A General Formulation for Safely Exploiting Weakly Supervised Data*

Lan-Zhe Guo and **Yu-Feng Li**
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
{guolz, liyf}@lamda.nju.edu.cn

Abstract

Weakly supervised data is an important machine learning data to help improve learning performance. However, recent results indicate that machine learning techniques with the usage of weakly supervised data may sometimes cause *performance degradation*. *Safely* leveraging weakly supervised data is important, whereas there is only very limited effort, especially on a general formulation to help provide insight to guide safe weakly supervised learning. In this paper we present a scheme that builds the final prediction results by integrating several weakly supervised learners. Our resultant formulation brings two advantages. i) For the commonly used convex loss functions in both regression and classification tasks, safeness guarantees exist under a mild condition; ii) Prior knowledge related to the weights of base learners can be embedded in a flexible manner. Moreover, the formulation can be addressed globally by simple convex quadratic or linear program efficiently. Experiments on multiple weakly supervised learning tasks such as label noise learning, domain adaptation and semi-supervised learning validate the effectiveness.

Introduction

Weakly supervised data is commonly appear in real applications (Zhou 2017). Compared to the data in traditional supervised learning, weakly supervised data does not require a large amount of precise label information. Examples includes label noise learning (Frénay and Verleysen 2014) where label information contains noise; domain adaptation (Pan and Yang 2010) where label information in target domain is not sufficient and one needs to exploit further label information from other domains; semi-supervised learning (Chapelle et al. 2006) where label information is scarce and one needs to leverage a number of additional unlabeled data. Because weakly supervised data loosens the constraint for the label information in learning tasks, it has a broad application prospect, such as image classification (Krishna et al. 2017), natural language processing (Alfonseca et al. 2012) and so on. Taking advantage of weakly supervised data to help build effective learning methods has gained extensive attention and obtained a lot of research progresses (Chapelle et al. 2006; Pan and Yang 2010; Frénay and Verleysen 2014; Zhou 2017).

It is often expected that, machine learning techniques exploiting weakly supervised data are able to improve learning performance. However, recent studies show that machine learning techniques with the use of weakly supervised data may sometimes lead to *performance degradation*. That is, the learning performance is even worse than that of baseline method without using weakly supervised data. For example, label noise learning may be worse than learning from only a small number of high-quality labeled data (Frénay and Verleysen 2014); domain adaptation methods may have the phenomenon of *negative transfer* (Pan and Yang 2010) that the source domain data contribute to the reduced performance of learning in the target domain; semi-supervised learning using unlabeled data may degenerate learning performance, which has been reported in a number of studies (Chapelle et al. 2006; Chawla and Karakoulas 2005; Li and Zhou 2015). How to *safely* exploit weakly supervised data so that machine learning technology often outperforms and never be worse than the simple baseline, has become an important yet unsolved problem. Recently there is a few effort, but they typically work on a specific scenario of weakly supervised learning (Li and Zhou 2015; Balsubramani and Freund 2015; Li, Zha, and Zhou 2017; Wei et al. 2017). The proposal on generic formulation for various weakly supervised data, to our best knowledge, has not been thoroughly studied.

In this paper, we present a scheme that builds the final prediction results by integrating several weakly supervised learners. The resultant formulation brings some advantages. Firstly, for multiple commonly used convex loss functions (e.g., square loss, hinge loss) in both regression and classification tasks of weakly supervised learning, it has safeness guarantees under a mild condition. Secondly, it can flexibly embed uncertain prior knowledge about the weights of weakly supervised learners in regression and classification tasks. Moreover, our formulation can be addressed globally via simple convex quadratic program or linear program in an efficient manner. Experiments on multiple weakly supervised learning tasks such as label noise learning, domain adaptation and semi-supervised learning validate the effectiveness of our proposed algorithms.

This paper is organized as follows. We first review related

*This research was supported by the National Natural Science Foundation of China (61772262, 61403186, 61422304).

works and then present the proposed formulation. Next we show the experiments. Finally we conclude this work.

Related Work

Effectively exploiting weakly supervised data to improve learning performance has been attracted much attention. In the aspect of label noise learning, quite many studies have indicated that without careful consideration, label noise may seriously affect the learning performance (Frénay and Verleysen 2014). Considerable efforts have been made to build models that are robust to the presence of label noise. For example, from the theoretical aspect, Manwani and Satry (2013) studied the robustness of loss functions in the empirical risk minimization framework and disclosed that 0-1 loss function is noise tolerant while the other loss functions are not naturally noise tolerant. From the practical aspect, ensemble methods, e.g., bagging and boosting are regarded to be robust to label noise (Frénay and Verleysen 2014) and bagging often achieves a better result than boosting in the presence of label noise (Dietterich 2000).

In the aspect of domain adaptation, there is little discussion on how to avoid *negative transfer* though it is regarded as an important issue in domain adaptation (Pan and Yang 2010). Bakker and Heskes (2003) presented a Bayesian method for a joint prior distribution of multiple domains and considered that some of the model parameters should be loosely connected among domains. Rosenstein et al. (2005) empirically showed that if two tasks are dissimilar, then brute-force transfer may hurt the performance of the target task. Argyriou et al. (2008) considered situations that the representations should be different among different groups of tasks and tasks with different group are hard to perform domain adaptation. Ge et al., (2014) proposed to weight source domains corresponding to the relatedness to the target domain and constructed the final target learner with the weights to attenuate the effects of negative transfer.

In the aspect of semi-supervised learning, most efforts on safely exploiting unlabeled data are raised in very recent. In terms of classification tasks, Li and Zhou (2015) aimed to build safe semi-supervised SVMs by assuming that the ground-truth decision boundary is realized by one of multiple diverse large-margin separations. Balsubramani and Freund (2015) proposed to learn a robust prediction with the highest accuracy given that the ground-truth label assignment is restricted to specific candidate set. In terms of regression tasks, it is indicated that safe semi-supervised regression is realized as an intuitive geometric projection issue and has an efficient solution (Li, Zha, and Zhou 2017).

Generally, safely exploiting weakly supervised data has become a crucial yet unsolved issue. Most of the previous studies were carried out for one specific scenario. In this work we propose to present a general formulation for safe weakly supervised learning and would like to provide some insight to understand safe weakly supervised learning.

