Centroid Estimation with Guaranteed Efficiency: A General Framework for Weakly Supervised Learning

Chen Gong, *Member, IEEE,* Jian Yang, *Member, IEEE,*
Jane You, Masashi Sugiyama, *Senior Member, IEEE*

Abstract—In this paper, we propose a general framework termed "Centroid Estimation with Guaranteed Efficiency" (CEGE) for Weakly Supervised Learning (WSL) with incomplete, inexact, and inaccurate supervision. The core of our framework is to devise an unbiased and statistically efficient risk estimator that is applicable to various weak supervision. Specifically, by decomposing the loss function (*e.g.*, the squared loss and hinge loss) into a label-independent term and a label-dependent term, we discover that only the latter is influenced by the weak supervision and is related to the *centroid* of the entire dataset. Therefore, by constructing two auxiliary pseudo-labeled datasets with synthesized labels, we derive unbiased estimates of centroid based on the two auxiliary datasets, respectively. These two estimates are further linearly combined with a properly decided coefficient which makes the final combined estimate not only unbiased but also statistically efficient. This is better than some existing methods that only care about the unbiasedness of estimation but ignore the statistical efficiency. The good statistical efficiency of the derived estimator is guaranteed as we theoretically prove that it acquires the minimum variance when estimating the centroid. As a result, intensive experimental results on a large number of benchmark datasets demonstrate that our CEGE generally obtains better performance than the existing approaches related to typical WSL problems including semi-supervised learning, positive-unlabeled learning, multiple instance learning, and label noise learning.

Index Terms—Weakly Supervised Learning, Centroid Estimation, Unbiasedness, Statistical Efficiency.

1 Introduction

As a classic yet important branch in machine learning, Weakly Supervised Learning (WSL) has been intensively studied for several decades, of which the target is to train a proper classifier/regressor based on a weakly labeled dataset. In contrast to the conventional fully supervised learning in which the training set is associated with sufficient, definite and precise labels, the supervision information for WSL can be incomplete, inexact, and inaccurate [1]. Therefore, WSL is quite useful when the strong supervision is difficult to obtain due to limited human resources, unaffordable monetary costs, or other unavoidable practical limitations.

According to [1], the existing WSL methods usually deal with the following three types of weak supervision:

- **Incomplete supervision**. This means that the supervision information available is inadequate to train a good model. This usually occurs when sufficient labeled training data cannot be acquired for the targeted task. The representative learning paradigms include semi-supervised learning [2], positive-unlabeled learning [3], unlabeled-unlabeled learning [4], active learning [5], domain adaptation [6], etc.
- **Inexact supervision**. This means that only a coarse-grained supervision is provided for every example, and the exact example-label pairs as in fully supervised learning are not available. This could happen when the annotators cannot provide definite labels of examples or are not sure about their labeling results. Typical learning paradigms include multiple instance learning [7], partial label learning [8] (*a.k.a.* superset label learning [9]), complementary label learning [10], positive-confidence learning [11], etc.
- **Inaccurate supervision**. This means that the labels assigned to the training data can be incorrect, which may mislead the learning algorithm and hurt the performance. This is usually due to labeler fatigue, measurement error, or other subjective or objective factors. For example, label noise learning [12]

- *C. Gong was supported by NSF of China (No: 61973162), the Fundamental Research Funds for the Central Universities (No: 30920032202), CCF-Tencent Open Fund (No: RAGR20200101), the "Young Elite Scientists Sponsorship Program" by CAST (No: 2018QNRC001), and Hong Kong Scholars Program (No: XJ2019036). J. Yang was supported by NSF of China (No: U1713208) and "111 Program" (No: AH92005MS). J. You was supported by the Hong Kong Polytechnic University grants (Nos: YZ3K, UAJP/UAGK, and ZVRH). M. Sugiyama was supported by KAKENHI (No: 20H04206).*

- *C. Gong is with the PCA Lab, the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, P.R. China; and is also with the Department of Computing, Hong Kong Polytechnic University, Hong Kong SAR, China.*
  E-mail: chen.gong@njust.edu.cn
- *J. Yang is with the PCA Lab, the Jiangsu Key Laboratory of Image and Video Understanding for Social Security, and the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, P.R. China.*
  E-mail: csjyang@njust.edu.cn
- *J. You is with the Department of Computing, Hong Kong Polytechnic University, Hong Kong, P.R. China.*
  E-mail: jane.you@polyu.edu.hk
- *M. Sugiyama is with the RIKEN Center for Advanced Intelligence Project, Tokyo, Japan; and is also with the Graduate School of Frontier Sciences, The University of Tokyo, Chiba, Japan.*
  E-mail: sugi@k.u-tokyo.ac.jp
- *Corresponding authors: J. Yang and J. You.*

and crowdsourcing [13] belong to this type.

From the above analyses, we learn that WSL is a broad topic and numerous WSL models have been developed to handle various kinds of weak supervision. Since the core of most WSL methods is to find suitable strategies to tackle weak supervision, it is natural to ask whether there exists a unified framework for the common WSL methods. To our best knowledge, there are two prior works targeting such a general framework, namely the WEakly LabeLed Support Vector Machines (WELLSVM) [14] and SAFE Weakly supervised learning (SAFEW) [15]. Among them, WELLSVM establishes a mixed integer programming framework by employing SVM with the large-margin property, and devises a label generation strategy to address the scalability issue. However, since this framework is presented as a constrained minmax optimization problem, the solution is quite complicated and the popular gradient-based solvers cannot be used. Besides, since WELLSVM was proposed in an early time, its applicability to some recent WSL paradigms, *e.g.*, label noise learning, still remains unclear. For SAFEW, it solves the safety problem in various WSL approaches by integrating multiple base learners, so that the final performance will not decrease with more weakly-labeled data. Nevertheless, because the goal of SAFEW is to guarantee safety, it simply assembles a set of existing weak models to achieve stable output, and does not delve into the essence of various weak supervision nor building specific models to tackle it directly. Therefore, the targets of SAFEW and this work are different.

Given the above reasons, this paper proposes a novel general framework for WSL termed "Centroid Estimation with Guaranteed Efficiency" (CEGE). Specifically, under the formulation of empirical risk minimization, we explicitly study the impact of weak supervision to the empirical risk, and devise an effective regularizer based on data centroid estimation to deal with various weakly-labeled data in a unified way. To make it clear, suppose the original strongly-supervised dataset is S, and we can only observe its weakly-labeled counterpart $\widetilde{S}$ [1]. To recover the real empirical risk of any classifier on S, we decompose the employed loss function (*e.g.*, the squared loss and hinge loss) into a label-independent term and a label-dependent term [16], [17], [18], among which only the latter is affected by imperfect labels. Moreover, by noting that the value of label-dependent part depends on the dataset centroid $\hat{\mu}(S)$ with "$\hat{\mu}(\cdot)$" denoting the centroid operator on a sample, we know that the empirical risk of any trained model on the unknown ground-truth labels can be acquired if $\hat{\mu}(S)$ is accurately estimated. To achieve this, we construct two auxiliary pseudo-labeled datasets $\widetilde{S_1}$ and $\widetilde{S_2}$ based on $\widetilde{S}$, and thus $\hat{\mu}(S)$ can be unbiasedly estimated (denoted as "$\breve{\mu}(S)$") by linearly combining the estimators $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ which are related to $\hat{\mu}(\widetilde{S_1})$ and $\hat{\mu}(\widetilde{S_2})$ accordingly (see Fig. 1). To improve the statistical efficiency and obtain a reliable estimator $\breve{\mu}(S)$, we minimize its variance which leads to the optimal coefficient β in linear combination. We strictly prove that the induced variance of $\breve{\mu}(S)$ in estimating $\hat{\mu}(S)$ is smaller than any of that from $\widetilde{S_1}$ or $\widetilde{S_2}$. Therefore, $\hat{\mu}(S)$ is estimated in an unbiased and statistically efficient way, which brings about precise and trustable estimation. Based on the unbiased centroid estimation with guaranteed statistical efficiency, we establish a convex optimization problem which can be easily solved by gradient descend. Our proposed CEGE framework is validated with the applications to several representative WSL paradigms such as Semi-Supervised Learning (SSL), Positive-Unlabeled Learning (PUL), Multiple Instance Learning (MIL), and Label Noise Learning (LNL). The experimental results on massive benchmark datasets reveal that CEGE generally achieves better or comparable performance when compared with major algorithms in SSL, PUL, MIL, and LNL.

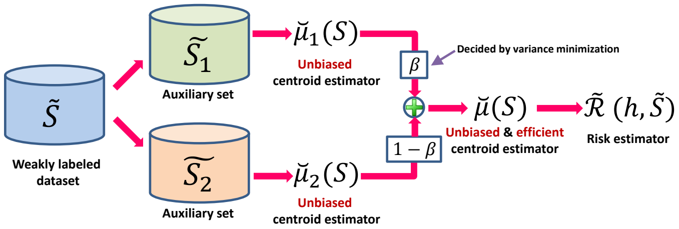

Fig. 1: The pipeline of our method. Based on a weakly-labeled dataset $\widetilde{S}$ corrupted from S, we build two auxiliary sets $\widetilde{S_1}$ and $\widetilde{S_2}$, which respectively yields an unbiased estimate of the centroid of S as $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$. Then $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ are linearly combined with the optimized coefficient β to form a both unbiased and statistically efficient centroid estimate $\breve{\mu}(S)$. Finally, an effective risk estimator of any hypothesis h on $\widetilde{S}$ is derived as $\widetilde{\mathcal{R}}(h, \widetilde{S})$.

In fact, the technique of centroid estimation has been utilized in [16] and [18] for addressing LNL and PUL, respectively, and this paper extends them by showing that a similar technique is applicable to a wide range of related WSL topics. More importantly, [16] and [18] only used one pseudo-labeled set to estimate the data centroid $\hat{\mu}(S)$, so the variance of the estimator $\breve{\mu}(S)$ could be high and the estimated centroid can sometimes be imprecise. To avoid such an undesirable estimate, they treated the real centroid $\hat{\mu}(S)$ as a variable to be learned, and introduced an additional constraint to allow the learned $\hat{\mu}^*(S)$ to slightly differ from the estimated centroid $\breve{\mu}(S)$. However, the tightness between $\breve{\mu}(S)$ and $\hat{\mu}^*(S)$ is controlled by a hyperparameter which is difficult to manually tune, and the introduced constraint also makes the optimization problem difficult to solve. In contrast, this paper remedies this defect by introducing two auxiliary pseudo-labeled datasets and employing variance reduction during centroid estimation, of which both the unbiasedness and statistical efficiency are theoretically guaranteed.

2 The General CEGE Framework

In this section, we start by reformulating the empirical risk in traditional fully supervised setting (Section 2.1), and then introduce our CEGE framework for weakly-supervised setting (Section 2.2). Finally, the basic and the improved ways for computing the involved risk estimator (Sections 2.3 and 2.4), as well as the optimal parameter calculation (Section 2.5) are detailed. The major mathematical notations used hereinafter are summarized in Table 2.

2.1 Fully Supervised Setting

Let $\mathbf{x} \in \mathbb{R}^d$ be a d-dimensional input variable in the feature space $\mathcal{X}$ and y be an output variable in the label space $\mathcal{Y} = \{+1, -1\}$, then a sample $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{\bar{n}}$ with size $\bar{n}$ is assumed to be independently generated from an unknown distribution $\mathcal{D}$ defined on $\mathcal{X} \times \mathcal{Y}$. Denoting the hypothesis space as $\mathcal{H}$, then the target of a fully supervised algorithm is to find a suitable decision function $h \in \mathcal{H} : \mathbb{R}^d \rightarrow \mathbb{R}$ expressed as $h(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ on S with $\mathbf{w}$ being

[1] In this paper, a variable with superscript "$\sim$", "$\hat{}$" and "$\breve{}$" means that it is a weakly-labeled quantity (*e.g.*, unknown or noisy), empirical quantity, and estimated quantity correspondingly.

TABLE 1: The decomposition of commonly adopted loss functions [17], [18], where $z = yh(\mathbf{x})$ is the functional margin, and $[\cdot]_+ = \max(\cdot, 0)$. The decomposition of the hinge loss is actually conducted on its upper bound as revealed by [18].

loss	$\ell(z)$	label-independent term	label-dependent term	$\varphi(h)$	Q
squared loss	$(z-1)^2$	z^2+1	$-2z$	h^2+1	-2
logistic loss	$\log(1+e^{-z})$	$\frac{1}{2}\log(2+e^{z}+e^{-z})$	$-z/2$	$\frac{1}{2}\log(2+e^{h}+e^{-h})$	$-1/2$
perceptron loss	$\max(0,-z)$	$\frac{1}{2}z\cdot\mathrm{sgn}(z\geq 0)$	$-z/2$	$\frac{1}{2}h\cdot\mathrm{sgn}[h\geq 0]$	$-1/2$
unhinged loss	$1-z$	1	$-z$	1	-1
Matsushita loss	$\sqrt{1+z^2}-z$	$\sqrt{1+z^2}$	$-z$	$\sqrt{1+h^2}$	-1
hinge loss	$[1-z]_+$	$\frac{1}{2}([1-z]_+ + [1+z]_+)$	$\frac{1}{2}(1-z)$	$\frac{1}{2}([1-h]_+ + [1+h]_+)$	$-1/2$

TABLE 2: Summary of main mathematical notations.

