# M³MIML: A Maximum Margin Method for Multi-Instance Multi-Label Learning

Min-Ling Zhang
College of Computer and Information
Engineering, Hohai University, China
zhangml@hhu.edu.cn

Zhi-Hua Zhou
National Key Laboratory for Novel Software
Technology, Nanjing University, China
zhouzh@lamda.nju.edu.cn

## Abstract

*Multi-instance multi-label learning (MIML) deals with the problem where each training example is associated with not only multiple instances but also multiple class labels. Previous MIML algorithms work by identifying its equivalence in degenerated versions of multi-instance multi-label learning. However, useful information encoded in training examples may get lost during the identification process. In this paper, a maximum margin method is proposed for MIML which directly exploits the connections between instances and labels. The learning task is formulated as a quadratic programming (QP) problem and implemented in its dual form. Applications to scene classification and text categorization show that the proposed approach achieves superior performance over existing MIML methods.*

## 1. Introduction

Multi-instance multi-label learning (MIML) is a newly proposed framework, where each example in the training set is associated with multiple instances as well as multiple labels [32, 33]. Many real-world problems involving ambiguous objects can be properly formalized under MIML. For instance, in image classification, an image generally contains several naturally-partitioned patches each can be represented as an instance, while such an image can correspond to multiple semantic classes simultaneously, such as *clouds*, *grassland* and *lions*; In bioinformatics, an gene sequence generally encodes a number of segments each can be expressed as an instance, while this sequence may be associated with several functional classes, such as *metabolism*, *transcription* and *protein synthesis*; In text categorization, each document usually consists of several sections or paragraphs each can be regarded as an instance, while the document may be assigned to a set of predefined topics, such as *sports*, *Beijing Olympics* and even *torch relay*.

The *traditional supervised learning*, i.e. single-instance single-label learning (SISL), can be viewed as a *degenerated* version of MIML. In SISL, each example is restricted to have only one instance and only one label. Existing approaches solve MIML problem by identifying its equivalence in SISL via problem reduction. Although this kind of identification strategy is feasible, the performance of the acquired algorithm may suffer from the loss of information incurred during the reduction process. Therefore, one open problem for MIML is that whether this learning framework can be tackled directly by exploiting connections between the instances and the labels of an MIML example [32].

In this paper, a novel algorithm named M³MIML, i.e. Maximum Margin Method for Multi-Instance Multi-Label learning, is proposed. Briefly, M³MIML assumes a linear model for each class, where the output on one class is set to be the maximum prediction of all the MIML example's instances with respect to the corresponding linear model. Subsequently, the outputs on all possible classes are combined to define the margin of the MIML example over the classification system. Obviously, each instance is involved in determining the output on each possible class and the correlations between different classes are also addressed in the combination phase. Therefore, the connections between the instances and the labels of an MIML example are *explicitly* exploited by M³MIML.

The rest of this paper is organized as follows. Section 2 gives the formal definition of MIML and reviews the related works. Section 3 proposes the new MIML approach. Section 4 reports experimental results on two real-world MIML data sets. Finally, Section 5 concludes and indicates several issues for future work.

## 2. Related Work

Let $\mathcal{X} = \mathbb{R}^d$ denote the input space of instances and $\mathcal{Y} = \{1, 2, \ldots, Q\}$ the set of class labels. The task of MIML is to learn a function $f_{\text{MIML}} : 2^{\mathcal{X}} \rightarrow 2^{\mathcal{Y}}$ from a set of

MIML training examples $\{(X_i, Y_i)|1 \leq i \leq N\}$, where $X_i \subseteq \mathcal{X}$ is a bag of instances $\{\mathbf{x}_1^i, \mathbf{x}_2^i, \ldots, \mathbf{x}_{n_i}^i\}$ and $Y_i \subseteq \mathcal{Y}$ is a set of labels $\{y_1^i, y_2^i, \ldots, y_{l_i}^i\}$ associated with $X_i$. Here $n_i$ is the number of instances in $X_i$ and $l_i$ the number of labels in $Y_i$. The MIML framework is closely related to the learning frameworks of *multi-instance learning* [10], *multi-label learning* [18, 21] and traditional supervised learning.

Multi-instance learning [10], or multi-instance single-label learning (MISL), was coined by Dietterich *et al.* in their investigation of drug activity prediction problem. The task of MISL is to learn a function $f_{\mathrm{MISL}} : 2^{\mathcal{X}} \to \{+1, -1\}$ from a set of MISL training examples $\{(X_i, y_i)|1 \leq i \leq N\}$, where $X_i \subseteq \mathcal{X}$ is a bag of instances $\{\mathbf{x}_1^i, \mathbf{x}_2^i, \ldots, \mathbf{x}_{n_i}^i\}$ and $y_i \in \{+1, -1\}$ is the binary label of $X_i$. After the seminal work of Dietterich *et al.* [10], numerous MISL learning algorithms have been proposed [2, 8, 16, 19, 25, 26, 29] and successfully applied to many applications especially in image categorization and retrieval [6, 7, 17, 30]. More works on MISL can be found in [31].

Multi-label learning [18, 21], or single-instance multi-label learning (SIML), originated from the investigation of text categorization problems. The task of SIML is to learn a function $f_{\mathrm{SIML}} : \mathcal{X} \to 2^{\mathcal{Y}}$ from a set of SIML training examples $\{(\mathbf{x}_i, Y_i)|1 \leq i \leq N\}$, where $\mathbf{x}_i \in \mathcal{X}$ is an instance and $Y_i \subseteq \mathcal{Y}$ is a set of labels $\{y_1^i, y_2^i, \ldots, y_{l_i}^i\}$ associated with $\mathbf{x}_i$. A number of SIML learning algorithms have been proposed by exploiting the relationships between different labels [5, 12, 15, 28, 34]. SIML techniques have been successfully applied to applications including text and image categorization [3, 14, 18, 21, 24]. More works on SIML can be found in [23].

According to the above definitions, it is clear that traditional supervised learning (SISL) can be regarded as a degenerated version of either MISL or SIML. Furthermore, SISL, MISL and SIML are all degenerated versions of MIML. Therefore, an intuitive way of solving MIML problem is to identify its equivalence in SISL, using either MISL or SIML as the bridge. Actually, while formalizing the MIML framework, Zhou and Zhang [32] adopted this strategy and proposed two MIML algorithms named MIML-BOOST and MIMLSVM.

MIMLBOOST reduces the MIML problem into an SISL one using MISL as the bridge. Specifically, MIMLBOOST firstly transforms the original MIML task into an MISL one by converting each MIML example $(X_i, Y_i)$ into $|\mathcal{Y}|$ number of MISL examples $\{([X_i, y], \Phi[X_i, y]) | y \in \mathcal{Y}\}$. Here, $[X_i, y]$ contains $n_i$ instances $\{(\mathbf{x}_1^i, y), \ldots, (\mathbf{x}_{n_i}^i, y)\}$ formed by concatenating each of $X_i$'s instance with label $y$, while $\Phi[X_i, y] = +1$ if $y \in Y_i$ and $-1$ otherwise. After that, MIMLBOOST solves the derived MISL problem by employing a specific algorithm named MIBOOSTING [26]. This algorithm deals with MISL problem by reducing it into an SISL one under the assumption that each instance in the bag

contributes equally and independently to a bag's label.