Proposed Formulation

In this section, we first present the problem setting and derive our formulation, then study the safeness, next we discuss the incorporation of prior knowledge, finally we present the optimization algorithms.

Problem Setting and Formulation

In weakly supervised learning, due to the lack of sufficient and accurate label information, ensemble learning (Zhou 2012) was recognized as a popular learning technology to derive robust performance. Many ways can be employed to generate multiple weakly supervised learners, such as through different learning models, different sampling strategies, different model parameters, etc. Although previous studies typically work on deriving good performance from multiple learners, they may suffer from *unsafeness*. One underlying reason is that the *good* performance derived by previous studies is not explicitly compared with the baseline method, and may sometimes mislead the learning process. These motivate us to derive a new formulation.

Formally, suppose we have obtained n predictive results $\{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ of unlabeled instances from multiple weakly supervised learners $\{f_1, \dots, f_n\}$, where $\mathbf{y}_i \in \mathbb{H}^u$, $i = 1, \dots, n$ and u is the number of unlabeled instances. We let $\mathbb{H} = \mathbb{R}$ for regression task and $\mathbb{H} = \{+1, -1\}$ for classification task. Meanwhile, we let $\mathbf{y}_0 \in \mathbb{H}^u$ denote the baseline result, e.g., obtained by training a supervised model with only limited labeled data. Our goal is to derive a safe prediction $\mathbf{y} = g(\{\mathbf{y}_1, \dots, \mathbf{y}_n\}, \mathbf{y}_0)$, which often outperforms, meanwhile would not be worse than $\mathbf{y}_0$.

We first consider a simpler case that the ground-truth label assignment on unlabeled instances, denoted by $\mathbf{y}^*$, is known. In this case, one can easily have the objective function that maximizes the performance gain against the baseline $\mathbf{y}_0$, as

$$\max_{\mathbf{y} \in \mathbb{H}^u} \ \ell(\mathbf{y}_0, \mathbf{y}^*) - \ell(\mathbf{y}, \mathbf{y}^*)$$

Here $\ell(\cdot, \cdot)$ is a loss function, e.g., mean square loss, hinge loss, etc. The smaller the value of the loss function, the better the performance. Table 1 summarizes some commonly used loss functions for regression and classification. Obviously $\mathbf{y}^*$ is unknown and otherwise the solution is trivial. To alleviate it, we consider that $\mathbf{y}^*$ is a convex combination of base learners. Specifically, $\mathbf{y}^* = \sum_{i=1}^n \alpha_i \mathbf{y}_i$ where $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; \dots; \alpha_n] \geq \mathbf{0}$ be the non-negative weights of base learners and $\sum_{i=1}^n \alpha_i = 1$. We then have the following objective by replacing the definition of $\mathbf{y}^*$,

$$\max_{\mathbf{y} \in \mathbb{H}^u} \ell(\mathbf{y}_0, \sum_{i=1}^n \alpha_i \mathbf{y}_i) - \ell(\mathbf{y}, \sum_{i=1}^n \alpha_i \mathbf{y}_i)$$

In practice, however, one may still be hard to know the precise weights of base learners. We further consider that $\boldsymbol{\alpha}$ is from a convex set $\mathcal{M}$ and make our proposal more practical, where $\mathcal{M}$ reflects the prior knowledge for the importance of base learners. The setup of $\mathcal{M}$ will be discussed in the later section. Without further information, we aim to optimize the worst-case performance gain. We then can have a general formulation with respect to regression as well as classification task as,

$$\max_{\mathbf{y} \in \mathbb{H}^u} \min_{\boldsymbol{\alpha} \in \mathcal{M}} \ell(\mathbf{y}_0, \sum_{i=1}^n \alpha_i \mathbf{y}_i) - \ell(\mathbf{y}, \sum_{i=1}^n \alpha_i \mathbf{y}_i) \qquad (1)$$

Table 1: Some commonly used loss functions $\ell(\mathbf{p}, \mathbf{q})$ for regression and classification tasks. The prediction $\mathbf{q} = [q_1; \ldots; q_u] \in \mathbb{R}^u$ and the label $\mathbf{p} = [p_1; \ldots; p_u] \in \mathbb{H}^u$ where $\mathbb{H}^u = \mathbb{R}^u$ is for regression and $\mathbb{H}^u = \{+1, -1\}^u$ is for classification.

Loss function	Definition	Task
Mean square loss	$\ell(\mathbf{p}, \mathbf{q}) = \frac{1}{u} \sum_{i=1}^{u} (p_i - q_i)^2 = \frac{1}{u} \|\mathbf{p} - \mathbf{q}\|_2^2$	Regression & Classification
Mean absolute loss	$\ell(\mathbf{p}, \mathbf{q}) = \frac{1}{u} \sum_{i=1}^{u} \|p_i - q_i\| = \frac{1}{u} \|\mathbf{p} - \mathbf{q}\|_1$	Regression
Mean ϵ-insensitive loss	$\ell(\mathbf{p}, \mathbf{q}) = \frac{1}{u} \sum_{i=1}^{u} \max\{\|p_i - q_i\| - \epsilon, 0\}$	Regression
Hinge loss	$\ell(\mathbf{p}, \mathbf{q}) = \frac{1}{u} \sum_{i=1}^{u} \max\{1 - p_i q_i, 0\}$	Classification

Study the Safeness

We show that for the commonly used convex loss functions as listed in Table 1 in both regression and classification tasks, safeness guarantees exist for Eq.(1) under a mild condition. We first introduce a result as follows.