Notation	Mathematical meaning
$(\mathbf{x}, y)$, $(\mathbf{x}, \tilde{y})$	A pair of random variables $(\mathbf{x}, y)$ and the observed weakly-labeled counterpart $(\mathbf{x}, \tilde{y})$.
$S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{\bar{n}}$	The unobserved fully-supervised sample S with $\bar{n}$ data $(\mathbf{x}_i, y_i)$.
$\widetilde{S} = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^{\bar{n}}$	The observed weakly-labeled sample $\widetilde{S}$ with $\bar{n}$ corrupted training data $(\mathbf{x}_i, \tilde{y}_i)$.
$\mathcal{D}$, $\widetilde{\mathcal{D}}$	The unknown original distribution $\mathcal{D}$ and its corrupted version $\widetilde{\mathcal{D}}$.
$\widetilde{S_1}$, $\widetilde{S_2}$	The two introduced auxiliary pseudo-labeled datasets.
$\mu(\mathcal{D})$	The (expected) centroid of some random variable defined on the distribution $\mathcal{D}$.
$\hat{\mu}(S)$, $\breve{\mu}(S)$	$\hat{\mu}(S)$ is the (empirical) centroid of a sample S, and $\breve{\mu}(S)$ is its estimator.
$\hat{\mathcal{R}}(h, S)$	Empirical risk of hypothesis h on a sample S.
$\mathbb{E}[\cdot]$	Mathematical expectation of some random variable.
$\Sigma[\cdot]$, $\hat{\Sigma}[\cdot]$	$\Sigma[\cdot]$ is the covariance of some random variable, and $\hat{\Sigma}[\cdot]$ is its corresponding empirical quantity.
$D[\cdot,\cdot]$, $\hat{D}[\cdot,\cdot]$	$D[\cdot,\cdot]$ is the cross-covariance of two random variables, and $\hat{D}[\cdot,\cdot]$ is its corresponding empirical quantity.

model parameter, which can decide the label of any unseen test data $\mathbf{x}_{\text{test}}$ as $y_{\text{test}} = \mathrm{sgn}(h(\mathbf{x}_{\text{test}}))$.

As S is with strong supervision in which all ground-truth labels $\{y_i\}_{i=1}^{\bar{n}}$ are explicitly and correctly provided, the empirical risk of any hypothesis h on S is thus formulated as

$$\hat{\mathcal{R}}(h, S) = \frac{1}{\bar{n}} \sum_{(\mathbf{x}_i, y_i)\in S} \ell(h(\mathbf{x}_i), y_i), \tag{1}$$

where $\ell : \mathbb{R}\times\mathbb{R}\to\mathbb{R}$ is some loss function which evaluates the difference between the model output $h(\mathbf{x})$ and the ground-truth label y.

According to [16], [17], [18], many existing loss functions or their upper bounds can be decomposed as a label-independent term plus a label-dependent term (see Table 1), so $\hat{\mathcal{R}}(h, S)$ in Eq. (1) is converted to the following form after simple mathematical derivations, namely

$$\hat{\mathcal{R}}(h, S) = \frac{1}{\bar{n}} \left[\sum_{(\mathbf{x}_i, y_i)\in S} \varphi(h(\mathbf{x}_i)) + Q \sum_{(\mathbf{x}_i, y_i)\in S} y_i \langle \mathbf{w}, \mathbf{x}_i \rangle \right], \tag{2}$$

where Q is a scalar, and $\varphi(h(\mathbf{x}_i))$ is some function irrelevant to the labels y_i $(i = 1, 2, \ldots, \bar{n})$. The specific forms of $\varphi(h(\mathbf{x}_i))$ and Q depend on the adopted loss function, and they are summarized in Table 1. It can be found that only the second term in the bracket of Eq. (2) is related to the label value, while the first term is related to the computable model output $h(\mathbf{x}_i)$.

Now we introduce the notion of the *centroid* with respect to the sample S and distribution $\mathcal{D}$, which are respectively represented as $\hat{\mu}(S) = \frac{1}{\bar{n}} \sum_{(\mathbf{x}_i, y_i)\in S} y_i \mathbf{x}_i$ and $\mu(\mathcal{D}) = \mathbb{E}_{(\mathbf{x}, y)\sim\mathcal{D}}[y\mathbf{x}]$ with "$\mathbb{E}[\cdot]$" denoting the expectation. Then, by further investigating the second term of Eq. (2), we see that it has a close relationship with the centroid of S. As such, Eq. (2) can be further transformed to

$$\hat{\mathcal{R}}(h, S) = \frac{1}{\bar{n}} \sum_{(\mathbf{x}_i, y_i)\in S} \varphi(h(\mathbf{x}_i)) + Q\langle \mathbf{w}, \hat{\mu}(S) \rangle. \tag{3}$$

2.2 Our Framework for Weakly Supervised Setting

In the WSL setting, we are only accessible to the weakly-labeled version of S, i.e., $\widetilde{S} = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^{\bar{n}}$ generated from the corrupted distribution $\widetilde{\mathcal{D}}$, where the labels $\{\tilde{y}_i\}_{i=1}^{\bar{n}}$ might be unobserved, ambiguous, or inaccurate under different weakly supervised conditions. We hope that we can still find a robust h on $\widetilde{S}$ which has good generalizability on the test data.

However, due to the imperfect labels $\{\tilde{y}_i\}_{i=1}^{\bar{n}}$ and unavailability of $\{y_i\}_{i=1}^{\bar{n}}$, Eq. (3) cannot be directly computed. Therefore, we need to find an effective estimator for $\hat{\mathcal{R}}(h, S)$ based on $\widetilde{S}$, which are subsequently denoted by $\widetilde{\mathcal{R}}(h, \widetilde{S})$. Besides, considering that in some WSL problems such as SSL, PUL and MIL, there also exists a strong supervision set S_{strong} in which the labels of examples are definite and correct. Then we may define the empirical risk on S_{strong} to enforce the model output to be consistent with the given labels, namely

$$\hat{\mathcal{R}}_{\text{strong}}(h, S_{\text{strong}}) = \frac{1}{|S_{\text{strong}}|} \sum_{(\mathbf{x}_i, y_i)\in S_{\text{strong}}} \ell(h(\mathbf{x}_i), y_i), \tag{4}$$

where $|S_{\text{strong}}|$ is the size of S_{strong}.

Based on the above considerations, the final objective of our CEGE framework is expressed as

$$\min_{\mathbf{w}} \ \widetilde{\mathcal{R}}(h, \widetilde{S}) + \gamma_1 \hat{\mathcal{R}}_{\text{strong}}(h, S_{\text{strong}}) + \gamma_2 \|\mathbf{w}\|^2, \tag{5}$$

where γ_1 and γ_2 are nonnegative trade-off parameters, and the term $\|\mathbf{w}\|^2$ is added to avoid overfitting[2].

Remark: Since the main purpose of this paper is to show the generality and applicability of our developed CEGE for WSL, we simply use the linear model $h(\mathbf{x}) = \mathbf{w}^\top\mathbf{x}$ in Eq. (5) to present our model under the framework of empirical risk minimization. However, CEGE can also render nonlinear classifiers if we apply the kernel trick [19] to Eq. (5), or replace the cross-entropy loss with $\widetilde{\mathcal{R}}(h, \widetilde{S}) + \gamma_1 \hat{\mathcal{R}}_{\text{strong}}(h, S_{\text{strong}})$ in a deep neural network.

In Eq. (5), the form of $\widetilde{\mathcal{R}}(h, \widetilde{S})$ remains unspecified as it involved the estimation for $\hat{\mu}(S)$. We will detail it in the following sections.

[2] The feature vector $\mathbf{x}$ is often practically augmented as $\mathbf{x} = [\mathbf{x}^\top 1]^\top$ and $\mathbf{w}$ is also expanded accordingly. In this case, the regularizer $\|\mathbf{w}\|^2$ is replaced by $\|\mathbf{J}\mathbf{w}\|^2$ where $\mathbf{J}$ is a $d\times(d+1)$ matrix with $\mathbf{J}_{i,i} = 1$ for $i = 1, \ldots, d$, and $\mathbf{J}_{i,j} = 0$ for other elements.

2.3 Estimating $\hat{\mu}(S)$ from Single Auxiliary Set

To accurately estimate the real centroid $\hat{\mu}(S)$, we propose to utilize the observed dataset $\widetilde{S}$ to build an auxiliary pseudo-labeled dataset $\widetilde{S_1}$ with synthesized label $\tilde{y}^{(1)}$, and then use its centroid to unbiasedly estimate $\hat{\mu}(S)$.

Given an auxiliary dataset $\widetilde{S_1} = \{(\mathbf{x}_i, \tilde{y}_i^{(1)})\}_{i=1}^{\bar{n}}$ that is supposed to be generated from a distribution $\widetilde{\mathcal{D}_1}$, on which the random variable is $(\mathbf{x}, \tilde{y}^{(1)})$, we denote the label flipping probabilities from the real label y to the synthesized label $\tilde{y}^{(1)}$ as $\eta_P^{(1)} = P(\tilde{y}^{(1)} = -1|y = +1)$ and $\eta_N^{(1)} = P(\tilde{y}^{(1)} = +1|y = -1)$, where "$P(\cdot)$" denotes probability throughout this paper. Therefore, as will be detailed in Section 4, we can show that $\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y)] = y\mathbf{x}/\tau_1$ where τ_1 is some constant related to $\eta_P^{(1)}$ and $\eta_N^{(1)}$. By assuming that the densities $P(\mathbf{x})$ on $\mathcal{D}$ and $\widetilde{\mathcal{D}_1}$ are identical, and similarly defining the centroid of $\widetilde{S_1}$ and $\widetilde{\mathcal{D}_1}$ as $\hat{\mu}(\widetilde{S_1}) = \frac{1}{\bar{n}}\sum_{(\mathbf{x}_i,\tilde{y}_i^{(1)})\in\widetilde{S_1}} \tilde{y}_i^{(1)}\mathbf{x}_i$ and $\mu(\widetilde{\mathcal{D}_1}) = \mathbb{E}_{(\mathbf{x},\tilde{y}^{(1)})\sim\widetilde{\mathcal{D}_1}}[\tilde{y}^{(1)}\mathbf{x}]$, we learn that

$$\begin{aligned}\mu(\widetilde{\mathcal{D}_1}) &= \mathbb{E}_{(\mathbf{x},\tilde{y}^{(1)})\sim\widetilde{\mathcal{D}_1}}[\tilde{y}^{(1)}\mathbf{x}] \\ &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y)]] \qquad (6) \\ &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[y\mathbf{x}/\tau_1] \\ &= \mu(\mathcal{D})/\tau_1, \qquad (7)\end{aligned}$$

where Eq. (6) is due to the law of iterated expectations. As a sequel, we know $\mathbb{E}_{\tilde{y}^{(1)}}[\hat{\mu}(\widetilde{S_1})] = \hat{\mu}(S)/\tau_1$, which suggests that an unbiased estimator of $\hat{\mu}(S)$ dubbed $\breve{\mu}_1(S)$ is $\breve{\mu}_1(S) = \tau_1\hat{\mu}(\widetilde{S_1})$.

2.4 Estimating $\hat{\mu}(S)$ from Two Auxiliary Sets for Variance Reduction

As will be later illustrated in the experiments, the output of single estimator $\breve{\mu}_1(S)$ might be inaccurate as its variance might be large in some cases. To solve this problem, here we propose to introduce another auxiliary dataset $\widetilde{S_2}$ with synthesized label $\tilde{y}^{(2)}$ to decrease the variance when estimating $\hat{\mu}(S)$.

Suppose that the second auxiliary dataset $\widetilde{S_2} = \{(\mathbf{x}_i, \tilde{y}_i^{(2)})\}_{i=1}^{\bar{n}}$ is generated from a distribution $\widetilde{\mathcal{D}_2}$, on which the random variable is $(\mathbf{x}, \tilde{y}^{(2)})$, then we acquire the label flipping probabilities from y to the synthesized label $\tilde{y}^{(2)}$ as $\eta_P^{(2)} = P(\tilde{y}^{(2)} = -1|y = +1)$ and $\eta_N^{(2)} = P(\tilde{y}^{(2)} = +1|y = -1)$, which lead to $\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y)] = y\mathbf{x}/\tau_2$ with τ_2 being a constant related to $\eta_P^{(2)}$ and $\eta_N^{(2)}$ (see Section 4 for details). Similarly to the notations and derivations in Eq. (7), we have $\mathbb{E}_{\tilde{y}^{(2)}}[\hat{\mu}(\widetilde{S_2})] = \hat{\mu}(S)/\tau_2$, which means that another unbiased estimator of $\hat{\mu}(S)$ dubbed $\breve{\mu}_2(S)$ can be written as $\breve{\mu}_2(S) = \tau_2\hat{\mu}(\widetilde{S_2})$. Akin to massive prior WSL works [12], [18], [20], [21], [22], [23], we also assume that the class prior $\pi = P(y = +1)$ is known, therefore the relationship between $\eta_P^{(1)}$, $\eta_N^{(1)}$, $\eta_P^{(2)}$, $\eta_N^{(2)}$ appeared above and $\pi = P(y = +1)$ can be built. Practically, the value of π can be obtained via cross-validation [12], [18], or via some specific class prior estimation methods such as [24], [25], [26], [27], [28].

Consequently, due to the fact that the linear combination of two unbiased estimators is still unbiased, an unbiased estimator of the real dataset centroid $\hat{\mu}(S)$ (namely $\breve{\mu}(S)$) is linearly represented by $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ as

$$\begin{aligned}\breve{\mu}(S) &= \beta\breve{\mu}_1(S) + (1-\beta)\breve{\mu}_2(S) \\ &= \beta\tau_1\hat{\mu}(\widetilde{S_1}) + (1-\beta)\tau_2\hat{\mu}(\widetilde{S_2}), \qquad (8)\end{aligned}$$

where $\beta \in \mathbb{R}$ is a weighting parameter. Here the auxiliary datasets $\widetilde{S_1}$ and $\widetilde{S_2}$ can be established via different ways, which will be later explained in Section 4 for various WSL problems.

