MIML：多示例多标记学习*

周 志 华[1]　　张 敏 灵[1,2]

[1]南京大学计算机软件新技术国家重点实验室，南京 210093

[2]河海大学计算机及信息工程学院，南京 210098

1. 引言

在利用机器学习技术解决实际问题时，常见的做法是先对真实对象进行特征提取，用一个特征向量来描述这个对象，这样就得到了一个示例（instance），然后把示例与该对象所对应的类别标记（label）关联起来，就得到了一个例子（example）。在拥有了一个较大的例子集合之后，就可以利用某种学习算法来学得示例空间与标记空间之间的一个映射，该映射可以预测未见示例（unseen instance）的标记。假设每个对象只有一个类别标记，那么形式化地来说，令$\mathcal{X}$为示例空间、$\mathcal{Y}$为标记空间，则学习任务是从数据集$\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$中学得函数$f: \mathcal{X} \to \mathcal{Y}$，其中$\boldsymbol{x}_i \in \mathcal{X}$为一个示例而 $y_i \in \mathcal{Y}$为示例$\boldsymbol{x}_i$所属的类别标记。在待学习对象具有明确的、单一的语义时，上面的学习框架已经取得了巨大的成功。

然而，真实世界的对象往往并不只具有唯一的语义，而是可能具有多义性的。例如，图 1(a)中的这幅图像，既可认为它属于“大象”这个类别，也可认为它属于“狮子”、“草地”、甚至“热带”、“非洲”；图 1(b)中的这个网页，既可认为它属于“体育”这个类别，也可因为贝克汉姆娱乐明星味十足而认为它属于“娱乐”类，甚至可以因为皇家马德里足球队出访的旅游、赚钱性质远大于比赛性质而认为它属于“旅游”类、“经济”类。由于这样的多义性对象不再只具有唯一的语义，这就使得前述的只考虑明确的、单一的语义的学习框架难以取得好的效果。

值得注意的是，对多义性对象进行学习是一个非常重要的问题。目前实际应用中遇到的很多难题都是由对象的多义性所造成的。例如在基于内容的图像检索中，众所周知的难题是“语义鸿沟”，即从图像的低层特征到高层语义之间存在难以逾越的障碍。笔者认为，这一语义鸿沟存在的本质原因之一，就是因为图像是一种多义性对象：同样的特征描述、不同的语义。试想，如果一幅图像只具有唯一的语义，那么哪里还会有什么语义鸿沟呢？笔者认为，要解决多义性造成的问题，首先需要从某个任务所涉及的众多“可能语义”中把某个具体的多义性对象所能具有的“合适语义”找出来，然后再根据具体的上下文从这些“合适语义”中确定当前的“语境语义”。而其中第一步，实际

*本文得到国家自然科学基金（60635030）、江苏省自然科学基金（BK2008018）和江苏省 333 高层次人才培养工程基金的资助

(a) 一幅图像

(b) 一个网页

图 1　多义性对象的两个例子

上就是要为对象赋予合适的类别标记子集，而不再是唯一的类别标记。针对这个目的，笔者提出了 MIML—— 即“多示例多标记学习”（Multi-Instance Multi-Label learning）这一学习框架[1][2]。本章将对这方面的研究进展做一个简介，主要内容及更详细的介绍可参见[2]。

2. MIML 框架

提出 MIML 的基本考虑，是多义性对象往往具有复杂的内涵，只用一个示例（即一个特征向量）来进行表示是一种过度简化，在表示阶段就丢失了有用的信息，后续的学习阶段将面临极大的困难。事实上，一个多义性对象往往可以用多个示例来描述。例如对图像来说，如果使用某种技术将图像划分为若干个区域，那么每个区域都可以用一个示例来描述，这样，一幅图像就可表示成多个示例组成的一个集合；对文档来说，如果使用某种技术将其划分为若干部分，例如不同的章节段落，那么每个部分都可以用一个示例来描述，这样，一个文档就可表示成多个示例的集合。考虑到多义性对象具有多个语义，我们所要学习的实际上就是从示例集合到类别标记集合上的一个映射。形式化地来说，令$\mathcal{X}$表示示例空间，$\mathcal{Y}$表示类别标记空间，则

多示例多标记学习：给定数据集$\{(X_1, Y_1), (X_2, Y_2), , \cdots, (X_m, Y_m)\}$，目标是学得$f : 2^{\mathcal{X}} \to 2^{\mathcal{Y}}$。其中，$X_i \subseteq \mathcal{X}$为一组示例$\{\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \cdots, \boldsymbol{x}_{i,n_i}\}$，$\boldsymbol{x}_{ij} \in \mathcal{X}$ $(j = 1, 2, \cdots, n_i)$，而$Y_i \subseteq \mathcal{Y}$为$X_i$的一组合适类别标记$\{y_{i1}, y_{i2}, \cdots, y_{i,l_i}\}$，$y_{ik} \in \mathcal{Y}$ $(k = 1, 2, \cdots, l_i)$。n_i为X_i中所含示例的个数，l_i为Y_i中所含标记的个数。

对多义性对象的学习，机器学习界在多标记学习（multi-label learning）[3] 这一框架（framework）下已经有一些研究。在这一框架下，每个对象由一个示例描述，该示例具有多个类别标记，学习的

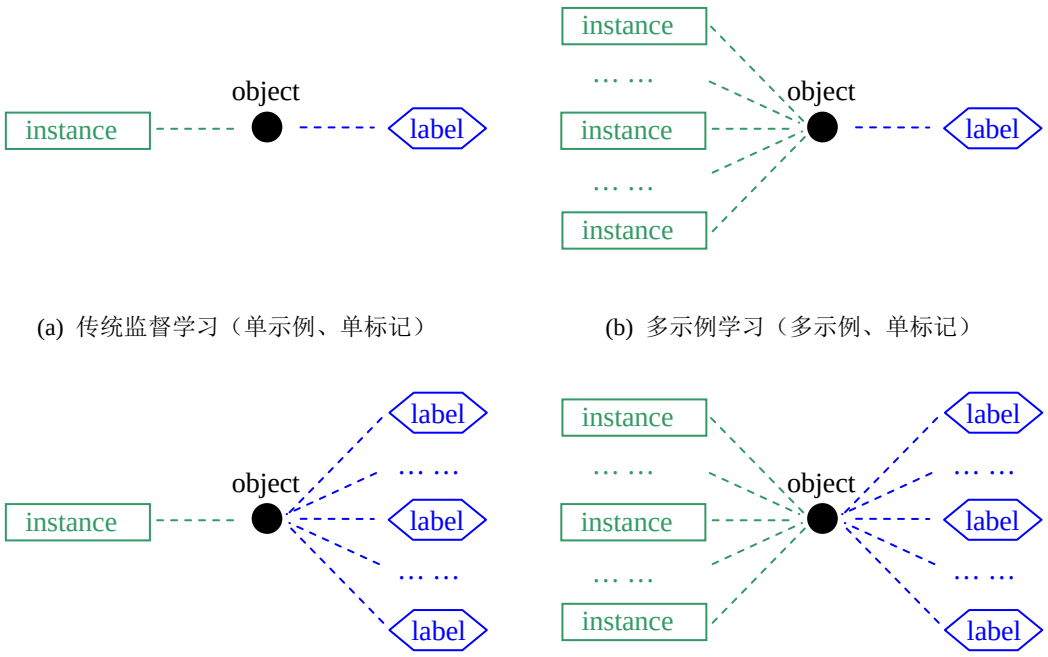
目的是将所有合适的类别标记赋予未见示例。形式化地来说，

多标记学习：给定数据集$\{(\boldsymbol{x}_1, Y_1), (\boldsymbol{x}_2, Y_2), , \cdots, (\boldsymbol{x}_m, Y_m)\}$，目标是学得 $f : \mathcal{X} \to 2^{\mathcal{Y}}$。其中，$\boldsymbol{x}_i \in \mathcal{X}$为一个示例，$Y_i \subseteq \mathcal{Y}$为$\boldsymbol{x}_i$的一组合适类别标记$\{y_{i1}, y_{i2}, \cdots, y_{i,l_i}\}$，$y_{ik} \in \mathcal{Y}$ $(k = 1, 2, \cdots, l_i)$。l_i为Y_i中所含类别标记的个数。