Theorem 1. *Suppose that the ground-truth $\mathbf{y}^*$ can be constructed by the base learners, i.e., $\mathbf{y}^* \in \{\mathbf{y} | \sum_{i=1}^{b} \alpha_i \mathbf{y}_i, \boldsymbol{\alpha} \in \mathcal{M}\}$. Let $\hat{\mathbf{y}}$ and $\hat{\boldsymbol{\alpha}}$ be the optimal solution to Eq.(1), we then have $\ell(\hat{\mathbf{y}}, \mathbf{y}^*) \leq \ell(\mathbf{y}_0, \mathbf{y}^*)$ and $\hat{\mathbf{y}}$ has already achieved the maximal performance gain against $\mathbf{y}_0$.*

Proof. We define

$$L(\mathbf{y}, \boldsymbol{\alpha}) = \ell(\mathbf{y}_0, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) - \ell(\mathbf{y}, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i).$$

The following inequality holds for any feasible $\mathbf{y}$ and $\boldsymbol{\alpha}$:

$$L(\mathbf{y}, \hat{\boldsymbol{\alpha}}) \leq L(\hat{\mathbf{y}}, \hat{\boldsymbol{\alpha}}) \leq L(\hat{\mathbf{y}}, \boldsymbol{\alpha})$$

According to the assumption, $\mathbf{y}^* \in \{\mathbf{y} | \sum_{i=1}^{b} \alpha_i \mathbf{y}_i, \boldsymbol{\alpha} \in \mathcal{M}\}$ and let $\boldsymbol{\alpha}^*$ makes $\mathbf{y}^* = \sum_{i=1}^{n} \alpha_i^* \mathbf{y}_i$. By setting $\mathbf{y}$ and $\boldsymbol{\alpha}$ to be $\mathbf{y}_0$ and $\boldsymbol{\alpha}^*$, then we can deduce that,

$$\ell(\mathbf{y}_0, \mathbf{y}^*) \geq \ell(\hat{\mathbf{y}}, \mathbf{y}^*)$$

Moreover, since we already maximize the performance gain in the worst case, $\hat{\mathbf{y}}$ has already achieved the maximal performance gain against $\mathbf{y}_0$. □

Theorem 1 indicates that Eq.(1) is reasonable for our purpose, that is, the derived optimal solution $\hat{\mathbf{y}}$ from Eq.(1) often outperforms and won't be worse than $\mathbf{y}_0$. In comparison to previous studies (Li and Zhou 2015; Balsubramani and Freund 2015; Li, Zha, and Zhou 2017), the formulation in Eq.(1) brings some advantages. In contrast to (Li and Zhou 2015) which requires the ground-truth is from one of the learners, the condition required in Theorem 1 is looser and more practical. We explicitly consider to maximize the performance gain in Eq.(1), which is not taken into account in (Balsubramani and Freund 2015). In contrast to (Li, Zha, and Zhou 2017) that focuses on regression, our work is readily applicable for both regression and classification tasks.

One question unclear in Theorem 1 is how to derive the optimal solution of Eq.(1). Eq.(1) is the subtraction of two loss functions, which is known to be non-convex and nontrivial to derive the global optima (Yuille and Rangarajan 2003). Fortunately, we find that for a class of commonly used convex loss functions, Eq.(1) could be equivalently rewritten as a convex optimization problem and thus the global optimal solution is achievable. First, we present the result for regression task.

Lemma 1. *When $\ell(\cdot, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i)$ is convex to $\boldsymbol{\alpha}$ and there exists $\mathbf{y} \in \mathbb{R}^u$ such that $\ell(\mathbf{y}, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) = 0$, for any $\boldsymbol{\alpha}$. In optimality, the optimal solution $\hat{\mathbf{y}}$ and $\hat{\boldsymbol{\alpha}}$ have the following relation, i.e., $\ell(\hat{\mathbf{y}}, \sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i) = 0$.*

Proof. Assume, to the contrary, $\ell(\hat{\mathbf{y}}, \sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i) \neq 0$. According to the assumption, there exist $\tilde{\mathbf{y}}$ such that $\ell(\tilde{\mathbf{y}}, \sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i) = 0$. Obviously, $0 = \ell(\tilde{\mathbf{y}}, \sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i) < \ell(\hat{\mathbf{y}}, \sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i)$. Hence, $\hat{\mathbf{y}}$ is not optimal, a contradiction. □

Theorem 2. *Under the same condition in Lemma 1, Eq.(1) is a convex optimization.*

Proof. With Lemma 1, the form of Eq.(1) for regression task can be rewritten as,

$$\min_{\boldsymbol{\alpha} \in \mathcal{M}} \ \ell(\mathbf{y}_0, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) \qquad (2)$$

Remind that $\ell(\cdot, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i)$ is convex to $\boldsymbol{\alpha}$, obviously, Eq.(1) is a convex optimization. □

The condition in Theorem 2 is rather mild. Many regression loss functions, for example, mean square loss, mean absolute loss (Willmott and Matsuura 2005) and mean ϵ-insensitive loss (Smola and Schölkopf 2004), satisfy such a mild condition in Theorem 2.

Due to the noncontinuous feasible field of $\mathbf{y}$, Lemma 1 does not hold for most of the classification loss functions. It could not simply apply or extend the result in regression task to classification. Fortunately, we find that for some particular classification loss function like the hinge loss, the optimal solution of Eq.(1) is still possible.

Lemma 2. *When $\ell(\cdot, \cdot)$ is realized as the hinge loss, in optimality, the optimal $\hat{\mathbf{y}}$ and $\hat{\boldsymbol{\alpha}}$ meet a relation $\hat{\mathbf{y}} = sign(\sum_{i=1}^{n} \hat{\alpha}_i \mathbf{y}_i))$ where $sign(s)$ is the sign of value s.*

The proof is in the supplementary material. We then have,

Theorem 3. *Suppose $\mathbf{y}_i \in \{+1, -1\}^u$, $\forall i = 1, \ldots, n$, Eq.(1) is a convex optimization when $\ell(\cdot, \cdot)$ is realized as the hinge loss.*

Proof. With Lemma 2, Eq.(1) is thus rewritten as,

$$\min_{\boldsymbol{\alpha} \in \mathcal{M}} \ell(\mathbf{y}_0, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) - \ell(sign(\sum_{i=1}^{n} \alpha_i \mathbf{y}_i), \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) \quad (3)$$

Since $\mathbf{y}_i \in \{+1, -1\}^u$, $\forall i = 1, \ldots, n$ and $\ell(\cdot, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i)$ is the hinge loss, the form $\ell(sign(\sum_{i=1}^{n} \alpha_i \mathbf{y}_i), \sum_{i=1}^{n} \alpha_i \mathbf{y}_i)$

can be equivalently rewritten as $1 - \frac{1}{u}\|\sum_{i=1}^{n} \alpha_i \mathbf{y}_i\|_1$ using $\sum_{i=1}^{n} \alpha_i = 1$. Therefore, Eq.(3) is equal to,

$$\min_{\boldsymbol{\alpha}\in\mathcal{M}} \ell(\mathbf{y}_0, \sum_{i=1}^{n} \alpha_i \mathbf{y}_i) + \frac{1}{u}\|\sum_{i=1}^{n} \alpha_i \mathbf{y}_i\|_1 - 1 \tag{4}$$

Evidently, Eq.(4) is a convex optimization. □

Though hinge loss is the only loss function to help derive a global solution of Eq.(1) for classification task, hinge loss is well-known as one powerful loss function in classification. The reason for the particularity of hinge loss mainly lies in its linearity to the predictive results (the term $\frac{1}{u}\|\sum_{i=1}^{n} \alpha_i \mathbf{y}_i\|_1$). Such a property, unfortunately, does not hold for other loss functions such as logistic loss, exponential loss, cross-entropy loss, etc.