2.5 Optimal β Determination

An advantage of the proposed CEGE over existing methods [16], [18] in estimating $\hat{\mu}(S)$ is that CEGE uses the linear combination of the estimators from two auxiliary sets, so that the variance of the estimator can be reduced apart from the unbiasedness. However, this brings about a new question, namely how to decide the optimal weighting coefficient β in Eq. (8). Here we propose to choose β that minimizes the variance of the estimator $\breve{\mu}(S)$, so as to make the estimation accurate and reliable. To this end, we define the random variables of the (estimated) sample means $\breve{\mu}(S)$, $\hat{\mu}(\widetilde{S_1})$ and $\hat{\mu}(\widetilde{S_2})$ as $\breve{\mathbf{m}}$, $\hat{\mathbf{m}}_1$ and $\hat{\mathbf{m}}_2$ correspondingly, and also respectively define $\mathbf{m}_1 = \tilde{y}^{(1)}\mathbf{x}$ and $\mathbf{m}_2 = \tilde{y}^{(2)}\mathbf{x}$ as the variables corresponding to $\{\tilde{y}_i^{(1)}\mathbf{x}_i\}_{i=1}^{\bar{n}}$ and $\{\tilde{y}_i^{(2)}\mathbf{x}_i\}_{i=1}^{\bar{n}}$, of which their expectations are $\mu(\mathbf{m}_1) = \mathbb{E}_{(\mathbf{x},\tilde{y}^{(1)})\sim\widetilde{\mathcal{D}_1}}[\mathbf{m}_1]$ and $\mu(\mathbf{m}_2) = \mathbb{E}_{(\mathbf{x},\tilde{y}^{(2)})\sim\widetilde{\mathcal{D}_2}}[\mathbf{m}_2]$ accordingly. Then according to Eq. (8), the covariance of $\breve{\mathbf{m}}$ is derived as

$$\begin{aligned}\Sigma[\breve{\mathbf{m}}] &= \Sigma[\beta\tau_1\hat{\mathbf{m}}_1 + (1-\beta)\tau_2\hat{\mathbf{m}}_2] \\ &= \beta^2\tau_1^2\Sigma[\hat{\mathbf{m}}_1] + (1-\beta)^2\tau_2^2\Sigma[\hat{\mathbf{m}}_2] + 2\beta(1-\beta)\tau_1\tau_2 D[\hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2] \\ &= \beta^2\tau_1^2\Sigma\left[\frac{1}{\bar{n}}\sum_{i=1}^{\bar{n}}\mathbf{m}_{1,i}\right] + (1-\beta)^2\tau_2^2\Sigma\left[\frac{1}{\bar{n}}\sum_{i=1}^{\bar{n}}\mathbf{m}_{2,i}\right] \\ &\quad + 2\beta(1-\beta)\tau_1\tau_2 D\left[\frac{1}{\bar{n}}\sum_{i=1}^{\bar{n}}\mathbf{m}_{1,i}, \frac{1}{\bar{n}}\sum_{i=1}^{\bar{n}}\mathbf{m}_{2,i}\right] \\ &= \frac{1}{\bar{n}}\beta^2\tau_1^2\Sigma[\mathbf{m}_1] + \frac{1}{\bar{n}}(1-\beta)^2\tau_2^2\Sigma[\mathbf{m}_2] \\ &\quad + \frac{2}{\bar{n}}\beta(1-\beta)\tau_1\tau_2 D[\mathbf{m}_1, \mathbf{m}_2], \qquad (9)\end{aligned}$$

where $\Sigma[\mathbf{m}_j] = \Sigma_{(\mathbf{x},\tilde{y}^{(j)})\sim\widetilde{\mathcal{D}_j}}[\tilde{y}^{(j)}\mathbf{x}]$ for $j = 1, 2$ compute the covariance of $\mathbf{m}_j$; $\Sigma[\hat{\mathbf{m}}_j] = \Sigma_{(\mathbf{x},\tilde{y}^{(j)})\sim\widetilde{\mathcal{D}_j}}[\mu(\mathbf{m}_j)]$ for $j = 1, 2$ calculate the covariance of $\hat{\mathbf{m}}_j$; $\mathbf{m}_{j,i}$ ($i = 1, \ldots, \bar{n}$) are the i-th duplicates of $\mathbf{m}_j$ ($j = 1, 2$); and "$D[\cdot,\cdot]$" denotes the cross-covariance matrix between two random vectors.

Therefore, the empirical version of $\Sigma[\breve{\mathbf{m}}]$, namely $\hat{\Sigma}[\breve{\mu}(S)]$, is represented by

$$\begin{aligned}\hat{\Sigma}[\breve{\mu}(S)] = &\frac{1}{\bar{n}}\beta^2\tau_1^2\hat{\Sigma}[\widetilde{S_1}] + \frac{1}{\bar{n}}(1-\beta)^2\tau_2^2\hat{\Sigma}[\widetilde{S_2}] \\ &+ \frac{2}{\bar{n}}\beta(1-\beta)\tau_1\tau_2 \cdot \hat{D}[\widetilde{S_1}, \widetilde{S_2}], \qquad (10)\end{aligned}$$

where $\hat{\Sigma}[\widetilde{S_1}]$, $\hat{\Sigma}[\widetilde{S_2}]$ and $\hat{D}[\widetilde{S_1}, \widetilde{S_2}]$ are empirical estimates of $\Sigma[\mathbf{m}_1]$, $\Sigma[\mathbf{m}_2]$ and $D[\mathbf{m}_1, \mathbf{m}_2]$ in Eq. (9), respectively.

Theorem 1. *Given two sets $\widetilde{S_1}$ and $\widetilde{S_2}$ respectively generated from $\widetilde{\mathcal{D}_1}$ and $\widetilde{\mathcal{D}_2}$, and $(\mathbf{x}, \tilde{y}^{(1)}) \sim \widetilde{\mathcal{D}_1}$ and $(\mathbf{x}, \tilde{y}^{(2)}) \sim \widetilde{\mathcal{D}_2}$. By denoting $\mathbf{m}_1 = \tilde{y}^{(1)}\mathbf{x}$ and $\mathbf{m}_2 = \tilde{y}^{(2)}\mathbf{x}$, then the empirical estimate $\hat{D}[\widetilde{S_1}, \widetilde{S_2}]$ for $D[\mathbf{m}_1, \mathbf{m}_2]$ is*

$$\hat{D}[\widetilde{S_1}, \widetilde{S_2}] = \frac{1}{\bar{n}}\sum_{i=1}^{\bar{n}}\tilde{y}_i^{(1)}\tilde{y}_i^{(2)}\mathbf{x}_i\mathbf{x}_i^{\top} - \left(\sum_{i=1}^{\bar{n}}\frac{\tilde{y}_i^{(1)}\mathbf{x}_i}{\bar{n}}\right)\left(\sum_{i=1}^{\bar{n}}\frac{\tilde{y}_i^{(2)}\mathbf{x}_i}{\bar{n}}\right)^{\top}, \qquad (11)$$

where $\tilde{y}_i^{(j)}$ ($i = 1, \ldots, \bar{n}$; $j = 1, 2$) are the synthesized labels of the i-th example in $\widetilde{S_j}$.

Algorithm 1 The summarization of CEGE.

```
Input: The weakly labeled dataset $\widetilde{S}$; the trade-off parameters $\gamma_1$ and $\gamma_2$;
Output: The optimal classifier parameter $\mathbf{w}$;
1: Construct $\widetilde{S_1}$ and $\widetilde{S_2}$, compute their centroids $\hat{\mu}(\widetilde{S_1})$, $\hat{\mu}(\widetilde{S_2})$ as well as $\tau_1$ and $\tau_2$;
2: Compute the empirical variances $\hat{\Sigma}[\widetilde{S_1}]$, $\hat{\Sigma}[\widetilde{S_2}]$ via Eq. (13), and $\hat{D}[\widetilde{S_1}, \widetilde{S_1}]$ via Eq. (11);
3: Compute the optimal $\beta$ via Eq. (14);
4: Compute $\breve{\mu}(S)$ via Eq. (8);
5: Solve Eq. (5) via gradient-based solvers;
6: return The parameters of classifier $\mathbf{w}$.
```

Proof. The proof is straightforward. According to the definition of cross-covariance matrix, we have

$$D[\mathbf{m}_1, \mathbf{m}_2] = \mathbb{E}[\mathbf{m}_1 \mathbf{m}_2^\top] - \mathbb{E}[\mathbf{m}_1] \left(\mathbb{E}[\mathbf{m}_2]\right)^\top, \tag{12}$$

of which the empirical version is directly the right-hand side of Eq. (11). □

According to Theorem 1, the empirical estimates $\hat{\Sigma}[\widetilde{S_j}]$ ($j = 1, 2$) appeared in Eq. (10) for the covariances $\Sigma[\mathbf{m}_j]$ are

$$\hat{\Sigma}[\widetilde{S_j}] = \frac{1}{\bar{n}} \sum_{i=1}^{\bar{n}} \mathbf{x}_i \mathbf{x}_i^\top - \left(\sum_{i=1}^{\bar{n}} \frac{\tilde{y}_i^{(j)} \mathbf{x}_i}{\bar{n}} \right) \left(\sum_{i=1}^{\bar{n}} \frac{\tilde{y}_i^{(j)} \mathbf{x}_i}{\bar{n}} \right)^\top. \tag{13}$$

To improve the statistical efficiency, we should minimize the trace of $\hat{\Sigma}[\breve{\mu}(S)]$ in Eq. (10), i.e., $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}(S)])$, and find a proper β that leads to the minimum variance. By computing the gradient of $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}(S)])$ to β and then setting the result to zero, we obtain the optimal β as

$$\beta = \frac{\tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}]) - \tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}])}{\tau_1^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}]) + \tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}]) - 2\tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}])}. \tag{14}$$

Note that the above expression of β has a clear interpretation. Firstly, $\tau_1^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}]) = \mathrm{tr}(\hat{\Sigma}[\breve{\mu}_1(S)])$ and $\tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}]) = \mathrm{tr}(\hat{\Sigma}[\breve{\mu}_2(S)])$ respectively evaluate the empirical variances of the estimators $\breve{\mu}_1(S)$ and $\breve{\mu}_1(S)$; and $\tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}])$ depicts the correlation between $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$. Secondly, if the variance of the estimator $\breve{\mu}_2(S) = \tau_2 \hat{\mu}(\widetilde{S_2})$ is large, which means that $\breve{\mu}_2(S)$ might be an imperfect estimator, then β would become large, and thus the estimator $\breve{\mu}_1(S) = \tau_1 \hat{\mu}(\widetilde{S_1})$ will contribute more in estimating $\hat{\mu}(S)$ than $\breve{\mu}_2(S)$. In Section 3, we will strictly prove that the resulting variance of $\breve{\mu}(S)$ under the selected β is smaller than that of any of $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$.

The entire CEGE is summarized in algorithm 1, from which we see that CEGE can be easily implemented.

3 Proof of Statistical Efficiency

A prominent advantage of our CEGE over existing methods is that CEGE takes statistical efficiency in addition to unbiasedness into consideration in estimating $\hat{\mu}(S)$, therefore the improved performance is obtained. The following theorem theoretically demonstrates this point:

Theorem 2. *Given β as defined in Eq.* (14)*, the estimator $\breve{\mu}(S) = \beta \tau_1 \hat{\mu}(\widetilde{S_1}) + (1 - \beta) \tau_2 \hat{\mu}(\widetilde{S_2})$ is more statistically efficient than $\breve{\mu}_1(S) = \tau_1 \hat{\mu}(\widetilde{S_1})$ and $\breve{\mu}_2(S) = \tau_2 \hat{\mu}(\widetilde{S_2})$.*

Proof. To study the statistical efficiency of $\breve{\mu}(S)$, we investigate the trace of its empirical covariance $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}(S)])$, which is quadratic regarding β and has the form

$$\mathrm{tr}(\hat{\Sigma}[\breve{\mu}(S)]) = \frac{1}{\bar{n}} (A_0 \beta^2 + A_1 \beta + A_2), \tag{15}$$

where $A_0 = \tau_1^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}]) + \tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}]) - 2\tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}])$, $A_1 = 2[\tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}]) - \tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])]$, and $A_2 = \tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])$. It is easy to know that the minimum value of Eq. (15) is achieved when β is computed as Eq. (14), and the minimum value $\mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)])$ is

$$\begin{aligned} &\mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)]) \\ &\quad = \frac{1}{\bar{n}} \frac{\tau_1^2 \tau_2^2 [\mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}]) \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}]) - (\mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}]))^2]}{A_0}. \end{aligned} \tag{16}$$

Since the empirical variances of $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ are evaluated by $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}_1(S)]) = \frac{1}{\bar{n}} \tau_1^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}])$ and $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}_2(S)]) = \frac{1}{\bar{n}} \tau_2^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])$, respectively, we need to prove that $\mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)])$ is smaller than $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}_j(S)])$ when $j = 1, 2$. When $j = 1$, the difference Δ_1 between $\mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)])$ and $\mathrm{tr}(\hat{\Sigma}[\breve{\mu}_1(S)])$ is

$$\begin{aligned} \Delta_1 &= \mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)]) - \mathrm{tr}(\hat{\Sigma}[\breve{\mu}_1(S)]) \\ &= -\frac{1}{\bar{n}} \frac{\left(\tau_1^2 \mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}]) - \tau_1 \tau_2 \mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}])\right)^2}{A_0}. \end{aligned} \tag{17}$$

Now we investigate the non-negativeness of the denominator A_0. By noting that $\mathrm{tr}(\hat{D}[\widetilde{S_1}, \widetilde{S_2}]) = \rho \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}])} \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])}$ where $\rho \in [-1, 1]$ is the correlation coefficient between $\mathbf{m}_1$ and $\mathbf{m}_2$, we have

$$\begin{aligned} A_0 = &\left(\tau_1 \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}])} - \tau_2 \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])} \right)^2 \\ &+ 2(1 - \rho) \tau_1 \tau_2 \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_1}])} \sqrt{\mathrm{tr}(\hat{\Sigma}[\widetilde{S_2}])} \end{aligned} \tag{18}$$

which is obvious non-negative. Therefore, we have $\Delta_1 \leq 0$ in Eq. (17). Similarly, we get $\Delta_2 = \mathrm{tr}_{\min}(\hat{\Sigma}[\breve{\mu}(S)]) - \mathrm{tr}(\hat{\Sigma}[\breve{\mu}_2(S)]) \leq 0$. Therefore, this theorem is proved. □

Theorem 2 informs us that by constructing two auxiliary sets and linearly combining the induced estimators $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$, our proposed estimator $\breve{\mu}(S)$ reveals smaller variance than any of $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$. This fact will also be empirically verified by the experiments in Section 6.5.1.