利用一个示例集合来描述一个对象，这一技术在多示例学习（multi-instance learning）[4] 框架下已有很多研究。在多示例学习框架下，每个对象由一组示例（即一个“示例包”）描述，该示例具有一个类别标记，学习的目的是预测未见示例包的类别标记。形式化地来说，

多示例学习：给定数据集$\{(X_1, y_1), (X_2, y_2), , \cdots, (X_m, y_m)\}$，目标是学得$f : 2^{\mathcal{X}} \to \mathcal{Y}$。其中，$X_i \subseteq \mathcal{X}$为一组示例$\{\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \cdots, \boldsymbol{x}_{i,n_i}\}$，$\boldsymbol{x}_{ij} \in \mathcal{X}$ $(j = 1, 2, \cdots, n_i)$，$y_i \in \mathcal{Y}$为与X_i的类别标记。n_i为X_i中所含示例的个数。

如果再考虑本章开头时所提到的传统监督学习（单示例、单标记）框架，那么我们就有了四种学习框架。图 2 给出了一个直观的对比。



(a) 传统监督学习（单示例、单标记）　　(b) 多示例学习（多示例、单标记）

(c) 多标记学习（单示例、多标记）　　(d) 多示例多标记学习

图 2　四种机器学习框架 [1][2]

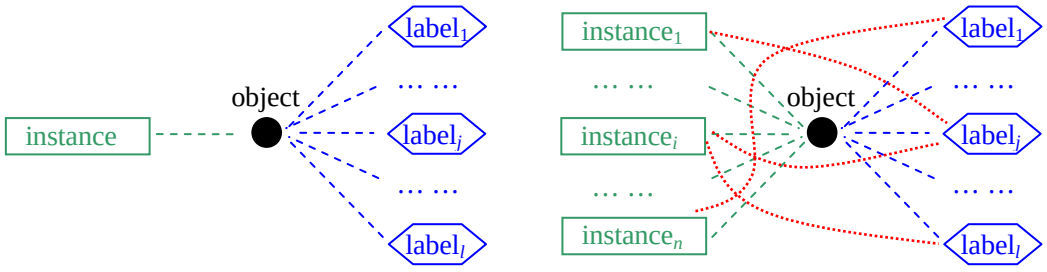
既然已经有了好几个学习框架，为什么我们还需要 MIML 呢？

首先，从表示能力上来看，传统监督学习框架可视为多示例学习框架或者多标记学习框架的特例，而传统监督学习框架、多示例学习框架以及多标记学习框架均可视为 MIML 的特例。换句话说，

其他三种框架下所覆盖的情形，MIML 框架也覆盖了；而 MIML 所覆盖的一些情形，其他三种框架未必能够覆盖。在对真实世界学习问题求解时，好的表示往往至关重要，在一定程度上甚至直接决定了学习的成败。采用了合适的表示，有可能更好地捕获学习对象所含的信息，从而使得学习任务变得容易完成；采用不合适的表示，有可能已经丢失了重要信息，从而使得学习任务变得极其困难。使用 MIML 来对多义性对象进行表示，有助于明示示例与类别标记之间的联系，从而有助于学习任务的解决。

实际上，多标记学习框架所面临的困难，很大程度上是因其用一个示例来描述多义性对象所造成的。如前所述，在多标记学习框架下，学习目标是$f:\mathcal{X}\to 2^{\mathcal{Y}}$，从图 2(c)可以看出，这是一个一对多映射（从一个示例到多个类别标记），而一对多映射并不是一个合式函数！与多标记学习框架相比，MIML 框架更加合理一些，虽然多对多映射看起来比一对多映射复杂，但是多对多映射毕竟是一个合式函数，具有很多一对多映射所不具备的数学性质，这就使得学习任务可能得以较好地完成。

值得一提的是，与简单地对合适标记进行预测相比，了解一个对象为什么具有某个类别标记可能在某些场合具有更重要的意义，而 MIML 为此提供了一种可能性。如图 3 所示，与图 3(a)中难以了解类别标记的原因不同，在图 3(b)中，我们可能可以知道，对象具有label_1的原因是因为其含有$\mathrm{instance}_n$，具有label_l的原因是因为其含有$\mathrm{instance}_i$，而该对象同时包含$\mathrm{instance}_1$与$\mathrm{instance}_i$则使得其具有label_j。

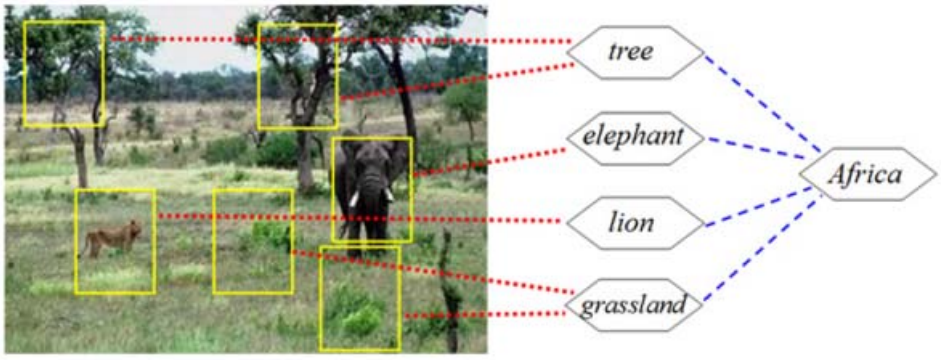


(a) 一个具有 l 个类别标记的对象　　(b) 示例与类别标记之间的关系

图 3　MIML 为理解示例与标记之间的关系提供了可能[2]

除了多标记学习问题，MIML 还有助于涉及到复杂概念的单标记学习问题的解决。如图 4(a)所示，对于“非洲”这个语义内涵丰富的概念，与之对应的图像在表现形式上具有很大的差异性。因此，对于图 4(a)左上角所示的图片，将其正确地分类为“非洲”是一个困难的问题。然而如图 4(b)所示，如果我们能够充分利用该图片包含的树木、狮子、大象、草地等“子概念”，由于这些子概念相对而言更加明确且易于学习，因此我们先利用 MIML 学习出子概念，然后再利用这些子概念导出“非洲”这一高层概念，这可能比直接对“非洲”进行学习要容易很多。

(a) “非洲”是一个复杂、难以学习的高层概念



b) 利用 MIML 学习“子概念”，再由“子概念”导出复杂高层概念

图 4　MIML 有助于学习复杂高层概念[2]

为了发挥 MIML 框架的能力，就需要设计出有效的算法。为此，我们基于退化策略提出了 MIMLBOOST 算法和 MIMLSVM 算法[1][2]，基于正则化（regularization）机制提出了 D-MIMLSVM 算法[2]和 M^{3}MIML 算法 [5]。本章第 3 节将对这些工作进行简介。

如果能够直接接触原始数据对象，那么我们可以利用 MIML 进行建模而获取更多的有用信息，但在不少应用中，尤其是数据挖掘应用中，我们往往只能得到第二手数据，这些数据已由他人进行了特征提取并将一个对象表示为一个特征向量。在这种情况下，虽然不能利用 MIML 表示的效力，但是 MIML 学习仍然能发挥重要的作用。我们提出了 INSDIF 算法[2][6]，将单示例多标记样本转化为 MIML 样本进行学习以获得更好的性能。本章第 4 节将对此进行简介。

如前所述，MIML 框架还有助于对复杂高层概念的学习，为此我们提出了 SUBCOD 算法[2]，通过发现目标概念的子概念来将单标记样本转化为多标记样本，从而利用 MIML 的帮助提高学习性能。本章第 5 节将对此进行简介。