Weight the Base Learners

One question remained is that how to setup $\mathcal{M}$. Obviously, the setup of $\mathcal{M}$ can be easily embedded with a variety of prior knowledge. For example, suppose base learner f_i is more reliable than f_j and the set of all such indexes (i, j) is denoted as $\mathcal{S}$, $\mathcal{M}$ could be set to $\{\boldsymbol{\alpha} | \alpha_i - \alpha_j \geq 0, (i, j) \in \mathcal{S}; \boldsymbol{\alpha}^\top \mathbf{1} = 1; \boldsymbol{\alpha} \geq \mathbf{0}\}$ where $\mathbf{1}$ and $\mathbf{0}$ refer to the all-one and all-zero vectors respectively; suppose the importance values of base learners are known and let $\{r_1, \ldots, r_n\}$ denote the importance values, one could set up $\mathcal{M}$ as $\{\boldsymbol{\alpha} | -\gamma \leq \alpha_i - r_i \leq \gamma, \forall i = 1, \ldots, n; \boldsymbol{\alpha}^\top \mathbf{1} = 1; \boldsymbol{\alpha} \geq \mathbf{0}\}$ where γ is a small constant, and so on and so forth.

In practice, one may be hard to obtain the precise prior knowledge of base learners, for example, the importance values. In this case we present to learn the weights of base learners. Before presenting the algorithms, we first investigate how the performance of our formulation is affected with the setup of $\mathcal{M}$. Assume that the loss function $\ell(\cdot, \cdot)$ is η-Lipschitz, i.e., $\|\ell(\mathbf{y}_1, \mathbf{y}_2) - \ell(\mathbf{y}_1, \mathbf{y}_3)\| \leq \eta \|\mathbf{y}_2 - \mathbf{y}_3\|_1$ for any $\mathbf{y}_1$, $\mathbf{y}_2, \mathbf{y}_3 \in [-1, 1]$. Most of commonly used loss functions including the ones in Table 1 satisfy such property (Rosasco et al. 2004). Let $\boldsymbol{\beta}^* = [\beta_1^*, \cdots, \beta_n^*] \in \mathcal{M}$ be the optimal solution to the objective,

$$\boldsymbol{\beta}^* = \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathcal{M}} \ \ell(\sum_{i=1}^{n} \beta_i \mathbf{y}_i, \mathbf{y}^*)$$

and $\boldsymbol{\epsilon}$ be the residual, i.e., $\boldsymbol{\epsilon} = \mathbf{y}^* - \sum_{i=1}^{n} \beta_i^* \mathbf{y}_i$. We have the following result,

Theorem 4. *The performance gain of $\hat{\mathbf{y}}$ against $\mathbf{y}_0$, i.e., $\ell(\mathbf{y}_0, \mathbf{y}^*) - \ell(\hat{\mathbf{y}}, \mathbf{y}^*)$, has a lower-bound $-2\eta\|\boldsymbol{\epsilon}\|_1$.*

The proof is in the supplementary material. Theorem 4 discloses that the worst-case of performance gain is lower-bounded by the norm of the residual. This motivates us to learn the weights of base learners such that the residual is minimized. We then present the learning approach for regression and classification through the idea of covariance matrix analysis (Bates and Granger 1969).

Regression Let $\mathbf{C}^{reg}$ be the $n \times n$ covariance matrix of the n base learners $\{f_1, \cdots, f_n\}$ with elements

$$C_{ij}^{reg} = \mathbb{E}[(f_i(X) - \mu_i)^\top (f_j(X) - \mu_j)]$$

where X refers to the set of unlabeled instances and $\mu_i = \mathbb{E}[f_i(X)]$. Let $\boldsymbol{\rho}^{reg} = [\rho_1^{reg}; \cdots; \rho_n^{reg}]$ be the vector of covariances between the base learners and the ground-truth label assignment $f^*(X)$,

$$\rho_i^{reg} = \mathbb{E}[(f^*(X) - \theta)^\top (f_i(X) - \mu_i)]$$

where $\theta = \mathbb{E}[f^*(X)]$. We minimize the residual for $\boldsymbol{\alpha}$ as,

$$\boldsymbol{\alpha}^* = \operatorname*{arg\,min}_{\boldsymbol{\alpha}} \ \mathbb{E}[\text{MSE}(\sum_{i=1}^{n} \alpha_i f_i(X), f^*(X))] \tag{5}$$

where MSE refers to the Mean Squared Error. Eq.(5) has a closed-form solution (Bates and Granger 1969).

Theorem 5. *(Bates and Granger, 1969) The optimal weights $\boldsymbol{\alpha}^*$ satisfies that*

$$\boldsymbol{\rho}^{reg} = \mathbf{C}^{reg} \boldsymbol{\alpha}^*.$$

With Theorem 5, one need to estimate $\mathbf{C}^{reg}$ and $\boldsymbol{\rho}$. For $\mathbf{C}^{reg}$, it is evident that $(\mathbf{y}_i - \mu_i)^\top (\mathbf{y}_j - \mu_j)$ is an unbiased estimation of C_{ij}^{reg}. Therefore, one could easily have $\hat{\mathbf{C}}^{reg}$ with elements

$$\hat{C}_{ij}^{reg} = (\mathbf{y}_i - \mu_i)^\top (\mathbf{y}_j - \mu_j)$$

be the unbiased estimation of $\mathbf{C}^{reg}$. For $\boldsymbol{\rho}$, the following proposition shows that it is closely related to the performance of base learners.