4 Applications of CEGE

In this section, we show that our proposed CEGE framework accommodates to a wide range of typical WSL problems including SSL, PUL, MIL, and LNL. Among them, SSL and PUL are learning problems with incomplete supervision, MIL is a learning problem with inexact supervision, and LNL represents the learning problem with inaccurate supervision.

4.1 Semi-Supervised Learning

SSL aims to train a suitable classifier based on very limited labeled examples and a large number of unlabeled examples. Mathematically, suppose that the entire training set $\widetilde{S} = S_L \cup S_U$ where $S_L = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^{l}$ is the labeled set consisted of l training data with known $\tilde{y}_i$ (i.e., $\tilde{y}_i = y_i$), and $S_U = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^{u}$ is the unlabeled set containing u examples with unknown y_i (i.e., y_i is corrupted to $\tilde{y}_i$ to form weak supervision). Moreover, assume that S_L contains l_p positive data and l_n negative data. Therefore, we have $\bar{n} = l + u$ and $l = l_p + l_n$. According to the explanations in Section 2, we need to establish two auxiliary sets $\widetilde{S_1}$ and $\widetilde{S_2}$ to get the primitive estimators $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$.

Construction of $\widetilde{S_1}$: To construct the auxiliary set $\widetilde{S_1}$, we regard S_U as a noisy negative set $\widetilde{S_N}$ by treating all the unlabeled examples in S_U as negative, namely $\tilde{y}_i^{(1)} = -1$ for $(\mathbf{x}_i, \tilde{y}_i^{(1)}) \in \widetilde{S_N}$. Therefore, we obtain $\widetilde{S_1} = S_L \cup \widetilde{S_N}$, and the corresponding label flipping probabilities from S to $\widetilde{S_1}$ are $\eta_P^{(1)} \in (0,1)$ and $\eta_N^{(1)} = 0$. Consequently, the positive class prior in the noisy distribution $\widetilde{\mathcal{D}_1}$ is $P(\tilde{y}^{(1)} = +1) = P(\tilde{y}^{(1)} = +1|y = +1)P(y = +1) + P(\tilde{y}^{(1)} = +1|y = -1)P(y = -1) = (1 - \eta_P^{(1)})\pi$ where $\pi = P(y = +1)$ has been defined in Section 2.3. Therefore, we have $\eta_P^{(1)} = \frac{\pi - P(\tilde{y}^{(1)}=+1)}{\pi}$ where $P(\tilde{y}^{(1)} = +1)$ can be estimated by $l_p/\bar{n}$. Moreover, we may have

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y)]\\
&= \pi\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=+1)] + (1-\pi)\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=-1)]\\
&= \pi(1 - 2\eta_P^{(1)})y\mathbf{x} + (1-\pi)y\mathbf{x}\\
&= (1 - 2\pi\eta_P^{(1)})y\mathbf{x}, \qquad (19)
\end{aligned}$$

which suggests that the unbiased estimate of $\hat{\mu}(S)$ based on $\widetilde{S_1}$ is $\breve{\mu}_1(S) = \frac{1}{1-2\pi\eta_P^{(1)}}\hat{\mu}(\widetilde{S_1})$. That is, $\tau_1 = \frac{1}{1-2\pi\eta_P^{(1)}}$ for SSL.

Construction of $\widetilde{S_2}$: To construct $\widetilde{S_2}$, we regard S_U as a noisy positive set $\widetilde{S_P}$ by treating all the unlabeled examples in S_U as positive, namely $\tilde{y}_i^{(2)} = +1$ for $(\mathbf{x}_i, \tilde{y}_i^{(2)}) \in \widetilde{S_P}$. That is to say, $\widetilde{S_2} = S_L \cup \widetilde{S_P}$, for which the label flipping probabilities are $\eta_P^{(2)} = 0$ and $\eta_N^{(2)} \in (0,1)$. Similarly as the above, we have $\eta_N^{(2)} = \frac{P(\tilde{y}^{(2)}=+1)-\pi}{1-\pi}$. Consequently, it can be derived that

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y)]\\
&= \pi\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=+1)] + (1-\pi)\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=-1)]\\
&= \pi y\mathbf{x} + (1-\pi)(1 - 2\eta_N^{(2)})y\mathbf{x}\\
&= [1 - 2(1-\pi)\eta_N^{(2)}]y\mathbf{x}. \qquad (20)
\end{aligned}$$

As a result, the unbiased estimate of $\hat{\mu}(S)$ from $\widetilde{S_2}$ is $\breve{\mu}_2(S) = \frac{1}{1-2(1-\pi)\eta_N^{(2)}}\hat{\mu}(\widetilde{S_2})$, namely $\tau_2 = \frac{1}{1-2(1-\pi)\eta_N^{(2)}}$.

Based on the obtained τ_1 and τ_2, we may compute the optimal β via Eq. (14), so the $\breve{\mu}(S)$ for SSL can be further calculated via Eq. (8). Therefore, according to Eq. (5), the model for SSL is expressed as

$$\begin{aligned}
\min_{\mathbf{w}}\ &\frac{1}{\bar{n}}\sum_{(\mathbf{x}_i,\tilde{y}_i)\in\widetilde{S}}\varphi(h(\mathbf{x}_i)) + Q\langle\mathbf{w},\breve{\mu}(S)\rangle\\
&+ \frac{\gamma_1}{l}\sum_{(\mathbf{x}_i,y_i)\in S_L}\ell(h(\mathbf{x}_i),y_i) + \gamma_2\left\|\mathbf{w}\right\|^2, \qquad (21)
\end{aligned}$$

where the formations of $\varphi(h(\mathbf{x}_i))$, Q and $\ell(h(\mathbf{x}_i), y_i)$ are governed by the selected loss function as illustrated in Table 1. Eq. (21) is an unconstrained optimization problem and can be easily solved via some off-the-shelf gradient-based solvers.

4.2 Positive-Unlabeled Learning

PUL has attracted intensive research interests in recent years due to the increasing demands in many practical problems. It aims to find a good binary classifier by training on a dataset that only contains positive data and unlabeled data. Consequently, PUL is quite useful when the negative training examples are not directly accessible. Formally, we have a weakly-labeled set $\widetilde{S} = S_P \cup S_U$ where $S_P = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^p$ denotes the positive set consisted of p positive training data (i.e., $\tilde{y}_i = y_i = +1$), and $S_U = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^u$ is the unlabeled set containing u examples with unspecified y_i (i.e., y_i degenerates to $\tilde{y}_i$). Here y_i for $(\mathbf{x}_i, \tilde{y}_i) \in S_U$ can be positive or negative, but its ground-truth value is not observed by the algorithm during training. Besides, the total number of examples is $\bar{n} = p + u$.

Construction of $\widetilde{S_1}$: Similarly to SSL, to construct the auxiliary set $\widetilde{S_1}$, we also regard S_U as a noisy negative set $\widetilde{S_N}$ by regarding all the unlabeled data in S_U as negative, i.e., $\tilde{y}_i^{(1)} = -1$ for $(\mathbf{x}_i, \tilde{y}_i^{(1)}) \in \widetilde{S_N}$. Therefore, $\widetilde{S_1}$ is formed as $\widetilde{S_1} = S_P \cup \widetilde{S_N}$, and thus the label flipping probabilities are $\eta_P^{(1)} \in (0,1)$ and $\eta_N^{(1)} = 0$. Consequently, we have $P(\tilde{y}^{(1)} = +1) = P(\tilde{y}^{(1)} = +1|y = +1)P(y = +1) + P(\tilde{y}^{(1)} = +1|y = -1)P(y = -1) = (1 - \eta_P^{(1)})\pi$. Therefore, as the derivations in Eq. (19) for SSL, we get the unbiased estimate of $\hat{\mu}(S)$ based on $\widetilde{S_1}$ as $\breve{\mu}_1(S) = \frac{1}{1-2\pi\eta_P^{(1)}}\hat{\mu}(\widetilde{S_1})$, which means that $\tau_1 = \frac{1}{1-2\pi\eta_P^{(1)}}$ for PUL.

Construction of $\widetilde{S_2}$: We treat S_U as a noisy positive set $\widetilde{S_P}$ by regarding all the examples in S_U as positive, namely $\tilde{y}_i^{(2)} = +1$ for $(\mathbf{x}_i, \tilde{y}_i^{(2)}) \in \widetilde{S_P}$. As a consequence, $\widetilde{S_2}$ is established as $\widetilde{S_2} = S_P \cup \widetilde{S_P}$, and the resulting label flipping probabilities are $\eta_P^{(2)} = 0$ and $\eta_N^{(2)} = 1$. This is because that as no positive examples are manually assigned synthesized negative labels, while all original negative data are labeled as positive in $\widetilde{S_2}$. Consequently, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y)]\\
&= \pi\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=+1)] + (1-\pi)\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=-1)]\\
&= \pi y\mathbf{x} + (1-\pi)(-y\mathbf{x})\\
&= (2\pi - 1)y\mathbf{x}. \qquad (22)
\end{aligned}$$

As a result, $\breve{\mu}_2(S) = \frac{1}{2\pi-1}\hat{\mu}(\widetilde{S_2})$, namely $\tau_2 = \frac{1}{2\pi-1}$ for PUL[3].

Therefore, by computing $\breve{\mu}(S)$ via Eq. (8) and plugging it to Eq. (5), the model for PUL is expressed as

$$\begin{aligned}
\min_{\mathbf{w}}\ &\frac{1}{\bar{n}}\sum_{(\mathbf{x}_i,\tilde{y}_i)\in\widetilde{S}}\varphi(h(\mathbf{x}_i)) + Q\langle\mathbf{w},\breve{\mu}(S)\rangle\\
&+ \frac{\gamma_1}{p}\sum_{(\mathbf{x}_i,y_i)\in S_P}\ell(h(\mathbf{x}_i),y_i) + \gamma_2\left\|\mathbf{w}\right\|^2, \qquad (23)
\end{aligned}$$

where $\varphi(h(\mathbf{x}_i))$, Q and $\ell(h(\mathbf{x}_i), y_i)$ can be chosen from Table 1.

4.3 Multiple Instance Learning

MIL is an important branch of WSL where the entire training set is composed of many bags, and a label is provided for each of the bags. A negative bag means that all its contained examples are negative, and a positive bag means that there is at least one positive example inside it. Formally, we are given a set of positive bags $S_P = \{B_1^+, \ldots, B_{p_+}^+\}$ with p_+ positive bags $\{B_i^+\}_{i=1}^{p_+}$, and a set of negative bags $S_N = \{B_1^-, \ldots, B_{n_-}^-\}$ with n_- negative

[3] If $\pi = 0.5$, we may write τ_2 as $\tau_2 = \frac{1}{2\pi-1+\epsilon}$ where ϵ is a small positive number.

bags $\{B_i^-\}_{i=1}^{n_-}$. The label of any $(\mathbf{x}_i, \tilde{y}_i) \in B_i^-$ is $\tilde{y}_i = y_i = -1$, while the $\tilde{y}_i$ for $(\mathbf{x}_i, \tilde{y}_i) \in B_i^+$ is unclear but at least one $\tilde{y}_i$ in B_i^+ is $+1$. Moreover, assume that S_P and S_N contains p and n examples correspondingly, then we have $\bar{n} = p + n$, and the target of MIL is to train a bag classifier based on $S = S_P \cup S_N$ such that it can determine the label of an unseen test bag.

Construction of $\widetilde{S_1}$: Since only a handful of examples in S_P are positive, we may firstly regard S_P as a noisy negative set $\widetilde{S_N}$ by assuming that all the examples in S_P as negative, i.e., the synthesized label $\tilde{y}_i^{(1)} = -1$ for $(\mathbf{x}_i, \tilde{y}_i^{(1)}) \in \widetilde{S_N}$. Therefore, $\widetilde{S_1}$ is established as $\widetilde{S_1} = S_N \cup \widetilde{S_N}$, where the labels of examples in S_N are definitely negative. Consequently, the label flipping probabilities are $\eta_P^{(1)} = 1$ and $\eta_N^{(1)} = 0$, as no examples are deemed as positive in $\widetilde{S_1}$. Consequently, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y)] \\
&=\pi\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=+1)]+(1-\pi)\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=-1)] \\
&= \pi(-y\mathbf{x}) + (1-\pi)y\mathbf{x} \\
&= (1-2\pi)y\mathbf{x}. \qquad (24)
\end{aligned}$$

As a result, we get the unbiased estimate of $\hat{\mu}(S)$ relying on $\widetilde{S_1}$ as $\breve{\mu}_1(S) = \frac{1}{1-2\pi}\hat{\mu}(\widetilde{S_1})$, and thus $\tau_1 = \frac{1}{1-2\pi}$ for MIL.

Construction of $\widetilde{S_2}$: in contrast to building $\widetilde{S_1}$, we then treat S_P as a noisy positive set $\widetilde{S_P}$ by labeling all the examples in S_P as positive, namely $\tilde{y}_i^{(2)} = +1$ for $(\mathbf{x}_i, \tilde{y}_i^{(2)}) \in \widetilde{S_P}$. As a consequence, we have $\widetilde{S_2} = S_N \cup \widetilde{S_P}$, and the induced label flipping probabilities are $\eta_P^{(2)} = 0$ and $\eta_N^{(2)} \in (0,1)$. Therefore, according to the equation $P(\tilde{y}^{(2)} = +1) = P(\tilde{y}^{(2)} = +1|y = +1)P(y = +1) + P(\tilde{y}^{(2)} = +1|y = -1)P(y = -1) = (1 - \eta_N^{(2)})\pi + \eta_N^{(2)}$, we have $\eta_N^{(2)} = \frac{P(\tilde{y}^{(2)}=+1)-\pi}{1-\pi}$ where $P(\tilde{y}^{(2)} = +1)$ can be estimated by $p/\bar{n}$. Consequently, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y)] \\
&=\pi\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=+1)]+(1-\pi)\mathbb{E}_{\tilde{y}^{(2)}}[\tilde{y}^{(2)}\mathbf{x}|(\mathbf{x},y=-1)] \\
&= \pi y\mathbf{x} + (1-\pi)(1-2\eta_N^{(2)})y\mathbf{x} \\
&= [1-2\eta_N^{(2)}(1-\pi)]y\mathbf{x}. \qquad (25)
\end{aligned}$$

Therefore, the unbiased estimate of $\hat{\mu}(S)$ relying on $\widetilde{S_2}$ is $\breve{\mu}_2(S) = \frac{1}{1-2(1-\pi)\eta_N^{(2)}}\hat{\mu}(\widetilde{S_2})$, which suggests that $\tau_2 = \frac{1}{1-2(1-\pi)\eta_N^{(2)}}$ for MIL.

