Partial Label Dimensionality Reduction via Confidence-Based Dependence Maximization

Wei-Xuan Bao
[†]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[‡]Key Laboratory of Computer Network and Information Integration, Ministry of Education, China
baowx@seu.edu.cn

Jun-Yi Hang
[†]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[‡]Key Laboratory of Computer Network and Information Integration, Ministry of Education, China
hangjy@seu.edu.cn

Min-Ling Zhang*
[†]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[‡]Key Laboratory of Computer Network and Information Integration, Ministry of Education, China
zhangml@seu.edu.cn

ABSTRACT

Partial label learning deals with training examples each associated with a set of *candidate* labels, among which only one is valid. Most existing works focus on manipulating the label space by estimating the labeling confidences of candidate labels, while the task of manipulating the feature space by dimensionality reduction has been rarely investigated. In this paper, a novel partial label dimensionality reduction approach named Cenda is proposed via confidence-based dependence maximization. Specifically, Cenda adapts the Hilbert-Schmidt Independence Criterion (HSIC) to help identify the projection matrix, where the dependence between projected feature information and confidence-based labeling information is maximized iteratively. In each iteration, the projection matrix admits closed-form solution by solving a tailored generalized eigenvalue problem, while the labeling confidences of candidate labels are updated by conducting kNN aggregation in the projected feature space. Extensive experiments over a broad range of benchmark data sets show that the predictive performance of well-established partial label learning algorithms can be significantly improved by coupling with the proposed dimensionality reduction approach.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning algorithms**; *Learning paradigms.*

KEYWORDS

Partial Label Learning, Dimensionality Reduction, Hilbert-Schmidt Independence Criterion

ACM Reference Format:
Wei-Xuan Bao, Jun-Yi Hang, and Min-Ling Zhang. 2021. Partial Label Dimensionality Reduction via Confidence-Based Dependence Maximization. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '21), August 14–18, 2021, Virtual Event, Singapore.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3447548.3467313

*Corresponding author

KDD '21, August 14–18, 2021, Virtual Event, Singapore

ACM ISBN 978-1-4503-8332-5/21/08…$15.00
https://doi.org/10.1145/3447548.3467313

1 INTRODUCTION

Partial label (PL) learning aims to induce a multi-class classifier from training examples each associated with a set of candidate labels, among which only one is valid [11, 29, 49, 53]. Compared with the ordinary multi-class classification problem where each training example is associated with only one valid label, partial label learning directly learns from ambiguously labeled examples [56], thus greatly reduces the cost of data annotation. As an emerging weakly supervised learning framework which arises in many real-world scenarios such as web mining [23], multimedia content analysis [8, 47], ecoinformatics [4, 42], natural language processing [34, 54], etc., partial label learning has been studied extensively in recent years.

In solving real-world tasks, dimensionality reduction is an effective technique to improve the generalization ability of learning system, i.e. alleviate the issue of *curse of dimensionality*. In partial label learning, the limited supervision information retrieved from training set often leads to the less satisfactory generalization performance. Although it is desirable to incorporate dimensionality reduction mechanism in partial label learning, existing works mainly focus on manipulating the label space by estimating the labeling confidences of candidate labels [6, 8, 16, 28, 42], while the manipulation of the feature space by dimensionality reduction is rarely investigated. To the best of our knowledge, Delin [43] is the only available partial label dimensionality reduction approach, which employs the linear discriminant analysis (LDA) technique to maximize the inter-class separability in the induced feature space. Nevertheless, due to the intrinsic properties of LDA, the essential dimensionality of the feature space induced by DELIN is upper-bounded by the number of class labels, which may lead to degenerated performance due to the excessively low dimensionality of the induced feature space.

In this paper, we propose a novel partial label dimensionality reduction method named Cenda, i.e. *partial label dimensionality reduction via ConfidENce-based Dependence mAximization*. Cenda performs dimensionality reduction by maximizing the dependence between the projected feature information and the confidence-based labeling information, where the dependence is measured by the *Hilbert-Schmidt Independence Criterion* (HSIC). Since the ground-truth label is not directly accessible to the learning algorithm, we

adapt HSIC to accommodate the exploitation of partial label training examples and employ an alternating procedure to optimize the projection matrix and update the confidences of candidate labels iteratively. Specifically, in each iteration, the projection matrix is identified by solving a tailored generalized eigenvalue problem, while the labeling confidences of candidate labels are updated by conducting kNN aggregation in the projected feature space. Comprehensive experiments over synthetic and real-world partial label data sets validate the effectiveness of CENDA as a dimensionality reduction method to improve the generalization performance of state-of-the-art partial label learning algorithms.

