# Co-Labeling for Multi-View Weakly Labeled Learning

Xinxing Xu, Wen Li, Dong Xu, *Senior Member, IEEE*, and Ivor W. Tsang

**Abstract**—It is often expensive and time consuming to collect labeled training samples in many real-world applications. To reduce human effort on annotating training samples, many machine learning techniques (e.g., semi-supervised learning (SSL), multi-instance learning (MIL), etc.) have been studied to exploit weakly labeled training samples. Meanwhile, when the training data is represented with multiple types of features, many multi-view learning methods have shown that classifiers trained on different views can help each other to better utilize the unlabeled training samples for the SSL task. In this paper, we study a new learning problem called multi-view weakly labeled learning, in which we aim to develop a unified approach to learn robust classifiers by effectively utilizing different types of weakly labeled multi-view data from a broad range of tasks including SSL, MIL and relative outlier detection (ROD). We propose an effective approach called co-labeling to solve the multi-view weakly labeled learning problem. Specifically, we model the learning problem on each view as a weakly labeled learning problem, which aims to learn an optimal classifier from a set of pseudo-label vectors generated by using the classifiers trained from other views. Unlike traditional co-training approaches using a single pseudo-label vector for training each classifier, our co-labeling approach explores different strategies to utilize the predictions from different views, biases and iterations for generating the pseudo-label vectors, making our approach more robust for real-world applications. Moreover, to further improve the weakly labeled learning on each view, we also exploit the inherent group structure in the pseudo-label vectors generated from different strategies, which leads to a new multi-layer multiple kernel learning problem. Promising results for text-based image retrieval on the NUS-WIDE dataset as well as news classification and text categorization on several real-world multi-view datasets clearly demonstrate that our proposed co-labeling approach achieves state-of-the-art performance for various multi-view weakly labeled learning problems including multi-view SSL, multi-view MIL and multi-view ROD.

**Index Terms**—Multi-view learning, multi-instance learning, semi-supervised learning, relative outlier detection, weakly labeled learning

## 1 Introduction

In many real-world applications, it is often expensive and time consuming to collect labeled training samples. In recent decades, researchers have been exploiting various learning scenarios for utilizing weakly labeled samples to reduce human effort on manually labeling the training samples. For example, in semi-supervised learning (SSL) [1], [2], the training data consists of a limited number of labeled training samples and a large number of unlabeled training samples. Similarly, in multi-instance learning (MIL) [3], [4], [5], [6], the training data is provided in the form of bags. Only the label of each bag is known, while the labels of instances in each bag remain unknown. In recent work [7], training data with uncertain labels was referred to as weakly labeled data, and those learning problems were uniformly referred to as the *weakly labeled learning* problem. A unified approach was proposed in [7] for solving various learning problems with weakly labeled data including SSL, MIL and clustering, in which they learnt an optimal classifier from all possible labelings of training data. However, their work [7] focused on single-view training data.

When training data is represented with multiple feature representations, researchers have developed many multi-view learning approaches to improve performance by utilizing information from different views [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. Most of those works (e.g., co-training [8]) in multi-view learning were proposed for the multi-view SSL scenario. It has been shown that, with multi-view information, the classifiers trained on different views can effectively help each other to better utilize the unlabeled training data. Nevertheless, those works were designed for SSL, and it is unclear how to extend them to the more general weakly labeled learning problem.

In this paper, we study the learning problem by using weakly labeled training data with multiple views, which is referred to as the *multi-view weakly labeled learning* problem. We aim to develop a unified approach to learn robust classifiers by effectively utilizing different types of weakly labeled training data with multiple views of features from a broad range of applications including the traditional multi-view SSL problem as well as the multi-view MIL and multi-view Relative Outlier Diction (ROD) problems. Specifically, we propose a novel **co-labeling** approach for multi-view weakly labeled learning, in which we consider two major challenges: how to effectively exchange information among different views, and how to effectively learn a robust classifier on each view.

To tackle the first challenge, we use pseudo-label vectors to pass information among different views similar to co-training

- *X. Xu is with the Institute of High Performance Computing (IHPC), the Agency for Science, Technology and Research, Singapore. E-mail: xuxinx@ihpc.a-star.edu.sg.*
- *W. Li is with the Computer Vision Laboratory, ETH Zürich, Zürich, Switzerland. E-mail: liwen@vision.ee.ethz.ch.*
- *D. Xu is with the School of Electrical and Information Engineering, The University of Sydney, Sydney, NSW 2006, Australia. E-mail: dong.xu@sydney.edu.au.*
- *I.W.H. Tsang is with the Center for Quantum Computation & Intelligent Systems, University of Technology, Sydney, NSW 2006, Australia. E-mail: ivor.tsang@gmail.com.*

*Manuscript received 26 Apr. 2014; revised 21 July 2015; accepted 16 Aug. 2015. Date of publication 3 Sept. 2015; date of current version 12 May 2016. Recommended for acceptance by F. Fleuret.*
*For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.*
*Digital Object Identifier no. 10.1109/TPAMI.2015.2476813*

based methods. In co-training based methods [8], [9], [17], [18], [19], the predictions from a classifier trained on one view are used to label the unlabeled samples for training the classifier on the other view. To handle more general weakly labeled learning scenarios including SSL, MIL and ROD, in our co-labeling approach, we first propose a projection operator, which converts the predictions (i.e., the decision values) to pseudo-label vectors by considering different constraints on weakly labeled data from different learning scenarios. Moreover, considering that a single pseudo-label vector in the co-training based approach may be sensitive to the threshold, we further propose different strategies to generate multiple pseudo-label vectors by using different biases to enhance the robustness of our co-labeling approach.

To learn the classifier for each view, traditional co-training based methods used supervised learning approaches by treating single pseudo-label vector as the ground-truth label vector, which may be sensitive to the noise in the pseudo-label vector. In Section 3, we formulate the learning problem on each view as a weakly labeled learning problem [20], [21], in which we learn an optimal classifier from a set of pseudo-label vectors and the combination of these pseudo-label vectors to the final classifier is automatically decided by the multiple kernel learning (MKL) method. Specifically, as discussed in Section 4, those pseudo-label vectors can be generated from the classifiers on other views, with different biases and from all previous iterations. Inspired by recent works [2], [5], [7], [22], in Section 5 we formulate this learning problem as the MKL problem [23], [24], in which each base kernel is associated with a pseudo-label vector. Moreover, by observing that these pseudo-label vectors are generated with different strategies, we further develop a novel multi-layer MKL method to effectively utilize the intrinsic group structure on those base kernels. An efficient alternating optimization algorithm is proposed to solve the new multi-layer MKL problem by using a recursive updating strategy for updating the kernel combination coefficients.

In Section 6, we conduct extensive experiments for different weakly labeled learning scenarios including multi-view SSL, multi-view MIL, and multi-view ROD, and also provide a detailed experimental analysis. The experimental results clearly demonstrate that our co-labeling approach for multi-view weakly labeled learning is not only better than the existing multi-view learning methods but also outperforms the recent weakly labeled learning work [7] as well as the related state-of-the-art methods for SSL, MIL or ROD.

Beyond our preliminary work [20], in this paper, we additionally propose a novel multi-layer MKL method to learn a more robust classifier on each view.

## 2 Related Work

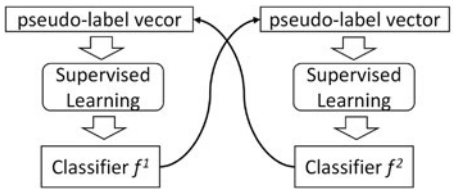
Our work is related to traditional multi-view learning approaches. Most traditional multi-view learning methods were proposed for SSL. One of the pioneering works is the co-training method [8], which was originally proposed for semi-supervised learning problems with two views of training data. It was further extended to co-EM [9], in which they label all the unlabeled data at each iteration without considering confidence. It was also extended by using SVMs as the base classifiers in [17]. Co-training was also extended to tri-training [18] and co-forest [19] to handle more than two views. However, the co-training style algorithms work under strict assumptions that each view is sufficient to train a low-error classifier and both views are conditionally independent, which might not be satisfied on real world datasets [25]. Many works attempted to relax those assumptions from various perspectives, such as weak dependence [26], $\alpha$-expansion [27], large diversity [25] and label propagation [28]. Recently, co-training with insufficient views has also been theoretically analyzed in [29].

Besides co-training style methods, other methods such as co-regularization based approaches [10], [11], [12], [13] were also proposed to train classifiers on different views based on the so-called *co-regularization* criterion, which is used to minimize the differences of decision values from the classifiers on different views. The similar idea has also been employed in multi-view clustering [14], [15]. However, similar to co-training style methods, all these methods are specifically designed for a specific learning scenario, thus they cannot be directly applied to handle general multi-view weakly labeled data.

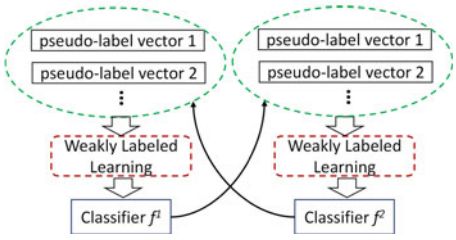
Our work is also related to various learning scenarios with different types of weakly labeled training data. Specifically, besides the above mentioned multi-view SSL works, SSL methods have also been widely studied for single view training data [1], [2]. A comprehensive survey on SSL can be found in [30]. MIL is another widely studied learning scenario, in which the weakly labeled training data is provided in the form of bags of instances. Only the labels of training bags are given, while the labels of instances inside each training bag are unknown. Many works have been proposed to solve the MIL problems in the literature [3], [4], [5], [6], [31], and a survey on MIL can be found in [32]. Another example is maximum margin clustering (MMC) [22], [33], in which the goal is to learn a discriminative classifier to partition unlabeled training samples into two disjoint clusters. Recent work [7] uniformly referred to the above learning scenarios as weakly labeled learning, and proposed a unified scheme called WellSVM to solve it. Another work that can handle different types of weakly labeled data was also proposed in [34]. Other learning scenarios related to weakly labeled data include relative outlier detection (ROD) [35], [36], [37] and multi-instance semi-supervised learning (MISSL) [38]. However, all those works were proposed for only single-view training data. In contrast, in this work, we study a new learning problem called multi-view weakly labeled learning, in which we further explore the multi-view information of different types of weakly labeled training data to learn more robust classifiers, and our proposed co-labeling approach is applicable for not only multi-view SSL, but also other tasks such as multi-view MIL and multi-view ROD.