In contrast to MIMLBOOST, MIMLSVM reduces the MIML problem into an SISL one using SIML as the bridge. Firstly, MIMLSVM transforms the original MIML task into an SIML one by converting each MIML example $(X_i, Y_i)$ into an SIML example $(\tau(X_i), Y_i)$. Here, the function $\tau(\cdot)$ maps a bag of instances $X_i$ into a single instance $\mathbf{z}_i$ using *constructive clustering* [32], where $k$-medoids clustering is performed on $\Lambda = \{X_1, X_2, \ldots, X_N\}$ at the level of bags and components of $\mathbf{z}_i$ correspond to the distances between $X_i$ and the medoids of the clustered groups. After that, MIMLSVM solves the derived SIML problem by employing a specific algorithm named MLSVM [3]. This algorithm deals with SIML problem by decomposing it into multiple SISL problems (one per class), where instance $\mathbf{x}_i$ associated with label set $Y_i$ will be regarded as positive instance when building classifier for class $y \in Y_i$ while regarded as negative instance when building classifier for class $y \notin Y_i$.

Obviously, the above approaches solve the MIML problem by reformulating it into its degenerated versions, while useful information encoded between instances and labels may get lost during the reduction process. Next we will present the M³MIML algorithm which explicitly exploit the connections between instances and labels.

## 3. The Proposed Approach

### 3.1. Primal form

Given an MIML training example $(X_i, Y_i)$, let $\vec{Y_i}$ denote the category vector for $X_i$ whose $l$-th component $\vec{Y_i}(l)$ equals $+1$ if $l \in Y_i$ and $-1$ otherwise. Suppose the classification system is composed of $Q$ linear models $\{(\mathbf{w}_l, b_l)|l \in \mathcal{Y}\}$, each corresponding to a possible class label. Here, $\mathbf{w}_l \in \mathbb{R}^d$ is the weight vector for the $l$-th class and $b_l \in \mathbb{R}$ is the corresponding bias.

M³MIML assumes that the system's output for $(X_i, Y_i)$ on the $l$-th class is determined by the *maximum* prediction of $X_i$'s instances with respect to $(\mathbf{w}_l, b_l)$. Note that this kind of strategy has been successfully employed to handle objects with multi-instance representations [1, 16]. Therefore, for an unseen bag $X \subseteq \mathcal{X}$, its associated label set is determined via:

$$Y = \{l| \max_{\mathbf{x} \in X}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l) \geq 0, l \in \mathcal{Y}\} \quad (1)$$

Based on the output, we define the margin of $(X_i, Y_i)$ on the $l$-th class as:

$$\frac{\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l)}{\|\mathbf{w}_l\|} \quad (2)$$

Here, $\langle \cdot, \cdot \rangle$ calculates the dot product between two vectors and $\| \cdot \|$ denotes the vector norm. Then, the margin of

$(X_i, Y_i)$ with respect to the classification system is set to be the minimum margin of $(X_i, Y_i)$ over all classes:

$$\min_{l \in \mathcal{Y}} \frac{\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l)}{\|\mathbf{w}_l\|} \qquad (3)$$

Consequently, the margin of the whole training set $S = \{(X_i, Y_i) | 1 \leq i \leq N\}$ (denoted as $\Delta_S$) with respect to the classification system corresponds to:

$$\Delta_S = \min_{1 \leq i \leq N} \min_{l \in \mathcal{Y}} \frac{\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l)}{\|\mathbf{w}_l\|} \qquad (4)$$

Suppose that all training examples in $S$ can be perfectly classified by the classification system, we can normalize the parameters $\{(\mathbf{w}_l, b_l) | l \in \mathcal{Y}\}$ such that $\forall i \in \{1, \ldots, N\}$ and $l \in \mathcal{Y}$, the following conditions are satisfied:

$$\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l) \geq 1 \qquad (5)$$

Furthermore, for each $l \in \mathcal{Y}$, the equality will hold for at least one $i \in \{1, \ldots, N\}$. Based on this, Eq.(4) can be rewritten as follows:

$$\begin{aligned}
\Delta_S &= \min_{1 \leq i \leq N} \min_{l \in \mathcal{Y}} \frac{\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l)}{\|\mathbf{w}_l\|} \\
&= \min_{l \in \mathcal{Y}} \min_{1 \leq i \leq N} \frac{\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l)}{\|\mathbf{w}_l\|} \\
&= \min_{l \in \mathcal{Y}} \frac{1}{\|\mathbf{w}_l\|} \qquad (6)
\end{aligned}$$

Note that maximizing $\min_{l \in \mathcal{Y}} \frac{1}{\|\mathbf{w}_l\|}$ is equivalent to minimizing $\frac{1}{2} \max_{l \in \mathcal{Y}} \|\mathbf{w}_l\|^2$. Therefore, the target maximum margin optimization problem can be formulated as:

**Problem 1**

$$\min_{\{(\mathbf{w}_l, b_l) | l \in \mathcal{Y}\}} \quad \frac{1}{2} \max_{l \in \mathcal{Y}} \|\mathbf{w}_l\|^2$$

subject to : $\forall i \in \{1, \ldots, N\}, l \in \mathcal{Y}$ such that

$$\begin{cases} \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l) \geq 1, & \text{if } l \in Y_i \\ \forall \mathbf{x} \in X_i : -\langle \mathbf{w}_l, \mathbf{x} \rangle - b_l \geq 1, & \text{if } l \in \overline{Y_i} \end{cases}$$

Here $\overline{Y_i}$ denotes the complementary set of $Y_i$ in $\mathcal{Y}$. Note that the constraint $\vec{Y_i}(l) \cdot \max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l) \geq 1$ as shown in Eq.(5) has been expressed in cases of $\vec{Y_i}(l) = +1$ and $\vec{Y_i}(l) = -1$ respectively. The above optimization problem is difficult to solve which involves the $\max(\cdot)$ function in both the objective function and constraints. To simplify the

problem, note that:

$$\max_{l \in \mathcal{Y}} \|\mathbf{w}_l\|^2 \leq \sum_{l=1}^{Q} \|\mathbf{w}_l\|^2 \qquad \text{and}$$

$$\max_{\mathbf{x} \in X_i}(\langle \mathbf{w}_l, \mathbf{x} \rangle + b_l) \geq \frac{\sum_{j=1}^{n_i}(\langle \mathbf{w}_l, \mathbf{x}_j^i \rangle + b_l)}{n_i} \qquad (7)$$

Therefore, Problem 1 can be approximated as:

**Problem 2**

$$\min_{\{(\mathbf{w}_l, b_l) | l \in \mathcal{Y}\}} \quad \frac{1}{2} \sum_{l=1}^{Q} \|\mathbf{w}_l\|^2$$

subject to : $\forall i \in \{1, \ldots, N\}, l \in \mathcal{Y}$ such that

$$\begin{cases} \frac{\sum_{j=1}^{n_i}(\langle \mathbf{w}_l, \mathbf{x}_j^i \rangle + b_l)}{n_i} \geq 1, & \text{if } l \in Y_i \\ -\langle \mathbf{w}_l, \mathbf{x}_j^i \rangle - b_l \geq 1 \, (1 \leq j \leq n_i), & \text{if } l \in \overline{Y_i} \end{cases}$$

In order to deal with practical situations where training set can not be perfectly classified, slack variables are introduced for all constraints to accommodate classification error. This leads to the specific optimization problem considered by M³MIML in its primal form:

**Problem 3**

$$\min_{\{W, \mathbf{b}, \Xi, \Theta\}} \quad \frac{1}{2} \sum_{l=1}^{Q} \|\mathbf{w}_l\|^2 + C \sum_{l=1}^{Q} \left( \sum_{i \in S_l} \xi_{il} + \sum_{i \in \overline{S}_l} \sum_{j=1}^{n_i} \theta_{ilj} \right)$$

subject to : $\forall i \in \{1, \ldots, N\}, l \in \mathcal{Y}$ such that

$$\begin{cases} \frac{\sum_{j=1}^{n_i}(\langle \mathbf{w}_l, \mathbf{x}_j^i \rangle + b_l)}{n_i} \geq 1 - \xi_{il}, & \text{if } l \in Y_i \\ -\langle \mathbf{w}_l, \mathbf{x}_j^i \rangle - b_l \geq 1 - \theta_{ilj} \, (1 \leq j \leq n_i), & \text{if } l \in \overline{Y_i} \end{cases}$$

$$\xi_{il} \geq 0, \quad \theta_{ilj} \geq 0 \, (1 \leq j \leq n_i)$$

Where $S_l = \{i | 1 \leq i \leq N, l \in Y_i\}$ is the index set for those MIML examples associated with label $l$. Correspondingly, $\overline{S}_l = \{i | 1 \leq i \leq N, l \in \overline{Y_i}\}$ is the index set for those MIML examples not associated with label $l$. $W = [\mathbf{w}_1, \ldots, \mathbf{w}_Q]$ is the matrix comprising all weight vectors while $\mathbf{b} = [b_1, \ldots, b_Q]$ is the vector comprising all bias values. Similarly, $\Xi = \{\xi_{il} | 1 \leq i \leq N, l \in Y_i\}$ and $\Theta = \{\theta_{ilj} | 1 \leq i \leq N, l \in \overline{Y_i}, 1 \leq j \leq n_i\}$ are the set of slack variables. In addition, constant $C$ in the objective function trades off the classification system's margin and its empirical loss on the training set.

### 3.2. Dual form

So far we have only assumed linear models to deal with MIML learning task. Note that the primal form of M³MIML is a standard quadratic programming (QP) problem, which has convex objective function and linear constraints. Therefore, by solving Problem 3 in its dual form, nonlinearity can

be incorporated into it via the well-known *kernel trick*. The Lagrangian of Problem 3 is:

$$\mathcal{L}(W, \mathbf{b}, \Xi, \Theta, A, B, \Gamma, \Phi) =$$

$$\frac{1}{2}\sum_{l=1}^{Q}\|\mathbf{w}_l\|^2 + C\sum_{l=1}^{Q}\left(\sum_{i\in S_l}\xi_{il} + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\theta_{ilj}\right)\right)$$

$$-\sum_{l=1}^{Q}\left(\sum_{i\in S_l}\alpha_{il}\left(\frac{\sum_{j=1}^{n_i}(\langle\mathbf{w}_l, \mathbf{x}_j^i\rangle + b_l)}{n_i} - 1 + \xi_{il}\right)\right.$$

$$\left. + \sum_{i\in\overline{S}_l}\sum_{j=1}^{n_i}\beta_{ilj}\left(-\langle\mathbf{w}_l, \mathbf{x}_j^i\rangle - b_l - 1 + \theta_{ilj}\right)\right)$$

$$-\sum_{l=1}^{Q}\left(\sum_{i\in S_l}\gamma_{il}\xi_{il} + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\phi_{ilj}\theta_{ilj}\right)\right) \quad (8)$$

Here $A = \{\alpha_{il}|1 \leq i \leq N, l \in Y_i\}$, $B = \{\beta_{ilj}|1 \leq i \leq N, l \in \overline{Y}_i, 1 \leq j \leq n_i\}$, $\Gamma = \{\gamma_{il}|1 \leq i \leq N, l \in Y_i\}$ and $\Phi = \{\phi_{ilj}|1 \leq i \leq N, l \in \overline{Y}_i, 1 \leq j \leq n_i\}$ are the set of *nonnegative* dual variables for different constraints.

To get the dual form of Problem 3, the derivatives of Lagrangian (8) with respect to the primal variables $\{W, \mathbf{b}, \Xi, \Theta\}$ are set to be zero. Consequently, setting $\frac{\partial\mathcal{L}}{\partial\mathbf{w}_l} = 0$ $(l \in \mathcal{Y})$ yields:

$$\mathbf{w}_l = \sum_{i\in S_l}\left(\alpha_{il}\frac{\sum_{j=1}^{n_i}\mathbf{x}_j^i}{n_i}\right) + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}-\beta_{ilj}\mathbf{x}_j^i\right) \quad (9)$$

Setting $\frac{\partial\mathcal{L}}{\partial b_l} = 0$ $(l \in \mathcal{Y})$ yields:

$$\sum_{i\in S_l}\alpha_{il} + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}-\beta_{ilj}\right) = 0 \quad (10)$$

Setting $\frac{\partial\mathcal{L}}{\partial\xi_{il}} = 0$ $(1 \leq i \leq N, l \in Y_i)$ yields:

$$\gamma_{il} = C - \alpha_{il} \quad (11)$$

Finally, setting $\frac{\partial\mathcal{L}}{\partial\theta_{ilj}} = 0$ $(1 \leq i \leq N, l \notin Y_i, 1 \leq j \leq n_i)$ yields:

$$\phi_{ilj} = C - \beta_{ilj} \quad (12)$$

Substituting Eqs.(9) to (12) back into Lagrangian (8) gives rise to the *Lagrangian dual form* of Problem 3:

**Problem 4**

$$\max_{A,B,\Gamma,\Phi} \Omega(A, B, \Gamma, \Phi)$$

subject to : $\forall i \in \{1, \ldots, N\}$, $l \in \mathcal{Y}$ such that

$$\begin{cases} 0 \leq \alpha_{il} \leq C, & \text{if } l \in Y_i \\ 0 \leq \beta_{ilj} \leq C \ (1 \leq j \leq n_i), & \text{if } l \notin Y_i \end{cases}$$

$$\sum_{i\in S_l}\alpha_{il} + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}-\beta_{ilj}\right) = 0$$