The rest of this paper is organized as follows. Section 2 briefly discusses related works on partial label learning. Section 3 introduces the technical procedure of the proposed CENDA approach. Section 4 reports detailed results of experimental studies. Finally, Section 5 concludes this paper.

2 RELATED WORKS

Partial label learning learns from *inaccurate* supervision information where the ground-truth label is concealed in the candidate label set of each training example. In terms of the formulation of the learning problem, partial label learning is related to several well-established weakly supervised learning frameworks such as multi-label learning [44, 50, 51], semi-supervised learning [55, 58] and multi-instance learning [2, 5].

Most existing partial label learning algorithms learn from partial label data via candidate label disambiguation, which is manipulated in the label space. There are two main types of disambiguation strategies, namely *disambiguation by identification* and *disambiguation by averaging*. For the strategy of disambiguation by identification, iterative optimization procedure such as EM is employed to estimate the unknown ground-truth label which is treated as the latent variable. Different methods such as maximizing the likelihood of observing the PL training examples over their candidate label sets [24, 27, 28], or maximizing the predictive margin between candidate labels and non-candidate labels of PL training examples [6, 30, 45] can be utilized to instantiate the optimization objective.

The strategy of disambiguation by averaging treats all candidate labels of the PL training example in an equal manner whose modeling outputs are averaged to yield the final prediction. Different methods are employed to instantiate the averaging procedure, such as distinguishing the averaged modeling outputs from candidate labels between the modeling outputs from non-candidate labels for discriminative models [11, 41], or aggregating the votes among candidate labels of the unseen instance's neighboring examples for distance-based models [16, 21, 39, 48].

As an effective technique to improve the generalization ability of the learning system by manipulating feature space, dimensionality reduction has been extensively studied in numerous machine learning paradigms, such as multi-label learning [22, 32, 33, 40, 46, 52] where each training example is associated with multiple valid class labels other than multiple candidate labels. However, the problem of partial label dimensionality reduction has been rarely investigated. To the best of our knowledge, DELIN [43] is the only available partial label dimensionality reduction approach which adapts the linear discriminant analysis technique to identify the projection matrix for dimensionality reduction.

Compared with DELIN that maximizes the inter-class separability in the induced feature space, CENDA makes better use of labeling information by adapting to maximize the dependence between projected feature information and confidence-based labeling information. Besides, the dimension of the projected feature space can be set arbitrarily in our approach other than being smaller than the number of class labels in DELIN.

3 THE PROPOSED APPROACH

Let $\mathcal{X} = \mathbb{R}^d$ denote the d-dimensional instance space and $\mathcal{Y} = \{l_1, l_2, ..., l_q\}$ denote the label space with q class labels. Given the partial label training set $\mathcal{D} = \{(\boldsymbol{x}_i, S_i) | 1 \leqslant i \leqslant m\}$, where $\boldsymbol{x}_i \in \mathcal{X}$ is a d-dimensional feature vector $(x_{i_1}, x_{i_2}, ..., x_{i_d})^\top$ and $S_i \subseteq \mathcal{Y}$ is the candidate label set associated with $\boldsymbol{x}_i$ among which only one is the ground-truth label. The task of partial label learning is to derive a multi-class classification model $f : \mathcal{X} \to \mathcal{Y}$ from the training set $\mathcal{D}$.

Let $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_m] \in \mathbb{R}^{d \times m}$ be the instance matrix formed by concatenating all feature vectors in the training set, the task of partial label dimensionality reduction is to find a projection matrix $\mathbf{P} = [\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_{d'}] \in \mathbb{R}^{d \times d'} (d' \ll d)$ which maps the training examples $\mathbf{X}$ into the d'-dimensional feature space, i.e. $\mathbf{X}' = \mathbf{P}^\top \mathbf{X}$. Considering that the feature description and the true label are characterizations of the same example from two perspectives, we attempt to find a lower-dimensional feature space where the dependence between the induced feature information and the confidence-based labeling information is maximized. Accordingly, CENDA adapts the Hilbert-Schmidt Independence Criterion to help identify the projection matrix iteratively via a two-stage alternating procedure consisting of *confidence-based dependence maximization* and *kNN-based candidate label confidence updating*.