## 3 Co-Labeling for Multi-View Weakly Labeled Learning

In this section, we first review the existing works on multi-view learning and weakly labeled learning, and then present our co-labeling approach for the multi-view weakly labeled learning problem.




(a) Co-training.




(b) Co-labeling.

Fig. 1. Comparison between co-training and our co-labeling.

### 3.1 Co-Training

In multi-view learning, the training data is represented with multiple views of features. Typically, a classifier $f^v$ (e.g., an SVM classifier) is trained on the $v$th view and the final classifier is obtained by fusing the classifiers from all views, i.e., $\tilde{f}(\mathbf{x}) = \frac{1}{V}\sum_{v=1}^{V} f^v(\mathbf{x}^v)$.

Co-training [8] was originally proposed for the SSL problem with two views. The basic idea of co-training is to iteratively add some pseudo-labeled samples into the pool of labeled training samples to re-train the classifiers on both views. The pseudo-labeled samples are selected from the pool of unlabeled training samples, and are labeled by at least one classifier which has a confident prediction. Finally, the classifiers from different views are fused to perform the classification.

While the original co-training algorithm feeds newly labeled training samples to each view, it can also be deemed as a learning process by iteratively updating the pseudo-labels of unlabeled data on each view [16]. We illustrate the co-training method in Fig. 1a. At each iteration, the classifier on one view (e.g., $f^1$ or $f^2$) generates pseudo-label vectors for learning the classifier on the other view (e.g., $f^2$ or $f^1$) by using the supervised learning approach.

### 3.2 Weakly Labeled Learning

To reduce human effort for labeling training data, various learning scenarios have been proposed in the literature to learn classifiers only based on weakly labeled data. For example, in SSL, one is given a limited number of labeled samples and a large amount of unlabeled samples. In MIL, the training data are given in the form of training bags, with each bag containing a certain number of training instances. While the label of each bag is given, the labels of instances inside each bag remain unknown.

Recently, the work in [7] studied the weakly labeled learning problem by unifying the above learning scenarios into a general learning problem with weakly labeled data. The weakly labeled learning problem is formulated as the following optimization problem [7]:

$$\min_{f,\mathbf{y}\in\mathcal{Y}} \quad \|f\|^2 + C\ell(f,\mathbf{y}), \tag{1}$$

where $\mathcal{Y}$ is the so-called *label candidate set*, $f$ is the target classifier, and $\ell(\cdot)$ is the loss function. The label candidate set $\mathcal{Y}$ contains all the possible labelings of the training samples. Intuitively, the weakly labeled learning problem aims to learn an optimal classifier from all the possible labelings.

By defining different constraints on the label candidate set $\mathcal{Y}$, the weakly labeled learning problem in (1) unifies various traditional learning scenarios with different weakly labeled data. Let us denote the label vector as $\mathbf{y} = [\mathrm{y}_1, \ldots, \mathrm{y}_n]'$ where $\mathrm{y}_i \in \{+1, -1\}$ is the possible label for $\mathbf{x}_i$ and $n$ is the number of training samples. We give several examples of the definition on the label candidate set $\mathcal{Y}$ corresponding to different traditional learning scenarios including MIL, SSL and ROD.

In MIL, the constraints are that all instances in the negative bags are negative, and at least one instance (or a portion of the instances) in each positive bag is positive [3], [5]. Let us denote $\mathcal{B}_I$ as the $I$th training bag and $Y_I$ as the corresponding bag label. Then we can represent the label candidate set as $\mathcal{Y} = \{\mathbf{y} | \sum_{i:\mathbf{x}_i\in\mathcal{B}_I}(\mathrm{y}_i+1)/2 \geq \varepsilon, \text{ if } Y_I = 1; \mathrm{y}_i = -1, \text{if } Y_I = -1\}$, where we have $\varepsilon = 1$ for the traditional MIL constraint [3], and $\varepsilon = \mu|\mathcal{B}_I|$ for the general MIL constraint [5] with $\mu$ being the portion parameter and $|\cdot|$ being the cardinality function.

In SSL [1], the training data is composed of $n_l$ labeled samples and a large number of unlabeled samples. Usually, the unlabeled samples are required to satisfy a balance constraint. Then the label candidate set can be represented as $\mathcal{Y} = \{\mathbf{y} | \mathrm{y}_i = g_i, i = 1, \ldots, n_l, \sum_{i=n_l+1}^{n}(\mathrm{y}_i+1)/2 = \sigma(n-n_l)\}$, where $g_i$ is the ground truth label of $\mathbf{x}_i$, and $\sigma$ is the parameter for the balance constraint.

In ROD [36], the training data consist of $n_l$ normal patterns and $n - n_l$ unlabeled samples. If we denote the label for the normal pattern and the outlier as 1 and $-1$, respectively, the label candidate set can be represented as $\mathcal{Y} = \{\mathbf{y} | \mathrm{y}_i = 1, i = 1, \ldots, n_l, \sum_{i=n_l+1}^{n}(\mathrm{y}_i+1)/2 = (n-n_l)(1-\kappa)\}$, where $\kappa$ is a parameter on the ratio of outliers. The label candidate set for other learning scenarios such as maximum margin clustering [22] can also be similarly defined.

### 3.3 Multi-View Weakly Labeled Learning

We observe that most traditional multi-view learning approaches are limited to semi-supervised learning, and the recently proposed weakly labeled learning works for semi-supervised learning and multi-instance learning are limited to single-view training data. In practice, traditional multi-view learning methods can benefit from the recent progress on weakly labeled learning, and weakly labeled learning methods can also be improved with multi-view information. Based on these observations and motivations, in this paper we propose to study a new learning problem called multi-view weakly labeled learning which aims to solve the weakly labeled learning problem with multi-view training data.

Specifically, we focus on the binary classification problem in this work. The multi-class classification problem can be converted into a set of binary classification problems using the one-versus-all strategy. Inspired by the work on co-training and weakly labeled learning, we formulate the multi-view weakly labeled learning problem as follows:

$$\min_{f^v,\mathbf{y}^v\in\mathcal{C}^v} \quad \sum_{v=1}^{V} \|f^v\|^2 + C\ell(f^v,\mathbf{y}^v), \tag{2}$$

where $\mathbf{y}^v$ is the pseudo-label vector for the training samples on the $v$th view, $\mathcal{C}^v$ is the set of possible pseudo-label vectors for the $v$th view generated by using the predictions from other views. Similar to co-training, those $V$ classifiers in (2) are not individually learnt. Specifically, we use the classifier trained on one view to help the training processes on the other views through the pseudo-label vector sets (see the details below). Finally those $V$ classifiers are equally fused for prediction as in co-training.

**Algorithm 1.** The Co-Labeling Algorithm

1: Initialize the pseudo-label vector set $\mathcal{C}^v$ for each view.
2: **repeat**
3: Train a classifier $f^v$ based on $\mathcal{C}^v$ by solving a weakly labeled learning problem for each view (see Section 5).
4: Obtain the predictions $\mathbf{z}_v$ of training samples using $f^v$ for each view.
5: Update the pseudo-label vector set $\mathcal{C}^v$ using $\mathbf{z}_j$'s ($j \neq v$).
6: **until** The stopping criterion is reached.
7: **return** $f^v$'s.

To solve the multi-view weakly labeled learning problem in (2), we propose a novel approach called "co-labeling" by tackling two major challenges: how to effectively use the classifier trained on one view to help the training processes on other views, and how to improve weakly labeled learning on each view for training a more robust classifier.

We use the pseudo-label vector set $\mathcal{C}^v$ to exchange information from the classifier trained on one view to another. Specifically, similar to co-training, in our co-labeling approach the pseudo-label vector set $\mathcal{C}^v$ is generated by using the predictions from classifiers trained on the other views. To cope with different weakly labeled data, we first propose a projection operator, which converts the predictions (i.e., the decision values) to pseudo-label vectors by considering the constraints associated with different weakly labeled learning scenarios. Moreover, in the traditional co-training methods, the classifiers exchange information through a single pseudo-label vector. In contrast, we use a set of pseudo-label vectors in (2), which contain more information for learning a robust classifier. Those pseudo-label vectors can be generated from different views, biases and iterations (see Section 4 for details).

To learn a classifier on each view, the traditional co-training algorithm adopted the supervised learning method by using single pseudo-label vector as the ground-truth of unlabeled data, which may be sensitive to noise in the pseudo-label vector. To overcome this problem, in our co-labeling approach, we treat the learning problem on each view as a weakly labeled learning problem, by learning an optimal classifier from a set of pseudo-label vectors (see Fig. 1). Moreover, to improve weakly labeled learning on each view, we further exploit the group structure within the pseudo-label vectors, which leads to a novel multi-layer MKL problem (see Section 5).

We list our co-labeling algorithm in Algorithm 1, in which we iteratively update the set of pseudo-label vectors and train a classifier on each view by solving a weakly labeled learning problem. The details of generating the pseudo-label vector set $\mathcal{C}^v$ for each view are introduced in Section 4, and the multi-layer MKL model for solving the weakly labeled learning problem on each view is presented in Section 5.

## 4 Generate the Pseudo-Label Vector Set

How to generate a set of pseudo-label vectors on one view using the information from the other views is the first key issue of our co-labeling approach. Inspired by co-training [8], we generate the pseudo-label vectors on one view by using the predictions (i.e., decision values) from the classifiers on other views. Unlike the co-training method, which uses only a single pseudo-label vector, we need to consider several issues as described below to better handle the general multi-view weakly labeled learning scenarios and train a more robust classifier.