Proposition 1. *Suppose $\{f_i(X)\}_{i=1}^{i=n}$ is normalized to the mean $\mu_i = 0, \forall i = 1, \ldots n$ and the standard deviation equal to 1. Consider mean squared error as the measurement, the bigger the value ρ_i^{reg}, the smaller the loss of f_i.*

Therefore, we can setup $\mathcal{M}$ as $\{\boldsymbol{\alpha} | \hat{\mathbf{C}}^{reg} \boldsymbol{\alpha} \geq \mathbf{1}\delta, \boldsymbol{\alpha}^\top \mathbf{1} = 1, \boldsymbol{\alpha} \geq \mathbf{0}\}$, where δ is a constant, indicating that the base learners have a low-bound performance (e.g., are better than random-guess) (Balsubramani and Freund 2015). It is easy to verify that $\mathcal{M}$ is a convex set.

Classification Similar to regression tasks, let $\mathbf{C}^{clf}$ be the $n \times n$ matrix represents the agreement between base learners with elements $C_{ij}^{clf} = \mathbb{E}[f_i(X)^\top f_j(X)]$. Let $\boldsymbol{\rho}^{clf} = [\rho_1^{clf}; \rho_2^{clf}; \cdots; \rho_n^{clf}]$ be the vector represents the agreement between base learner and the ground truth,

$$\rho_i^{clf} = \mathbb{E}[f^*(X)^\top f_i(X)]$$

With classification accuracy to be the performance measure, it can be shown that,

Theorem 6. *The optimal weights $\boldsymbol{\alpha}^*$ in classification satisfies that $\boldsymbol{\rho}^{clf} = \mathbf{C}^{clf} \boldsymbol{\alpha}^*$.*

We can setup $\mathcal{M}$ as $\{\boldsymbol{\alpha} | \hat{\mathbf{C}}^{clf} \boldsymbol{\alpha} \geq \mathbf{1}\delta, \boldsymbol{\alpha}^\top \mathbf{1} = 1, \boldsymbol{\alpha} \geq \mathbf{0}\}$ where $\hat{\mathbf{C}}^{clf}$ is the unbiased estimation of $\mathbf{C}^{clf}$, with elements $\hat{C}_{ij}^{clf} = \mathbf{y}_i^\top \mathbf{y}_j$. $\mathcal{M}$ is also a convex set.

In summary, our formulation is able to directly absorb the precise prior knowledge about the importance of learners if available. It is also capable of incorporating with the estimation results obtained by covariance matrix analysis on both regression and classification tasks, when the precise prior knowledge is unavailable.

Efficient Optimization Algorithms

The formulation in Eq. (1) can be globally and efficiently addressed. For regression, we adopt mean square loss as the implementation. According to Lemma 1 and Theorem 2 , Eq.(1) can be rewritten as,

$$\min_{\hat{\mathbf{C}}^{reg}\boldsymbol{\alpha}\geq\mathbf{1}\delta,\boldsymbol{\alpha}^\top\mathbf{1}=1,\boldsymbol{\alpha}\geq\mathbf{0}} \quad ||\textstyle\sum_{i=1}^{n}\alpha_i\mathbf{y}_i-\mathbf{y}_0||^2$$

which is equivalent to the following form,

$$\min_{\hat{\mathbf{C}}^{reg}\boldsymbol{\alpha}\geq\mathbf{1}\delta,\boldsymbol{\alpha}^\top\mathbf{1}=1,\boldsymbol{\alpha}\geq\mathbf{0}} \quad \boldsymbol{\alpha}^\top\mathbf{F}\boldsymbol{\alpha}-\mathbf{v}^\top\boldsymbol{\alpha} \tag{6}$$

where $\mathbf{F} \in \mathbb{R}^{n\times n}$ is a linear kernel matrix of $\mathbf{y}_i$'s, i.e, $F_{ij} = \mathbf{y}_i^\top\mathbf{y}_j$ and $\mathbf{v} = [2\mathbf{y}_1^\top\mathbf{y}_0;\cdots;2\mathbf{y}_n^\top\mathbf{y}_0]$. Obviously, Eq.(6) is a simple convex quadratic program (Boyd and Vandenberghe 2004) and can be efficiently addressed by off-the-shelf optimization package, such as MOSEK.

For classification, we adopt the hinge loss as the implementation. According to Lemma 2 and Theorem 3, Eq.(1) can be rewritten as

$$\min_{\hat{\mathbf{C}}^{clf}\boldsymbol{\alpha}\geq\mathbf{1}\delta,\boldsymbol{\alpha}^\top\mathbf{1}=1,\boldsymbol{\alpha}\geq\mathbf{0}} \ell(\mathbf{y}_0,\sum_{i=1}^{n}\alpha_i\mathbf{y}_i)+\frac{1}{u}\|\sum_{i=1}^{n}\alpha_i\mathbf{y}_i\|_1 \tag{7}$$

which is a simple linear program. The detail derivation is in the supplementary material. Eq. (7) can be globally addressed in an efficient manner via MOSEK as well.

Experiments

In this section, we conduct experiments on three weakly supervised learning tasks, i.e., label noise learning, domain adaptation and semi-supervised learning so as to evaluate the effectiveness of our proposed algorithms. We call our proposal as SAFEW (SAFE Weakly supervised learning)[1].

Label Noise Learning Task

We conduct experimental comparison for label noise learning tasks on a number of frequently-used classification datasets[2], i.e., *Australian*, *Breast-Cancer*, *Diabetes*, *Digit1*, *Heart*, *Ionosphere*, *USPS* and *Splice*. For each data set, 80% of instances are used for training and the rest ones are used for testing. In the training set, 70% of instances are randomly selected as the noisy or low-quality labeled data and the rest ones are high-quality labeled data. For the noisy labeled data, their labels are randomly reversed with a probability $p\%$ where p ranges from 10% to 40% with an interval 10%. Experiments are repeated for 30 times, and the average classification accuracy is reported.