Here the dual objective function $\Omega(A, B, \Gamma, \Phi)$ is:

$$\Omega(A, B, \Gamma, \Phi) =$$

$$-\frac{1}{2}\sum_{l=1}^{Q}\left(\sum_{i\in S_l}\sum_{i'\in S_l}\left(\frac{\alpha_{il}\alpha_{i'l}}{n_i n_{i'}}\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}\langle\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}\rangle\right)\right.$$

$$+ 2\sum_{i\in S_l}\sum_{i'\in\overline{S}_l}\left(\frac{\alpha_{il}}{n_i}\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}-\beta_{i'lj'}\langle\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}\rangle\right)$$

$$\left. + \sum_{i\in\overline{S}_l}\sum_{i'\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}\beta_{ilj}\beta_{i'lj'}\langle\mathbf{x}_j^i, \mathbf{x}_{j'}^{i'}\rangle\right)\right)$$

$$+ \sum_{l=1}^{Q}\left(\sum_{i\in S_l}\alpha_{il} + \sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\beta_{ilj}\right)\right) \quad (13)$$

In Problem 4, the first constraints (the inequalities) are inherited from the nonnegative properties of dual variables together with Eqs.(11) and (12), while the second constraints (the equalities) are inherited from Eq.(10).

As same as Problem 3 (primal form), it is evident that Problem 4 (dual form) also falls into the category of QP problems. Note that the constraints in Problem 4 only bound dual variables with *intervals* and *linear equalities*, such kind of QP problem can be solved by an efficient iterative approach named the Franke and Wolfe's method [13]. The fundamental idea of this method is to transform the difficult QP problem into a sequence of simpler linear programming (LP) problems. Due to page limit, the Franke and Wolfe' method is not elaborated here while its detailed description can be found in [12, 13]. To apply this method, the gradients of the dual objective function are indispensable:

$$\frac{\partial\Omega}{\partial\alpha_{il}} = 1 - \sum_{i'\in S_l}\left(\frac{\alpha_{i'l}}{n_{i'}n_i}\sum_{j'=1}^{n_{i'}}\sum_{j=1}^{n_i}\langle\mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i\rangle\right)$$

$$- \sum_{i'\notin\overline{S}_l}\left(\frac{1}{n_i}\sum_{j'=1}^{n_{i'}}\sum_{j=1}^{n_i}-\beta_{i'lj'}\langle\mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i\rangle\right) \quad (14)$$

$$\frac{\partial\Omega}{\partial\beta_{ilj}} = 1 - \sum_{i'\in S_l}\left(\frac{\alpha_{i'l}}{n_{i'}}\sum_{j'=1}^{n_{i'}}-\langle\mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i\rangle\right)$$

$$- \sum_{i'\notin\overline{S}_l}\left(\sum_{j'=1}^{n_{i'}}\beta_{i'lj'}\langle\mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i\rangle\right) \quad (15)$$

After solving Problem 4 with the Franke and Wolfe's method, parameters of the classification system can be determined with the help of Karush-Kuhn-Tucker (KKT) conditions [9]. Concretely, the weight vectors $\mathbf{w}_l$ $(l \in \mathcal{Y})$ are calculated using Eq.(9). One way to compute the bias

**Table 1. Characteristics of the data sets.**

| Data set | Number of examples | Number of classes | Number of features | Instances per bag | | | Labels per example ($k$) | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | min | max | mean±std. | $k$=1 | $k$=2 | $k \geq 3$ |
| Scene | 2,000 | 5 | 15 | 9 | 9 | 9.00±0.00 | 1,543 | 442 | 15 |
| Reuters | 2,000 | 7 | 243 | 2 | 26 | 3.56±2.71 | 1,701 | 290 | 9 |

values $b_l$ ($l \in \mathcal{Y}$) is as follows:

$$b_l = \varphi(i,l) = 1 - \sum_{i' \in S_l} \left( \frac{\alpha_{i'l}}{n_{i'} n_i} \sum_{j'=1}^{n_{i'}} \sum_{j=1}^{n_i} \langle \mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i \rangle \right)$$
$$- \sum_{i' \notin \overline{S}_l} \left( \frac{1}{n_i} \sum_{j'=1}^{n_{i'}} \sum_{j=1}^{n_i} -\beta_{i'lj'} \langle \mathbf{x}_{j'}^{i'}, \mathbf{x}_j^i \rangle \right) \quad (16)$$

Here $(i, l)$ is an index pair with $l \in Y_i$ and $0 < \alpha_{il} < C$.

Resorting to Eqs.(1), (9) and (16), the label set $Y$ for an unseen bag $X$ is determined as:

$$Y = \{l | \max_{\mathbf{x} \in X} f(\mathbf{x}, l) \geq 0, l \in \mathcal{Y}\}, \quad \text{where } f(\mathbf{x}, l) =$$

$$\sum_{i \in S_l} \left( \frac{\alpha_{il}}{n_i} \sum_{j=1}^{n_i} \langle \mathbf{x}_j^i, \mathbf{x} \rangle \right) - \sum_{i \in \overline{S}_l} \left( \sum_{j=1}^{n_i} \beta_{ilj} \langle \mathbf{x}_j^i, \mathbf{x} \rangle \right) + b_l \quad (17)$$

In addition, in case of $Y$ being empty, the label with highest (least negative) output is then assigned to $X$. This is actually the $T$-Criterion [3] used to treat learning problems with multi-label outputs.

To sum up, in training phase, parameters of $\mathrm{M}^3\mathrm{MIML}$ are learned by solving Problem 4 using the Franke and Wolfe's method[1]. In testing phase, the label set of unseen MIML example is determined via Eq.(17). To have the non-linear version of $\mathrm{M}^3\mathrm{MIML}$, it suffices to replace the dot products $\langle \cdot, \cdot \rangle$ with some kernel function $k(\cdot, \cdot)$ over $\mathcal{X} \times \mathcal{X}$.

# 4. Experiments

## 4.1. Experimental setup

In this section, the performance of $\mathrm{M}^3\mathrm{MIML}$ is evaluated with applications to two real-world MIML learning tasks. The first task is scene classification which was studied by Zhou and Zhang [32, 33] in their investigation of the MIML framework. The scene classification data contains

2,000 natural scene images collected from the COREL image collection and the Internet. All the possible class labels are *desert*, *mountains*, *sea*, *sunset* and *trees* and a set of labels is manually assigned to each image. Images belonging to more than one class comprise over 22% of the data set and the average number of labels per image is 1.24±0.44. Each image is represented as a bag of nine 15-dimensional instances using the SBN image bag generator [17], where each instance corresponds to an image patch.

In addition to scene classification, we have also tested $\mathrm{M}^3\mathrm{MIML}$ on text categorization problems. Specifically, the widely studied Reuters-21578 collection [22] is used in experiment [33]. The seven most frequent categories are considered. After removing documents whose label sets or main texts are empty, 8,866 documents are retained where only 3.37% of them are associated with more than one class labels. After randomly removing documents with only one label, a text categorization data set containing 2,000 documents is obtained. Around 15% documents with multiple labels comprise the resultant data set and the average number of labels per document is 1.15±0.37. Each document is represented as a bag of instances using the sliding window techniques [2], where each instance corresponds to a text segment enclosed in one sliding window of size 50 (overlapped with 25 words). "Function words" on the SMART stop-list [20] are removed from the vocabulary and the remaining words are stemmed. Instances in the bags adopt the *"Bag-of-Words"* representation based on term frequency [11, 22]. Without loss of effectiveness, dimensionality reduction is performed by retaining the top 2% words with highest document frequency [27]. Thereafter, each instance is represented as a 243-dimensional feature vector. Table 1 summarizes characteristics of both data sets.