### 4.1 Handling General Weakly Labeled Constraints

Let us denote the prediction from the classifier of the $v$th view $f^v$ as $\mathbf{z} = [z_1, \ldots, z_n]'$, where $z_i$ is the decision value of the $i$th training sample. To cope with general weakly labeled learning scenarios, we need to define a projection operator $\mathbf{y} = \pi(\mathbf{z})$, which converts each prediction $\mathbf{z}$ to a pseudo-label vector $\mathbf{y}$ by considering the constraints associated with different weakly labeled learning scenarios. Formally, the projection operator can be defined as follows:

$$\pi(\mathbf{z}) = \arg\min_{\mathbf{y} \in \mathcal{Y}} \|\mathbf{y} - \mathbf{z}\|, \tag{3}$$

where $\|\cdot\|$ is the $\ell_2$-norm.

We solve the above problem as follows. For training samples with known labels (e.g., the labeled samples in SSL and ROD as well as the instances in negative bags in MIL), we directly assign their ground-truth labels. For the weakly labeled samples, we use the thresholding function to convert their decision values into binary values (i.e., $+1$ or $-1$) as discussed below.

In MIL, the instances in each positive bag should satisfy the constraint $\sum_{i:\mathbf{x}_i \in \mathcal{B}_I} (\mathrm{y}_i + 1)/2 \geq \varepsilon$ (see Section 3.2). To solve Eq. (3), we first sort the instances in each positive bag based on their decision values in descending order, and obtain the pseudo-label vector $\mathbf{y}$ according to the sign of decision values (i.e., the threshold is set to zero). If the labeling for any positive bag does not satisfy the bag constraint, we then assign the top $\epsilon$ instances to be positive.

In SSL, the unlabeled samples are associated with a balance constraint $\sum_{i=n_l+1}^{n} (\mathrm{y}_i + 1)/2 = \sigma(n - n_l)$ (see Section 3.2). So we sort these unlabeled samples based on their decision values in descending order, and adjust the threshold to assign the first $\sigma(n - n_l)$ samples to be positive and the remaining ones to be negative. The same method can also be used for the unlabeled training samples in the ROD task. We can also similarly employ the projection operator for other weakly labeled training data with linear constraints.

### 4.2 Handling the Bias

Moreover, the learnt classifiers in the weakly labeled learning problems can be easily biased. For example, in MIL one common approach is to initialize all the instances in positive bags as positive samples, so it is more likely that the initial classifier will predict the negative samples to be positive. A possible solution is to adjust the bias term of the learnt classifier. However, it is a nontrivial task since we do not have the ground truth labels to decide the adjustment. Considering the learnt classifiers actually rely on the

pseudo-label vectors, we propose to perturb the predictions to obtain multiple pseudo-label vectors, and learn the optimal bias by using the MKL method (see Section 5) in the training process, as motivated by the recent work on domain adaptation [39].

Formally, for MIL, given any prediction $\mathbf{z}$, we can obtain a set of perturbed predictions $\{\tilde{\mathbf{z}}_s|_{s=1}^S\}$ by using different predefined biases, i.e., $\tilde{\mathbf{z}}_s = \mathbf{z} + b_s$, where $S$ is the number of biases and $b_s \in \mathbb{R}$ is a bias term to adjust the predictions. After that, we generate a set of pseudo-label vectors using the projection operator defined as in (3) on these perturbed predictions. In our experiments, we empirically set $b_s$ in the range of $[-0.5, -0.3]$ with an interval of $0.1$.

Similarly, the learnt classifier can be easily biased due to the limited number of labeled training samples in SSL. Moreover, the parameter $\sigma = \sigma_0$ in the balance constraint is usually estimated from a limited number of labeled samples, where $\sigma_0$ is the ratio of positive training samples over all labeled training samples [1]. The estimation may also be inaccurate, so we propose to use different constraints by perturbing $\sigma$ in SSL, which is similar to that we perturb the bias term in MIL. Specifically, we change the balance constraint in (3) by setting $\sigma = \sigma_0 + \sigma_s$, where $\sigma_s$ is a predefined perturbation term. After that, we generate a set of pseudo-label vectors using the projection operator defined as in (3) with different constraints decided by $\sigma_s$'s. In our experiments, we empirically set $\sigma_s$ in the range of $[-0.1, +0.1]$ with an interval of $0.02$. We use the same method to set $\kappa = \kappa_0 + \kappa_s$ for the ROD task, where $\kappa_s$ is also in the range of $[-0.1, +0.1]$ with an interval of $0.02$. We only have the normal samples for the ROD task, but $\kappa_0$ can still be decided according to our prior knowledge.

### 4.3 Combining the Pseudo-Label Vector Set from Previous Iterations

Thus far, we only consider the pseudo-label vectors obtained by using the predictions from the latest iteration, which means the pseudo-label vector set on each view can be changed at different iterations. As a result, one potential problem is that the algorithm may not converge.

Inspired by the recent weakly labeled learning works [2], [7], [22], we construct the pseudo-label vector set by using the pseudo-label vectors obtained from all previous iterations. In other words, at each iteration we augment the pseudo-label vector set by appending the newly obtained pseudo-label vectors into the previous pseudo-label vector set. Thus, our algorithm can converge (see Section 5.4 for more detailed discussions).

### 4.4 Summary and Discussion

We illustrate the entire process for generating the pseudo-label vectors in Algorithm 2. For MIL, we first perturb the prediction $\mathbf{z}_{v,t}$ at the $t$th iteration from the $v$th view with $S$ predefined biases and obtain a set of perturbed predictions $\{\tilde{\mathbf{z}}_{v,s,t}|_{s=1}^S\}$. Then, we use the projection operator to convert each $\tilde{\mathbf{z}}_{v,s,t}$ to a pseudo-label vector $\mathbf{y}_{v,s,t}$. For SSL and ROD, we obtain S perturbed constraints and then project the prediction into the pseudo-label vector based on each constraint. By defining $\mathcal{O}_{v,t} = \{\mathbf{y}_{v,s,t}|_{s=1}^S\}$ as the pseudo-label vectors generated from the $v$th classifier at the $t$th iteration, the pseudo-label vector set for training the classifier on the $v$th view at the next iteration can be obtained by combining the pseudo-label vectors from all the other views, different biases and all the previous iterations, i.e., $\mathcal{C}_{t+1}^v = \mathcal{C}_t^v \bigcup \{\mathcal{O}_{j,t}|_{j=1,j\neq v}^V\}$. Therefore, at the $t$th iteration, in total we have $(V-1) \times S \times t$ pseudo-label vectors for the $v$th view, where $V$ is the number of views, $S$ is the number of biases.

**Algorithm 2.** Algorithm for Generating Pseudo-label Vectors

**Input:** The current pseudo-label vector set $\mathcal{C}_t^v$, and the decision values from the classifiers on different views at the $t$th iteration $\mathbf{z}_{v,t}$, for $v = 1, \ldots, V$.

1: **for** $v = 1, \ldots, V$ **do**
2: **for** $s = 1, \ldots, S$ **do**
3: For MIL, obtain the perturbed prediction with the $s$th bias: $\tilde{\mathbf{z}}_{v,s,t} = \mathbf{z}_{v,t} + b_s$, and project the perturbed prediction into the pseudo-label vector: $\mathbf{y}_{v,s,t} = \pi(\tilde{\mathbf{z}}_{v,s,t})$.
4: For SSL (*resp.*, ROD), obtain the perturbed constraint by setting $\sigma = \sigma_0 + \sigma_s$ (*resp.*, $\kappa = \kappa_0 + \kappa_s$), and project the prediction into the pseudo-label vector based on the $s$th constraint: $\mathbf{y}_{v,s,t} = \pi(\mathbf{z}_{v,t})$.
5: **end for**
6: $\mathcal{O}_{v,t} = \{\mathbf{y}_{v,s,t}|_{s=1}^S\}$
7: **end for**
8: **for** $v = 1, \ldots, V$ **do**
9: Obtain the pseudo-label vector set on the $v$th view: $\mathcal{C}_{t+1}^v = \mathcal{C}_t^v \bigcup \{\mathcal{O}_{j,t}|_{j=1,j\neq v}^V\}$
10: **end for**

**Output:** The pseudo-label vector set on each view: $\mathcal{C}_{t+1}^v$ for $v = 1, \ldots, V$.

## 5 Multi-Layer Multiple Kernel Learning

After generating a pseudo-label vector set for each view, the remaining problem is to learn a robust classifier using the pseudo-label vector set. In this section, we formulate the weakly labeled learning problem on each view as an MKL problem by combining each pseudo-label vector with the input kernel. To train a more robust classifier, we further employ the intrinsic group structure inside the pseudo-label vector set, which leads to a novel multi-layer MKL formulation.

### 5.1 Weakly Labeled Learning via $\ell_1$-MKL

In supervised learning, one can learn a classifier based on regularized empirical risk minimization. Specifically, considering a max-margin classifier $f(\mathbf{x}) = \mathbf{w}'\boldsymbol{\phi}(\mathbf{x}) + b$ with $\boldsymbol{\phi}(\cdot)$ being the feature mapping function and $b$ being the bias, and $\rho$-SVM with the squared hinge loss, one can formulate an optimization problem for supervised learning as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\rho,\xi_i} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + b^2 + C\sum_{i=1}^n \xi_i^2\right) - \rho, \\ \text{s.t.} \quad & g_i(\mathbf{w}'\boldsymbol{\phi}(\mathbf{x}_i) + b) \geq \rho - \xi_i,\ i = 1, \ldots, n, \end{aligned} \tag{4}$$

where $C$ is a tradeoff parameter, $g_i$ is the given label of the $i$th training sample $\mathbf{x}_i$ and $n$ is the number of training

samples. By introducing the dual variable $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]'$ for the constraints in (4), one can write its dual form as:

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} -\frac{\boldsymbol{\alpha}'\boldsymbol{\alpha}}{2C} - \frac{1}{2}\boldsymbol{\alpha}'(\mathbf{K} \odot \mathbf{g}\mathbf{g}')\boldsymbol{\alpha}, \tag{5}$$

where $\mathbf{K} = \hat{\mathbf{K}} + \mathbf{1}\mathbf{1}'$, $\hat{\mathbf{K}} = [k(\mathbf{x}_i, \mathbf{x}_j)] = [\boldsymbol{\phi}(\mathbf{x}_i)'\boldsymbol{\phi}(\mathbf{x}_j)]$ is the input kernel matrix, $\mathbf{g} = [g_1, \ldots, g_n]'$ is the label vector, $\odot$ denotes the element-wise product between two matrices, and $\mathcal{A} = \{\boldsymbol{\alpha} | \boldsymbol{\alpha} \geq \mathbf{0}, \boldsymbol{\alpha}'\mathbf{1} = 1\}$ is the feasible set of $\boldsymbol{\alpha}$.