After combining $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ to form $\breve{\mu}(S)$ based on Eq. (8), and also introducing the bag constraints proposed in [7], we achieve the following model for MIL:

$$\begin{aligned}
\min_{\mathbf{w},\{y_i\}_{i=1}^{\bar{n}}} \ & \frac{1}{\bar{n}} \sum_{(\mathbf{x}_i,\tilde{y}_i)\in\widetilde{S}} \varphi(h(\mathbf{x}_i)) + Q\langle\mathbf{w},\breve{\mu}(S)\rangle \\
& + \frac{\gamma_1}{n} \sum_{(\mathbf{x}_i,\tilde{y}_i)\in S_N} \ell(h(\mathbf{x}_i),y_i) + \gamma_2\left\|\mathbf{w}\right\|^2 \\
\text{s.t.} \ & \sum_{(\mathbf{x}_i,\tilde{y}_i)\in B_i^+} \frac{y_i+1}{2} \geq 1, \quad \forall\, B_i^+ \ (i=1,\ldots,p_+); \\
& y_i = -1, \quad \forall\, B_i^- \ (i=1,\ldots,p_-). \qquad (26)
\end{aligned}$$

Note that the above model for MIL contains binary optimization variables $\{y_i\}_{i=1}^{\bar{n}}$ and the classifier parameter $\mathbf{w}$, so we deploy the alternating strategy between the subproblems of $\mathbf{w}$ and $\{y_i\}_{i=1}^{\bar{n}}$ as suggested by [7]. In the subproblem of $\mathbf{w}$, $\{y_i\}_{i=1}^{\bar{n}}$ is deemed as constants, so $\mathbf{w}$ can be easily updated via some gradient-based methods. In the subproblem of $\{y_i\}_{i=1}^{\bar{n}}$, $\mathbf{w}$ is treated as known, and $\{y_i\}_{i=1}^{\bar{n}}$ are updated according to the output of current classifier as well as the constraints. To be specific, for every positive bag, if all data points inside it are classified as negative which violates the constraint, we manually set the label of the example that has the maximum classifier output to $+1$. The above two subproblems iterate until convergence.

4.4 Label Noise Learning

In LNL, the training data might be mistakenly labeled due to various human issues such as fatigue and limited knowledge, so LNL aims to generate a robust classifier which can resist the adverse impact of the noisy labels. Specifically, we have a noisy training set $\widetilde{S} = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^{\bar{n}}$ in which some of the labels $\{\tilde{y}_i\}_{i=1}^{\bar{n}}$ are incorrect, and hope that a score function h can be acquired on $\widetilde{S}$ of which the performance is close to the one that is trained on clean set S with correct $\{y_i\}_{i=1}^{\bar{n}}$.

Construction of $\widetilde{S_1}$: As the entire dataset $\widetilde{S}$ itself is weakly-labeled in LNL, we may directly let $\widetilde{S_1} = \widetilde{S}$ first. According to the routine setting of LNL [12], [26], here we also assume that the label flipping probabilities $\eta_P^{(1)}$ and $\eta_N^{(1)}$ are known[4]. Practically, they can be acquired via domain knowledge or cross-validation. Therefore, here we should estimate the class prior $\pi = P(y = +1)$ based on $\eta_P^{(1)}$ and $\eta_N^{(1)}$, which slightly differs from the determinative relationship between the class prior π and label flipping probabilities in SSL, PUL and MIL. Since $P(\tilde{y}^{(1)} = +1) = P(\tilde{y}^{(1)} = +1|y = +1)P(y = +1) + P(\tilde{y}^{(1)} = +1|y = -1)P(y = -1) = \eta_N^{(1)} + (1 - \eta_P^{(1)} - \eta_N^{(1)})\pi$, we immediately get $\pi = \frac{P(\tilde{y}^{(1)}=+1)-\eta_N^{(1)}}{1-\eta_P^{(1)}-\eta_N^{(1)}}$ where $P(\tilde{y}^{(1)} = +1)$ is estimated by $p/\bar{n}$. Therefore,

$$\begin{aligned}
&\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y)] \\
&=\pi\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=+1)]+(1-\pi)\mathbb{E}_{\tilde{y}^{(1)}}[\tilde{y}^{(1)}\mathbf{x}|(\mathbf{x},y=-1)] \\
&= \pi(1-2\eta_P^{(1)})y\mathbf{x} + (1-\pi)(1-2\eta_N^{(1)})y\mathbf{x} \\
&= [1-2\eta_N^{(1)} - 2\pi(\eta_P^{(1)} - \eta_N^{(1)})]y\mathbf{x}, \qquad (27)
\end{aligned}$$

which indicates that the unbiased estimate of $\hat{\mu}(S)$ from the perspective of $\widetilde{S_1}$ is $\breve{\mu}_1(S) = \frac{1}{1-2\eta_N^{(1)}-2\pi(\eta_P^{(1)}-\eta_N^{(1)})}\hat{\mu}(\widetilde{S_1})$. As a consequence, $\tau_1 = \frac{1}{1-2\eta_N^{(1)}-2\pi(\eta_P^{(1)}-\eta_N^{(1)})}$ for LNL.

Construction of $\widetilde{S_2}$: To construct $\widetilde{S_2}$, we assign the synthesized label $+1$ to all the examples in $\widetilde{S}$ and obtain a noisy positive set $\widetilde{S_2} = \widetilde{S_P}$, namely $\tilde{y}_i^{(2)} = +1$ for $(\mathbf{x}_i, \tilde{y}_i^{(2)}) \in \widetilde{S_P}$. This situation is the same as the construction of $\widetilde{S_2}$ in PUL, so we know $\breve{\mu}_2(S) = \frac{1}{2\pi-1}\hat{\mu}(\widetilde{S_2})$ and $\tau_2 = \frac{1}{2\pi-1}$ for LNL.

Based on τ_1 and τ_2, the optimal β can be computed via Eq. (14), which facilitates the calculation of $\breve{\mu}(S)$ via Eq. (8). Since LNL problem does not have an S_{strong} term with definite labels, the model for LNL is formulated as

$$\min_{\mathbf{w}} \frac{1}{\bar{n}} \sum_{(\mathbf{x}_i,\tilde{y}_i)\in\widetilde{S}} \varphi(h(\mathbf{x}_i)) + Q\langle\mathbf{w},\breve{\mu}(S)\rangle + \gamma_2\left\|\mathbf{w}\right\|^2. \qquad (28)$$

4.5 Summary of WSL Problems

From the explanations from Sections 4.1∼4.4, we learn that the models of many existing WSL problems can be written under

[4] As the class prior π is determined by $\eta_P^{(1)}$ and $\eta_N^{(1)}$ as will be later shown, we can also say that π in LNL is available in advance.

TABLE 3: Summary of various WSL tasks under CEGE framework.

WSL tasks	$\widetilde{S_1}$	$\widetilde{S_2}$	τ_1	τ_2
SSL	$S_L \cup \widetilde{S_N}$	$S_L \cup \widetilde{S_P}$	$\frac{1}{1-2\pi\eta_P^{(1)}}$	$\frac{1}{1-2(1-\pi)\eta_N^{(2)}}$
PUL	$S_P \cup \widetilde{S_N}$	$S_P \cup \widetilde{S_P}$	$\frac{1}{1-2\pi\eta_P^{(1)}}$	$\frac{1}{2\pi-1}$
MIL	$S_N \cup \widetilde{S_N}$	$S_N \cup \widetilde{S_P}$	$\frac{1}{1-2\pi}$	$\frac{1}{1-2(1-\pi)\eta_N^{(2)}}$
LNL	$\widetilde{S}$	$\widetilde{S_P}$	$\frac{1}{1-2\eta_N^{(1)}-2\pi(\eta_P^{(1)}-\eta_N^{(1)})}$	$\frac{1}{2\pi-1}$

the framework of CEGE, and their main differences lie in the constructions of auxiliary sets $\widetilde{S_1}$ and $\widetilde{S_2}$, and the computations of τ_1 and τ_2. The detailed formations of $\widetilde{S_1}$, $\widetilde{S_2}$, τ_1 and τ_2 are summarized in Table 3, which reveals that our CEGE is applicable to a variety of common WSL tasks.

5 Related Works

WSL is a very broad topic in machine learning which contains many learning paradigms such as semi-supervised learning [2], active learning [5], positive-unlabeled learning [21], partial label learning [8], [9], complementary label learning [10], positive-confidence learning [11], unlabeled-unlabeled learning [4], label noise learning [12], multiple instance learning [29], transfer learning [6], etc. Therefore, it is difficult to thoroughly review all related works on WSL in this paper. Since our paper tackles SSL, PUL, MIL and LNL under the framework of CEGE, here we simply review some related works on these topics.

SSL tries to establish an accurate classifier by harnessing the supervision information carried by the limited labeled data as well as the distribution information revealed by the massive unlabeled data. The existing SSL approaches are usually graph based, large-margin-theory based, and consistency regularization based. Graph-based methods usually build a graph to relate the training data, and hypothesize that the nearby data points in the graph will receive similar labels. The works [30], [31], [32], [33], [34], [35], [36], [37], [38] explicitly design various graph Laplacians, graph convolutions, or propagation sequences to achieve this purpose. Large-margin-theory based approaches assume that the examples in different classes form different clusters in the feature space, and they are usually built on SVM so that a discriminative classifier can be learned. The typical algorithms include [39], [40], [41], [42]. Consistency regularization based methods usually involve data augmentation or perturbation by leveraging the idea that a classifier should output the same class distribution for unlabeled examples even they are slightly changed. For example, [43], [44], [45] add disturbances to data and use temporal ensembling or consistency term to obtain stable label output. MixMatch [46] augment the dataset via linear interpolation between pairwise data which shows good performance.

PUL aims to find a suitable classifier simply based on positive and unlabeled data, among which the labels of the unlabeled data can be positive or negative but are unknown to the learning algorithm. The early-staged methods [3], [47] propose to extract some reliable negative examples from the unlabeled set, and then use these confident negative examples and the original positive examples to train a classifier. After that, some works formulate PUL as a cost-sensitive learning problem by imposing different weights on positive data and unlabeled data, such as [21], [22], [48], [49]. The risk estimators introduced by these methods can be unbiased or biased. Besides, there are also some works [18], [50], [51], [52] that treat the unlabeled data as negative ones, and then converting PUL problem into a one-side label noise learning problem. Other typical PUL approaches include [20], [53], [54] which exploit data distribution to reduce the adverse impact caused by the absence of negative examples.

MIL targets to train a bag classifier where a bag contains a set of training examples and a label is provided for the entire bag rather than individual example. Existing MIL models usually work on instance level or bag level. The methods on instance level [7], [55], [56] try to predict the label of every data point, and then the label of a bag can be decided. Due to the development of deep neural networks, many algorithms belonging to this category have achieved considerable performance gain, such as [57], [58]. The bag-level approaches treat the bag as a whole and directly find its label based on the representations or similarities of bags. For example, [59], [60] try to describe the closeness of examples via bag kernel or distance metrics, while [61], [62], [63] investigate precise bag representations to facilitate the subsequent bag classification.

LNL targets to build a robust classifier by training on the examples with possible incorrect annotations. The most natural choice for tackling noisy labels is arguably label correction, namely detecting the possible incorrect labels and then eliminating or fixing them. For example, [64], [65] remove the possible noise ahead of conducting the standard classification algorithms, while [66], [67], [68], [69] rely on the "small-loss" effect of neural network or "class prototypes" to conduct joint noise reduction and network training. Apart from label correction, some other algorithms spend efforts on loss correction, in which the traditional loss functions for fully supervised learning are repaired to combat with label noise. Under the framework of empirical risk minimization, a series of works [12], [16], [26], [70], [71], [72] have been done to design various surrogate loss functions based on label transition probability, so that the surrogate loss for noisy data is the same as the risk under the original loss for noise-free data. Lastly, there are also some works [17], [73], [74] targeting to improve the model robustness by modifying the optimization process.

6 Experiments

In this section, we compare our CEGE framework with existing representative methods in SSL, PUL, MIL and LNL on extensive benchmark datasets (Sections 6.1$\sim$6.4). The two WSL frameworks WELLSVM [14] and SAFEW [15] are also compared on different WSL problems if they are applicable. Moreover, we empirically study the important behaviors of CEGE such as the variance reduction in centroid estimation, and the sensitivity regarding the inaccurate input class prior π (Section 6.5).

For all the experiments below, we use hinge loss for CEGE, and actually there is no significant performance variation between different choices of loss functions. For the baseline methods, we use their linear models for direct comparison, as some of them achieve non-linearity via kernel or neural network, which makes them not directly comparable. The features for all methods have been normalized and standardized on every dataset. Ten-fold cross validation is applied to all compared approaches on all datasets, and the mean test accuracy as well as standard deviation of the ten independent trials on each dataset are reported for algorithm evaluation. Besides, the paired t-test with significance level 0.05 is employed to statistically compare the results of various methods.