$\mathrm{M}^3\mathrm{MIML}$ is compared with MIMLBOOST and MIMLSVM, both of which are set to take the best parameters as reported in [32]. Concretely, the number of boosting rounds for MIMLBOOST is set to 25, and Gaussian kernel with $\gamma = 0.2^2$ is used to implement MIMLSVM. For fair comparison, Gaussian kernel function is also used to yield the non-linear version of $\mathrm{M}^3\mathrm{MIML}$. Note that although parameters of MIMLBOOST and MIMLSVM are carefully chosen, on the other hand, those of $\mathrm{M}^3\mathrm{MIML}$ are not specifically tuned in any way. In particular, the values of $C$ (cost parameter as shown in Problem 4) and $\gamma$ are

---

[1]In each iterative round of the Franke and Wolfe's method, a LP problem with $M$ variables and $M + Q$ constraints needs to be resolved. Here $M = |A| + |B|$ is the number of dual variables optimized in Problem 4 and $Q$ is the number of possible class labels. It is well-known that the computational complexity of solving a standard LP problem with $p$ variables and $c$ constraints is $\mathcal{O}(p^3, cp^2)$ [4]. Considering that in general $Q \ll M$ holds, the computational complexity of solving Problem 4 with Franke and Wolfe's method is $\mathcal{O}(nM^3)$, here $n$ is the number of iterations involved.

[2]$k(\mathbf{x}, \mathbf{z}) = \exp(-\gamma \cdot \|\mathbf{x} - \mathbf{z}\|^2)$ for two vectors $\mathbf{x}$ and $\mathbf{z}$.

(a) *hamming loss*       (b) *one-error*       (c) *coverage*

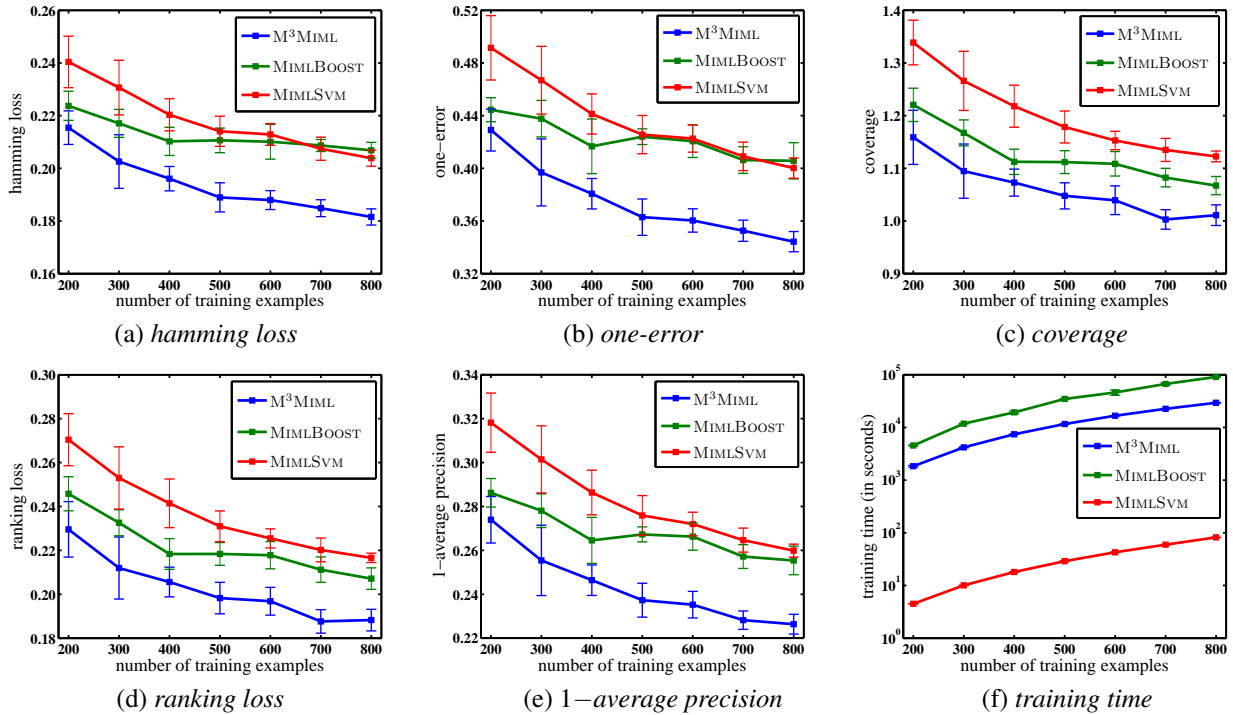(d) *ranking loss*     (e) *1−average precision*     (f) *training time*

**Figure 1. The performance of each compared algorithm (on the scene data) changes as the number of training examples increases. In each subfigure, the lower the curve the better the performance.**

both set to the default value of 1. Actually, in preliminary experiments, M$^3$MIML shows similar performance with $\gamma$ ranging from 0.6 to 1.4 by step 0.2.

Since MIML algorithms make multi-label predictions, the performance of each compared algorithm is evaluated according to five popular multi-label metrics, i.e. *hamming loss*, *one-error*, *coverage*, *ranking loss* and *average precision*. As for *average precision*, the *bigger* the value the better the performance. While for the other four metrics, the *smaller* the value the better the performance. Due to page limit, details on these metrics can be found in [21, 28].

Next, we will make comparative studies among MIML algorithms with two series of experiments. The first series concerns how the algorithms perform under different number of training examples. The other series investigates how the algorithms learn from data sets with varying percentage of examples associated with multiple labels.

### 4.2. Experimental results under varying training set size

To investigate the performance of each algorithm learned with different number of training examples, we create the training and test data as follows. For either of the scene or Reuters data, a test set is created by randomly choosing 1,000 examples from the original data set. The remaining

1,000 examples is then used to form the *potential training set*, where training set is formed by randomly picking up $N$ examples from the potential training set. In this paper, $N$ ranges from 200 to 800 with an interval of 100. For each value of $N$, ten different training sets are created by repeating the pickup procedure. The *average* test performance of each algorithm trained on the ten training sets is reported.

Figure 1 illustrates the performance of each compared algorithm on the scene classification data in terms of the five multi-label evaluation metrics as well as the time spent in training. For each algorithm, when the training set size is fixed, the average and standard deviation out of ten independent runs are depicted. Note that in Figure 1(e), we plot 1−*average precision* instead of *average precision* such that for all subfigures, the *lower* of one algorithm's curve the better its performance. Furthermore, the training time (measured in seconds) shown in Figure 1(f) is plotted in *log-linear* scale. Accordingly, Figure 2 reports the experimental results on the Reuters categorization data.