In the weakly labeled learning scenario, we have a set of pseudo-label vectors instead of a single pseudo-label vector as in co-training. Let us denote the pseudo-label vector set as $\mathcal{C} = \{\mathbf{y}_m | m = 1, \ldots, M\}$, similar to [2], [22], we extend the supervised learning problem in (5) to the weakly labeled learning scenario as follows:

$$\min_{\mathbf{d} \geq 0, \mathbf{1}'\mathbf{d} \leq 1} \max_{\boldsymbol{\alpha} \in \mathcal{A}} -\frac{\boldsymbol{\alpha}'\boldsymbol{\alpha}}{2C} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{m=1}^{M} d_m \mathbf{K} \odot \mathbf{y}_m\mathbf{y}_m'\right)\boldsymbol{\alpha}, \tag{6}$$

where $\mathbf{d} = [d_1, \ldots, d_M]'$ is the combination coefficient vector. In other words, when there are more than one pseudo-label vectors, we optimize the classifier parameter $\boldsymbol{\alpha}$ and simultaneously find an optimal linear combination $\sum_m^M d_m \mathbf{y}_m\mathbf{y}_m'$ to approximate $\mathbf{g}\mathbf{g}'$, where $\mathbf{g}$ is the same ground truth label vector as that in supervised learning.

The problem in (6) can be deemed as an $\ell_1$-norm MKL problem [24] with each base kernel as $\mathbf{K} \odot \mathbf{y}_m\mathbf{y}_m'$. To simplify notation, we define $\mathbf{Q}_m = \mathbf{K} \odot \mathbf{y}_m\mathbf{y}_m'$, which is referred to as an *input-output kernel* as in [21]. The traditional MKL problem [23], [24], [40], [41] aims to find an optimal linear combination of input base kernels $\mathbf{K}_m$'s with a single label vector. In contrast, we aim to learn the optimal linear combination of input-output kernels in our weakly labeled learning scenario in order to effectively integrate these kernels that are decided by a set of pseudo-label vectors. The problem in (6) can be solved by using existing solvers such as [24].

### 5.2 Weakly Labeled Learning via Multi-Layer MKL

As shown in Section 4, the pseudo-label vectors on each view are generated according to different views, biases, and iterations. In Section 5.1, we formulate the weakly labeled learning problem on each view as an $\ell_1$-norm MKL problem, in which we ignore the different ways that the pseudo-label vectors are generated. By putting the pseudo-label vectors generated in the same way into a group, we can organize the pseudo-label vectors into three dimensions in terms of view, bias, and iteration.

To effectively utilize and capture such a group structure when learning the classifier, we propose to use different regularizers on the combination coefficients at different layers, which leads to a multi-layer MKL problem. In this section, we study a general multi-layer MKL problem and also propose an efficient solution. Since our multi-view weakly labeled learning problem has a three-layer structure, we take the three-layer structure as an example (see Fig. 2) to introduce the objective function and the solution of our multi-layer MKL.

Considering the three-layer case, we have the pseudo-label vector set as discussed in Section 4. For each view, we will learn an MKL classifier by using the corresponding $(V-1) \times S \times t$ pseudo-label vectors. Let us denote the total number of iterations at the current iteration as $T$, the total number of the other views as $J$ with $J = V - 1$. We can organize the pseudo-label vectors and uniformly refer to the pseudo-label vector set for the $t$th iteration and the $v$th view as $\mathcal{C} = \{\mathbf{y}_{j,s,l}\}$, where $l = 1, \ldots, T$, $s = 1, \ldots, S$, and $j = 1, \ldots, J$ are the indices for the iteration, bias and view, respectively. Correspondingly, the input-output kernel is defined as $\mathbf{Q}_{j,s,l} = \mathbf{K} \odot (\mathbf{y}_{j,s,l}\mathbf{y}_{j,s,l}')$.

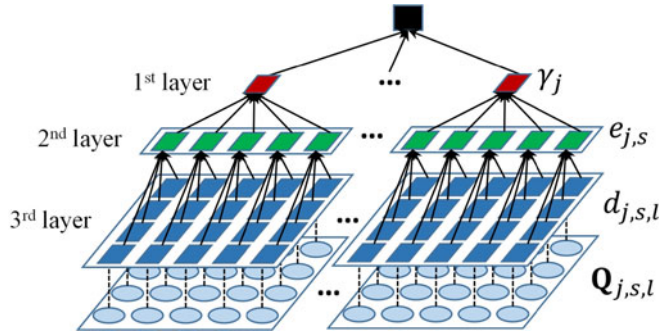



Fig. 2. Organization of the three-layer structure for the multi-layer MKL in our co-labeling framework. The circles denote the input-output kernels, and the rectangles with different colors denote the combination coefficients at different layers, respectively. We impose different regularizers on the combination coefficients at each layer when combining them to obtain their parent node (e.g., from the blue rectangles to the green rectangles).

Inspired by [21], we propose to exploit the inherent group structure on the input-output kernels by enforcing a multi-layer group regularization on the kernel combination coefficients. We formulate our three-layer Multiple Kernel Learning problem as follows:

$$\min_{\mathbf{D} \in \mathcal{D}} \max_{\boldsymbol{\alpha}} -\frac{\boldsymbol{\alpha}'\boldsymbol{\alpha}}{2C} - \frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}\boldsymbol{\alpha}, \tag{7}$$

where $\mathbf{Q} = \sum_{j=1}^{J}\sum_{s=1}^{S}\sum_{l=1}^{T} d_{j,s,l}\mathbf{Q}_{j,s,l}$, $\mathbf{D} \in \mathbb{R}^{J \times S \times T}$ is a third-order tensor with each element $\mathbf{D}(j,s,l) = d_{j,s,l}$, and $\mathcal{D} = \{\mathbf{D} | d_{j,s,l} \geq 0, \Omega(\mathbf{D}) \leq 1\}$ with $\Omega(\mathbf{D}) = \|\mathbf{D}\|_{p_1,p_2,p_3}$ as the $\ell_{p_3,p_2,p_1}$-norm on $\mathbf{D}$ defined as follows: $\Omega(\mathbf{D}) = \|\mathbf{D}\|_{p_1,p_2,p_3} = (\sum_{j=1}^{J}(\sum_{s=1}^{S}(\sum_{l=1}^{T}(d_{j,s,l})^{p_3})^{\frac{p_2}{p_3}})^{\frac{p_1}{p_2}})^{\frac{1}{p_1}}$.

The $\ell_{p_3,p_2,p_1}$-norm on $\mathbf{D}$ allows us to impose different regularizers on different layers to cope with different prior information. As shown in Fig. 2, each blue circle is an input-output kernel $\mathbf{Q}_{j,s,l}$ in our multi-view weakly labeled learning problem, and each rectangle in the third layer is the combination coefficient $d_{j,s,l}$. At the third layer, for any given $j, s$, we impose the $\ell_{p_3}$-norm on the coefficients $\{d_{j,s,l}|_{l=1}^{T}\}$ that share the same parent, i.e., $(\sum_{l=1}^{T}(d_{j,s,l})^{p_3})^{1/p_3} = e_{j,s}$, where $e_{j,s}$ is denoted by a green rectangle in Fig. 2. Similarly, at the second layer, for any given $j$, we impose the $\ell_{p_2}$-norm on $\{e_{j,s}|_{s=1}^{S}\}$ that share the same parent, $(\sum_{s=1}^{S}(e_{j,s})^{p_2})^{1/p_2} = \gamma_j$ with $\gamma_j$ being denoted by red rectangles in Fig. 2. And we impose the $\ell_{p_1}$-norm on $\{\gamma_j|_{j=1}^{J}\}$ at the first layer. In our work, the base input-output kernels in the first layer, second layer and third layer are constructed by using the pseudo-label vectors from different views, biases and iterations, respectively. Other possible orders for the multi-layer

structure can also be used, but we found that they achieve similar results. Therefore we fix the order as in Fig. 2 in all the experiments.

Different norms may introduce different levels of sparsity on the kernel combination coefficients [24]. For example, if one uses $\ell_1$-norm as in the traditional $\ell_1$-norm MKL (see Section 5.1), it usually leads to a sparse solution for the kernel combination coefficients. Similarly, an $\ell_p$-norm ($1 < p < \infty$) may lead to a denser solution, and the $\ell_\infty$-norm will result in a uniform solution for the kernel combination coefficients. Therefore, with different norm parameters for different layers, our model can better cope with the intrinsic structure in these input-output kernels, making us learn a more robust classifier.

### 5.3 Solution to Multi-layer MKL

By defining $\tilde{\varphi}_{j,s,l}(\cdot)$ as the corresponding non-linear mapping function induced from the input-output kernel matrix $\mathbf{Q}_{j,s,l}$ (*i.e.*, $\mathbf{Q}_{j,s,l}(i,\tilde{i}) = \tilde{\varphi}_{j,s,l}(\mathbf{x}_i)'\tilde{\varphi}_{j,s,l}(\mathbf{x}_{\tilde{i}})$), we write the primal form of (7) as follows:

$$\begin{aligned} \min_{\tilde{\mathbf{w}}_{j,s,l},\mathbf{D},\rho,\xi_i} \quad & \frac{1}{2}\left(\sum_{j,s,l}\frac{||\tilde{\mathbf{w}}_{j,s,l}||^2}{d_{j,s,l}} + C\sum_{i=1}^{n}\xi_i^2\right) - \rho, \\ \text{s.t.} \quad & \sum_{j,s,l}\tilde{\mathbf{w}}'_{j,s,l}\tilde{\varphi}_{j,s,l}(\mathbf{x}_i) \geq \rho - \xi_i, \quad \forall i, \\ & \Omega(\mathbf{D}) \leq 1, \quad d_{j,s,l} \geq 0, \quad \forall j,s,l. \end{aligned} \tag{8}$$

The derivation follows the Lagrangian multiplier method used in [24] and thus it is omitted here. The formulation in (8) is a convex optimization problem, therefore the global optimum is guaranteed. To solve this problem, we alternatingly optimize two subproblems with respect to the two sets of variables $\{\tilde{\mathbf{w}}_{j,s,l}, \rho, \xi_i\}$ and $\{\mathbf{D}\}$ as in [23], [24], [42], but we propose a new recursive updating strategy to solve $\{\mathbf{D}\}$.