TABLE 4: Comparison of the mean test accuracies of various approaches on SSL task. The best two records on each dataset are highlighted in red and blue, respectively. The "√" ("×") denotes that our CEGE is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

Dataset	$(\bar{n}, d)$	ζ	LapRLS [32]	LapSVM [32]	LPDGL [34]	PNU [23]	WELLSVM [14]	SAFEW [15]	CEGE
BreastCancer	(683,9)	5%	0.947±0.028	0.925±0.045 √	0.922±0.055 √	0.927±0.060 √	0.933±0.032	0.963±0.021	0.966±0.020
		10%	0.962±0.029	0.955±0.034	0.956±0.023	0.944±0.028 √	0.949±0.026	0.967±0.018	0.969±0.021
Phoneme	(5404,6)	5%	0.734±0.034 √	0.718±0.033 √	0.719±0.036 √	0.737±0.026 √	0.690±0.030 √	0.721±0.038 √	0.767±0.027
		10%	0.729±0.017 √	0.718±0.014 √	0.718±0.022 √	0.753±0.034 √	0.699±0.020 √	0.694±0.037 √	0.770±0.023
Spambase	(4601,57)	5%	0.874±0.017 √	0.812±0.068 √	0.875±0.019 √	0.886±0.025 √	0.863±0.027 √	0.769±0.027 √	0.900±0.019
		10%	0.891±0.009 √	0.864±0.031 √	0.895±0.008 √	0.877±0.016 √	0.884±0.009 √	0.783±0.033 √	0.910±0.007
GermanCredit	(1000,24)	5%	0.659±0.055 √	0.629±0.012 √	0.650±0.051 √	0.659±0.063 √	0.649±0.069 √	0.543±0.100 √	0.727±0.029
		10%	0.665±0.046 √	0.653±0.051 √	0.674±0.042 √	0.701±0.052 √	0.659±0.041 √	0.624±0.051 √	0.738±0.051
PhishingWebsites	(2456,30)	5%	0.990±0.007	0.961±0.010 √	0.971±0.021 √	0.991±0.006	0.913±0.036 √	0.995±0.005	0.992±0.008
		10%	0.988±0.004 √	0.960±0.012 √	0.990±0.007	0.991±0.007	0.967±0.017 √	0.996±0.006	0.992±0.006
Magic	(19020,18)	5%	0.785±0.016	0.783±0.013	0.780±0.015 √	0.777±0.013 √	0.774±0.012 √	0.700±0.025 √	0.787±0.014
		10%	0.786±0.007 √	0.784±0.007 √	0.782±0.008 √	0.780±0.005 √	0.741±0.049 √	0.681±0.014 √	0.792±0.006
Average			0.834	0.814	0.828	0.835	0.810	0.786	0.859

6.1 Experiments on Semi-Supervised Learning

We first compare our CEGE with several representative SSL approaches including Laplacian Regularized Least Squares (LapRLS) [32], Laplacian SVM (LapSVM) [32], Label Propagation via Deformed Graph Laplacian (LPDGL) [34], and Positive-Negative-Unlabeled classification (PNU) [23]. Besides, since both WELLSVM and SAFEW are applicable to SSL, they are also compared here. Following [23], we also adopt the *BreastCancer*, *Phoneme*, *Spambase*, *GermanCredit*, *PhishingWebsites* and *Magic* datasets from UCI machine learning repository [75] for our experiments, of which the amount of examples ranges from 683∼19020 and the data dimensionality is within $[6, 57]$. For each dataset, we investigate the performances of various methods when 5% and 10% training examples are labeled (i.e., labeling rate $\zeta = l/\bar{n} \in \{5\%, 10\%\}$), and the split of labeled set, unlabeled set and test set is kept identical for all comparators on every dataset.

In CEGE, the trade-off parameters γ_1 and γ_2 in Eq. (5) are selected from $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, , 10^3\}$, and the values are $\{\gamma_1, \gamma_2\} = \{10^2, 10\}, \{10, 10^{-1}\}, \{10, 10\}, \{10, 10\}, \{10, 1\}, \{10, 10^{-3}\}$ on *BreastCancer*, *Phoneme*, *Spambase*, *GermanCredit*, *PhishingWebsites* and *Magic*, respectively. For the graph-based methods such as LapRLS, LapSVM and LPDGL, we build a K-NN graph with $K = 10$ on these datasets as a sparse graph usually leads to good performance. Besides, the trade-off parameters γ_A and γ_I in LapRLS and LapSVM, β and γ in LPDGL, and C_1 and C_2 in WELLSVM are also carefully tuned via searching the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3\}$. In PNU, the trade-off parameter γ is chosen from $[-1, 1]$ with interval 0.1 as suggested by the authors. For SAFEW, we use the self-KNN classifiers [76] with Euclidean and Cosine distances as the base learners as indicated in [15].

The classification accuracies of all compared methods under $\zeta = 5\%$ and $\zeta = 10\%$ are reported in Table 4, from which we see that CEGE achieves the highest test accuracy among the compared methods on all datasets except on *PhishingWebsites*. On *PhishingWebsites*, CEGE actually achieves comparable outputs with SAFEW as revealed by t-test. Regarding the average accuracy on the adopted six dataset, CEGE touches the highest record, which leads the second best method PNU by a margin of 2.4%. Therefore, the effectiveness of CEGE in dealing with SSL is demonstrated. Comparatively, some conventional methods such as LapRLS, LapSVM and LPDGL perform unsatisfactorily as they assume that the similar examples will receive similar label assignments, which might not hold in some complicated datasets.

6.2 Experiments on Positive-Unlabeled Learning

This part shows the effectiveness of CEGE in conducting PUL. To this end, we follow [18] and use *BreastCancer*, *Australian*, *HockeyFight*, *NBA*, *Banknote*, and *Mushroom* datasets for algorithm comparison. These datasets cover a variety of domains such as biology, banking, sports, medical science, etc, and their detailed configurations can be found in Table 5. For each dataset, we study two situations with different numbers of originally positive data in the unlabeled set. That is to say, the unlabeled set is composed of the original negative set as well as 20% or 40% (i.e., $\eta_P = \{0.2, 0.4\}$) of the positive training examples that are randomly selected.

The compared baselines include several state-of-the-art PU methods such as Weighted SVM (WSVM) [48], unbiased PU (uPU) model [21], non-negative PU risk estimator (nnPU) [22], Loss Decomposition and Centroid Estimation (LDCE) [18], and Large-margin Label-calibrated SVM (LLSVM) [20]. Note that these methods design various regularizers to achieve the proper training with PU data, so the comparison with them is fair. In CEGE, the parameters γ_1 and γ_2 are respectively set to $\{\gamma_1, \gamma_2\} = \{1, 10^{-2}\}, \{1, 1\}, \{1, 1\}, \{10^{-1}, 10^{-1}\}, \{1, 10^{-1}\}, \{10, 10^{-1}\}$ on *BreastCancer*, *Australian*, *HockeyFight*, *NBA*, *Banknote*, and *Mushroom* datasets. For nnPU, β is set to the recommended value 0. The tuning parameter λ in LDCE, and the weights α, β, γ in LLSVM, are also selected by searching the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3\}$. Among the baseline methods, uPU, nnPU, LDCE and LLSVM also require the class prior π as our CEGE for algorithm implementation, so here we simply send real π to them for generating the results.

The performances of various methodologies are presented in Table 5, which reveals that CEGE obtains top two performance on most of the datasets. CEGE performs comparably with nnPU, and slightly worse than WSVM and uPU on *Mushroom* dataset, but the performance gap to the leading method is within 1%. Overall, the average accuracy of CEGE is the highest among the involved methods, which leads the second best method LDCE with a margin of 0.9%. As mentioned above, LDCE also estimates the dataset

TABLE 5: Comparison of the mean test accuracies of various approaches on PUL task. The best two records on each dataset are highlighted in red and blue, respectively. The "√" ("×") denotes that our CEGE is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

Dataset	$(\bar{n}, d)$	η_P	WSVM [48]	uPU [21]	nnPU [22]	LDCE [18]	LLSVM [20]	CEGE
BreastCancer	(683,9)	0.2	0.966±0.022	0.909±0.022 √	0.955±0.022 √	0.952±0.027	0.959±0.021 √	0.969±0.016
		0.4	0.967±0.019	0.937±0.029 √	0.972±0.018	0.966±0.034	0.941±0.033 √	0.974±0.024
Australian	(690,14)	0.2	0.861±0.055	0.813±0.036 √	0.812±0.042 √	0.858±0.035 √	0.861±0.040	0.874±0.037
		0.4	0.859±0.048	0.855±0.036	0.840±0.034	0.842±0.028 √	0.855±0.035	0.865±0.034
HockeyFight	(1000,100)	0.2	0.857±0.027 √	0.817±0.032 √	0.875±0.020 √	0.882±0.036	0.881±0.022 √	0.904±0.030
		0.4	0.833±0.019 √	0.851±0.045 √	0.873±0.044 √	0.886±0.027	0.883±0.037	0.889±0.025
NBA	(1340,19)	0.2	0.676±0.018 √	0.696±0.030	0.616±0.029 √	0.694±0.033	0.655±0.048 √	0.698±0.028
		0.4	0.664±0.025 √	0.693±0.023	0.629±0.027 √	0.687±0.050	0.641±0.071 √	0.693±0.047
Banknote	(1372,4)	0.2	0.966±0.016 √	0.950±0.024 √	0.953±0.015 √	0.988±0.006	0.937±0.023 √	0.983±0.013
		0.4	0.956±0.016 √	0.967±0.034	0.962±0.016 √	0.985±0.014	0.894±0.034 √	0.985±0.014
Mushroom	(8124,112)	0.2	0.998±0.004 ×	0.998±0.002 ×	0.991±0.001	0.983±0.004 √	0.897±0.014 √	0.989±0.003
		0.4	0.996±0.002 ×	0.998±0.002 ×	0.993±0.005	0.984±0.005 √	0.892±0.010 √	0.990±0.006
Average			0.883	0.874	0.873	0.892	0.858	0.901

centroid to design an unbiased risk estimator to fully supervised case, which is similar to this work. However, it only utilizes one auxiliary set and thus cannot obtain statistically efficient or reliable estimate of the centroid. Therefore, it is no better than CEGE as revealed by the average test accuracy. For other baselines such as WSVM, uPU and nnPU, they all follow a cost-sensitive learning framework by allocating different weights to labeled data and unlabeled data, in which the weight calculation is based on some heuristic assumptions and might be imprecise, so they yield lower average accuracy than CEGE.

6.3 Experiments on Multiple Instance Learning

To study the performance of CEGE on MIL, we employ the widely-used MIL benchmark datasets[5] including *MUSK1*, *MUSK2*, *Fox*, *Tiger*, and *Elephant* for our experiments, where the first two datasets are regarding drug activity prediction, and the last three focus on content-based image retrieval. The compared baselines are miSVM [7], MISVM [7], MissSVM [77], EM-DD [55], WELLSVM [14], and SAFEW [15]. Here ten-fold cross validation is conducted on bag level, and we examine the mean bag test accuracies of ten trials generated by the compared methods.

The trade-off parameter C in miSVM, MISVM, MissSVM and WELLSVM are selected from $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}\}$. In SAFEW, we use miSVM and MISVM with carefully tuned parameters as base learners. For our CEGE, the trade-off parameters γ_1 and γ_2 are adjusted via the similar way as in Sections 6.1 and 6.2. However, unlike the previous SSL and PUL datasets which provide the ground-truth labels of examples, the MIL datasets here only provide real bag labels without indicating the instance labels, so we regard the class prior π as a tuning parameter for CEGE, and choose π from 0.1 to 0.7 with interval 0.1. Generally, the positive examples will not account for a large proportion in the entire training set.

Table 6 presents the dataset configurations and the bag-level test accuracies achieved by various approaches. We see that CEGE obtains comparable performance with miSVM, MISVM and SAFEW, but is slightly inferior to EM-DD. Especially, EM-DD shows very encouraging performance on *MUSK1* and *MUSK2* datasets and is significantly better than CEGE. We conjecture that the reason may be that our CEGE works on instance level rather than bag level, so one incorrect classification on the example in negative bag will lead to the erroneous bag label assignment. Besides, actually EM-DD is time-consuming which takes more than ten times CPU seconds than miSVM, MISVM and our CEGE on these datasets. Moreover, by comparing CEGE with another popular WSL framework named WELLSVM, we see that CEGE acquires very promising classification results.

6.4 Experiments on Label Noise Learning

LNL is a challenging problem in WSL as the supervision information is imprecise and this will probably mislead the training process of an algorithm. To test the ability of CEGE in dealing with LNL, we adopt the benchmark datasets employed by [26] for our experiment, which include *Thyroid*, *Heart*, *BreastCancer*, *Diabetes*, *GermanCredit*, and *Image*[6]. By defining that the label noise rates in positive data and negative data are $\eta_P = P(\tilde{y} = -1|y = +1)$ and $\eta_N = P(\tilde{y} = +1|y = -1)$ (*a.k.a.* $\eta_P^{(1)}$ and $\eta_N^{(1)}$ in Section 4.4) correspondingly, we evaluate the test accuracies of various methods under three cases of noise rates, namely the symmetric noise rates $(\eta_P = 0.2, \eta_N = 0.2)$, $(\eta_P = 0.4, \eta_N = 0.4)$, and the asymmetric one $(\eta_P = 0.3, \eta_N = 0.1)$. To achieve fair comparison, the contaminated examples in each fold are also identical for all compared methods on every dataset.

The compared LNL methods include Unbiased Estimator (UE) [12], μ Stochastic Gradient Descent (μSGD) [17], Labeled Instance Centroid Smoothing (LICS) [16], and Spectral Cluster Discovery (SCD) [78]. Besides, SAFEW [15] is also compared here as it is able to handle the noisy labels. In the implementations of UE, μSGD, LICS and our CEGE, the label noise rates η_P and η_N are treated as known. The parameters of CEGE and other baseline methods are carefully tuned as described above. For SAFEW, the base learners are composed of Logistic regression, SVM, and Nearest Neighbor classifier as recommended by [15].

Table 7 shows the experimental results. We can see that CEGE performs robustly under both symmetric and asymmetric noise rates on the six datasets. Compared with existing methodologies, CEGE obtains the best performance in terms of the average test accuracy, which is 81.5% that leads SAFEW by a margin of 1.7%.