3. MIML 学习算法

3.1 基于退化策略的 MIML 学习算法

如第 2 节所述，传统监督学习是多示例学习或者多标记学习的特例，而传统监督学习、多示例学习以及多标记学习均是多示例多标记学习的特例。因此，一种简单的 MIML 求解策略是以多示例学习或者多标记学习为桥梁，将 MIML 问题退化为传统监督学习问题进行求解。

- 策略 1 - 以多示例学习为桥梁：多示例多标记学习的目标是学得$f : 2^{\mathcal{X}} \to 2^{\mathcal{Y}}$，该目标可以简化为一个多示例学习问题，即学习相应的目标函数$f_{MIL} : 2^{\mathcal{X}} \times \mathcal{Y} \to \{-1, +1\}$。此时，对于任意的$y \in \mathcal{Y}$，$f_{MIL}(X_i, y) = +1$当且仅当$y \in Y_i$否则$f_{MIL}(X_i, y) = -1$。基于此，给定新样本$X^*$，与之对应的类别标记集合为$Y^* = \{y | sign[f_{MIL}(X^*, y)] = +1\}$。该多示例学习问题还可进一步转化为传统监督学习问题，其目标是学得$f_{SISL} : \mathcal{X} \times \mathcal{Y} \to \{-1, +1\}$并指定如何由$f_{SISL}(\boldsymbol{x}_{ij}, y)$ $(j = 1, 2, \cdots, n_i)$的取值确定$f_{MIL}(X_i, y)$的取值。此时，对于任意的$y \in \mathcal{Y}$，$f_{SISL}(\boldsymbol{x}_{ij}, y) = +1$当且仅当$y \in Y_i$否则$f_{SISL}(\boldsymbol{x}_{ij}, y) = -1$。特别地，我们采用文献[7]中的方法将多示例学习问题转化为传统监督学习问题，即$f_{MIL}(X_i, y) = sign[\sum_{j=1}^{n_i} f_{SISL}(\boldsymbol{x}_{ij}, y)]$。值得注意的是，上述转化过程也可采用其他方法实现。

- 策略 2 - 以多标记学习为桥梁：多示例多标记学习的目标是学得$f : 2^{\mathcal{X}} \to 2^{\mathcal{Y}}$，该目标可以简化为一个多标记学习问题，即学习相应的目标函数$f_{MLL} : \mathcal{Z} \to 2^{\mathcal{Y}}$。此时，对于任意的$\boldsymbol{z}_i \in \mathcal{Z}$，$f_{MLL}(\boldsymbol{z}_i) = f(X_i)$当且仅当$\boldsymbol{z}_i = \phi(X_i)$，$\phi : 2^{\mathcal{X}} \to \mathcal{Z}$。基于此，给定新样本$X^*$，与之对应的类别标记集合为$Y^* = f_{MLL}(\phi(X^*))$。该多标记学习问题还可进一步转化为传统监督学习问题，其目标是学得$f_{SISL} : \mathcal{Z} \times \mathcal{Y} \to \{-1, +1\}$。此时，对于任意的$y \in \mathcal{Y}$，$f_{SISL}(\boldsymbol{z}_i, y) = +1$当且仅当$y \in Y_i$否则$f_{SISL}(\boldsymbol{z}_i, y) = -1$。基于此，$f_{MLL}(\boldsymbol{z}_i) = \{y | f_{SISL}(\boldsymbol{z}_i, y) = +1\}$。本文采用文献[8]中的“构造性聚类(constructive clustering)”方法实现所需的映射函数ϕ。值得注意的是，上述转化过程也可采用其他方法实现。

基于策略 1，我们设计了多示例多标记学习算法 MIMLBOOST[1][2]。该算法以多示例学习为桥梁，将 MIML 问题退化为传统监督学习问题求解。首先，MIMLBOOST 算法将每个多示例多标记样本(X_i, Y_i)转化为$|\mathcal{Y}|$个多示例单标记样本$\{([X_i, y], \Phi[X_i, y]) | y \in \mathcal{Y}\}$。其中，$[X_i, y]$包含$n_i$个示例$\{(\boldsymbol{x}_{i1}, y), \cdots, (\boldsymbol{x}_{i,n_i}, y)\}$，每个示例由$X_i$所含示例与类别标记$y$拼接而来。此外，$\Phi[X_i, y] = +1$当且仅当$y \in Y_i$，否则$\Phi[X_i, y] = -1$。上述转化过程完成后，MIMLBOOST 算法利用文献[7]中的 MIBOOSTING 算法对转化得到的多示例学习问题进行求解。MIBOOSTING 算法假设包中每个示例对包的类别标记具有独立且同等的影响，将多示例学习问题进一步退化为传统监督学习问题求解。需要

说明的是，基于策略 1 还可以设计出其它类型的 MIML 学习算法。

基于策略 2，我们设计了另一种多示例多标记学习算法 MIMLSVM [1][2]。该算法以多标记学习为桥梁，将多示例多标记学习问题退化为传统监督学习问题求解。首先，MIMLSVM 算法将每个多示例多标记样本(X_i, Y_i)转化为一个单示例多标记样本$(\phi(X_i), Y_i)$。其中，函数ϕ基于构造性聚类[8]将每个包X_i转化为一个示例$\boldsymbol{z}_i$。该聚类过程将集合$\Lambda = \{X_1, \cdots, X_m\}$中的每个包看作一个原子对象，基于 Hausdorff 距离度量包之间的距离并利用 *k*-medoids 算法将集合Λ划分为若干个聚类。此时，$\boldsymbol{z}_i$的每一维即为包X_i与各聚类中心之间的距离。上述转化过程完成后，MIMLSVM 算法利用文献[9]中的 MLSVM 算法对转化得到的多标记学习问题进行求解。该算法针对每个可能的类别标记构造一个二类学习问题，从而将多标记学习问题进一步退化为传统监督学习问题求解。需要说明的是，基于策略 2 还可以设计出其它类型的 MIML 学习算法。

为了考察 MIMLBOOST 与 MIMLSVM 算法的性能，我们在场景分类以及文本分类问题上进行了实验。结果表明[1][2]，在 MIML 框架下对这些学习问题进行建模和学习，可以取得比使用多标记学习框架和多示例学习框架更好的结果。MIMLBOOST 算法与 MIMLSVM 算法的具体细节请参见[1][2]。

3.2 基于正则化的 MIML 学习算法

基于正则化机制以及最大化间隔（margin）策略，我们提出了两种多示例多标记学习算法，即 D-MIMLSVM [2]以及 M^{3}MIML [5]。本小节将首先介绍 D-MIMLSVM 算法，然后介绍 M^{3}MIML 算法。

3.2.1 D-MIMLSVM 算法

假设类别标记集合$\mathcal{Y}$中共含有T个不同的类别标记，即$|\mathcal{Y}| = T$。此外，假设分类系统由T个函数$\boldsymbol{f} = (f_1, f_2, \cdots, f_T)$构成。其中，每个函数是一个由示例集合到实值空间的映射，即$f_t : 2^{\mathcal{X}} \to \mathcal{R},\ t = 1, \cdots, T$。此时，给定新样本$X^* \subseteq \mathcal{X}$，分类系统预测与之对应的类别标记集合为$Y^* = \{t | f_t(X^*) > 0,\ t \in \mathcal{Y}\}$，其中$f_t(X^*)$用于判断$\mathcal{Y}$中第$t$个类别标记是否属于$X^*$。值得注意的是，如果我们将包$X_i$中的每个示例$\boldsymbol{x}_{ij}$看作一个仅含一个示例的包$\{\boldsymbol{x}_{ij}\}$，那么我们就可以用$\boldsymbol{f}(\boldsymbol{x}_{ij})$表示分类系统在示例$\boldsymbol{x}_{ij}$上的输出(即$\boldsymbol{f}(\{\boldsymbol{x}_{ij}\})$)。