#### 5.3.1 Updating SVM Variables with Fixed $\mathbf{D}$

With a fixed $\mathbf{D}$, we introduce the Lagrangian multipliers $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]'$ and write the dual of (8) with respect to other primal variables $\{\tilde{\mathbf{w}}_{j,s,l}, \rho, \xi_i\}$ as:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{\boldsymbol{\alpha}'\boldsymbol{\alpha}}{2C} - \frac{1}{2}\boldsymbol{\alpha}'\left(\sum_{j,s,l} d_{j,s,l}\mathbf{Q}_{j,s,l}\right)\boldsymbol{\alpha}, \tag{9}$$

which is a standard quadratic programming (QP) problem with $\mathcal{A} = \{\boldsymbol{\alpha}|\boldsymbol{\alpha}'\mathbf{1} = 1, \mathbf{0} \leq \boldsymbol{\alpha}\}$. Thus it can be efficiently solved by any existing QP solvers. Then, the primal variables $\tilde{\mathbf{w}}_{j,s,l}, \rho, \xi_i$ can be recovered accordingly. Thus, the squared $\ell_2$-norm of $\tilde{\mathbf{w}}_{j,s,l}$ can be obtained as:

$$||\tilde{\mathbf{w}}_{j,s,l}||^2 = (d_{j,s,l})^2\boldsymbol{\alpha}'\mathbf{Q}_{j,s,l}\boldsymbol{\alpha}. \tag{10}$$

#### 5.3.2 Updating $\mathbf{D}$ with Fixed SVM Variables

With fixed SVM variables $\{\tilde{\mathbf{w}}_{j,s,l}, \rho, \xi_i\}$, the problem for updating $\mathbf{D}$ is as follows:

$$\begin{aligned} \min_{\mathbf{D}} \quad & \frac{1}{2}\sum_{j=1}^{J}\sum_{s=1}^{S}\sum_{l=1}^{T}\frac{||\tilde{\mathbf{w}}_{j,s,l}||^2}{d_{j,s,l}} \\ \text{s.t.} \quad & \|\mathbf{D}\|_{p_1,p_2,p_3} \leq 1, \quad d_{j,s,l} \geq 0, \forall j,s,l. \end{aligned} \tag{11}$$

The most challenging problem in solving (11) comes from the $\ell_{p_3,p_2,p_1}$-norm constraint on $\mathbf{D}$. We observe that there is a multi-layer structure for our MKL problem (see Fig. 2), namely, some of the nodes at the higher layers (e.g., the third layer) share the same parent at the lower layer (e.g., the second layer). So we propose a new recursive updating strategy, which can also be applied to multi-layer MKL with more than three layers. We sketch the main idea here, and also provide the detailed derivations and proof in Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2015.2476813.

Let us denote $\nu_{j,s,l} = \|\tilde{\mathbf{w}}_{j,s,l}\|^2$ for simplicity, and define a matrix $\mathbf{E} \in \mathbb{R}^{J\times S}$ with $\mathbf{E}(j,s) = e_{j,s}$ where $e_{j,s}$ is defined as in Section 5.2 and Fig. 2. Then the regularizer in (11) can be written as $\Omega(\mathbf{D}) = \|\mathbf{E}\|_{p_1,p_2}$. Correspondingly, the problem in (11) can be reduced to a two-layer MKL as follows:

$$\begin{aligned} \min_{\mathbf{E}} \quad & \frac{1}{2}\sum_{j=1}^{J}\sum_{s=1}^{S}\frac{\lambda_{j,s}}{e_{j,s}} \\ \text{s.t.} \quad & \|\mathbf{E}\|_{p_1,p_2} \leq 1, \ e_{j,s} \geq 0, \ \forall j,s, \end{aligned} \tag{12}$$

where we have

$$\lambda_{j,s} = \left(\sum_{l=1}^{T}\nu_{j,s,l}^{\frac{p_3}{1+p_3}}\right)^{\frac{1+p_3}{p_3}}. \tag{13}$$

Similarly, let us define a vector $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_J]'$, where $\gamma_j$ is defined in Section 5.2 and Fig. 2. We further reduce the problem in (12) to a one-layer MKL as: $\min_{\|\boldsymbol{\gamma}\|_{p_1}\leq 1,\boldsymbol{\gamma}\geq\mathbf{0}}\frac{1}{2}\sum_{j=1}^{J}\frac{\eta_j}{\gamma_j'}$ where we also have

$$\eta_j = \left(\sum_{s=1}^{S}\lambda_{j,s}^{\frac{p_2}{1+p_2}}\right)^{\frac{1+p_2}{p_2}}. \tag{14}$$

For the given $\eta_j$'s, we can easily obtain $\gamma_j$ by solving the one-layer problem in closed form as in [24].

Therefore, to solve the multi-layer problem in (11), we first calculate $\nu_{j,s,l} = \|\tilde{\mathbf{w}}_{j,s,l}\|^2$ for the nodes in the third layer by using (10). Then we use $\nu_{j,s,l}$ to calculate $\lambda_{j,s}$ and $\eta_j$ for the nodes in the second and first layers by using (13) and (14), respectively. After reaching the first layer, we can calculate $\gamma_j$ in closed form by solving the one-layer MKL problem using [24], and then we recursively solve $e_{j,s}$ and $d_{j,s,l}$ as described in the following proposition:

**Proposition 1.** *Let us define a function $G(a,b,p) = \left(\frac{a}{b}\right)^{\frac{1}{p+1}}$. Then we obtain the optimal solution for the subproblem (11) as the following analytical form:*

$$d_{j,s,l} = G(\nu_{j,s,l}, \ \lambda_{j,s}, \ p_3)\, e_{j,s}, \tag{15}$$

$$where \qquad e_{j,s} = G(\lambda_{j,s}, \ \eta_j, \ p_2)\ \gamma_j, \tag{16}$$

$$\gamma_j = G(\eta_j, \ \tau, \ p_1), \tag{17}$$

*where $\nu_{j,s,l} = \|\tilde{\mathbf{w}}_{j,s,l}\|^2$, $\lambda_{j,s}$ and $\eta_j$ are calculated using (13) and (14), respectively, and $\tau = (\sum_{j=1}^{J}\eta_j^{\frac{p_1}{1+p_1}})^{\frac{1+p_1}{p_1}}$.*

The optimization methods for learning the kernel combination coefficients in [24] and [42] are all based on the analysis of the KKT conditions, but how to apply the method directly to structures with more than two layers is unclear as it is too complex to directly calculate the derivatives with respect to the kernel combination coefficients. In contrast, our solution is based on the new recursive updating strategy from the introduction of the intermediate variables, thus it can be easily used for three or more layer structures. Moreover, the objective function in (8) for our multi-layer MKL is jointly convex with respect to the kernel combination coefficients and the SVM primal variables, therefore our algorithm enjoys similar convergence properties as $\ell_p$-MKL [24].

### 5.4 Overall Optimization Procedure for Co-Labeling

The whole optimization procedure for our co-labeling algorithm is listed in Algorithm 3. A total number of $V$ classifiers are trained from our co-labeling algorithm. The initial pseudo-label vector set $\mathcal{C}_1^v$ for the $v$th view is problem dependent. For example, in SSL and ROD we can use the predictions from the classifiers trained on the labeled data to generate the pseudo-label vectors as in Section 4, while in MIL we initialize all instances in positive bags as positive and all instances in negative bags as negative. The algorithm is composed of two main loops. In the outer loop, we iteratively update $\mathcal{C}^v$ for each view by using Algorithm 2, while in the inner loop we learn the classifiers by using our proposed multi-layer MKL as discussed in Section 5.3. After obtaining the final classifier for each view, the final decision value is calculated by fusing the decision values from all the $V$ views.

**Algorithm 3.** Co-Labeling Algorithm

1: Initialize the pseudo-label vector set $\mathcal{C}_1^v$ for each view and set $t = 1$.
2: **repeat**
3: **for** $v = 1 : V$ **do**
4: Construct $\mathbf{Q}_{j,s,l}$ using $\mathbf{K}$ and $\mathcal{C}_t^v$ as in Section 5.2 and initialize $\mathbf{D}$ with equal weights.
5: **repeat**
6: Obtain $\boldsymbol{\alpha}$ by solving the subproblem (9) using the standard QP solver with $\mathbf{D}$.
7: Calculate $||\tilde{\mathbf{w}}_{j,s,l}||^2$ according to (10) and update $\mathbf{D}$ by solving (11).
8: **until** The objective in (8) converges.
9: **end for**
10: Update the pseudo-label vector set $\mathcal{C}_{t+1}^v$ using Algorithm 2.
11: Set $t \leftarrow t + 1$
12: **until** The objectives of all views converge.
13: **return** The classifiers $\{f^v|_{v=1}^V\}$ and the final classifier is obtained by fusing all the V classifiers.

*Convergence:* For each view, we solve an MKL problem which minimizes the objective function in (8) with respect to the SVM primal variables and $\mathbf{D}$ (see Algorithm 3). Note we add the new pseudo-label vectors into the set $\mathcal{C}^v$ at each iteration. So, in the worst case the optimal solution of MKL at the current iteration should be the same one at the previous iteration by setting the entries in the coefficient vector $\mathbf{D}$ corresponding to the newly added pseudo-label vectors to zeros. Therefore the objective values of our MKL problem on each view in (8) should not increase as the number of iterations increases. According to our experiments, our algorithm often stops within around 10 iterations.

*Time complexity:* The main cost in Algorithm 3 is from the training process of multi-layer MKL. Let us denote the time complexity for training MKL as $O(\mathrm{MKL})$. Then the total time complexity of Algorithm 3 is $T \cdot V \cdot O(\mathrm{MKL})$, where $V$ is the total number of views and T is the number of iterations.