[5] http://www.cs.columbia.edu/~andrews/mil/datasets.html

[6] These datasets are preprocessed by Gunnar Rätsch at http://theoval.cmp.uea.ac.uk/matlab

TABLE 6: Comparison of the mean bag test accuracies of various approaches on MIL task. Here $n_{bag} = p_+ + n_-$ is the total number of bags. The best two records on each dataset are highlighted in red and blue, respectively. The "√" ("×") denotes that our CEGE is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

Dataset	$(\bar{n}, n_{bag}, d)$	miSVM [7]	MISVM [7]	MissSVM [77]	EM-DD [55]	WELLSVM [14]	SAFEW [15]	CEGE
MUSK1	(476,92,166)	0.746±0.086 √	0.771±0.111	0.731±0.130 √	0.857±0.076 ×	0.753±0.137	0.763±0.138	0.772±0.078
MUSK2	(6598,102,166)	0.723±0.134	0.762±0.099	0.627±0.076 √	0.823±0.074 ×	0.617±0.033 √	0.714±0.124	0.724±0.111
Fox	(1320,200,230)	0.595±0.076	0.530±0.054 √	0.610±0.113	0.560±0.120	0.500±0.061 √	0.570±0.072	0.585±0.043
Tiger	(1220,200,230)	0.800±0.064	0.735±0.085 √	0.720±0.079 √	0.690±0.077 √	0.705±0.040 √	0.790±0.050	0.775±0.069
Elephant	(1391,200,230)	0.775±0.034 √	0.780±0.068	0.715±0.110 √	0.770±0.051 √	0.805±0.034	0.775±0.035 √	0.812±0.054
Average		0.728	0.716	0.681	0.740	0.676	0.722	0.729

TABLE 7: Comparison of the mean test accuracies of various approaches on LNL task. The best two records on each dataset are highlighted in red and blue, respectively. The "√" ("×") denotes that our CEGE is significantly better (worse) than the corresponding existing methods revealed by the paired t-test with significance level 0.05.

Dataset	$(\bar{n}, d)$	(η_P, η_N)	UE [12]	μSGD [17]	LICS [16]	SCD [78]	SAFEW [15]	CEGE
Thyroid	(215,5)	(0.2,0.2)	0.842±0.066	0.852±0.117	0.809±0.062	0.856±0.067	0.865±0.077	0.846±0.063
		(0.3,0.1)	0.800±0.065	0.828±0.071	0.782±0.051 √	0.786±0.049 √	0.815±0.064	0.851±0.046
		(0.4,0.4)	0.772±0.132	0.660±0.132 √	0.745±0.126	0.740±0.139	0.786±0.099	0.808±0.154
Heart	(270,13)	(0.2,0.2)	0.833±0.044	0.833±0.053	0.752±0.078 √	0.841±0.039	0.844±0.038	0.848±0.051
		(0.3,0.1)	0.822±0.080 √	0.811±0.077	0.722±0.059 √	0.796±0.070 √	0.782±0.066 √	0.859±0.057
		(0.4,0.4)	0.789±0.082	0.719±0.107 √	0.670±0.092 √	0.737±0.093 √	0.767±0.070	0.800±0.063
BreastCancer	(683,9)	(0.2,0.2)	0.969±0.023	0.975±0.027	0.948±0.089 √	0.953±0.028 √	0.961±0.023 √	0.974±0.019
		(0.3,0.1)	0.977±0.019	0.962±0.028	0.939±0.048 √	0.965±0.024	0.962±0.022	0.972±0.018
		(0.4,0.4)	0.956±0.037	0.875±0.054 √	0.863±0.036 √	0.912±0.046 √	0.909±0.055 √	0.955±0.036
Diabetes	(768,28)	(0.2,0.2)	0.757±0.040	0.723±0.057	0.703±0.054 √	0.764±0.033	0.766±0.030	0.750±0.038
		(0.3,0.1)	0.741±0.035	0.718±0.028 √	0.688±0.041 √	0.728±0.037	0.709±0.049 √	0.742±0.026
		(0.4,0.4)	0.720±0.035	0.681±0.049 √	0.638±0.096 √	0.715±0.060	0.721±0.065	0.728±0.055
GermanCredit	(1000,24)	(0.2,0.2)	0.754±0.035	0.653±0.029 √	0.681±0.030 √	0.753±0.034	0.759±0.032	0.740±0.045
		(0.3,0.1)	0.687±0.048 √	0.660±0.075 √	0.618±0.086 √	0.668±0.059 √	0.674±0.053 √	0.735±0.063
		(0.4,0.4)	0.680±0.044 √	0.600±0.059 √	0.581±0.066 √	0.677±0.046 √	0.684±0.040	0.714±0.036
Image	(2086,18)	(0.2,0.2)	0.785±0.022	0.819±0.022	0.751±0.096 √	0.831±0.019 ×	0.833±0.012 ×	0.795±0.046
		(0.3,0.1)	0.722±0.021 √	0.814±0.026	0.763±0.064 √	0.742±0.032 √	0.771±0.032 √	0.817±0.016
		(0.4,0.4)	0.722±0.039	0.740±0.022	0.707±0.079 √	0.772±0.031 ×	0.760±0.038	0.742±0.042
Average			0.796	0.774	0.742	0.791	0.798	0.815

Note that SAFEW actually assembles the outputs of multiple base learners via a proper way, so it is quite competitive. In this sense, it is an encouraging result that our CEGE relying on single well-designed model outperforms SAFEW. Besides, compared with our CEGE that employs two auxiliary sets to achieve statistically unbiased and efficient estimation, LICS only utilizes single auxiliary set $\widetilde{S_1}$ for centroid estimation, and introduces a hard constraint to heuristically reduce the estimation error, therefore its performance is far behind that of our method. Similarly, UE also only considers the unbiasedness in designing risk estimator without considering the variance reduction, so its average accuracy is also below 80%. SCD yields better performance than CEGE on *Image* dataset as it considers the spectral property that often resides in vision data in eliminating the label noise.

6.5 Algorithm Validation

From the above experiments, we see that our CEGE generally achieves satisfactory performance on various WSL settings. Here we empirically investigate the reasons behind and also examine some important behaviors of CEGE.

6.5.1 Statistical Efficiency Improvement

In Theorem 2, we have theoretically prove that the estimator $\breve{\mu}(S)$ of our method is more statistically efficient than $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$, which helps to accurately estimate the centroid $\hat{\mu}(S)$. Here we empirically show this by comparing the variances as well as the estimation errors of $\breve{\mu}(S)$, $\breve{\mu}_1(S)$, and $\breve{\mu}_2(S)$. Specifically, the estimation error is computed by the ℓ_2 distance between the estimated value and the real data centroid $\hat{\mu}(S)$.

To this end, we use *GermanCredit* and *PhishingWebsites* datasets with $\zeta = 5\%$ of SSL, *Australian* and *Banknote* datasets with $\eta_P = 5\%$ of PUL, *MUSK2* and *Tiger* datasets of MIL, and *Heart* and *Image* datasets with $(\eta_P = 0.3, \eta_N = 0.1)$ of LNL, and investigate the variances and estimation errors of $\breve{\mu}(S)$, $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ in the first fold on these datasets (see Fig. 2). We choose the above labeling rates or label flipping probabilities as they are the most difficult setting in the corresponding learning task tested in Sections 6.1∼6.4. Note that the estimation error is not reported on MIL datasets as the instance-level ground-truth labels are not provided, so the real centroid $\hat{\mu}(S)$ cannot be computed.

The experimental results suggest that the proposed $\breve{\mu}(S)$ consistently obtains the lowest variance among the three estimators (see blue bars), which verifies the theoretical result in Theorem 2. Besides, we see that the estimation error induced by $\breve{\mu}(S)$ is also smaller than those brought by $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ on four of the six SSL, PUL and LNL datasets (see red bars). Therefore, the usefulness of our strategy that linearly combines $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$ with the optimal β computed in Eq. (14) is demonstrated.

6.5.2 Influence of Inaccurate Class Prior

Note that in the above experiments, we assume that the class prior π is known and directly send the real π to our algorithm.

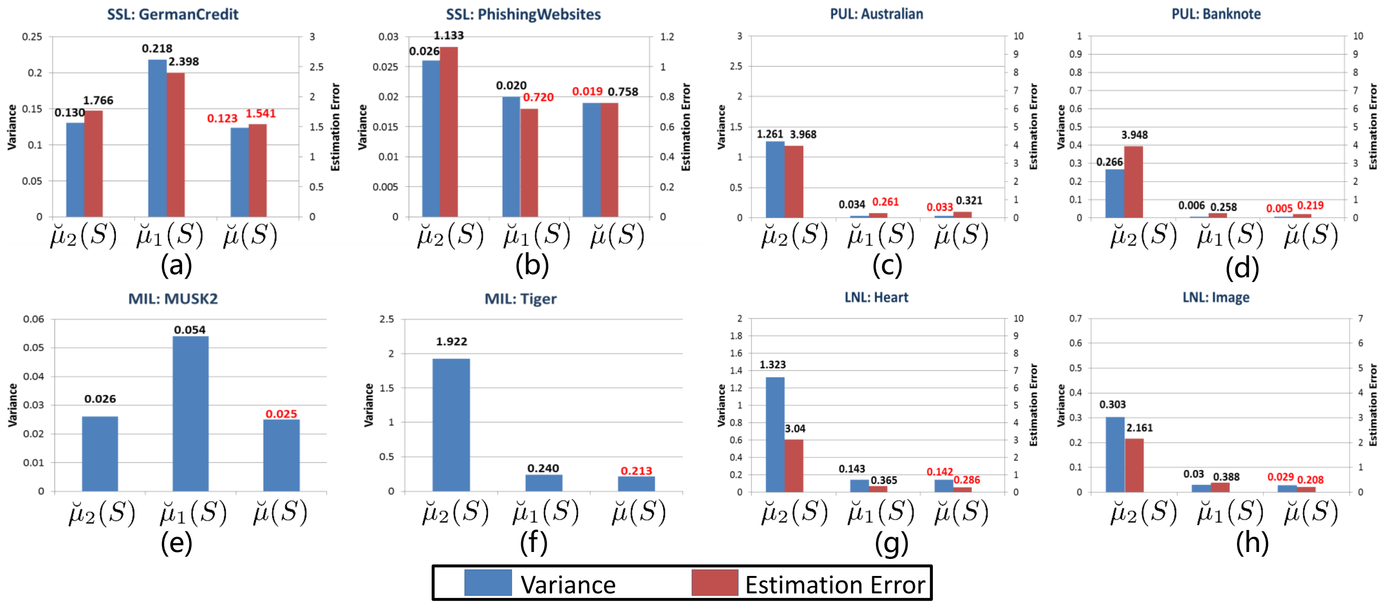

Fig. 2: The comparison on variances and estimation errors of the three estimators $\breve{\mu}(S)$, $\breve{\mu}_1(S)$ and $\breve{\mu}_2(S)$, respectively. The numerical values are annotated above the bars, and the smallest record among the three estimators in variance comparison or estimation error comparison on a certain dataset is highlighted in red color.

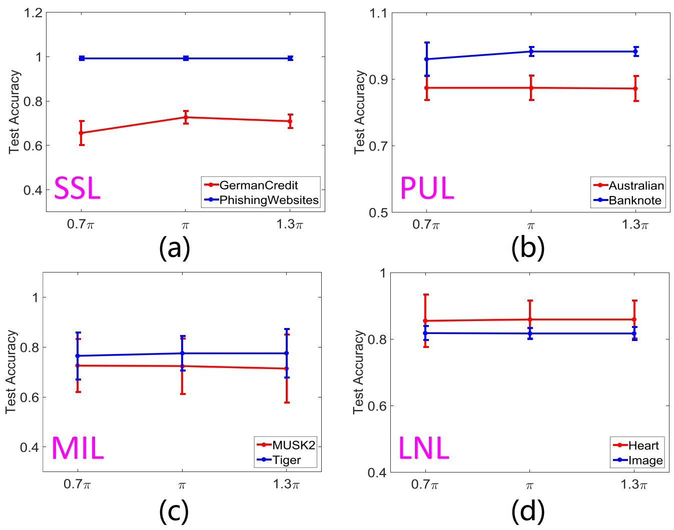

Fig. 3: Performance variation of CEGE w.r.t. inaccurate π.

Practically, this prior needs to be pre-estimated, so here we discuss how the inaccurate estimate of π influences the algorithm output. To be specific, we also use the eight datasets belonging to SSL, PUL, MIL and LNL appeared in Section 6.5.1, and examine the mean test accuracy of ten trials of our model when the prior is set to $\{0.7\pi, \pi, 1.3\pi\}$ where π denotes the precise value. From Fig. 3, we easily observe that the slight deviations of class prior to its real value will have little impact on our model output. Consequently, our CEGE can be used safely in various practical applications across SSL, PUL, MIL and LNL.

7 Conclusion and Future Work

In this paper, we show that a variety of typical WSL problems can be perfectly tackled by our developed general framework dubbed "CEGE". Specifically, we reveal that the key to address various kinds of weak supervision is to accurately estimate the centroid of dataset. Importantly, with the aid of two auxiliary sets, CEGE finds such an estimate which is both unbiased and statistically efficient, enabling our method to produce precise and reliable classification results. The generality of our CEGE lies in three aspects: 1) CEGE is able to cover many popular WSL problems within a unified framework; 2) CEGE directly accommodates to many common loss functions as indicated in Table 1; and 3) CEGE can be equipped with both SVM and neural networks to reach linear or nonlinear classifier.

Although this paper only presents the applications of CEGE on four WSL problems, we believe that it is also suitable for other WSL paradigms such as similarity-unlabeled learning [79] and semi-supervised clustering [80], by treating a pair of examples as a data point in Eq. (5). Besides, the instantiation of CEGE to distributionally inconsistent problems such as domain adaptation and transfer learning still remains unclear. These potential usages of CEGE needs further investigation.

References

[1] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, vol. 5, no. 1, pp. 44–53, 2018.

[2] X. Zhu and B. Goldberg, *Introduction to semi-supervised learning*. Morgan & Claypool Publishers, 2009.

[3] B. Liu, W. S. Lee, P. S. Yu, and X. Li, "Partially supervised classification of text documents," in *International Conference on Machine Learning*, 2002, pp. 387–394.

[4] N. Lu, G. Niu, A. K. Menon, and M. Sugiyama, "On the minimal supervision for training any binary classifier from only unlabeled data," in *International Conference on Learning Representations*, 2019, pp. 1–13.