To fullfill the alternative procedure, we construct the labeling confidence matrix $\mathbf{Y} = [Y_{i,j}]_{m \times q}$ where each element Y_{ij} denotes the estimated confidence of l_j being the ground-truth label for $\boldsymbol{x}_i$ and initialize it as follows:

$$\forall\, 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant q:\ Y_{ij} = \begin{cases} \frac{1}{|S_i|}, & \text{if } l_j \in S_i \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Here, the constraints $\sum_{j=1}^{q} Y_{ij} = 1 (1 \leqslant i \leqslant m)$ hold for each iteration of CENDA.

For the stage of confidence-based dependence maximization, we firstly introduce two kernel matrices for the projected feature space and the label space respectively. Given partial label training set $\mathcal{D} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_1, \boldsymbol{y}_m)\}$, the kernel matrix for the projected feature space is defined as $\mathbf{K} = [K_{ij}]_{m \times m}$, where K_{ij} is formulated as:

$$K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j) \triangleq \left\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \right\rangle = \left\langle \mathbf{P}^\top \boldsymbol{x}_i, \mathbf{P}^\top \boldsymbol{x}_j \right\rangle \tag{2}$$

For label space, the kernel matrix is defined as $\mathbf{L} = [L_{ij}]_{m \times m}$, where L_{ij} corresponds to the linear kernel function for simplicity:

$$L_{ij} = l(\boldsymbol{y}_i, \boldsymbol{y}_j) \triangleq \left\langle \boldsymbol{y}_i, \boldsymbol{y}_j \right\rangle \tag{3}$$

where $\boldsymbol{y}_i = (Y_{i1}, Y_{i2}, ..., Y_{iq})^\top$ denotes the ith row of the labeling confidence matrix $\mathbf{Y}$.

Then we attempt to maximize the dependence between the projected feature information and the confidence-based labeling information. The Hilbert-Schmidt Independence Criterion [18] is an effective measure of dependence which has been successfully applied to solve various machine learning tasks [7, 14, 17]. HSIC computes the square of the norm of the cross-covariance operator over the domain $\mathcal{X} \times \mathcal{Y}$ in Hilbert space. Considering its simplicity and neat theoretical properties, we adapt HSIC to deal with the inaccurate supervision information and formulate the empirical estimate of HSIC as:

$$\text{HSIC}(\mathcal{F}, \mathcal{G}, \mathcal{D}) = (m-1)^{-2}\text{tr}\,(\mathbf{HKHL}) \tag{4}$$

where $\text{tr}(\cdot)$ is the trace of a matrix and $\mathbf{H} = \mathbf{I} - \frac{1}{m}\boldsymbol{e}\boldsymbol{e}^\top$ with $\boldsymbol{e}$ being an all-one column vector. $\mathcal{F}$ and $\mathcal{G}$ are the reproducing kernel Hilbert space (RKHS) mapped from $\mathcal{X}$ and $\mathcal{Y}$ respectively.

Substituting $\mathbf{K} = \mathbf{X}^\top \boldsymbol{p}\boldsymbol{p}^\top \mathbf{X}$ into Eq.(4) and dropping the normalization term, we obtain the objective function as follows:

$$\begin{aligned} \boldsymbol{p}^* &= \arg\max_{\boldsymbol{p}} \text{tr}(\mathbf{H}\mathbf{X}^\top \boldsymbol{p}\boldsymbol{p}^\top \mathbf{X}\mathbf{H}\mathbf{L}) \\ &= \arg\max_{\boldsymbol{p}} \boldsymbol{p}^\top (\mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top)\boldsymbol{p} \end{aligned} \tag{5}$$