Note the time complexity for MKL training has not been theoretically analyzed. Usually, the MKL solver needs to train an SVM for a few iterations. The empirical analysis shows the time complexity for optimizing the QP problem in SVM is $O(n^{2.3})$, where $n$ is the number of training samples. Therefore, the complexity of MKL is $O(\tilde{t}n^{2.3})$, where $\tilde{t}$ is the number of iterations in MKL.

*Generalization bound analysis:* We also analyze the generalization bound of our co-labeling algorithm in Appendix B, available online. Specifically, we first give the generalization bound of our weakly labeled learning method on each view, and then present the generalization bound for the final classifier.

## 6 Experiments

In this section, we evaluate our co-labeling approach on five real-world datasets for four cases: 1) Two-view Multiple Instance Learning, 2) Two-view Semi-Supervised Learning, 3) Multi-view Semi-Supervised Learning and 4) Multi-view Relative Outlier Detection. Note Well-SVM [7] can handle different types of weakly labeled data in the single view setting, we treat it as the most related baseline for comparison. We also compare our co-labeling approach with the related state-of-the-art methods for each learning task (see details of those baseline methods in Sections 6.1, 6.2, and 6.3).

For our co-labeling approach with multi-layer MKL, different norm parameters for different layers can represent different prior information for the corresponding layer. First, we iteratively update the label candidate sets. The labels from different iterations may be quite different and only the labels from a limited number of iterations are close to the ground-truth labels. Thus we prefer a sparse regularizer for the iterations, and we use an $\ell_1$-norm on the iteration layer so that only the pseudo-label vectors from a few iterations should be used for the final prediction. Second, in our co-labeling algorithm, multiple bias terms are used to cope with the prediction biases. Considering that the pseudo-label vectors from different biases within a certain region should be somewhat similar, we use an $\ell_2$-norm on the biases, which leads to a denser solution. So we can better utilize multiple bias terms to enhance the robustness of the learnt classifier. Third, as the labels from other views may contain complementary information, we thus expect that they are equally important. Therefore, an $\ell_\infty$-norm is utilized for the view layer in our model. While it is possible to tune other values, we finally utilize the $\ell_{1,2,\infty}$-norm MKL on each view.

To extensively evaluate our proposed method, we report the results using $\ell_1$-MKL as in our preliminary work [20] as *CoL(1-layer)*, and we also refer to our co-labeling

TABLE 1
Mean Average Precisions over 81 Concepts from Different Methods on the NUS-WIDE Dataset

| | #bag = 15 | #bag = 20 | #bag = 25 |
|---|---|---|---|
| MIL-CPB | 72.24 | 73.00 | 73.22 |
| mi-SVM | 79.68 | 79.96 | 80.11 |
| sMIL | 73.24 | 74.16 | 74.80 |
| WellSVM | 74.91 | 75.49 | 76.12 |
| CoL(1-layer) | 80.62 | 80.95 | 81.64 |
| CoL(2-layer) | **81.15** | **81.33** | **82.03** |

*#bag denotes the number of positive/negative training bags.*

algorithm with $\ell_{1,2,2}$-norm MKL as *CoL(2-layer)*, and the one with $\ell_{1,2,\infty}$-norm MKL as *CoL(3-layer)*. As mentioned in Section 5, in the two-view settings, the pseudo-label vector on one view is constructed by using the predictions from only one view (i.e., the other view), and there is only one node at the first layer (see Fig. 2). As a result, our three-layer MKL for solving the weakly labeled learning problem on each view becomes a two layer-MKL, thus CoL(3-layer) reduces to CoL(2-layer) in this case. So we only report CoL(1-layer) and CoL(2-layer) for the two view settings.

### 6.1 Two-View Multiple Instance Learning

MIL has been successfully used for Text-Based Image Retrieval (TBIR) [5], so we evaluate our co-labeling approach for TBIR under the two-view setting. Similar to [5], we conduct the experiment on the large-scale NUS-WIDE dataset [43], which consists of 269,648 images from 81 annotated concepts collected from the website *Flickr.com*. Two types of features are extracted,

- 1) **The textual feature** is extracted from the tags associated with each image, in which the vocabulary is constructed by using the top 1,000 words with the highest frequency. Then, a 1,000 dimensional term-frequency feature is extracted for each image.
- 2) **The DeCAF$_6$ feature** is extracted by using the output from the sixth layer of the CNN model in [44].

We treat each type of feature as one view, and use the Gaussian kernel for each view with the bandwidth parameter as the mean of squared distances between all training samples.

We compare our co-labeling approach with WellSVM and other state-of-the-art MIL methods, MIL-CPB [5], mi-SVM [3] and sMIL [4], which have achieved the best performances on the NUS-WIDE dataset as reported in [5]. Since those works are single-view methods, we use the late fusion strategy to average the decision values from the classifiers on different views. We also employ the early fusion strategy for these methods by using the average kernel, which are worse than or only comparable to the results using the late fusion strategy.

For all methods, we construct $15, 20$, and $25$ positive bags using the top-ranked relevant images and the same number of negative bags using randomly selected irrelevant images, in which each bag contains 15 instances. For performance evaluation, the non-interpolated Average Precision (AP) is used as the performance metric. Mean Average Precision (MAP) is the mean of the APs over all the concepts/classes. A binary classifier is trained for each concept, and the top-100 MAPs are reported in the experiments as in [5].

TABLE 2
Summarization of the Datasets Used in Two-View SSL

| Datasets | d1 | d2 | $\#c$ | $\#l$ | $\#u$ | $\#t$ |
|---|---|---|---|---|---|---|
| BBC | 4,817 | 4,818 | 5 | 10 | 1,104 | 1,111 |
| BBCSport | 2,306 | 2,307 | 5 | 10 | 360 | 367 |

*d1 and d2 are the feature dimensions of two views. $\#c$, $\#l$, $\#u$ and $\#t$ are the numbers of classes, labeled training data, unlabeled training data and test data, respectively.*

The MAPs over 81 concepts for different methods on the NUS-WIDE dataset are reported in Table 1. The mi-SVM method outperforms other baseline methods MIL-CPB, sMIL, and WellSVM, which indicates the simple approach works well on this dataset by iteratively training the SVM classifier and inferring the labels of training instances. We also observe that the results of all methods become higher, when the number of positive/negative training bags increases. Our co-labeling approaches consistently outperform the existing MIL methods when using different number of positive/negative training bags, which clearly demonstrates the effectiveness of our methods for combining information from two views. Moreover, CoL(2-layer) is better than CoL(1-layer), which indicates it is beneficial to employ group structure information among the base input-output kernels associated with the pseudo-label vectors.

### 6.2 Semi-Supervised Learning

For SSL, we compare our proposed co-labeling approach with WellSVM [7] as well as the following state-of-the-art baselines:

- SVM, the standard SVM trained with the labeled training data, which is a commonly used baseline in semi-supervised learning;
- Co-Training [8], the original Co-Training algorithm;
- Co-LapSVM [11], the Laplacian SVM method for the multi-view setting,[1] in which the Laplacian matrices from all views are averaged to obtain a common Laplacian matrix for each view;
- TSVM [45], the transductive SVM[2] method trained with the labeled and unlabeled data for each view;
- PMC [46], an improved version of co-training[3] which is designed to automatically split the single feature vector into two views for sample selection, and finally it only outputs one classifier for prediction. We apply PMC on the concatenated feature vector from all the views.

For all the methods except PMC, the classifiers from all views are fused with equal weights to obtain the final prediction in the late fusion fashion, unless stated otherwise.

1. Code available at: http://manifold.cs.uchicago.edu/manifold_regularization/software.html
2. Code available at: http://mloss.org/software/view/19/
3. Code available at: www.cse.wustl.edu/~mchen/code/pmc.tar

TABLE 3
MAPs (Means $\pm$ Standard Deviations (%)) over Five Classes for Different Methods on the BBC and BBCSport Datasets

| | BBC | | | BBCSport | | |
|---|---|---|---|---|---|---|
| | View1 | View2 | View1+View2 | View1 | View2 | View1+View2 |
| SVM | $68.37 \pm 3.80$ | $65.76 \pm 3.91$ | $76.46 \pm 3.45$ | $73.90 \pm 3.22$ | $68.90 \pm 2.62$ | $80.65 \pm 3.38$ |
| TSVM | $71.99 \pm 5.48$ | $66.83 \pm 3.54$ | $76.33 \pm 4.06$ | $74.62 \pm 5.73$ | $65.51 \pm 3.36$ | $78.22 \pm 3.05$ |
| WellSVM | $80.41 \pm 5.57$ | $74.95 \pm 3.16$ | $85.24 \pm 3.24$ | $75.57 \pm 6.37$ | $72.67 \pm 5.79$ | $81.29 \pm 5.21$ |
| Co-LapSVM | $77.44 \pm 3.36$ | $75.44 \pm 4.82$ | $84.09 \pm 3.56$ | $74.57 \pm 3.50$ | $70.84 \pm 2.46$ | $81.70 \pm 3.63$ |
| Co-Training | $88.47 \pm 7.79$ | $84.44 \pm 8.05$ | $88.62 \pm 8.04$ | $89.25 \pm 4.38$ | $85.60 \pm 4.06$ | $90.31 \pm 4.89$ |
| PMC | — | — | $89.64 \pm 5.16$ | — | — | $88.44 \pm 4.38$ |
| CoL(1-layer) | $92.77 \pm 4.19$ | $91.60 \pm 3.10$ | $94.77 \pm 3.69$ | $91.87 \pm 2.20$ | $90.83 \pm 1.56$ | $94.81 \pm 1.41$ |
| CoL(2-layer) | $\mathbf{93.88 \pm 3.36}$ | $\mathbf{93.02 \pm 2.74}$ | $\mathbf{95.65 \pm 3.12}$ | $\mathbf{92.51 \pm 2.63}$ | $\mathbf{91.91 \pm 1.32}$ | $\mathbf{95.44 \pm 1.63}$ |

*Results in boldface are significantly better than the others, judged by the t-test with a significance level at 0.05.*

#### 6.2.1 Two-View SSL

We evaluate our co-labeling approach for two-view semi-supervised learning for news classification on the BBC and BBCSport datasets [47]. The details of these two datasets are summarized in Table 2 and described in the following.