It is evident from Figures 1 and 2 that, on both data sets, M$^3$MIML *consistently* outperforms MIMLBOOST and MIMLSVM in terms of each evaluation metric. As expected, the performance of each algorithm improves as the number of training examples increases. It is interesting to see that, as more training examples become available, the *performance gap* between M$^3$MIML and its compared counter-

(a) *hamming loss*    (b) *one-error*    (c) *coverage*

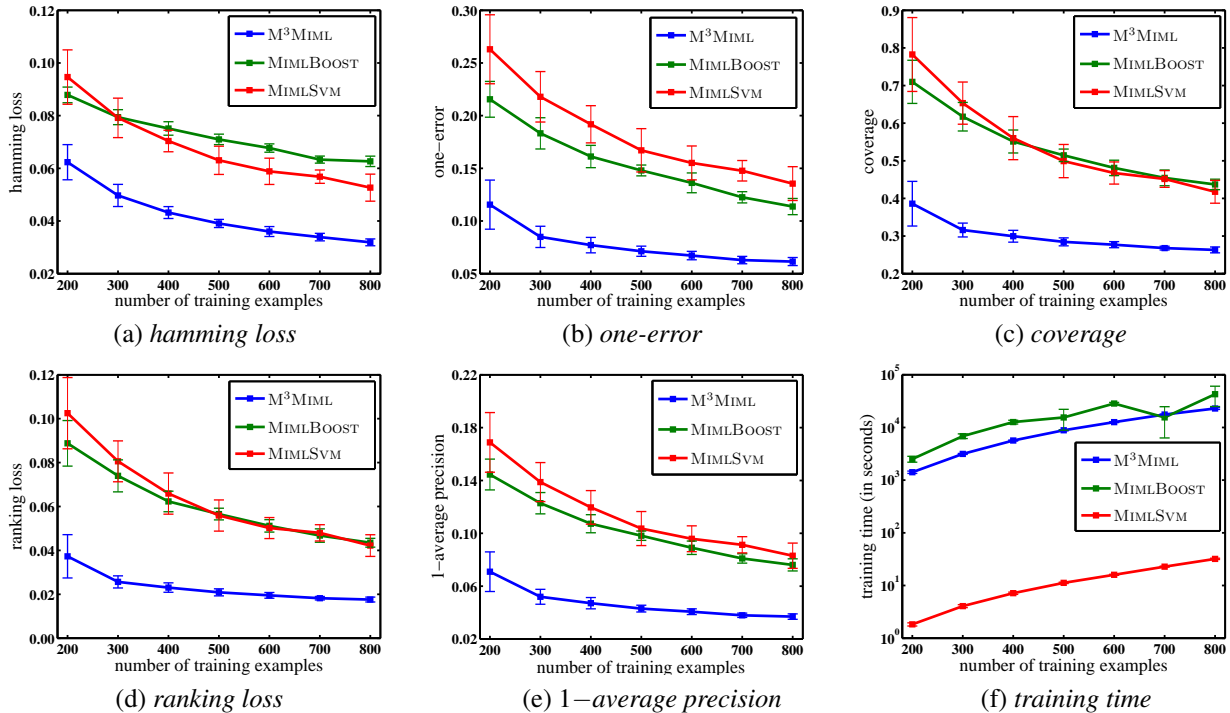(d) *ranking loss*    (e) 1−*average precision*    (f) *training time*

**Figure 2. The performance of each compared algorithm (on the Reuters data) changes as the number of training examples increases. In each subfigure, the lower the curve the better the performance.**

**Table 2. The win/tie/loss counts for** $\mathrm{M}^3\textsc{Miml}$ **against** $\textsc{MimlBoost}$ **and** $\textsc{MimlSvm}$ **with varying training set size.**

| Evaluation metric | $\mathrm{M}^3\textsc{Miml}$ against | | | |
| | $\textsc{MimlBoost}$ | | $\textsc{MimlSvm}$ | |
| | Scene | Reuters | Scene | Reuters |
|---|---|---|---|---|
| *hamming loss* | 7/0/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *one-error* | 7/0/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *coverage* | 7/0/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *ranking loss* | 7/0/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *average precision* | 7/0/0 | 7/0/0 | 7/0/0 | 7/0/0 |

parts tends to *increase* on the scene data but *decrease* on the Reuters data (while still remarkably large). Furthermore, when more and more training examples are used in classifier induction, the performance of $\textsc{MimlSvm}$ would gradually approaches that of $\textsc{MimlBoost}$ on both data sets.

Pairwise $t$-tests at 0.05 significance level are conducted to statistically measure the performance difference between the compared algorithms. The win/tie/loss counts based on pairwise $t$-test are reported in Table 2. For each metric, a win (or loss) is counted when $\mathrm{M}^3\textsc{Miml}$ is significantly better (or worse) than the compared algorithm on a specific training set size out of 10 runs. Otherwise, a tie is recorded.

As shown in Table 2, it is rather impressive that in terms of each multi-label metric, $\mathrm{M}^3\textsc{Miml}$ is statistically superior to $\textsc{MimlBoost}$ and $\textsc{MimlSvm}$ on both data sets under any number of training examples. As shown in Figures 1(f) and 2(f), although $\textsc{MimlSvm}$ runs greatly faster than both $\mathrm{M}^3\textsc{Miml}$ and $\textsc{MimlBoost}$, it has the worst performance among all the compared algorithms. In addition, $\textsc{MimlBoost}$ usually consumes 2 to 4 times of training period than $\mathrm{M}^3\textsc{Miml}$ in order to complete the learning procedure. The above results reveal that, compared to other $\textsc{Miml}$ algorithms, $\mathrm{M}^3\textsc{Miml}$ is a better choice for solving $\textsc{Miml}$ problems with balanced effectiveness and efficiency.

