q) \le \hat{k}_{\min}(P, Q) \le (1+\epsilon)k_{\min}(P, Q)\right] \ge 1 - \delta$.*

Proof: First note that an ϵ-good approximation of each (p, q) term of the summation yields an ϵ-good approximation of $k_{\min}(P, Q)$. Thus we focus on a single (p, q) pair. Given $b \in B(p \wedge q)$, let $X(b) = 1/\max(|P_b|, |Q_b|)$. Then

$$\mu = \mathrm{E}[X] = \frac{1}{|B(p \wedge q)|} \sum_{b \in B(p \wedge q)} 1/\max(|P_b|, |Q_b|) \ .$$

Thus $X, \mu \in [1/n, 1]$. Lemma 2 says that our approximation (using a sample of size S) is not ϵ-good with probability at most

$$2e^{-2\epsilon^2 \mu^2 S n^2/(n-1)^2} \le 2e^{-2\epsilon^2 S/(n-1)^2} \ ,$$

since $\mu \ge 1/n$. Setting this to be at most δ/n^2 (so we can apply the union bound over all n^2 failure probabilities) and solving for S, we get $S \ge (n-1)^2 \ln(2n^2/\delta)/(2\epsilon^2)$ as sufficient for an ϵ-good approximation of each term. Repeat this n^2 times (once per (p, q) pair) to approximate (2). The time complexity is $O(n^5 d \ln(n/\delta) \log s/\epsilon^2)$ since it takes time linear in n, d, and $\log s$ to compute each max. □

There is no guarantee that the Gram matrix computed by our approximation algorithm is positive semidefinite. However, it is reasonable to believe that if ϵ is small and $k_{\min}$'s Gram matrix (which is positive semidefinite) has no zero eigenvalues, the approximated matrix would not adversely affect SVM optimization. In our experiments, our approximate kernel works well with $\epsilon = 0.1$.

5. Experimental Results

To evaluate our new kernel, we tested it with SVM^{light} [4] on the following learning tasks: content-based image retrieval, biological sequence analysis, and the Musk data.

5.1. Content-Based Image Retrieval

In content-based image retrieval (CBIR), the user presents examples of desired images, and the task is to determine commonalities among the query images and retrieve similar ones from the database. In early work in applying conventional MIL for CBIR [9, 19], images were filtered and subsampled and then "blobs" (groups of m adjacent pixels) were extracted, which were each mapped to one point in a bag.

We experimented with the two CBIR tasks used by Tao et al. [13]. One is the "sunset" task: to distinguish images containing sunsets from those not containing sunsets. Like Zhang et al. [19], Tao et al. built 30 random testing sets of 720 examples (120 positives and 600 negatives): 150 negatives each from the waterfall, mountain, field, and flower sets. Each of 30 training sets consisted of 50 positives and 50 negatives.

Another CBIR task Tao et al. experimented with was to test a conjunctive CBIR concept, where the goal was to distinguish images containing a field with no sky from those containing a field and sky or containing no field. Zhang et al.'s field images that contained the sky were relabeled from positive to negative. Each training set had 6 bags of each of flower, mountain, sunset, and waterfall for negatives, and had around 30 fields, 6 of them negative and the rest positive. Each negative test set had 150 bags of each of flower, mountain, sunset, and waterfall. Also, each test set had 120 fields, around 50 serving as positives and the remainder as negatives.

The top two rows of each table in Table 1 summarize the prediction error of $k_{\min}$, Tao et al.'s kernel $k_\wedge$, and GMIL-2 [12], a faster version of GMIL-1 with comparable accuracy. For comparison purposes, we also give results for the algorithms Diverse Density [8] and EMDD [18] that operate in the conventional MIL model. The sunset task fits well into the conventional MIL model; hence error rates for EMDD and DD are only about 1% higher than ours, and there is little improvement of $\hat{k}_{\min}$ over $\hat{k}_\wedge$. But since the conjunctive task is more complex, we see that the generalized model helps significantly over the conventional model. Further, there is much improvement of $k_{\min}$ over $k_\wedge$ on the positives and a slight degradation on the negatives, yielding an overall improvement in generalization error.

IEEE COMPUTER SOCIETY

Total Error					
TASK	$\hat{k}_{\min}$	$\hat{k}_{\wedge}$	**GMIL-2**	**EMDD**	**DD**
sunset	0.084	0.088	0.098	0.096	0.099
conj.	0.084	0.108	0.147	0.215	0.181
protein	0.215	0.218	0.250	0.365	0.664
False Postive Error					
TASK	$\hat{k}_{\min}$	$\hat{k}_{\wedge}$	**GMIL-2**	**EMDD**	**DD**
sunset	0.112	0.120	0.082	0.082	0.078
conj.	0.198	0.192	0.140	0.213	0.173
protein	0.215	0.218	0.250	0.365	0.668
False Negative Error					
TASK	$\hat{k}_{\min}$	$\hat{k}_{\wedge}$	**GMIL-2**	**EMDD**	**DD**
sunset	0.078	0.080	0.157	0.166	0.168
conj.	0.075	0.102	0.244	0.244	0.282
protein	0.144	0.169	0.250	0.360	0.125

Table 1. Generalization errors for CBIR and protein learning tasks. $\hat{k}_{\wedge}$ and $\hat{k}_{\min}$ are based on approximations of the kernel with $\epsilon = 0.1$ and $\delta = 0.01$.