给定多示例多标记训练集$\{(X_1, Y_1), \cdots, (X_m, Y_m)\}$，我们首先从两个方面来考察分类系统$\boldsymbol{f}$的性质。一方面，我们要求$\boldsymbol{f}$在每个包$X_i$上预测的类别标记集合与其真实类别标记集合$Y_i$保持一致。另一方面，我们还考察$\boldsymbol{f}$在包$X_i$上的输出与$\boldsymbol{f}$在$X_i$所含示例上的输出之间的关系。具体来说，我们按如下方式定义分类系统在训练集上的损失函数V：

$$V(\{X_i\}_{i=1}^m, \{Y_i\}_{i=1}^m, \boldsymbol{f}) = \frac{1}{mT} \sum_{i=1}^{m} \sum_{t=1}^{T} (1 - y_{it} f_t(X_i))_+$$

$$+\frac{\lambda}{mT}\sum_{i=1}^{m}\sum_{t=1}^{T} l\left(f_t(X_i), \max_{j=1,\cdots,n_i} f_t(\boldsymbol{x}_{ij})\right) \tag{1}$$

由上可见，函数V包含两个由正则化参数λ进行平衡的损失项。具体来说，第一个损失项基于 hinge loss 考察包X_i的真实标记集合(即Y_i)与系统输出$\boldsymbol{f}(X_i)$之间的差异。其中，$y_{it}=+1$当且仅当$t\in Y_i$否则$y_{it}=-1$，且函数$(z)_+=\max(0,z)$。V中第二个损失项用于度量$\boldsymbol{f}$在包X_i上输出与其在X_i所含示例上的输出之间的差异。基于多示例学习中常见的假设，我们将$\boldsymbol{f}$在X_i的各个示例上的最大输出，即$\max_{j=1,\cdots,n_i} f_t(\boldsymbol{x}_{ij})$，看作$\boldsymbol{f}$在$X_i$上的最终输出。此外，用于度量$f_t(X_i)$与$\max_{j=1,\cdots,n_i} f_t(\boldsymbol{x}_{ij})$之间差异的函数$l(v_1,v2)$可以有多种实现方式，例如采用$l_1$损失函数：$l(v_1,v_2)=|v_1-v_2|$。

设k为某种定义在包空间 (即$2^{\mathcal{X}}$)上的核函数，如 set kernel [10]。假设$\boldsymbol{f}$中的每个成员函数f_t都是一个线性函数，即$f_t(\boldsymbol{x})=\langle \boldsymbol{w}_t,\phi(\boldsymbol{x})\rangle$。其中，$\phi$是由核函数$k$导出的属性映射，而$\langle\cdot,\cdot\rangle$是由核函数$k$导出的再生核希尔伯特空间(RKHS) $\mathcal{H}$所具有的点积运算。进一步地，记$\boldsymbol{w}_0=\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t$为各线性函数对应的平均权值向量。为了考察每个包所含各个类别标记之间的关系，受文献[11]中工作的启发，我们通过同时最小化$\sum_{i=1}^{T}||\boldsymbol{w}_i-\boldsymbol{w}_0||^2$以及$||\boldsymbol{w}_0||^2$的方式来实现该目标。然而，考虑到$\sum_{i=1}^{T}||\boldsymbol{w}_i-\boldsymbol{w}_0||^2=\sum_{t=1}^{T}||\boldsymbol{w}_t||^2-T||\boldsymbol{w}_0||^2$，同时最小化$\sum_{i=1}^{T}||\boldsymbol{w}_i-\boldsymbol{w}_0||^2$以及$||\boldsymbol{w}_0||^2$可以简化为同时最小化$\sum_{t=1}^{T}||\boldsymbol{w}_t||^2$以及$||\boldsymbol{w}_0||^2$。结合上述考虑以及式(1)，我们定义如下的 MIML 学习正则化框架：

$$\min_{\boldsymbol{f}\in\mathcal{H}}\frac{1}{T}\sum_{t=1}^{T}||f_t||_{\mathcal{H}}^2+\mu||\frac{\sum_{t=1}^{T}f_t}{T}||_{\mathcal{H}}^2+\gamma V(\{X_i\}_{i=1}^m,\{Y_i\}_{i=1}^m,\boldsymbol{f}) \tag{2}$$

其中，正则化参数γ用于平衡系统的模型复杂性和经验误差，而正则化参数μ用于平衡系统在各个类别上输出的差别和共性。直观上看，参数μ的取值越大，则系统更关注在各个类上输出的共性，反之则更关注在各个类上输出的差异。

基于上述设置，我们可以证明下述的表示定理：对于式(2)中的优化问题而言，其最优解$\boldsymbol{f}$必定具有如下的表示形式：

$$f_t(\boldsymbol{x})=\sum_{i=1}^{m}\left(\alpha_{t,i0}k(\boldsymbol{x},X_i)+\sum_{j=1}^{n_i}\alpha_{t,ij}k(\boldsymbol{x},\boldsymbol{x}_{ij})\right) \tag{3}$$

其中，$\alpha_{t,i0},\alpha_{t,ij}\in\mathcal{R}$。

假设采用l_1损失函数度量$l(v_1,v_2)$，则式(2)中的优化问题可以重写为：

$$\min_{\boldsymbol{f}\in\mathcal{H},\boldsymbol{\xi},\boldsymbol{\delta}}\frac{1}{T}\sum_{t=1}^{T}||f_t||_{\mathcal{H}}^2+\mu||\frac{\sum_{t=1}^{T}f_t}{T}||_{\mathcal{H}}^2+\frac{\gamma}{mT}\boldsymbol{\xi}'\mathbf{1}+\frac{\gamma\lambda}{mT}\boldsymbol{\delta}'\mathbf{1} \tag{4}$$

$$
\begin{aligned}
\text{s.t.} \quad & y_{it} f_t(X_i) \geq 1 - \xi_i \\
& \boldsymbol{\xi} \geq \mathbf{0} \\
& -\delta_{it} \leq f_t(X_i) - \max_{j=1,\cdots,n_i} f_t(\boldsymbol{x}_{ij}) \leq \delta_{it}
\end{aligned}
$$

其中，$\boldsymbol{\xi} = [\xi_{11}, \xi_{12}, \cdots, \xi_{it}, \cdots, \xi_{mT}]'$是与每个训练包在各个概念类上的误差对应的松弛变量，$\boldsymbol{\delta} = [\delta_{11}, \delta_{12}, \cdots, \delta_{it}, \cdots, \delta_{mT}]'$而$\mathbf{0}$与$\mathbf{1}$分别代表全零和全 1 向量。

不失一般性，假设我们按照$(X_1, \cdots, X_m, \boldsymbol{x}_{11}, \cdots, \boldsymbol{x}_{1,n_1}, \cdots, \boldsymbol{x}_{m,1}, \cdots, \boldsymbol{x}_{m,n_m})$的顺序将所有训练包以及训练包中的示例进行排列。此时，每个对象(包或者示例)可以用如下的函数$\mathcal{I}$进行索引：

$$
\forall\ 1 \leq i \leq m,\ 1 \leq j \leq n_i
$$

$$
\begin{cases}
\mathcal{I}(X_i) = i \\
\mathcal{I}(\boldsymbol{x}_{ij}) = m + \sum_{l=1}^{i-1} n_l + j
\end{cases}
$$

利用上述排序，我们可以得到定义在所有对象(所有训练包及其包中示例)上的大小为$(m+n) \times (m+n)$的核矩阵$\boldsymbol{K}$。设$\boldsymbol{k}_i$表示核矩阵$\boldsymbol{K}$的第i列，基于式(3)所示的表示定理可得$f_t(X_i) = \boldsymbol{k}'_{\mathcal{I}(X_i)}\boldsymbol{\alpha}_t + b_t$且$f_t(\boldsymbol{x}_{ij}) = \boldsymbol{k}'_{\mathcal{I}(\boldsymbol{x}_{ij})}\boldsymbol{\alpha}_t + b_t$。其中，$b_t$是与每个概念类对应的偏置(bias)而$\boldsymbol{\alpha}_t = [\alpha_{t,10}, \cdots, \alpha_{t,m0}, \alpha_{t,11}, \cdots, \alpha_{t,(1,n_1)}, \cdots, \alpha_{t,m1}, \cdots, \alpha_{t,(m,n_m)}]'$。