## 4.3. Experimental results under varying percentage of multi-label data

It is interesting to study the influence of the percentage of multi-label data (or equivalently the average number of labels per example) on the algorithms, so we do another series of experiments. We derive *seven* data sets from the scene data which contains around 22% images with multiple labels. By randomly removing some single-label images, we obtain a data set where 30% (or 40%, 50%, 60%, 70%) images belong to multiple classes simultaneously; by randomly removing some multi-label images, we obtain a data set where 10% (or 20%) images belong to multiple classes
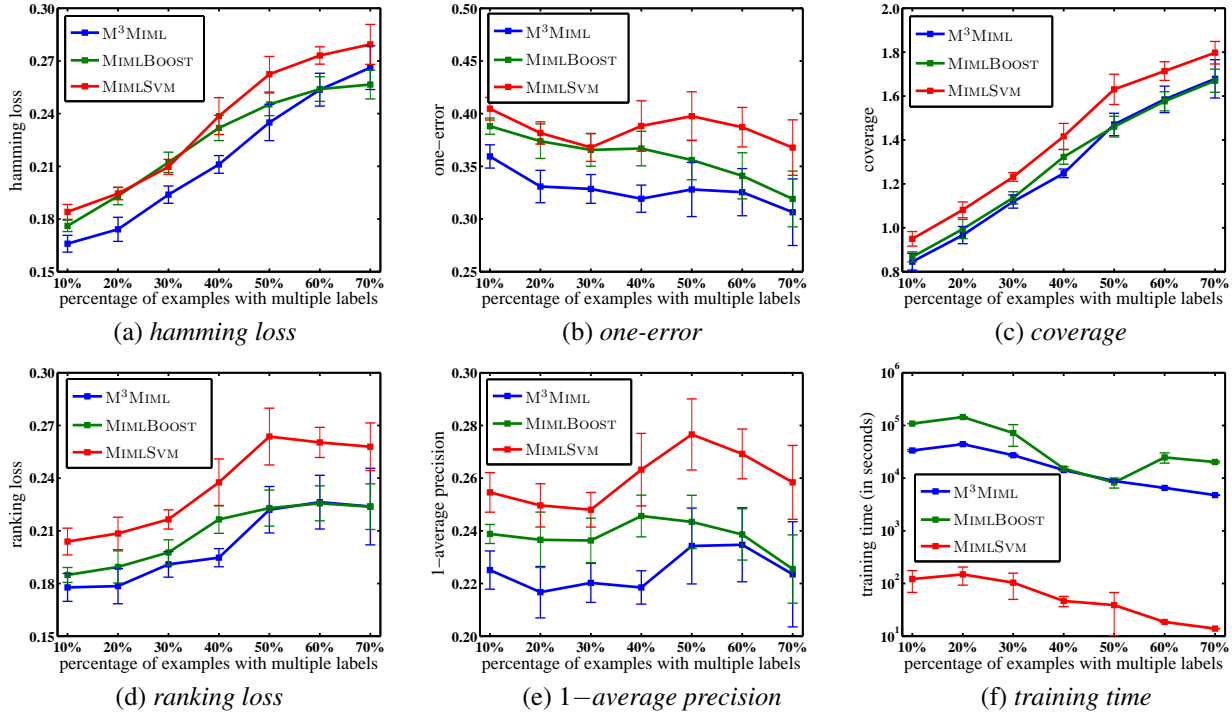
**Figure 3. The performance of each compared algorithm (on data sets derived from scene classification task) changes as the percentage of multi-label examples increases. In each subfigure, the lower the curve the better the performance.**

simultaneously. Note that the derived data set with high percentage of multi-label images would have a relatively small size, since during its generation process more single-label images are removed from the original scene data. Similarly, we also derive seven data sets with $P\%$ percentage of multi-label documents from the Reuters data, here $P\%$ ranges from $10\%$ to $70\%$ with an interval of $10\%$.

Ten times of hold-out tests are performed on each derived data set. In each hold-out test, the data set is randomly divided into two parts with equal size. Algorithms are trained on one part and then evaluated on the other part. Figure 3 illustrates the performance of each compared algorithm on data sets derived from the scene classification task. For each algorithm, when the percentage of multi-label examples is fixed, the average and standard deviation out of ten independent hold-out tests are depicted. The same as Subsection 4.2, we draw $1-average$ $precision$ instead of *average precision* and plot the training time (measured in seconds) in *log-linear* scale. Accordingly, Figure 4 reports the experimental results on data sets derived from the text categorization task.

It is evident from Figures 3 and 4 that, in most cases, $M^3M$IML is superior to MIMLBOOST and MIMLSVM. Specifically, on data sets derived from scene classification task, $M^3M$IML is indistinguishable from MIMLBOOST and

**Table 3. The win/tie/loss counts for $M^3M$IML against MIMLBOOST and MIMLSVM with varying percentage of multi-label examples.**

| Evaluation metric | $M^3M$IML against | | | |
|---|---|---|---|---|
| | MIMLBOOST | | MIMLSVM | |
| | Scene | Reuters | Scene | Reuters |
| *hamming loss* | 5/1/1 | 7/0/0 | 7/0/0 | 7/0/0 |
| *one-error* | 6/1/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *coverage* | 3/4/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *ranking loss* | 3/4/0 | 7/0/0 | 7/0/0 | 7/0/0 |
| *average precision* | 4/3/0 | 7/0/0 | 7/0/0 | 7/0/0 |

slightly outperforms MIMLSVM in terms of *coverage*. In terms of other evaluation metrics, $M^3M$IML performs consistently better than MIMLSVM, while the performance gap between $M^3M$IML and MIMLBOOST gradually *ceases* toward zero as the percentage of multi-label examples approaches $70\%$; On data sets derived from text categorization task, $M^3M$IML achieves consistently superior performance over MIMLBOOST and MIMLSVM in terms of all evaluation metrics. In addition, as the fraction of multi-label examples increases, the performance gap between $M^3M$IML and its compared counterparts tends to steadily *increase*.

(a) *hamming loss*  (b) *one-error*  (c) *coverage*

(d) *ranking loss*  (e) 1−*average precision*  (f) *training time*
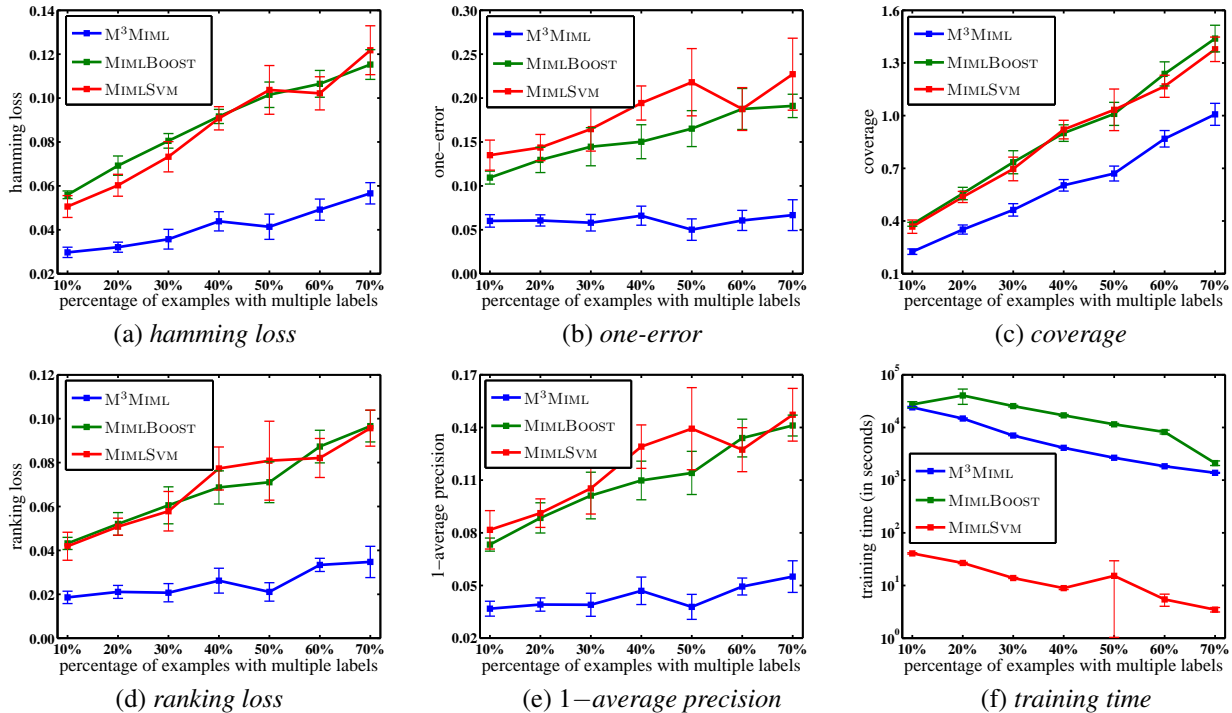
**Figure 4. The performance of each compared algorithm (on data sets derived from text categorization task) changes as the percentage of multi-label examples increases. In each subfigure, the lower the curve the better the performance.**