5.2. Identifying Trx-fold Proteins

The low conservation of primary sequence in protein superfamilies such as Thioredoxin-fold (Trx-fold) makes conventional modeling methods difficult to use. Wang et al. [14] propose using multiple-instance learning as a tool for identification of new Trx-fold proteins. They mapped each protein's primary sequence to a bag in the following way. First, they found in each sequence the primary sequence motif (typically CxxC) that is known to exist in all Trx-fold proteins. They then extracted a window of size 214 around it (30 residues upstream, 180 downstream) and aligned these windows around the motif. They then mapped all sequences to 8-dimensional profiles based on the numeric properties of Kim et al. [6] and used them as inputs to the multiple-instance learning algorithm.

Wang et al. used GMIL-2 to perform cross-validation tests: 20-fold CV on 20 positives and 8-fold CV on 160 negatives. So in each round, they trained GMIL-2 on 19 positive proteins plus one of 8 sets of negative proteins, and tested on the held-out positive protein plus the remaining 7 sets of negative proteins. They repeated this for each of the 8 sets of negative proteins. To compare with their results, we performed the same tests with $k_{\min}$, comparing to $k_{\wedge}$, EMDD and DD (bottom row of each table in Table 1). Here we see that applying $k_{\min}$ instead of $k_{\wedge}$ yields little improvement in false positive error, but there is a noticeable improvement in false negative error.

5.3. Musk Data Sets

Finally, we tested on the Musk data sets from the UCI repository[3], which represent different conformations of various molecules, labeled according to whether they exhibit a "musk-like" odor when smelled by a human expert. We performed 10-fold cross-validation experiments on the same 10 partitions used by Dietterich et al. [3].

For the Musk experiments, the ratio of diagonal entries in the kernel matrix to the off-diagonal entries was often around 10^{50}. So we applied the method of Schölkopf et al. [10] to solve this problem. We used the sub-polynomial function $x^{1/50}$ to reduce the range of each entry in the Gram matrices. We then let SVM^{light} work with the original Gram matrices as well as transduction empirical kernels and non-transduction empirical kernels.

Table 2 summarizes our results and those from Andrews et al. [1] with mi-SVM and MI-SVM and their results with EMDD[4]. Results for DD come from [8], TLC results come from the generalized MIL algorithm of Weidmann et al. [16], DD-SVM is from [2], and "IAPR" is the iterative axis-parallel rectangle algorithm from Dietterich et al.

While we see that the empirical kernels based on $k_{\wedge}$ and $k_{\min}$ provide some of the best results on both Musk sets, there is no improvement in $k_{\min}$ over $k_{\wedge}$. In fact, the results exactly match except for false positive error on Musk 1 for the transduction case (not shown), in which $k_{\wedge}$ is better. One possible explanation for this is that there is no room for improvement via a richer hypothesis class for these data sets. Another explanation for this phenomenon is that since $k_{\min}(P,Q)/k_{\wedge}(P,Q) \in [1,n]$ for all P,Q and the kernel is so diagonally dominant for such high-dimensional input data, the kernel values are too similar to each other to make a difference in training and testing. Thus in cases like Musk when there is diagonal dominance, there is probably little reason to choose $k_{\min}$ over $k_{\wedge}$.

6. Conclusions

Tao et al.'s kernel $k_{\wedge}$ has been shown to be very powerful and well-suited for MIL learning tasks. We extended this kernel by increasing its representational power to generalize "count-based" MIL of Weidmann et al. Empirical results show noticeable improvements in generalization error for $k_{\min}$ over $k_{\wedge}$ for the conjunctive CBIR task and protein classification. In particular, our new kernel reduced false negative error over $k_{\wedge}$ while not increasing false positive error. For the sunset and Musk learning tasks, there was little improvement in any error rate, but there was little degrada-

3 `http://www.ics.uci.edu/~mlearn/MLRepository.html`

4 Andrews et al. point out that the EMDD experiments of Zhang et al. [18] were optimistically biased since they used the test set to choose the final hypotheses. Thus Andrews et al. reran them.