基于上述描述，优化问题式(4)可以重写为：

$$
\begin{aligned}
\min_{\boldsymbol{A},\boldsymbol{\xi},\boldsymbol{\delta},\boldsymbol{b}} \quad & \frac{1}{2T}\sum_{t=1}^{T} \boldsymbol{\alpha}_t' \boldsymbol{K} \boldsymbol{\alpha}_t + \frac{\mu}{T^2}\mathbf{1}'\boldsymbol{A}'\boldsymbol{A}\mathbf{1} + \frac{\gamma}{mT}\boldsymbol{\xi}'\mathbf{1} + \frac{\gamma\lambda}{mT}\boldsymbol{\delta}'\mathbf{1} \\
\text{s.t.} \quad & y_{it}\left(\boldsymbol{k}'_{\mathcal{I}(X_i)}\boldsymbol{\alpha} + b_t\right) \geq 1 - \xi_i \\
& \boldsymbol{\xi} \geq \mathbf{0} \\
& \boldsymbol{k}'_{\mathcal{I}(\boldsymbol{x}_{ij})}\boldsymbol{\alpha}_t - \delta_{it} \leq \boldsymbol{k}'_{\mathcal{I}(X_i)}\boldsymbol{\alpha}_t \\
& \boldsymbol{k}'_{\mathcal{I}(X_i)}\boldsymbol{\alpha}_t - \max_{j=1,\cdots,n_i} \boldsymbol{k}'_{\mathcal{I}(\boldsymbol{x}_{ij})}\boldsymbol{\alpha}_t \leq \delta_{it}
\end{aligned}
\tag{5}
$$

其中，$\boldsymbol{b} = [b_1, b_2, \cdots, b_T]'$且$\boldsymbol{A} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \cdots, \boldsymbol{\alpha}_T]$。

由于优化问题式(5)中含有非凸约束条件，因此该优化问题并不是常见的凸优化问题。然而另一方面，考虑到优化问题式(5)中的非凸约束条件是由两个凸函数的差所构成的，对于这种形式的优化问题我们可以利用 CCCP (Constrained Concave-Convex Procedure) [12][13]这一成熟的迭代优化策略对其求解。已有的研究结果表明，CCCP 方法可以保证收敛到相应优化问题的局部最优解 [14]，有时甚至可以收敛到全局最优解 [15]。

此外，考虑到对于每一个可能的概念类，属于该概念的样本数通常远少于不属于该概念的样本

数，我们还在 D-MIMLSVM 算法中引入了相应的机制来处理类别不平衡问题。另一方面，为了提高基于 CCCP 策略求解式(5)的优化问题时每一步迭代过程的效率，我们还使用割平面（cutting plane）算法提高效率。具体细节请参见[2]。

3.2.2 M^{3}MIML 算法

给定多示例多标记样本(X_i, Y_i)，我们用$\vec{Y}_i$表示相应的类别向量。其中，当$l \in Y_i$时，其第l维$\vec{Y}_i(l)$取值为$+1$否则取值为-1。M^{3}MIML 算法假设分类系统包含T个线性模型$\{(\boldsymbol{w}_l, b_l)|l \in \mathcal{Y}\}$，每个线性模型对应于一个可能的概念类。其中，$\boldsymbol{w}_l \in \mathcal{R}^d$与$b_l \in \mathcal{R}$分别为与第$l$类对应的权值向量和偏置。

M^{3}MIML 算法假设样本在第l类上的输出由其所含示例在对应模型$(\boldsymbol{w}_l, b_l)$上的最大输出决定[16][17]。因此，给定新样本X^*，对应的类别标记集合为$Y^* = \{l|\max_{\boldsymbol{x}\in X^*}(\langle \boldsymbol{w}_l, \boldsymbol{x}\rangle + b_l) \geq 0, l \in \mathcal{Y}\}$。基于样本在每一个类上的输出，M^{3}MIML 算法定义(X_i, Y_i)在第l类上的间隔为：

$$\frac{\vec{Y}_i(l) \cdot \max_{\boldsymbol{x}\in X_i}(\langle \boldsymbol{w}_l, \boldsymbol{x}\rangle + b_l)}{||\boldsymbol{w}_l||} \tag{6}$$

其中，$\langle \cdot, \cdot \rangle$用于计算向量之间的点积。M^{3}MIML 算法进一步假设模型在(X_i, Y_i)上的间隔由各个类上间隔的最小值确定，并且模型在整个训练集上的间隔(记为Δ)由所有样本间隔的最小值确定。在理想情况下，假设模型可以对训练集中的所有样本正确分类。则$\forall\ i \in \{1, 2, \ldots, m\},\ l \in \mathcal{Y}$，存在模型$\{(\boldsymbol{w}_l, b_l)|l \in \mathcal{Y}\}$使得下式成立：

$$\vec{Y}_i(l) \cdot \max_{\boldsymbol{x}\in X_i}(\langle \boldsymbol{w}_l, \boldsymbol{x}\rangle + b_l) \geq 1 \tag{7}$$

并且对于任意的$l \in \mathcal{Y}$，最少存在一个$i \in \{1, 2, \ldots, m\}$使得式(7)取等号。由此，$\Delta$即为：

$$\begin{aligned}\Delta &= \min_{1\leq i\leq m} \min_{l\in\mathcal{Y}} \frac{\vec{Y}_i(l)\cdot\max_{\boldsymbol{x}\in X_i}(\langle \boldsymbol{w}_l,\boldsymbol{x}\rangle+b_l)}{||\boldsymbol{w}_l||} \\ &= \min_{l\in\mathcal{Y}} \min_{1\leq i\leq m} \frac{\vec{Y}_i(l)\cdot\max_{\boldsymbol{x}\in X_i}(\langle \boldsymbol{w}_l,\boldsymbol{x}\rangle+b_l)}{||\boldsymbol{w}_l||} \\ &= \min_{l\in\mathcal{Y}} \frac{1}{||\boldsymbol{w}_l||}\end{aligned} \tag{8}$$

理想情况下，M^{3}MIML 算法求解的最大化间隔问题为：

$$\min_{\{(\boldsymbol{w}_l, b_l)|l\in\mathcal{Y}\}} \frac{1}{2} \max_{l\in\mathcal{Y}} ||\boldsymbol{w}_l||^2 \tag{9}$$

$$\text{s.t.: } \forall\ i \in \{1, 2, \ldots, m\},\ l \in \mathcal{Y} \text{ such that}$$

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

其中，$\overline{Y}_i$表示Y_i在$\mathcal{Y}$中的补集。式(7)所示的不等式按照$\vec{Y}_i(l)$取值为$+1$或-1两种不同的情况在式(9)中对应于不同的约束条件。而最大化式(8)所示的间隔$\min\limits_{l\in\mathcal{Y}} \frac{1}{||\boldsymbol{w}_l||}$相当于最小化$\max\limits_{l\in\mathcal{Y}} ||\boldsymbol{w}_l||^2$，对应于式(9)中的优化目标。

式(9)在优化目标和约束条件中均涉及max函数，难以使用优化技术直接寻优。为此，我们利用如下所示的不等式在一定程度上放宽优化目标和约束条件：

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

$$\max_{\boldsymbol{x}\in X_i} (\langle \boldsymbol{w}_l, \boldsymbol{x}\rangle + b_l) \geq \frac{\sum_{j=1}^{n_i} (\langle \boldsymbol{w}_l, \boldsymbol{x}_{ij}\rangle + b_l)}{n_i} \tag{10}$$