Our proposed algorithm is compared with the following methods, including 1 baseline method Sup-SVM that trains a supervised SVM only on the high-quality labeled data; 2 state-of-the-art label noise learning methods: i) Bagging which is regarded as to be robust with label noisy (Frénay and Verleysen 2014); ii) rLR (Robust Logistic Regression) (Bootkrajang and Kabán 2012) that enhances the logistic regression model to handle label noise; 3 traditional classification methods (i.e., SVM, LR (Logistic Regression), k-NN (k-Nearest Neighbor)) with regardless of label noise. For LR, the *glmfit* function in Matlab is used. For k-NN method, k is set to 3. For Sup-SVM and SVM method, Libsvm package (Chang and Lin 2011) is adopted and the kernel is set to RBF kernel. For Bagging method, we adopt decision tree as the base learner. For rLR method, the parameter is set to the recommended one. For SAFEW, LR, SVM and k-NN are used as base learners and parameter δ is set by 5-fold cross validation from the range $[0.5u, 0.7u]$.

The results are shown in Figure 1 and we can have the following observations. i) As the noise ratio increases, the accuracies of compared methods generally decrease; ii) Compared with the baseline method, all the compared methods performs worse than Sup-SVM in many cases, especially when the noise ratio becomes larger, while our proposed SAFEW does not suffer from such deficiency. Moreover, SAFEW achieves best average performance (see detail results in the supplementary material). These demonstrate the effectiveness of SAFEW method.

Domain Adaptation Task

We conduct compared experiments for domain adaptation on two benchmark datasets[3], i.e., *20newsgroup* and *Landmine*. The *20Newsgroups* dataset (Lang 1995) contains 19,997 documents and is partitioned into 20 different newsgroups. Following the setup in (Dai et al. 2007; Li, Jin, and Long 2012), we generate six different cross-domain data sets by utilizing its hierarchical structure. Specifically, the learning task is defined as the top-category binary classification, where our goal is to classify documents into one of the top-categories. For each data set, two top categories are chosen, one as positive and another as negative. Then we select sub-categories under the positive and negative classes respectively to form a domain.

The *Landmine* dataset is a detection dataset which contains 29 domains and 9 features. The data from domain 1 to domain 5 are collected from a leafy area; the data from Domain 20 to domain 24 are collected from a sand area. We use the whole data from domain 1 to domain 5 as the source domain and the data from domain 20 to domain 24 as five target domains. For *20newsgroup*, following (Xue et al. 2008), we randomly select 10% instances in target domain as the labeled data and use 300 most important features as the representation. For *Landmine*, 5% instances in the target domain are used as the labeled data. Experiments are repeated for 30 times and the average accuracies on the unlabeled instances are reported.

Our method is compared with one baseline supervised method LR that trains a supervised logistic regression model for the labeled data in target domain, one naive domain adaptation method called as Original method that combines the data in source and target domain to train a supervised model, and three state-of-the-art domain adaptation methods, i) Maximum Independence Domain Adaptation (MIDA) method (Yan, Kou, and Zhang 2016); ii) Transfer Component Analysis (TCA) method (Pan et al. 2011); iii)

[1] `http://lamda.nju.edu.cn/code_SAFEW.ashx`

[2] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

[3] `http://www.cse.ust.hk/TL/`

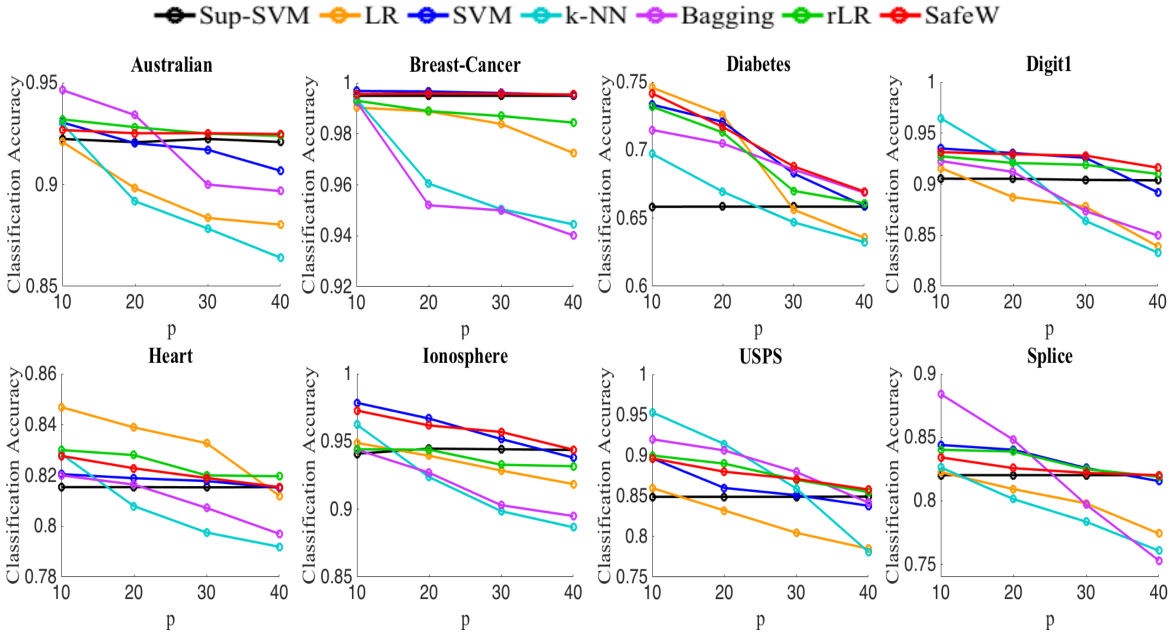

Figure 1: Classification accuracy of compared methods with different numbers of noise ratio.

TrAdaBoost method (Dai et al. 2007). MIDA and TCA are two feature-level learning algorithms that learns a domain-invariant subspace between source domain and target domain. TrAdaBoost method is a transfer learner based on AdaBoost. For MIDA and TCA, the kernel type is set to linear kernel and the dimension of the subspace is set to 30. For MIDA, TCA and the Original method, Logistic Regression model is employed as the supervised model on the feature space. For TrAdaBoost, SVM is adopted as the base learner and the number of iterations is set to 20. MIDA, TCA and the Original method are used as our base learners. Parameter δ is set by 5-fold cross validation from the range $[0.5u, 0.7u]$.

Results are shown in Tables 2 and 3. Original, MIDA and TCA methods degenerate the performance in many cases. SAFEW does not suffer such a deficiency. Moreover, in terms of average performance, SAFEW achieves the best result. Therefore, our proposal achieves highly competitive performance with compared methods while more importantly, unlike previous methods that will hurt performance in some cases, it does not degenerate the performance.