The two datasets contain news articles collected from the BBC.[4] The BBC dataset contains 2,225 documents from five topics (business, entertainment, politics, sports and technology) and the BBCSport dataset consists of 737 sports news documents from five classes (athletics, cricket, football, rugby and tennis), respectively. Following [14], we randomly partition each original feature into two views, and each view is normalized such that its $\ell_1$-norm is equal to one. For each view, we use the linear kernel for all the methods. We partition the datasets into the training set and the test set, each of which contains 50 percent of the documents per class. Two labeled samples from each class are further selected from the training set, and all the remaining data in the training set are utilized as unlabeled data for the training process. We perform the experiments ten times based on different data partitions, and report the MAPs (means$\pm$standard deviations) in Table 3.

From the results, we have the following observations in terms of the means of MAPs. Our co-labeling methods outperform the existing SSL methods for each single view and for the joint view, which demonstrates the effectiveness of our co-labeling approach for two-view semi-supervised learning. Moreover, we also observe that CoL(2-layer) outperforms CoL(1-layer), which again demonstrates the effectiveness of our proposed multi-layer MKL for utilizing the group structure among the base kernels associated with the pseudo-label vectors.

We also conduct the experiments for the single-view learning methods SVM, TSVM and WellSVM by using the original features. The results for SVM, TSVM and WellSVM are $76.85 \pm 3.61$, $75.72 \pm 3.16$, and $90.91 \pm 3.36$ (*resp.*, $80.80 \pm 3.61$, $79.21 \pm 6.43$, and $84.82 \pm 7.28$) on the BBC (*resp.*, BBCSport) dataset. It is interesting that SVM and WellSVM achieve better results by using the early fusion strategy with the original features on both datasets, which demonstrates the two methods can benefit from accessing all the features in the learning process for this application. However, those results are still worse than our co-labeling methods.

4. Features available at: http://mlg.ucd.ie/datasets/bbc.html

#### 6.2.2 Multi-View SSL

We also evaluate our co-labeling approach for multi-view semi-supervised learning on the Reuters multilingual dataset [48], which is from the Reuters RCV1 and RCV2 collections. The task is to classify the documents written in five languages, *English*, *French*, *German*, *Italian* and *Spanish* into different categories. Following [48], a total number of 6 classes (*e.g.*, *C15 (Performance)*, *CCAT (Corporate/Industrial)*, *E21 (Government Finance)*, *ECAT (Economics)*, *GCAT (Government Social)*, *M11 (Equity Markets)*) are utilized for performance evaluation. The documents belonging to more than one class are annotated using the label of their smallest class. Each document from the corresponding corpus has been translated to the other four languages by using the statistical machine translation system PORTAGE. Detailed information of this dataset is shown in Table 4.

In order to perform the multi-view learning task, we uniformly divide the feature vector of the original language and the four corresponding translated languages into three parts. For each view, we use the linear kernel for all the methods. In this way, we can obtain a total number of 15 views for the learning problem. For each class of each language, a total number of 14 documents are selected as the labeled training data, thus a total number of 84 documents are used as the labeled training data for each language. Moreover, another 2,916 samples are utilized as the unlabeled data. So a total number of 3,000 documents are used to train the classifiers. For each class of each language, the binary one-versus-others classifiers are trained for performance evaluation, and the experiments are repeated five times with different data partitions. The numbers of training and testing samples are also summarized in Table 4.

TABLE 4
Summarization of the Reuters Multilingual Dataset Used in Multi-View SSL

| Language | $\#dim$ | $\#docs$ | $\#c$ | $\#l$ | $\#u$ | $\#t$ |
|---|---|---|---|---|---|---|
| English | 21,531 | 18,758 | 6 | 84 | 2,916 | 18,674 |
| French | 24,893 | 26,648 | 6 | 84 | 2,916 | 26,564 |
| Spanish | 11,547 | 12,342 | 6 | 84 | 2,916 | 12,258 |
| German | 34,279 | 29,953 | 6 | 84 | 2,916 | 29,869 |
| Italian | 15,506 | 24,039 | 6 | 84 | 2,916 | 23,955 |

*$\#dim$ is the dimension of the document, while $\#docs$, $\#c$, $\#l$, $\#u$ and $\#t$ are the numbers of documents, classes, labeled training instances, unlabeled training instances and test instances, respectively.*

TABLE 5
MAPs (Means $\pm$ Standard Deviations (%)) over Six Classes and All Five Languages from Different Methods on the Reuters Multilingual Dataset

| | SVM | TSVM | WellSVM | Co-LapSVM | Co-Training | PMC | CoL(1-layer) | CoL(2-layer) | CoL(3-layer) |
|---|---|---|---|---|---|---|---|---|---|
| MAP | $66.79 \pm 1.11$ | $69.34 \pm 1.22$ | $50.02 \pm 0.92$ | $69.34 \pm 0.82$ | $42.00 \pm 2.36$ | $67.20 \pm 0.76$ | $69.33 \pm 1.71$ | $71.73 \pm 1.25$ | $\mathbf{72.45 \pm 1.12}$ |

*Results in boldface are significantly better than the others, judged by the t-test with a significance level at 0.05.*

TABLE 6
APs (Means $\pm$ Standard Deviations (%)) from Different Methods on the 20 Newsgroups Data Set When Using Different Number of Normal Training Documents (i.e., $N$)

| $N$ | WellSVM-ROD | LSOD | MLOD | CoL(1-layer) | CoL(2-layer) | CoL(3-layer) |
|---|---|---|---|---|---|---|
| 400 | $45.35 \pm 0.78$ | $42.41 \pm 0.21$ | $44.45 \pm 0.93$ | $45.80 \pm 1.31$ | $46.79 \pm 1.23$ | $\mathbf{47.30 \pm 1.38}$ |
| 1,200 | $45.08 \pm 1.24$ | $42.29 \pm 0.17$ | $44.83 \pm 0.27$ | $46.82 \pm 0.91$ | $47.26 \pm 1.24$ | $\mathbf{47.65 \pm 1.51}$ |
| 2,000 | $47.33 \pm 4.36$ | $42.29 \pm 0.23$ | $44.74 \pm 0.22$ | $47.90 \pm 2.06$ | $49.90 \pm 2.07$ | $\mathbf{50.47 \pm 1.94}$ |

*Results in boldface are significantly better than the others, judged by the t-test with a significance level at 0.05.*

The means and the standard deviations of MAPs over six classes and all five languages for different methods on the multilingual dataset are reported in Table 5. We have the following observations in terms of the means of MAPs over six classes and all five languages:

- Our co-labeling approaches CoL(2-layer) and CoL(3-layer) after considering the group structure outperforms the existing semi-supervised learning methods, which demonstrates the effectiveness of the proposed methods.
- On the multi-view SSL setting, we also observe that CoL(2-layer), which partially employs the group structure, outperforms CoL(1-layer). By fully considering the group structure on all the three layers, CoL (3-layer) achieves the best results. These results again demonstrate that it is beneficial to employ different regularizers at different layers to capture the inherent group structure among the base input-output kernels.

We also conduct the experiments for the single view learning methods by concatenating the features from all views. The results for SVM, TSVM, and WellSVM are $66.62 \pm 0.96$, $66.63 \pm 0.98$, and $67.83 \pm 0.99$, respectively. Those results are still worse than our co-labeling approaches.

### 6.3 Multi-View Relative Outlier Detection

The 20 Newsgroups Data Set[5] contains 18,774 news documents from 20 subcategories, in which 11,269 news documents are used as the training set, and the remaining 7,505 news documents are used as the test set. Each news document is represented using the word-frequency feature and its feature dimension is 61,188. If we regard the news documents from some groups as the normal patterns, and the news documents from other groups as outliers, the news document classification problem can be treated as an outlier detection task. By additionally using a set of labeled normal news documents in the training set, we can further formulate this problem as a relative outlier detection problem.

In our experiments, we treat the samples from the first 10 subcategories as the normal documents, and the samples from the remaining 10 subcategories as the outliers. We then construct the labeled reference set by using $N$ normal training documents, and use another $N$ normal training documents as unlabeled data. Also, another $N/9$ outlier documents in the training set are used as unlabeled training data, such that the outlier ratio for unlabeled training data is $1/10$. In our experiments, we set $N = 400, 1{,}200$, and $2{,}000$, respectively. The test data set is used to evaluate all the algorithms, and the mean of APs over 10 rounds of experiments is reported for performance evaluation. In order to perform the multi-view learning task, we also uniformly divide the feature vector of each news document into three parts. In this way, we can obtain a total number of three views for the learning algorithms, and the linear kernel is used for each view in our method.

While the ROD task is not discussed in [7], we can still apply WellSVM to this task, which is referred to as WellSVM-ROD here. We also compare our work with the state-of-the-art ROD methods MLOD [36] and LSOD [37] for the ROD task. The parameters of MLOD and LSOD are set by using their leave-one-out cross validation strategies [36], [37].

From the results shown in Table 6, we have the following observations. First, our co-labeling approach CoL(3-layer) again achieves the best performances for multi-view relative outlier detection when using different number of normal news documents. Second, our CoL(3-layer) outperforms CoL(2-layer) and CoL(1-layer).

## 7 Conclusions

To effectively utilize different types of multi-view weakly labeled data, in this paper we have studied a new problem called multi-view weakly labeled learning, which covers various weakly labeled learning problems including SSL, MIL and ROD under the multi-view setting. We firstly propose a co-labeling framework to solve the multi-view weakly labeled learning problem using pseudo-label vectors. For each view, we propose a novel multi-layer MKL formulation to train a more robust classifier based on a set of input-output kernels associated with the pseudo-label vectors generated from different iterations, biases and views. Extensive experimental results for MIL, SSL and ROD on the real-world multi-view datasets demonstrate that our proposed approach achieves state-of-the-art results.