[5] B. Settles, "Active learning literature survey," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.

[6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[7] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems*, 2003, pp. 577–584.

[8] T. Cour, B. Sapp, C. Jordan, and B. Taskar, "Learning from ambiguously labeled images," in *IEEE International Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 919–926.

[9] C. Gong, T. Liu, Y. Tang, J. Yang, J. Yang, and D. Tao, "A regularization approach for instance-based superset label learning," *IEEE Transactions on Cybernetics*, vol. 48, no. 3, pp. 967–978, 2017.

[10] T. Ishida, G. Niu, W. Hu, and M. Sugiyama, "Learning from complementary labels," in *Advances in Neural Information Processing Systems*, 2017, pp. 5639–5649.

[11] T. Ishida, G. Niu, and M. Sugiyama, "Binary classification from positive-confidence data," in *Advances in Neural Information Processing Systems*, 2018, pp. 5917–5928.

[12] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Advances in Neural Information Processing Systems*, 2013, pp. 1196–1204.

[13] D. C. Brabham, "Crowdsourcing as a model for problem solving: An introduction and cases," *Convergence*, vol. 14, no. 1, pp. 75–90, 2008.

[14] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, "Convex and scalable weakly labeled svms," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2151–2188, 2013.

[15] Y.-F. Li, L.-Z. Guo, and Z.-H. Zhou, "Towards safe weakly supervised learning," *IEEE transactions on Pattern Analysis and Machine Intelligence*, pp. 1–14, 2019.

[16] W. Gao, L. Wang, Y.-F. Li, and Z.-H. Zhou, "Risk minimization in the presence of label noise," in *AAAI Conference on Artificial Intelligence*, 2016, pp. 1575–1581.

[17] G. Patrini, F. Nielsen, R. Nock, and M. Carioni, "Loss factorization, weakly supervised learning and label noise robustness," in *International Conference on Machine Learning*, 2016, pp. 708–717.

[18] C. Gong, H. Shi, T. Liu, C. Zhang, J. Yang, and D. Tao, "Loss decomposition and centroid estimation for positive and unlabeled learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[19] B. Schölkopf, A. J. Smola, F. Bach *et al.*, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[20] C. Gong, T. Liu, J. Yang, and D. Tao, "Large-margin label-calibrated support vector machines for positive and unlabeled learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2019.

[21] M. Du Plessis, G. Niu, and M. Sugiyama, "Convex formulation for learning from positive and unlabeled data," in *International Conference on Machine Learning*, 2015, pp. 1386–1394.

[22] R. Kiryo, G. Niu, M. C. du Plessis, and M. Sugiyama, "Positive-unlabeled learning with non-negative risk estimator," in *Advances in Neural Information Processing Systems*, 2017, pp. 1674–1684.

[23] T. Sakai, M. C. du Plessis, G. Niu, and M. Sugiyama, "Semi-supervised classification based on classification from positive and unlabeled data," in *International Conference on Machine Learning*, 2017, pp. 2998–3006.

[24] J. Bekker and J. Davis, "Estimating the class prior in positive and unlabeled data through decision tree induction," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 2712–2719.

[25] C. Scott, G. Blanchard, and G. Handy, "Classification with asymmetric label noise: Consistency and maximal denoising," in *Annual Conference on Learning Theory*, 2013, pp. 489–511.

[26] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 447–461, 2016.

[27] H. Ramaswamy, C. Scott, and A. Tewari, "Mixture proportion estimation via kernel embeddings of distributions," in *International Conference on Machine Learning*, 2016, pp. 2052–2060.

[28] M. Du Plessis, G. Niu, and M. Sugiyama, "Class-prior estimation for learning from positive and unlabeled data," in *Asian Conference on Machine Learning*, 2015, pp. 221–236.

[29] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[30] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *International Conference on Machine Learning*, Washington, DC, USA, 2003, pp. 912–919.

[31] D. Zhou and O. Bousquet, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, Vancouver, Canada, 2003, pp. 321–328.

[32] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, Nov 2006.

[33] C. Gong, D. Tao, K. Fu, and J. Yang, "Fick's law assisted propagation for semisupervised learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 9, pp. 2148–2162, 2014.

[34] C. Gong, T. Liu, D. Tao, K. Fu, E. Tu, and J. Yang, "Deformed graph laplacian for semisupervised learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2261–2274, 2015.

[35] J. Wang, F. Wang, C. Zhang, H. C. Shen, and L. Quan, "Linear neighborhood propagation and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1600–1615, 2008.

[36] C. Gong, D. Tao, W. Liu, L. Liu, and J. Yang, "Label propagation via teaching-to-learn and learning-to-teach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 6, pp. 1452–1465, 2016.

[37] C. Gong, D. Tao, S. J. Maybank, W. Liu, G. Kang, and J. Yang, "Multi-modal curriculum learning for semi-supervised image classification," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3249–3260, 2016.

[38] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[39] T. Joachims, "Transductive inference for text classification using support vector machines," in *International Conference on Machine Learning*, vol. 99, 1999, pp. 200–209.

[40] R. Collobert, F. Sinz, J. Weston, and L. Bottou, "Large scale transductive svms," *Journal of Machine Learning Research*, vol. 7, no. Aug, pp. 1687–1712, 2006.

[41] Y.-F. Li, J. T. Kwok, and Z.-H. Zhou, "Semi-supervised learning using label mean," in *International Conference on Machine Learning*, 2009, pp. 633–640.

[42] Y.-F. Li and Z.-H. Zhou, "Towards making unlabeled data never hurt," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 1, pp. 175–188, 2014.

[43] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *International Conference on Learning Representations*, 2017, pp. 1–14.

[44] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017, pp. 1195–1204.

[45] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.

[46] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 5050–5060.

[47] X. Li and B. Liu, "Learning to classify texts using positive and unlabeled data," in *International Joint Conference on Artificial Intelligence*, vol. 3, 2003, pp. 587–592.

[48] C. Elkan and K. Noto, "Learning classifiers from only positive and unlabeled data," in *Proceedings of The 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 213–220.

[49] M. Du Plessis, G. Niu, and M. Sugiyama, "Analysis of learning from positive and unlabeled data," in *Advances in Neural Information Processing Systems*, 2014, pp. 703–711.

[50] W. S. Lee and B. Liu, "Learning with positive and unlabeled examples using weighted logistic regression," in *International Conference on Machine Learning*, vol. 3, 2003, pp. 448–455.

[51] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu, "Building text classifiers using positive and unlabeled examples," in *IEEE International Conference on Data Mining*, vol. 2, 2003, pp. 179–186.

[52] H. Shi, S. Pan, J. Yang, and C. Gong, "Positive and unlabeled learning via loss decomposition and centroid estimation." in *International Joint Conferences on Artificial Intelligence*, 2018, pp. 2689–2695.

[53] C. Zhang, D. Ren, T. Liu, J. Yang, and C. Gong, "Positive and unlabeled learning with label disambiguation," in *International Joint Conference on Artificial Intelligence*, 2019, pp. 4250–4256.

[54] C. Gong, H. Shi, J. Yang, J. Yang, and J. Yang, "Multi-manifold positive and unlabeled learning for visual analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[55] Q. Zhang and S. A. Goldman, "Em-dd: An improved multiple-instance learning technique," in *Advances in Neural Information Processing Systems*, 2002, pp. 1073–1080.

[56] M. Peng and Q. Zhang, "Address instance-level label prediction in multiple instance learning," *arXiv preprint arXiv:1905.12226*, 2019.

[57] J. Wu, Y. Yu, C. Huang, and K. Yu, "Deep multiple instance learning for image classification and auto-annotation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3460–3469.

[58] X. Wang, Y. Yan, P. Tang, X. Bai, and W. Liu, "Revisiting multiple instance neural networks," *Pattern Recognition*, vol. 74, pp. 15–24, 2018.

[59] J. Wang and J.-D. Zucker, "Solving multiple-instance problem: A lazy learning approach," in *International Conference Machine Learning*, 2000, pp. 1119–1126.

[60] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels," in *International Conference Machine Learning*, vol. 2, no. 3, 2002, pp. 179–486.

[61] X.-S. Wei, J. Wu, and Z.-H. Zhou, "Scalable algorithms for multi-instance learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 975–987, 2016.

[62] M. Ilse, J. M. Tomczak, and M. Welling, "Attention-based deep multiple instance learning," in *International Conference on Machine Learning*, 2018, pp. 3376–3391.

[63] X. Wang, Y. Yan, P. Tang, W. Liu, and X. Guo, "Bag similarity network for deep multi-instance learning," *Information Sciences*, vol. 504, pp. 578–588, 2019.

[64] C. Brodley and M. Friedl, "Identifying mislabeled training data," *Journal of Artificial Intelligence Research*, vol. 11, pp. 131–167, 1999.

[65] X. Zhu, X. Wu, and Q. Chen, "Eliminating class noise in large datasets," in *International Conference on Machine Learning*, vol. 3, 2003, pp. 920–927.

[66] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *International Conference on Machine Learning*, 2018, pp. 2304–2313.

[67] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Advances in Neural Information Processing Systems*, 2018, pp. 8527–8537.

[68] K.-H. Lee, X. He, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5447–5456.

[69] Y. Lyu and I. W. Tsang, "Curriculum loss: Robust learning and generalization against label corruption," in *International Conference on Learning Representations*, 2020, pp. 1–14.

[70] B. Rooyen, A. Menon, and R. Williamson, "Learning with symmetric label noise: The importance of being unhinged," in *Advances in Neural Information Processing Systems*, 2015, pp. 10–18.

[71] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1944–1952.

[72] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" in *Advances in Neural Information Processing Systems*, 2019, pp. 6835–6846.

[73] B. Han, I. Tsang, and L. Chen, "On the convergence of a family of robust losses for stochastic gradient descent," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2016, pp. 665–680.

[74] B. Han, G. Niu, X. Yu, Q. Yao, M. Xu, I. W. Tsang, and M. Sugiyama, "Sigua: Forgetting may make learning with noisy labels more robust." International Conference on Machine Learning, 2020.

[75] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[76] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Annual meeting of the Association for Computational Linguistics*, 1995, pp. 189–196.

[77] Z.-H. Zhou and J.-M. Xu, "On the relation between multi-instance learning and semi-supervised learning," in *International Conference on Machine Learning*, 2007, pp. 1167–1174.

[78] Y. Luo, B. Han, and C. Gong, "A bi-level formulation for label noise learning with spectral cluster discovery," in *International Joint Conference on Artificial Intelligence*, 2020, pp. 1–7.

[79] H. Bao, G. Niu, and M. Sugiyama, "Classification from pairwise similarity and unlabeled data," in *International Conference on Machine Learning*, 2018, pp. 452–461.

[80] B. Kulis, S. Basu, I. Dhillon, and R. Mooney, "Semi-supervised graph clustering: a kernel approach," *Machine learning*, vol. 74, no. 1, pp. 1–22, 2009.

Chen Gong received his doctoral degree from the University of Technology Sydney in 2017. Currently, he is a professor with Nanjing University of Science and Technology. His research interests mainly include machine learning and data mining. He has published more than 100 technical papers at prominent journals and conferences such as JMLR, IEEE T-PAMI, IEEE T-NNLS, IEEE T-IP, ACM T-IST, CVPR, ICML, NeurIPS, AAAI, IJCAI, ICDM, etc. He also serves as the reviewer for more than 20 international journals such as JMLR, AIJ, IEEE T-PAMI, IEEE T-NNLS, IEEE T-IP, and also the (S)PC member of several top-tier conferences such as ICML, NeurIPS, ICLR, CVPR, AAAI, IJCAI, ICDM, etc. He received the "Excellent Doctorial Dissertation" awarded by Chinese Association for Artificial Intelligence, and was enrolled by the "Young Elite Scientists Sponsorship Program" of CAST. He was also the recipient of "Wu Wen-Jun AI Excellent Youth Scholar Award".

Jian Yang received the PhD degree from Nanjing University of Science and Technology (NUST), on the subject of pattern recognition and intelligence systems in 2002. In 2003, he was a Postdoctoral researcher at the University of Zaragoza. From 2004 to 2006, he was a Postdoctoral Fellow at Biometrics Centre of Hong Kong Polytechnic University. From 2006 to 2007, he was a Postdoctoral Fellow at Department of Computer Science of New Jersey Institute of Technology. Now, he is a Chang-Jiang professor in the School of Computer Science and Technology of NUST. He is the author of more than 200 scientific papers in pattern recognition and computer vision. His papers have been cited more than 20000 times in the Google Scholar. His research interests include pattern recognition, computer vision and machine learning. Currently, he is/was an associate editor of Pattern Recognition, Pattern Recognition Letters, IEEE Trans. Neural Networks and Learning Systems, and Neurocomputing. He is a Fellow of IAPR.

Jane You received the B.Eng. degree in electronics engineering from Xi'an Jiaotong University, Xi'an, China, in 1986, and the Ph.D. degree in computer science from La Trobe University, Melbourne, VIC, Australia, in 1992. She was a Lecturer with the University of South Australia, Adelaide SA, Australia, and a Senior Lecturer with Griffith University, Nathan, QLD, Australia, from 1993 to 2002. She is currently a Full Professor with The Hong Kong Polytechnic University, Hong Kong. Her current research interests include image processing, pattern recognition, medical imaging, biometrics computing, multimedia systems, and data mining.

Masashi Sugiyama received the degrees of Bachelor, Master, and Doctor of Engineering in Computer Science from Tokyo Institute of Technology, Japan in 1997, 1999, and 2001, respectively. In 2001, he was appointed Assistant Professor in the same institute, and was promoted to Associate Professor in 2003. Then he moved to the University of Tokyo as Professor in 2014. Since 2016, he have concurrently served as the Director of the RIKEN Center for Advanced Intelligence Project. His research interests include theories and algorithms of machine learning and data mining, and a wide range of applications such as signal processing, image processing, and robot control. His works have been published on prestigious journals and conferences such as JMLR, TPAMI, MLJ, ICML, NeurIPS, etc. He received the Japan Society for the Promotion of Science Award and the Japan Academy Medal in 2017.