To avoid the scaling problem, we add the constraint that the l_2-norm of $\boldsymbol{p}$ should be 1, i.e. $\boldsymbol{p}^\top \boldsymbol{p} = 1$. Note that $\mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top$ is symmetric, its eigenvalues are all real and the eigenvectors are orthogonal to each other. After sorting the eigenvalues $\lambda_1 \geqslant \cdots \geqslant \lambda_d$, the optimal $\boldsymbol{p}^*$ is the eigenvector corresponding to the largest eigenvalue λ_1. Furthermore, the projection matrix can be set to $\mathbf{P} = [\boldsymbol{p}_1, ..., \boldsymbol{p}_{d'}]$ where $\boldsymbol{p}_i$ is the eigenvector associated with the ith largest eigenvalue. The corresponding HSIC value is:

$$\text{HSIC} = \sum_{i=1}^{d'} \lambda_i \tag{6}$$

Since the eigenvalue reflects the contribution of the corresponding dimension, we can control the dimensionality of the projected feature space, i.e. d' by setting a threshold thr ($0 \leqslant thr \leqslant 1$) and choose the first d' eigenvectors such that:

$$\sum_{i=1}^{d'} \lambda_i \geqslant thr \times (\sum_{i=1}^{d} \lambda_i) \tag{7}$$

In the above procedure, we obtain a subspace where the projection bases are orthonormal, i.e. $\boldsymbol{p}_i^\top \boldsymbol{p}_j = \delta_{ij}$[1]. However, such projection strategy still remains some redundant information in the lower-dimensional feature space as the feature vectors are still correlated after projection [9]. Specifically, we expect the projected features to be uncorrelated, i.e. $\text{Cor}(\boldsymbol{p}_i^\top \mathbf{X}, \boldsymbol{p}_j^\top \mathbf{X}) = \delta_{ij}, 1 \leqslant i, j \leqslant d'$. Thus we introduce the new constraint:

$$\boldsymbol{p}_i^\top \mathbf{X}\mathbf{X}^\top \boldsymbol{p}_j = \delta_{ij} \tag{8}$$

By jointly considering the orthonormal constraints over projection bases and the uncorrelatedness constraints over projected features, we rewrite the optimization problem as:

$$\begin{aligned} &\max_{\mathbf{P}} \ \text{tr}(\mathbf{H}\mathbf{X}^\top \mathbf{P}\mathbf{P}^\top \mathbf{X}\mathbf{H}\mathbf{L}) \\ &\text{s.t.} \quad \boldsymbol{p}_i^\top (\mu \mathbf{X}\mathbf{X}^\top + (1-\mu)\mathbf{I})\boldsymbol{p}_j = \delta_{ij} \end{aligned} \tag{9}$$

[1] δ_{ij} represents the Kronecker delta where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise.

Table 1: The pseudo-code of CENDA.

Inputs:

$\mathcal{D}$	:	the PL training set $\{(\boldsymbol{x}_i, S_i) \mid 1 \leq i \leq m\}$ ($\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \{l_1, l_2, \ldots, l_q\}, \boldsymbol{x}_i \in \mathcal{X}, S_i \subseteq \mathcal{Y}$)
thr	:	the threshold parameter in Eq.(7)
μ	:	the trade-off parameter in Eq.(9)
k	:	the number of nearest neighbors used for candidate label disambiguation

Outputs:

$\mathcal{D}'$	:	the transformed lower-dimensional PL training set $\{(\boldsymbol{x}_i', S_i) \mid 1 \leqslant i \leqslant m\}$

Process:

Initialize the $m \times q$ labeling confidence matrix $\mathbf{Y}$ according to Eq.(1);
Specify the instance matrix $\mathbf{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_m]$;
repeat
  Calculate the kernel matrix for label space $\mathbf{L} = [L_{ij}]_{m \times m}$ according to Eq.(3);
  Calculate $\mathbf{H} = \mathbf{I} - \frac{1}{m}\boldsymbol{e}\boldsymbol{e}^\top$;
  Solve the generalized eigenvalue problem of Eq.(12), and then form the projection matrix $\mathbf{P}$ by concatenating the d' eigenvectors w.r.t. the top d' eigenvalues satisfying Eq.(7);
  Derive the lower-dimensional PL training set $\mathcal{D}' = \{(\boldsymbol{x}_i', S_i) \mid \boldsymbol{x}_i' = \mathbf{P}^\top \boldsymbol{x}_i, 1 \leqslant i \leqslant m\}$;
  for i=1 to m **do**
    Identify the k nearest neighbors of $\boldsymbol{x}_i'$ in $\mathcal{D}'$ as $\mathcal{N}(\boldsymbol{x}_i')$;
  end for
  Calculate the intermediate matrix $\mathbf{B}$ according to Eq.(13);
  Derive the updated labeling confidence matrix $\mathbf{Y}'$ according to Eq.(14);
  $\mathbf{Y} = \mathbf{Y}'$;
until convergence

where $\mu \in (0, 1)$ is a trade-off parameter which balances the importance of the above two constraints.