与此同时，引入松弛变量以反映真实世界中训练集不可完全正确分类的情况，M^{3}MIML 算法对应的优化问题即转化为：

$$\min_{\{\boldsymbol{W},\boldsymbol{b},\boldsymbol{\Xi},\boldsymbol{\Theta}\}} \ \frac{1}{2}\sum_{l=1}^{T} \|\boldsymbol{w}_l\|^2 + C\sum_{l=1}^{T} \left(\sum_{i\in S_l} \xi_{il} + \sum_{i\in \overline{S}_l} \sum_{j=1}^{n_i} \theta_{ilj}\right) \tag{11}$$

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

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

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

其中$S_l = \{i | 1 \leq i \leq m,\ l \in Y_i\}$是具有标记$l$的样本对应的索引集合。相应地，$\overline{S}_l = \{i | 1 \leq i \leq m,\ l \in \overline{Y_i}\}$为不具有标记$l$的样本对应的索引集合。$\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_T]$为所有权值向量构成的参数矩阵而$\mathbf{b} = [b_1, \ldots, b_T]$为所有偏置构成的参数向量。$\boldsymbol{\Xi} = \{\xi_{il} | 1 \leq i \leq m,\ l \in Y_i\}$和$\boldsymbol{\Theta} = \{\theta_{ilj} | 1 \leq i \leq m,\ l \in \overline{Y_i},\ 1 \leq j \leq n_i\}$为相应的松弛变量集合。此外，目标函数中的参数$C$用于平衡系统在训练集上的经验误差和间隔。

优化问题式(11)是一个具有凸目标函数和线性约束条件的二次规划问题，但仅仅假设了线性模型用于样本分类。为了使得系统具有非线性分类能力，我们将式(11)在其对偶形式下利用核技巧求解，相应的优化问题变为：

$$\max_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{\Gamma},\boldsymbol{\Phi}} \ \Omega(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\Gamma}, \boldsymbol{\Phi}) \tag{12}$$

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

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

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

其中，集合$\boldsymbol{A}=\{\alpha_{il}|1\le i\le m,\ l\in Y_i\}$，$\boldsymbol{B}=\{\beta_{ilj}|1\le i\le m,\ l\in\overline{Y_i},\ 1\le j\le n_i\}$，$\boldsymbol{\Gamma}=\{\gamma_{il}|1\le i\le m,\ l\in Y_i\}$以及$\boldsymbol{\Phi}=\{\phi_{ilj}|1\le i\le N,\ l\in\overline{Y_i},\ 1\le j\le n_i\}$分别为与优化问题式(11)中约束条件相对应的对偶变量。此外，优化问题(式 12)中的目标函数为：

$$\begin{aligned}\Omega(\boldsymbol{A},\boldsymbol{B},\boldsymbol{\Gamma},\boldsymbol{\Phi})=&-\tfrac{1}{2}\sum_{l=1}^{T}\left(\sum_{i\in S_l}\sum_{i'\in S_l}\left(\frac{\alpha_{il}\alpha_{i'l}}{n_i n_{i'}}\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}\langle\boldsymbol{x}_{ij},\boldsymbol{x}_{i'j'}\rangle\right)\right.\\&+2\sum_{i\in S_l}\sum_{i'\in\overline{S}_l}\left(\frac{\alpha_{il}}{n_i}\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}-\beta_{i'lj'}\langle\boldsymbol{x}_{ij},\boldsymbol{x}_{i'j'}\rangle\right)\\&\left.+\sum_{i\in\overline{S}_l}\sum_{i'\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\sum_{j'=1}^{n_{i'}}\beta_{ilj}\beta_{i'lj'}\langle\boldsymbol{x}_{ij},\boldsymbol{x}_{i'j'}\rangle\right)\right)\\&+\sum_{l=1}^{T}\left(\sum_{i\in S_l}\alpha_{il}+\sum_{i\in\overline{S}_l}\left(\sum_{j=1}^{n_i}\beta_{ilj}\right)\right)\end{aligned}$$

式(12)的对偶优化问题仍是一个具有凸目标函数和线性约束条件的二次规划问题。值得注意的是，该二次规划问题的约束条件仅含有区间形式的不等式约束以及线性的等式约束，由此我们可以利用文献[18]中的方法对其进行快速优化。该方法的基本思想是将二次规划问题分解为一系列相对简单的线性规划问题求解。在完成式(12)的优化过程后，我们可以利用 KKT 条件来获得模型参数$\{(\boldsymbol{w}_l,b_l)|l\in\mathcal{Y}\}$。这样，给定新样本$X^*$，与之对应的类别标记集合可按如下方式确定：

$$Y^*=\{l|\max_{\boldsymbol{x}\in X^*}\ f(\boldsymbol{x},l)\ge 0,\ l\in\mathcal{Y}\},\quad\text{where}$$

$$f(\boldsymbol{x},l)=\sum_{i\in S_l}\left(\frac{\alpha_{il}}{n_i}\sum_{j=1}^{n_i}\langle\boldsymbol{x}_j^i,\boldsymbol{x}\rangle\right)-\sum_{i\in\overline{S}_l}\left(\textstyle\sum_{j=1}^{n_i}\beta_{ilj}\langle\boldsymbol{x}_{ij},\boldsymbol{x}\rangle\right)+b_l\tag{13}$$

M^{3}MIML 算法的具体细节请参见[5]。

4. 利用 MIML 学习单示例样本

如前所述，如果能够直接接触原始数据对象，那么我们可以利用 MIML 进行建模而获取更多的有用信息，但在不少应用中，我们只能得到他人进行特征提取后的数据，一个对象由一个特征向量表示。事实上，对于采用单示例表示的对象，此时该对象多个类别标记所蕴含的多样性信息仅仅内嵌于单一的示例中。如果能将对象单一示例的表示形式合适地转化为包（一组示例）的表示形式，使得包中的每个示例能从特定方面反映对象所包含的某种信息，那么将有助于学习问题的解决。

基于上述考虑，我们设计了 INSDIF (INStance DIFferentiation)方法[2][6]。该方法将单示例多标记样本转化为多示例多标记样本，从而利用 MIML 框架获得更好的学习结果。总的来看，INSDIF 采用了基于“示例区分”策略的两阶段学习算法。在算法的第一阶段，INSDIF 将每个样本转化为包的表示形式从而在输入空间中显式地描述对象歧义性。在算法的第二阶段，INSDIF 利用多示例多标记学习器对转化后的数据集进行学习。

令$S=\{(\boldsymbol{x}_1,Y_1),\cdots,(\boldsymbol{x}_m,Y_m)\}$为训练集。其中，$\boldsymbol{x}_i\in\mathcal{X}$为一个示例，而$Y_i\subseteq\mathcal{Y}$为与$\boldsymbol{x}_i$对应的一组类别标记。此外，设每个示例都是一个$d$维的特征向量。在算法的第一阶段，INSDIF 为每个可能的概念类$l\in\mathcal{Y}$计算一个原型向量$\boldsymbol{v}_l$，该向量为具有类别l的所有训练样本对应的均值向量：

$$\boldsymbol{v}_l=\frac{1}{|U_l|}\left(\sum_{\boldsymbol{x}_i\in U_l}\boldsymbol{x}_i\right)$$
$$\text{where}\quad U_l=\{\boldsymbol{x}_i|\{\boldsymbol{x}_i,Y_i\}\in S,\ l\in Y_i\} \tag{14}$$

INSDIF 基于上述原型向量将对象转化为包的表示形式。具体来说，在求得每一类的原型向量后，INSDIF 将每个样本$\boldsymbol{x}_i$转化为一组示例构成的包B_i，包中的每个示例对应于样本$\boldsymbol{x}_i$与某个原型向量之间的差值：

$$B_i=\{\boldsymbol{x}_i-\boldsymbol{v}_l|l\in\mathcal{Y}\} \tag{15}$$

基于式(15)，每个样本由单一示例的表示形式$\boldsymbol{x}_i$转化为包的表示形式B_i，且包的大小等于所有可能的概念类别数。特别地，包中的每个示例(即$\boldsymbol{x}_i-\boldsymbol{v}_l$)考察了给定样本与类别$l$之间的空间关系，从而蕴含了该样本与此类别相关的某种信息。实际上，除了利用上述方式实现单示例表示向多示例表示的转化，还可采用其它策略来实现该目标。