Semi-Supervised Learning Task

For semi-supervised learning, we do experiments on regression tasks with a broad range of datasets[4] that cover diverse domains including physical measurements (*abalone*), health (*bodyfat*), economics (*cadata*), activity recognition (*mpg*), etc. The sample size ranges from around 100 (*pyrim*) to more than 20,000 (*cadata*). All the features and labels are normalized into $[0, 1]$. For each dataset, we randomly select 10 data as the labeled instances. Experiment for each dataset is repeated for 30 times, and the average performance (mean±std) on the unlabeled data is reported.

The compared methods include 1NN method which is a direct supervised nearest neighbor algorithm with only labeled data, Self-kNN method which is a semi-supervised extension of the supervised kNN method based on self-training (Yarowsky 1995), Self-LS method which is a semi-supervised extension of the supervised least square method (Hastie, Tibshirani, and Friedman 2001), Average method which is a simple ensemble method, Safer method (Li, Zha, and Zhou 2017) which is a method proposed for semi-supervised regression. For Self-kNN, we use two distance measures: Euclidean and Cosine, and k is set to 3, the maximum number of iterations is set to 5. For Self-LS method, the parameters related to the importance for the labeled and unlabeled instances are set to 1 and 0.1. For SAFEW, Average and Safer methods, Self-kNN(Euclidean), Self-kNN(Cosine) and Self-LS are adopted as base learners. Parameter δ is set by 5-fold cross validation from the range $[0.5u, 0.7u]$.

According to the results in Table 4, we can see that SAFEW and Safer are always better than the baseline, while the other compared methods will be outperformed by the baseline method in many cases. Moreover, in terms of average performance, SAFEW performs better than Safer. The reason owes to a tight set $\mathcal{M}$ learned for base learners. Again, the results validate the effectiveness of SAFEW.

Conclusion

In this paper, we study to *safely* exploit weakly supervised data. That is, learning methods with the usage of weakly supervised data could often improve learning performance, meanwhile in the worst case it wont be worse than the base-

[4] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

Table 2: Classification Accuracy of domain adaptation on 20newsgroup. For the compared methods, if the performance is significantly better/worse than the baseline method, the corresponding entries are then bolded/boxed (paired t-tests at 95% significance level). The average performance is listed for comparison. The win/tie/loss counts against the baseline method are summarized, and the method with the smallest number of losses is bolded.

Dataset	Logistic Regression	Original	MIDA	TCA	TrAdaBoost	SafeW
Comp vs Rec	.7028 ± .0091	**.7492 ± .0135**	**.7961 ± .0197**	**.7940 ± .0162**	**.8077 ± .0155**	**.7956 ± .0170**
Comp vs Sci	.8225 ± .0662	$\boxed{.7985 \pm .0194}$	**.8946 ± .0188**	.8255 ± .0172	**.8583 ± .0201**	**.8925 ± .0212**
Comp vs Talk	.8423 ± .0685	$\boxed{.8022 \pm .0182}$	$\boxed{.8231 \pm .0164}$	.8434 ± .0110	$\boxed{.8247 \pm .0143}$	.8451 ± .0158
Sci vs Talk	.7294 ± .1045	.7100 ± .0121	**.7456 ± .0164**	$\boxed{.7022 \pm .0092}$	.7166 ± .0213	**.7468 ± .0153**
Rec vs Sci	.8006 ± .0758	$\boxed{.7754 \pm .0161}$	.8033 ± .0151	**.8440 ± .0118**	.8016 ± .0151	**.8435 ± .0157**
Rec vs Talk	.8278 ± .0446	.8276 ± .0115	**.8566 ± .0105**	**.8580 ± .0128**	**.8415 ± .0113**	**.8579 ± .0105**
Average	.7876	.7805	.8199	.8112	.8084	.8302
Win/Tie/Loss against LR		1/2/3	4/1/1	3/2/1	3/2/1	**5/1/0**

Table 3: Classification Accuracy of domain adaptation on Landmine data.

Dataset	Logistic Regression	Original	MIDA	TCA	TrAdaBoost	SafeW
Domain 20	.9215 ± .0173	.9237 ± .0034	**.9265 ± .0039**	.9255 ± .0045	.9183 ± .0029	**.9271 ± .0035**
Domain 21	.9360 ± .0095	$\boxed{.9310 \pm .0047}$	.9384 ± .0045	$\boxed{.9304 \pm .0051}$	$\boxed{.9261 \pm .0033}$	.9396 ± .0038
Domain 22	.9594 ± .0051	.9555 ± .0038	$\boxed{.9506 \pm .0065}$	**.9650 ± .0017**	$\boxed{.9095 \pm .0026}$	**.9648 ± .0016**
Domain 23	.9361 ± .0095	$\boxed{.9310 \pm .0041}$	**.9424 ± .0045**	$\boxed{.9314 \pm .0051}$	**.9627 ± .0043**	**.9426 ± .0038**
Domain 24	.9535 ± .0052	.9524 ± .0029	$\boxed{.9447 \pm .0025}$	$\boxed{.9432 \pm .0029}$	.9535 ± .0034	.9550 ± .0024
Average	.9413	.9387	.9405	.9391	.9340	.9458
Win/Tie/Loss against LR		0/3/2	2/1/2	1/1/3	1/2/2	**3/2/0**

Table 4: Mean Square Error (mean±std) for the compared methods and SafeW on a number of regression data sets.