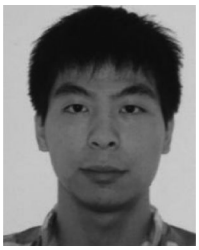
5. Data available at: http://qwone.com/~jason/20Newsgroups/

## ACKNOWLEDGMENTS

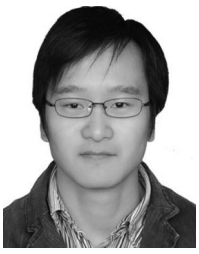
This research was supported by funding from the Faculty of Engineering & Information Technologies, The University of Sydney, under the Faculty Research Cluster Program. This research was also partially supported by the Australian Research Council Future Fellowship FT130100746. Xinxing Xu and Wen Li contributed equally to this work.

## REFERENCES

[1] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th Int. Conf. Mach. Learn.*, 1999, pp. 200–209.
[2] Y.-F. Li, J. T. Kwok, and Z.-H. Zhou, "Semi-supervised learning using label mean," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 633–640.
[3] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 561–568.
[4] R. C. Bunescu and R. J. Mooney, "Multiple instance learning for sparse positive bags," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 105–112.
[5] W. Li, L. Duan, D. Xu, and I. W.-H. Tsang, "Text-based image retrieval using progressive multi-instance learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2049–2055.
[6] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels," in *Proc. Int. Conf. Mach. Learn.*, 2002, pp. 179–186.
[7] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, "Convex and scalable weakly labeled SVMs," *J. Mach. Learn. Res.*, vol. 14, pp. 1391–1445, 2013.
[8] A. Blum and T. M. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. 11th Annu. Conf. Comput. Learn. Theory*, 1998, pp. 92–100.
[9] K. Nigam and R. Ghani, "Analyzing the effectiveness and applicability of co-training," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2000, pp. 86–93.
[10] B. Krishnapuram, D. Williams, Y. Xue, A. Hartemink, L. Carin, and M. Figueiredo, "On semi-supervised classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 721–728.
[11] V. Sindhwani and P. Niyogi, "A co-regularized approach to semi-supervised learning with multiple views," in *Proc. ICML Workshop Learn. Multiple Views*, 2005.
[12] V. Sindhwani and D. Rosenberg, "An RKHS for multi-view learning and manifold co-regularization," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 976–983.
[13] S. Yu, B. Krishnapuram, R. Rosales, H. Steck, and R. B. Rao, "Bayesian co-training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1665–1672.
[14] A. Kumar and H. Daumé III, "A co-training approach for multi-view spectral clustering," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 393–400.
[15] A. Kumar, P. Rai, and H. Daumé III, "Co-regularized multi-view spectral clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1413–1421.
[16] S. Dasgupta, M. L. Littman, and D. McAllester, "PAC generalization bounds for co-training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 375–382.
[17] U. Brefeld and T. Scheffer, "Co-EM support vector learning," in *Proc. Int. Conf. Mach. Learn.*, 2004.
[18] Z.-H. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 11, pp. 1529–1541, Nov. 2005.
[19] M. Li and Z.-H. Zhou, "Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples," *IEEE Trans. Syst., Man, Cybern., Part A*, vol. 37, no. 6, pp. 1088–1098, Nov. 2007.
[20] W. Li, L. Duan, I. W.-H. Tsang, and D. Xu, "Co-labeling: A new multi-view learning approach for ambiguous problems," in *Proc. 11th Int. Conf. Data Mining*, 2012, pp. 419–428.
[21] X. Xu, I. W.-H. Tsang, and D. Xu, "Handling ambiguity via input-output kernel learning," in *Proc. 11th Int. Conf. Data Mining*, 2012, pp. 725–734.
[22] Y.-F. Li, I. W. Tsang, J. T.-Y. Kwok, and Z.-H. Zhou, "Tighter and convex maximum margin clustering," *Proc. 12th Int. Conf. Artif. Intell. Stat.*, 2009, pp. 344–351.
[23] X. Xu, I. W. Tsang, and D. Xu, "Soft margin multiple kernel learning," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 24, no. 5, pp. 749–761, May 2013.
[24] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien, "$\ell_p$-norm multiple kernel learning," *J. Mach. Learn. Res.*, vol. 12, pp. 953–997, 2011.
[25] W. Wang and Z.-H. Zhou, "Analyzing co-training style algorithms," in *Proc. 18th Eur. Conf. Mach. Learn.*, 2007, pp. 454–465.
[26] S. P. Abney, "Bootstrapping," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 360–367.
[27] M.-F. Balcan, A. Blum, and K. Yang, "Co-training and expansion: Towards bridging theory and practice," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 89–96.
[28] W. Wang and Z.-H. Zhou, "A new analysis of co-training," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 1135–1142.
[29] W. Wang and Z.-H. Zhou, "Co-training with insufficient views," in *Proc. Asian Conf. Mach. Learn.*, 2013, pp. 467–482.
[30] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin at Madison, Madison, WI, USA, Tech. Rep. 1530, 2006.
[31] Y.-F. Li, J. T. Kwok, I. W. Tsang, and Z.-H. Zhou, "A convex method for locating regions of interest with multi-instance learning," in *Proc. Eur. Conf. Mach. Learn./Knowl. Discovery Database*, 2009, pp. 15–30.
[32] Z. Zhou, "Multi-instance learning: A survey," AI Lab, Dept. Comput. Sci. Technol., Nanjing Univ., Nanjing, China, 2004.
[33] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1537–1544.
[34] A. Joulin and F. Bach, "A convex relaxation for weakly supervised classifiers," in *Proc. Int. Conf. Mach. Learn.*, 2012, pp. 1279–1286.
[35] A. J. Smola, L. Song, and C. H. Teo, "Relative novelty detection," *J. Mach. Learn. Res. - Proc. Track*, vol. 5, pp. 536–543, 2009.
[36] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, "Inlier-based outlier detection via direct density ratio estimation," in *Proc. Int. Conf. Data Mining*, 2008, pp. 223–232.
[37] T. Kanamori, S. Hido, and M. Sugiyama, "A least-squares approach to direct importance estimation," *J. Mach. Learn. Res.*, vol. 10, pp. 1391–1445, 2009.
[38] R. Rahmani and S. A. Goldman, "MISSL: Multiple-instance semi-supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 705–712.
[39] C.-W. Seah, I. W.-H. Tsang, and Y.-S. Ong, "Healing sample selection bias by source classifier selection," in *Proc. 11th Int. Conf. Data Mining*, 2011, pp. 577–586.
[40] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.
[41] L. Duan, D. Xu, I. W.-H. Tsang, and J. Luo, "Visual event recognition in videos by learning from web data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1667–1680, Sep. 2012.
[42] M. Szafranski, Y. Grandvalet, and A. Rakotomamonjy, "Composite kernel learning," *Mach. Learn.*, vol. 79, nos. 1/2, pp. 73–103, 2010.
[43] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "NUS-WIDE: A real-world web image database from national university of singapore," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, p. 48.
[44] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.
[45] R. Collobert, F. H. Sinz, J. Weston, and L. Bottou, "Large scale transductive SVMs," *J. Mach. Learn. Res.*, vol. 7, pp. 1687–1712, 2006.
[46] M. Chen, K. Q. Weinberger, and Y. Chen, "Automatic feature decomposition for single view co-training," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 953–960.
[47] D. Greene and P. Cunningham, "Practical solutions to the problem of diagonal dominance in kernel document clustering," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 377–384.
[48] M.-R. Amini, N. Usunier, and C. Goutte, "Learning from multiple partially observed views—An application to multilingual text categorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 28–36.

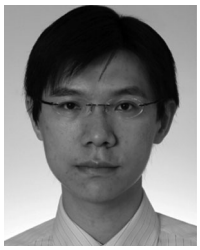


**Xinxing Xu** received the BE degree from the University of Science and Technology of China, Hefei, China, in 2009. He received the PhD degree in computer engineering from the School of Computer Engineering, Nanyang Technological University, Singapore, in 2015. He is currently a scientist with the Institute of High Performance Computing (IHPC), the Agency for Science, Technology and Research, Singapore. His current research interests include machine learning and its applications to computer vision.



**Wen Li** received the BS and MEng degrees from the Beijing Normal University, Beijing, China, in 2007 and 2010, respectively. He received the PhD degree from the Nanyang Technological University, Singapore, in 2015. Currently, he is a postdoctoral researcher at the Computer Vision Laboratory, ETH Zürich, Switzerland. His main interests include weakly supervised learning, domain adaptation, and multiple kernel learning.

**Dong Xu** (M'07-SM'13) received the BE and PhD degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2005, respectively. He was with Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Hong Kong, for over two years, while pursuing the PhD degree. He was a Post-Doctoral Research scientist with Columbia University, New York, NY, for one year. He was a faculty member with the School of Computer Engineering, Nanyang Technological University, Singapore. He is currently a faculty member with the School of Electrical and Information Engineering, The University of Sydney, Sydney, NSW, Australia. His current research interests include computer vision, statistical learning, and multimedia content analysis. He has coauthored a paper that received the Best Student Paper Award in the IEEE International Conference on Computer Vision and Pattern Recognition in 2010. His coauthored work also won the *IEEE Transactions on Multimedia* Prize Paper Award in 2014. He is a senior member of the IEEE.



**Ivor Wai-Hung Tsang** received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2007. He is an Australian Future fellow and associate professor with the Centre for Quantum Computation & Intelligent Systems, at the University of Technology, Sydney. He has published more than 100 research papers in refereed international journals and conference proceedings, including *JMLR*, *TPAMI*, TNN/TNNLS, NIPS, ICML, UAI, SIGKDD, ICCV and CVPR. In 2009, he was conferred the 2008 Natural Science Award (Class II) by the Ministry of Education, China, which recognized his contributions to kernel methods. In 2013, he received the prestigious Australian Research Council Future Fellowship for his research regarding Machine Learning on Big Data. In addition, he received the prestigious *IEEE Transactions on Neural Networks* Outstanding 2004 Paper Award in 2006, the 2014 *IEEE Transactions on Multimedia* Prized Paper Award, and a number of best paper awards and honors from reputable international conferences, including the Best Student Paper Award at CVPR 2010, and the Best Paper Award at ICTAI 2011, etc. He was also awarded the ECCV 2012 Outstanding Reviewer Award.