在算法的第二阶段，INSDIF 采用 MIML 学习算法对转化后的数据集$S^*=\{(B_1,Y_1),\cdots,(B_m,Y_m)\}$进行学习。在提出该算法时[6]，我们使用了一种类似于 RBF 神经网络的两层分类结构来实现该目标，但其他的 MIML 学习算法，例如本章第 3 节中所述的算法都可用于此处。

具体地说，该结构的输入为一个包含n个示例的包$B=\{\boldsymbol{b}_1,\boldsymbol{b}_2,\cdots,\boldsymbol{b}_n\}$，包中的每个示例$\boldsymbol{b}_k$为一个 d 维的属性向量$[\boldsymbol{b}_{k1},\boldsymbol{b}_{k2},\cdots,\boldsymbol{b}_{kd}]'$。该结构的输出包含了$T$个实值$\{y_1,y_2,\cdots,y_T\}$，其中每个实值输出$y_l$与标记$l$相对应。该结构的第一层由$M$个包$\{C_1,C_2,\cdots,C_M\}$组成，其中每个包$C_j$对应于簇 G_j 的中心且 $\{G_1,G_2,\cdots,G_M\}$ 将训练集划分为 M 个互不相交的子集，即$\bigcup_{j=1}^{M}G_j=\{B_1,B_2,\cdots,B_m\}$且$G_i\bigcap_{i\neq j}G_j=\emptyset$。该结构的第二层对应于权值矩阵$\boldsymbol{W}=[w_{jl}]_{M\times T}$，其中$w_{jl}$为连接包$C_j$与输出$y_l$的权值。

我们将每个包看作一个原子对象，基于 Hausdorff 距离度量包之间的距离并利用 *k*-medoids 算法

将集合$\{B_1, B_2, \cdots, B_m\}$划分为$M$个不相交的簇$\{G_1, G_2, \cdots, G_M\}$。这样，每个子集$G_j$对应的中心$C_j$即为：

$$C_j = \underset{A \in G_j}{\operatorname{argmin}} \sum_{B \in G_j} d_H(A, B) \tag{16}$$

其中，$d_H(A, B)$用于计算包A与包B之间的 Hausdorff 距离。

由于聚类过程有助于发现数据集的内在结构信息，因此基于上式求得的子集中心可能蕴含了不同包的分布信息。由此，每个包B可以转化为一个M维的属性向量$[\phi_1(B), \phi_2(B), \cdots, \phi_M(B)]'$，其中$\phi_i(B) = d_H(B, C_i)$。INSDIF 算法所需的第二层权值矩阵$\boldsymbol{W} = [w_{jl}]_{M \times T}$可通过最小化如下的误差平方和函数得到：

$$E = \frac{1}{2} \sum_{i=1}^{m} \sum_{l=1}^{T} \{y_l(B_i) - d_l^i\}^2 \tag{17}$$

其中，$y_l(B_i) = \sum_{j=1}^{M} w_{jl} \phi_j(B_i)$为分类结构相对于包$B_i$在第$l$类上的实际输出。此外，$d_l^i$为算法相对于包$B_i$在第$l$类上的期望输出，当$l \in Y_i$时$d_l^i$取值为$+1$否则取值为$-1$。将上式相对于变量$w_{jl}$求导并设导数值为0，则最小化上述误差平方和函数等价于求解如下的方程组：

$$\left(\boldsymbol{\Phi}'\boldsymbol{\Phi}\right) \boldsymbol{W} = \boldsymbol{\Phi}'\mathbf{T} \tag{18}$$

其中，矩阵$\boldsymbol{\Phi} = [\phi_{ij}]_{m \times M}$且含有元素$\phi_{ij} = \phi_j(B_i)$，矩阵$\mathbf{T} = [t_{il}]_{m \times T}$且含有元素$t_{il} = d_l^i$。这里，我们使用奇异值分解来对上式求解。

在 INSDIF 算法的两阶段训练过程完成后，给定新样本$\boldsymbol{x}^*$，与之对应的类别标记集合为$Y^* = \{l | y_l(B^*) = \sum_{j=1}^{M} w_{jl} \phi_j(B^*) > 0,\ l \in \mathcal{Y}\}$。其中，$B^* = \{\boldsymbol{x}^* - \boldsymbol{v}_l | l \in \mathcal{Y}\}$为与$\boldsymbol{x}^*$对应的包的表示形式。

INSDIF 算法的具体细节请参见[2][6]。

5. 利用 MIML 学习复杂高层概念

如前所述，MIML 框架还有助于对复杂高层概念的学习，为此我们提出了 SUBCOD (sub-concept discovery) 算法[2]，通过发现目标概念的子概念来将单标记样本转化为多标记样本，从而利用 MIML 的帮助提高学习性能。

SUBCOD 采用了基于“子概念发现”策略的两阶段学习算法。在算法的第一阶段，SUBCOD 基于训练包中的所有示例进行聚类分析。由此，算法发现与高层概念对应的一组低层子概念，并将多示例单标记样本转化为多示例多标记样本。在算法的第二阶段，SUBCOD 利用监督学习器获得低层子概念与高层概念之间的映射关系。由此，基于 MIML 学习器对转化后的数据集进行学习，并利用

监督学习器所得的映射关系对新样本的类别标记进行预测。

令$S=\{(X_1,y_1),\cdots,(X_m,y_m)\}$为训练集。其中，$X_i\subseteq\mathcal{X}$为一组示例构成的包，而$y_i\in\mathcal{Y}$为与$X_i$对应的类别标记。在算法的第一阶段，SUBCOD 将所有训练包中的示例构成数据集$D=\{\boldsymbol{x}_{11},\cdots,\boldsymbol{x}_{1,n_1},\cdots,\boldsymbol{x}_{m1},\cdots,\boldsymbol{x}_{m,n_m}\}$。为了方便起见，我们将$D$中所有示例重新索引并记为$\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_N\}$。其中，$N=\sum_{i=1}^{m}n_i$。

我们利用具有M个混合成分的混合高斯模型对数据集D进行建模，并将所得模型中的每个混合成分作为相应的低层子概念。我们基于标准的 EM 算法对高斯混合模型中的参数进行学习。简要地说，我们首先随机初始化各个高斯混合成分的均值向量$\boldsymbol{\mu}_k$，协方差矩阵Σ_k以及混合系数$\pi_k\ (k=1,2,\cdots,M)$。在 EM 算法迭代的每一轮中，我们首先求得D中每个样本隶属于各混合成分的概率：

$$\gamma_{ik}=\frac{\pi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k,\Sigma_k)}{\sum_{j=1}^{M}\pi_j\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_j,\Sigma_j)}\quad(i=1,2,\cdots,N)\tag{19}$$

然后，基于所得数值对模型参数进行更新：

$$\boldsymbol{\mu}_k^{new}=\frac{\sum_{i=1}^{N}\gamma_{ik}\boldsymbol{x}_i}{\sum_{i=1}^{N}\gamma_{ik}}\tag{20}$$

$$\Sigma_k^{new}=\frac{\sum_{i=1}^{N}\gamma_{ik}(\boldsymbol{x}_i-\boldsymbol{\mu}_k^{new})(\boldsymbol{x}_i-\boldsymbol{\mu}_k^{new})'}{\sum_{i=1}^{N}\gamma_{ik}}\tag{21}$$

$$\pi_k^{new}=\frac{\sum_{i=1}^{N}\gamma_{ik}}{N}\tag{22}$$

在上述 EM 过程收敛或迭代达到指定轮数后，我们通过如下方式确定D中每个样本$\boldsymbol{x}_i$对应的低层子概念：

$$sc(\boldsymbol{x}_i)=\underset{k}{\operatorname{argmax}}\,\gamma_{ik}\quad(k=1,2,\cdots,M)\tag{23}$$