The same as Subsection 4.2, the win/tie/loss counts based on pairwise $t$-test are reported in Table 3. For each evaluation metric, a win (or loss) is counted when M$^3$MIML is significantly better (or worse) than the compared algorithm on a specific percentage of multi-label examples out of 10 hold-out runs. Otherwise, a tie is recorded.

As shown in Table 3, it is quite impressive that in terms of each multi-label metric, M$^3$MIML statistically outperforms MIMLSVM on both scene and text learning tasks. M$^3$MIML also performs statistically better than MIML-BOOST on the text categorization task. On the scene classification task, our approach is inferior to MIMLBOOST in terms of *hamming loss* in only one case, while it is superior or at least comparable to MIMLBOOST on the other metrics. The series of experiments reported in this subsection further confirm the superiority of our proposed approach.

## 5. Conclusion

In this paper, a novel MIML approach named M$^3$MIML is proposed. This method directly considers the connections between the instances and the labels of an MIML example through defining a specific margin on it. The corresponding maximum margin learning task is formulated as a QP problem and solved in its dual form with kernel imple-

mentation. Comparative studies with existing MIML algorithms are carried out with applications to scene classification and text categorization. Experimental results show that M$^3$MIML achieves significantly better performance than existing methods together with a good balance between effectiveness and efficiency.

Designing other kinds of MIML algorithms and perform comparative studies on more and larger MIML data sets are important issues for future work.

## Acknowledgements

## References

[1] R. A. Amar, D. R. Dooly, S. A. Goldman, and Q. Zhang. Multiple-instance learning of real-valued data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 3–10, Williamstown, MA, 2001.

[2] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances*

*in Neural Information Processing Systems 15*, pages 561–568. MIT Press, Cambridge, MA, 2003.

[3] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[5] K. Brinker, J. Fürnkranz, and E. Hüllermeier. A unified model for multilabel classification and ranking. In *Proceedings of the 17th European Conference on Artificial Intelligence*, pages 489–493, Riva del Garda, Italy, 2006.

[6] Y. Chen, J. Bi, and J. Z. Wang. MILES: multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006.

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

[8] Y. Chevaleyre and J.-D. Zucker. Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. Application to the mutagenesis problem. In *Lecture Notes in Artificial Intelligence 2056*, pages 204–214. Springer, Berlin, 2001.

[9] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, Cambridge, UK, 2000.

[10] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple-instance problem with axis-parallel rectangles. *Artificial Intelligence*, 39(1-2):31–71, 1997.

[11] S. T. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representation for text categorization. In *Proceedings of the 7th ACM International Conference on Information and Knowledge Management*, pages 148–155, Bethesda, MD, 1998.

[12] A. Elisseeff and J. Weston. Kernel methods for multi-labelled classification and categorical regression problems. Technical report, BIOwulf Technologies, 2001.

[13] M. Franke and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

[14] S. Gao, W. Wu, C.-H. Lee, and T.-S. Chua. A MFoM learning approach to robust multiclass multi-label text categorization. In *Proceedings of the 21st International Conference on Machine Learning*, pages 329–336, Banff, Canada, 2004.

[15] H. Kazawa, T. Izumitani, H. Taira, and E. Maeda. Maximal margin labeling for multi-topic text categorization. In *Advances in Neural Information Processing Systems 17*, pages 649–656. MIT Press, Cambridge, MA, 2005.

[16] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *Advances in Neural Information Processing Systems 10*, pages 570–576. MIT Press, Cambridge, MA, 1998.

[17] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proceedings of the 15th International Conference on Machine Learning*, pages 341–349, Madison, WI, 1998.

[18] A. McCallum. Multi-label text classification with a mixture model trained by EM. In *Working Notes of the AAAI'99 Workshop on Text Learning*, Orlando, FL, 1999.

[19] G. Ruffo. *Learning single and multiple decision trees for security applications*. PhD thesis, Department of Computer Science, University of Turin, Italy, 2000.

[20] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, Reading, Pennsylvania, 1989.

[21] R. E. Schapire and Y. Singer. Boostexter: a boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

[22] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[23] G. Tsoumakas and I. Katakis. Multi-label classification: an overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2007.

[24] N. Ueda and K. Saito. Parametric mixture models for multi-label text. In *Advances in Neural Information Processing Systems 15*, pages 721–728. MIT Press, Cambridge, MA, 2003.

[25] J. Wang and J.-D. Zucker. Solving the multiple-instance problem: a lazy learning approach. In *Proceedings of the 17th International Conference on Machine Learning*, pages 1119–1125, Stanford, CA, 2000.

[26] X. Xu and E. Frank. Logistic regression and boosting for labeled bags of instances. In *Lecture Notes in Computer Science 3056*, pages 272–281. Springer, Berlin, 2004.

[27] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Learning*, pages 412–420, Nashville, TN, 1997.

[28] M.-L. Zhang and Z.-H. Zhou. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338–1351, 2006.

[29] Q. Zhang and S. A. Goldman. EM-DD: an improved multiple-instance learning technique. In *Advances in Neural Information Processing Systems 14*, pages 1073–1080. MIT Press, Cambridge, MA, 2002.

[30] Q. Zhang, W. Yu, S. A. Goldman, and J. E. Fritts. Content-based image retrieval using multiple-instance learning. In *Proceedings of the 19th International Conference on Machine Learning*, pages 682–689, Sydney, Australia, 2002.

[31] Z.-H. Zhou. Multi-instance learning: A survey. Technical report, AI Lab, Department of Computer Science & Technology, Nanjing University, Nanjing, China, 2004.

[32] Z.-H. Zhou and M.-L. Zhang. Multi-instance multi-label learning with application to scene classification. In *Advances in Neural Information Processing Systems 19*, pages 1609–1616. MIT Press, Cambridge, MA, 2007.

[33] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li. A framework for learning with ambiguous objects. *CORR abs/0808.3231*, 2008.

[34] S. Zhu, X. Ji, W. Xu, and Y. Gong. Multi-labelled classification using maximum entropy method. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 274–281, Salvador, Brazil, 2005.