According to the properties of the trace of matrix, by Lagrange method [3, 15], we construct Lagrange function:

$$\mathcal{L}(\mathbf{P}) = \text{tr}\left(\mathbf{P}^\top \mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top \mathbf{P}\right) + \text{tr}\left(\mathbf{\Lambda}\left(\mathbf{I} - \mathbf{P}^\top\left(\mu \mathbf{X}\mathbf{X}^\top + (1-\mu)\,\mathbf{I}\right)\mathbf{P}\right)\right) \tag{10}$$

where $\mathbf{\Lambda} = \text{diag}(\lambda_1, ..., \lambda_{d'}) \in \mathbb{R}^{d' \times d'}$ is a diagonal matrix whose entries are the Lagrange multipliers. Then we set the derivative of the Lagrange function to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{P}} = 2\mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top \mathbf{P} - 2(\mu \mathbf{X}\mathbf{X}^\top + (1-\mu)\mathbf{I})\mathbf{P}\mathbf{\Lambda} \overset{\text{set}}{=} 0 \tag{11}$$

Further, we obtain the projection matrix $\mathbf{P} = [\boldsymbol{p}_1, ..., \boldsymbol{p}_{d'}]$, where $\boldsymbol{p}_i$ is the eigenvector associated with the ith largest eigenvalue of the following generalized eigenvalue problem:

$$\mathbf{X}\mathbf{H}\mathbf{L}\mathbf{H}\mathbf{X}^\top \boldsymbol{p} = \lambda(\mu \mathbf{X}\mathbf{X}^\top + (1-\mu)\mathbf{I})\boldsymbol{p} \tag{12}$$

Table 2: Characteristics of the synthetic experimental data sets.

Data Set	# Examples	# Features	# Class Labels	# False Positive Labels (r)	Task Domain
mediamill	2,854	120	10	$r = 1, 2, 3$	*video semantic detection* [36]
tmc2007	8,670	981	18	$r = 1, 2, 3$	*text anomaly detection* [38]
slashdot	3,142	1,079	19	$r = 1, 2, 3$	*text classification* [26]
amazon	1,500	1,326	50	$r = 1, 2, 3$	*authorship identification* [12]
DeliciousMIL	1,409	1,389	20	$r = 1, 2, 3$	*sentence labeling* [37]
bookmark	2,500	1,413	57	$r = 1, 2, 3$	*automatic tag suggestion* [25]
sports	9,120	1,738	19	$r = 1, 2, 3$	*human activity recognition* [1]
sector	6,412	6,104	105	$r = 1, 2, 3$	*text classification* [35]

After inducing the projection matrix $\mathbf{P}$, we construct a new partial label training set $\mathcal{D}' = \{(\boldsymbol{x}'_i, S_i) \mid 1 \leqslant i \leqslant m\}$ in the projected feature space, where $\boldsymbol{x}'_i = \mathbf{P}^\top \boldsymbol{x}_i$. Then, we update the candidate label confidence by conducting kNN aggregation in the projected feature space. An intermediate matrix $\mathbf{B} = [B_{ij}]_{m \times q}$ is specified by exploiting the labeling information of the k neareast neightbors of each instance in $\mathcal{D}'$. Specifically, each row vector $\boldsymbol{b}_i = [B_{i1}, ..., B_{iq}]$ of matrix $\mathbf{B}$ is calculated by:

$$\boldsymbol{b}_i = \alpha \boldsymbol{y}_i^\top + \sum_{\boldsymbol{x}'_j \in \mathcal{N}(\boldsymbol{x}'_i)} \boldsymbol{y}_j^\top \tag{13}$$

where $\mathcal{N}(\boldsymbol{x}'_i)$ denotes the k neareast neightbors of instance $\boldsymbol{x}'_i$ identified in $\mathcal{D}'$, $\boldsymbol{y}_j$ denotes the jth row of the labeling confidence matrix $\mathbf{Y}$, and parameter α is used to balance the importance of the labeling information of the instance itself and its k neareast neighbors. In this paper, α is set to the default value of 1.