基于上述结果，我们为每个训练包X_i赋予一个M维的二值类别向量$\boldsymbol{c}_i$以表达其隶属的一组低层子概念。其中，$c_{ij}=+1$代表X_i具有第j个高斯混合成分所代表的子概念，否则$c_{ij}=-1$。特别地，基于式(23)，$c_{ij}=+1$当且仅当$\exists\boldsymbol{x}\in X_i:sc(\boldsymbol{x})=j$，否则$c_{ij}=-1$。值得注意的是，对于两个具有相同高层概念的包而言，由于它们所含的示例不同，因此其对应的低层子概念有可能不同。

由于上述确定子概念的过程基于非监督聚类的方式实现，因此并未考虑每个包所含的高层概念。为此，我们通过考察子概念与X_i的高层概念(即y_i)之间的关系对二值类别向量$\boldsymbol{c}_i$做进一步的修正。具体来说，我们采用最大化间隔策略来实现该目标。设$\boldsymbol{z}_i$为用于子类别标记修正的M维实值向量，向量的每一维$z_{ij}\ (j=1,\cdots,M)$位于$[-1.0,+1.0]$区间之内。其中，$z_{ij}=+1$代表标记c_{ij}的取值应保持

不变而$z_{ij}=-1$则代表应翻转标记c_{ij}的取值。此外，设向量$\boldsymbol{q}_i=\boldsymbol{c}_i\odot\boldsymbol{z}_i$，其中$q_{ij}=c_{ij}z_{ij}\ (j=1,2,\cdots,M)$。另设$\{\boldsymbol{c}_1,\boldsymbol{c}_2,\cdots,\boldsymbol{c}_m\}$中至少有$\theta$个标记不能被翻转。基于上述表示，SUBCOD 算法将求解如下的优化问题：

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{Z}} \frac{1}{2}||\boldsymbol{w}||_2^2+C\cdot\boldsymbol{\xi}'\mathbf{1} \tag{24}$$

$$\begin{aligned}\text{s.t.}\ \ & y_i(\boldsymbol{w}'(\boldsymbol{c}_i\odot\boldsymbol{z}_i)+b)\geq 1-\xi_i,\ \ \forall\ 1\leq i\leq m\\ & \boldsymbol{\xi}\geq\mathbf{0},\\ & \sum_{i,j} z_{ij}\geq 2\theta-1\end{aligned}$$

其中，$\boldsymbol{Z}=[\boldsymbol{z}_1,\boldsymbol{z}_2,\cdots,\boldsymbol{z}_m]$。

通过优化上述问题，我们可以得到最大化间隔意义下的修正值$\boldsymbol{Z}$。我们迭代地求解式(24)。在迭代过程开始前，我们将$\boldsymbol{Z}$的每一个元素初始化为 1。在迭代优化的每一轮中，我们首先固定$\boldsymbol{Z}$的取值来优化变量$\boldsymbol{w}$与b(二次规划问题)；然后，我们固定变量$\boldsymbol{w}$与b的取值来优化修正值$\boldsymbol{Z}$(线性规划问题)。上述迭代过程不断重复直至收敛或达到指定迭代轮数。

此后，我们利用修正值$\boldsymbol{Z}$将每个训练包X_i对应的二值类别向量$\boldsymbol{c}_i$修正为$\tilde{\boldsymbol{c}}_i$。其中，$\tilde{c}_{ij}=+1$当且仅当$c_{ij}z_{ij}>0$，否则$\tilde{c}_{ij}=-1$。上述修正过程完成后，初始的多示例单标记训练集$S=\{(X_1,y_1),\cdots,(X_m,y_m)\}$即可转化为相应的多示例多标记数据集$S^*=\{(X_1,\tilde{\boldsymbol{c}}_1),\cdots,(X_m,\tilde{\boldsymbol{c}}_m)\}$。基于转化后的数据集$S^*$，我们可以学习得到一个 MIML 学习器$h:2^{\mathcal{X}}\to 2^{\tilde{\mathcal{Y}}}\ (\tilde{\mathcal{Y}}=\{1,2,\cdots,M\})$。

在算法的第二阶段，为了将测试样本在h上的多标记输出映射到所需的单标记，SUBCOD 使用一个监督学习算法从$\{(\tilde{\boldsymbol{c}}_1,y_1),\cdots,(\tilde{\boldsymbol{c}}_m,y_m)\}$中学习得到一个分类器$f:2^{\tilde{\mathcal{Y}}}\to\mathcal{Y}$。在 SUBCOD 算法的两阶段训练过程完成后，给定新样本X^*，与之对应的类别标记即为$y^*=f(h(X^*))$。

SUBCOD 算法的具体细节请参见[2]。

6. 结束语

MIML 是一个有潜力的面向多义性对象的学习框架，本章对这方面的一些初步工作[1][2][5][6]进行了介绍。最近，我们在 MIML 的距离度量学习方面又取得了一些进展[19]。作为一个新框架，MIML 还有很多内容需要进一步探索。我们相信，在今后的几年中，在 MIML 的学习理论、高效算法、新型应用等方面都会有新成果出现。

参 考 文 献

[1] Zhou Z-H, Zhang M-L. Multi-instance multi-label learning with application to scene classification. In: Schölkopf B, Platt J, Hofmann T, eds. *Advances in Neural Information Processing Systems 19 (NIPS'06)*, Cambridge, MA: MIT Press, 2007, 1609-1616.

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

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

[4] 周志华. 多示例学习. 见: 刘大有 主编, 知识科学中的基本问题研究, 北京: 清华大学出版社, 2006, 322-336.

[5] Zhang M-L, Zhou Z-H. M^{3}MIML: A maximum margin method for multi-instance multi-label learning. In: *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM'08)*, Pisa, Italy, 2008, 688-697.

[6] Zhang M-L, Zhou Z-H. Multi-label learning by instance differentiation. In: *Proceedings of the 22nd Conference on Artificial Intelligence (AAAI'07)*, Vancouver, Canada, 2007, 669-674.

[7] Xu X, Frank E. Logistic regression and boosting for labeled bags of instances. In: *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'04)*, Sydney, Australia, LNAI 3056, 2004, 272-281.

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

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

[10] Gärtner T, Flach P A, Kowalczyk A, Smola A J. Multi-instance kernels. In: *Proceedings of the 19th International Conference on Machine Learning (ICML'02)*, Sydney, Australia, 2002, 179-186.

[11] Evgeniou T, Micchelli C A, Pontil M. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 2005, 6: 615-637.

[12] Cheung P M, Kwok J T. A regularization framework for multiple-instance learning. In: *Proceedings of the 23rd International Conference on Machine Learning (ICML'06)*, Pittsburgh, PE, 2006, 193-200.

[13] Smola A J, Vishwanathan S V N, Hofmann T. Kernel methods for missing variables. In: *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS'05)*, Barbados, 2005, 325-332.

[14] Yuille A L, Rangarajan A. The concave-convex procedure. *Neural Computation*, 2003, 15(4): 915-936.

[15] Pham Dinh T, Le Thi H A. A D. C. optimization algorithm for solving the trust-region subproblem. *SIAM Journal on Optimization*, 1998, 8(2): 476-505.

[16] Maron O, Lozano-Pérez T. A framework for multiple-instance learning. In: Jordan M I, Kearns M J, Solla S A, eds. *Advances in Neural Information Processing Systems 10 (NIPS'07)*, Cambridge, MA: MIT Press, 1998, 570-576.

[17] Ray S, Page D. Multiple instance regression. In: *Proceedings of the 18th International Conference on Machine Learning (ICML'01)*, Williamstown, MA, 2001, 425-432.

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

[19] Wang S, Jin R, Zhou Z-H. Learn a distance metric from multi-instance multi-label data. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'09)*, Miami, FL, 2009.