IEEE COMPUTER SOCIETY

ALGORITHM	Total Error MUSK1	MUSK2
$\hat{k}_{\min}$	0.176	0.227
$\hat{k}_{\min\ emp}$ non-transduction	0.120	0.118
$\hat{k}_{\min\ emp}$ transduction	0.098	0.097
$\hat{k}_{\wedge}$	0.176	0.227
$\hat{k}_{\wedge\ emp}$ non-transduction	0.120	0.118
$\hat{k}_{\wedge\ emp}$ transduction	0.088	0.097
TLC	0.113	0.169
EMDD	0.152	0.151
DD	0.120	0.160
mi-SVM	0.126	0.164
MI-SVM	0.221	0.157
DD-SVM	0.142	0.087
IAPR	0.076	0.108

Table 2. Classification error on the Musk data sets. EMDD, mi-SVM, and MI-SVM are from [1], DD is from [8], TLC is from [16], DD-SVM is from [2], and IAPR is from [3].

tion either. This implies that our new kernel does not overfit despite its richer representation.

Acknowledgments

The authors thank Tom Dietterich for his Musk partitionings, Qi Zhang, Sally Goldman, and James Wang for the CBIR data (indirectly from Corel and webshots.com), and Qi Zhang for his EMDD/DD code. This research was funded in part by NSF grants CCR-0092761 and EPS-0091900, and a grant from the NU Foundation. It was also supported in part by NIH Grant Number RR-P20 RR17675 from the IDeA program of the National Center for Research Resources. This work was completed in part utilizing the Research Computing Facility of the University of Nebraska.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems 15*, 2002.

[2] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939, 2004.

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

[4] T. Joachims. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*, chapter 11, pages 169–184. MIT Press, 1999.

[5] R. Karp, M. Luby, and N. Madras. Monte-Carlo approximation algorithms for enumeration problems. *Journal of Algorithms*, 10:429–448, 1989.

[6] J. Kim, E. N. Moriyama, C. G. Warr, P. J. Clyne, and J. R. Carlson. Identification of novel multi-transmembrane proteins from genomic databases using quasi-periodic structural properties. *Bioinformatics*, 16(9):767–775, 2000.

[7] N. Littlestone. Redundant noisy attributes, attribute errors, and linear threshold learning using Winnow. In *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, pages 147–156, San Mateo, CA, 1991.

[8] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *Advances in Neural Information Processing Systems 10*, 1998.

[9] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proc. 15th International Conf. on Machine Learning*, pages 341–349, 1998.

[10] B. Schölkopf, J. Weston, E. Eskin, C. Leslie, and W. S. Noble. A kernel approach for learning from almost orthogonal patterns. In *Proceedings of the 13th European Conference on Machine Learning*, pages 511–528, 2002.

[11] S. D. Scott, J. Zhang, and J. Brown. On generalized multiple-instance learning. Technical Report UNL-CSE-2003-5, Dept. of Comp. Sci., University of Nebraska, 2003.

[12] Q. Tao and S. Scott. A faster algorithm for generalized multiple-instance learning. In *Proceedings of the 17th International Florida Artificial Intelligence Research Society Conference (FLAIRS)*, pages 550–555, 2004.

[13] Q. Tao, S. Scott, N. V. Vinodchandran, and T. Osugi. SVM-based generalized multiple-instance learning via approximate box counting. In *Proc. of the Twenty-First International Conference on Machine Learning*, pages 799–806, 2004.

[14] C. Wang, S. Scott, J. Zhang, Q. Tao, D. E. Fomenko, and V. N. Gladyshev. A study in modeling low-conservation protein superfamilies. Technical Report TR-UNL-CSE-2004-3, Dept. of Computer Science, University of Nebraska, 2004.

[15] J. Wang and J. D. Zucker. Solving the multiple-instance problem: A lazy learning approach. In *Proc. 17th International Conf. on Machine Learning*, pages 1119–1125, 2000.

[16] N. Weidmann, E. Frank, and B. Pfahringer. A two-level learning method for generalized multi-instance problems. In *Proceedings of the European Conference on Machine Learning*, pages 468–479, 2003.

[17] C. Yang and T. Lozano-Pérez. Image database retrieval with multiple-instance learning techniques. In *Proc. of the 16th Int. Conf. on Data Engineering*, pages 233–243, 2000.

[18] Q. Zhang and S. A. Goldman. EM-DD: An improved multiple-instance learning technique. In *Neural Information Processing Systems 14*, pages 1073–1080, 2001.

[19] Q. Zhang, S. A. Goldman, W. Yu, and J. E. Fritts. Content-based image retrieval using multiple-instance learning. In *Proc. 19th International Conf. on Machine Learning*, pages 682–689. Morgan Kaufmann, San Francisco, CA, 2002.

[20] Z. Zhou, M. Zhang, and K. Chen. A novel bag generator for image database retrieval with multi-instance learning techniques. In *Proc. of the 15th IEEE International Conference on Tools with Artificial Intelligence*, pages 565–569, 2003.

IEEE COMPUTER SOCIETY