Dataset	1NN	Self-kNN(Euclidean)	Self-kNN(Cosine)	Self-LS	Average	Safer	SafeW
abalone	.020 ± .010	**.014 ± .005**	**.014 ± .003**	**.013 ± .004**	**.012 ± .003**	**.013 ± .005**	**.013 ± .005**
bodyfat	.019 ± .005	.018 ± .006	.019 ± .005	$\boxed{.041 \pm .013}$	$\boxed{.023 \pm .009}$	.018 ± .007	.017 ± .005
cadata	.083 ± .029	**.063 ± .012**	**.058 ± .009**	**.056 ± .007**	**.057 ± .009**	**.060 ± .013**	**.057 ± .005**
cpusmall	.024 ± .012	$\boxed{.027 \pm .011}$	$\boxed{.028 \pm .009}$	.025 ± .010	.024 ± .005	.025 ± .011	.024 ± .009
housing	.039 ± .010	.036 ± .009	.033 ± .006	.036 ± .009	**.034 ± .008**	**.034 ± .009**	**.033 ± .005**
mg	.051 ± .009	**.039 ± .006**	**.038 ± .006**	**.035 ± .015**	**.038 ± .014**	**.038 ± .006**	**.038 ± .006**
mpg	.022 ± .007	.020 ± .006	.018 ± .006	.021 ± .008	.020 ± .006	**.019 ± .004**	**.018 ± .004**
pyrim	.023 ± .006	**.021 ± .005**	.022 ± .005	$\boxed{.052 \pm .014}$	**.020 ± .007**	**.020 ± .006**	**.020 ± .006**
Ave. Mse.	.035	.030	.029	.035	.029	.030	.028
Win/Tie/Loss against 1NN		4/3/1	3/4/1	3/3/2	5/2/1	**6/2/0**	**6/2/0**

line method without using weakly supervised data. This problem is important whereas no general formulation has been proposed to guide safe weakly supervised learning. In this paper we present a scheme that builds the final prediction result by integrating multiple weakly supervised learners. The resultant formulation has safeness guarantees for many commonly used convex loss functions for both regression and classification tasks. Besides, it is capable of embedding prior knowledge on the weights of base learners. The resultant formulation is globally solved by simple convex optimization efficiently. Experiments on three weakly supervised learning tasks including label noise learning, domain adaptation and semi-supervised learning validate the effectiveness of our proposed algorithms.

There are many interesting future works. For example, our method is a two-stage method and may loss some information, whereas directly one stage method that takes the generation of base learners into account, would be worth studying. Moreover, the study of other weakly supervised setting such as new class detection (Mu, Ming, and Zhou 2017), is an interesting issue in future.

References

Alfonseca, E.; Filippova, K.; Delort, J.-Y.; and Garrido, G. 2012. Pattern learning for relation extraction with a hierarchical topic model. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 54–59.

Argyriou, A.; Maurer, A.; and Pontil, M. 2008. An algorithm for transfer learning in a heterogeneous environment. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases*, 71–85.

Bakker, B., and Heskes, T. 2003. Task clustering and gating for bayesian multitask learning. *Journal of Machine Learning Reserch* 4:83–99.

Balsubramani, A., and Freund, Y. 2015. Optimally combining classifiers using unlabeled data. In *Proceedings of International Conference on Learning Theory*, 211–225.

Bates, J. M., and Granger, C. W. 1969. The combination of forecasts. *Operational Research* 20(4):451–468.

Bootkrajang, J., and Kabán, A. 2012. Label-noise robust logistic regression and its applications. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases*, 143–158.

Boyd, S., and Vandenberghe, L. 2004. *Convex optimization*. Cambridge University Press.

Chang, C.-C., and Lin, C.-J. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2(3):27.

Chapelle, O.; Schölkopf, B.; Zien, .; et al. 2006. *Semi-supervised learning*. MIT Press.

Chawla, N. V., and Karakoulas, G. 2005. Learning from labeled and unlabeled data: An empirical study across techniques and domains. *Journal of Artificial Intelligence Research* 23:331–366.

Dai, W.; Yang, Q.; Xue, G.-R.; and Yu, Y. 2007. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning*, 193–200.

Dietterich, T. G. 2000. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning* 40(2):139–157.

Frénay, B., and Verleysen, M. 2014. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems* 25(5):845–869.

Ge, L.; Gao, J.; Ngo, H.; Li, K.; and Zhang, A. 2014. On handling negative transfer and imbalanced distributions in multiple source transfer learning. *Statistical Analysis and Data Mining* 7(4):254–271.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123(1):32–73.

Lang, K. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th International Cconference on Machine Learning*, 331–339.

Li, Y.-F., and Zhou, Z.-H. 2015. Towards making unlabeled data never hurt. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(1):175–188.

Li, L.; Jin, X.; and Long, M. 2012. Topic correlation analysis for cross-domain text classification. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 998–1004.

Li, Y.-F.; Zha, H.-W.; and Zhou, Z.-H. 2017. Learning safe prediction for semi-supervised regression. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 2217–2223.

Manwani, N., and Sastry, P. 2013. Noise tolerance under risk minimization. *IEEE Transactions on Cybernetics* 43(3):1146–1151.

Mu, X.; Ming, K.; and Zhou, Z.-H. 2017. Classification under streaming emerging new classes: A solution using completely-random trees. *IEEE Transactions on Knowledge and Data Engineering*.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22(10):1345–1359.

Pan, S. J.; Tsang, I. W.; Kwok, J. T.; and Yang, Q. 2011. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks* 22(2):199–210.

Rosasco, L.; De Vito, E.; Caponnetto, A.; Piana, M.; and Verri, A. 2004. Are loss functions all the same? *Neural Computation* 16(5):1063–1076.

Rosenstein, M. T.; Marx, Z.; and Kaelbling, L. P. 2005. To transfer or not to transfer. In *a NIPS-05 Workshop on Inductive Transfer: 10 Years Later*.

Smola, A. J., and Schölkopf, B. 2004. A tutorial on support vector regression. *Statistics and Computing* 14(3):199–222.

Wei, T.; Guo, L.-Z.; Li, Y.-F.; and Gao, W. 2017. Learning safe multi-label prediciton for weakly labeled data. *Machine Learning*.

Willmott, C. J., and Matsuura, K. 2005. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research* 30(1):79–82.

Xue, G.-R.; Dai, W.; Yang, Q.; and Yu, Y. 2008. Topic-bridged plsa for cross-domain text classification. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 627–634.

Yan, K.; Kou, L.; and Zhang, D. 2016. Domain adaptation via maximum independence of domain features. *arXiv preprint arXiv:1603.04535*.

Yarowsky, D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*, 189–196.

Yuille, A., and Rangarajan, A. 2003. The concave-convex procedure. *Neural Computation* 15(4):915–936.

Zhou, Z.-H. 2012. *Ensemble Methods: Foundations and Algorithms*. Boca Raton: FL: Chapman & Hall.

Zhou, Z.-H. 2017. A brief introduction to weakly supervised learning. *National Science Review*.