Based on the intermediate matrix $\mathbf{B}$, we set the labeling confidence of labels outside the candidate label set to be 0 and then normalize the sum of the labeling confidence for each instance to 1:

$$\forall\, 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant q:\ Y'_{ij} = \begin{cases} \frac{B_{ij}}{\sum_{l_c \in S_i} B_{ic}}, & \text{if } l_j \in S_i \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

where $\mathbf{Y}'$ denotes the updated labeling confidence matrix.

Table 1 summarizes the complete procedure of Cenda. Firstly, the labeling confidence matrix is initialized (step 1) and the instance matrix is specified based on the assignment of training data set (step 2). After that, an iterative procedure alternating between confidence-based dependence maximization (steps 4-7) and kNN-based candidate label confidence updating (steps 8-13) is conducted. Here, the iterative procedure terminates if the projection matrix $\mathbf{P}$ does not change or the maximum number of iteration is reached.[2] Finally, the transformed lower-dimensional PL training set is constructed and ready for follow-up model training.

4 EXPERIMENTS

4.1 Experimental Setup

To evaluate the effectiveness of the proposed dimensionality reduction approach for partial label data, state-of-the-art partial label learning algorithms are coupled with Cenda for performance evaluation. To the best of our knowledge, Delin is the only available partial label dimensionality reduction approach which employs LDA to perform dimensionality reduction. For any partial label learning algorithm $\mathcal{A}$, its coupling versions with Cenda and Delin are denoted as $\mathcal{A}$-Cenda and $\mathcal{A}$-Delin respectively. The performance of $\mathcal{A}$-Cenda is compared against that of $\mathcal{A}$-Delin and $\mathcal{A}$ to verify the effectiveness of the proposed dimensionality reduction approach in improving the generalization ability of partial label learning system.

In this paper, five well-established partial label learning algorithms are utilized to instantiate $\mathcal{A}$ with suggested parameter configuration in respective literatures:

- Pl-knn [21]: An averaging-based partial label learning algorithm which makes prediction on unseen instance via weighted kNN voting strategy [suggested configuration: k=10].
- Pl-svm [30]: An identification-based partial label learning algorithm which makes prediction on unseen instance by maximizing the classification margin over candidate label set and non-candidate label set [suggested configuration: regularization parameter pool with $\{10^{-3}, \ldots, 10^{3}\}$].
- Pl-ecoc [49]: A transformation-based partial label learning algorithm which makes prediction on unseen instance by decomposing the PL learning problem into a group of binary learning problems via adapting the error-correcting output codes (ECOC) techniques [suggested configuration: ECOC coding length $\lceil 10 \cdot \log_2(q) \rceil$].
- Ipal [48]: Another instance-based partial label learning algorithm which makes prediction on unseen instance via adapting label propagation for graph-based disambiguation [suggested configuration: balancing parameter $\alpha = 0.95$].
- Sure [13]: A self-training based partial label learning algorithm which trains the desired model and performs pseudo-labeling jointly by solving a convex-concave optimization problem [suggested configuration: regularization parameters $\lambda = 0.3, \beta = 0.05$].

For Cenda, thr is employed to control the dimension of the projected lower-dimensional space according to Eq.(7). In this paper, the parameters for Cenda are set as: $thr = 0.999$, $\mu = 0.5$, $k = 8$. The parameter k for Delin is set to be $k = 8$. For the sake of fairness, the parameter d' for Delin is adjusted to ensure that both algorithms reduce the data to the same dimension.

In following subsections, comparative studies are conducted over both synthetic and real-world data sets. On each data set, ten-fold cross-validation is performed and the detailed experimental results(mean classification accuracy with standard deviation) are presented.

[2] In this paper, the maximum number of iterations is set to be 50 which suffices to yield stable performance for the proposed approach.

Table 3: Classification accuracy (mean±std) of each comparing algorithm on controlled synthetic data sets ($r \in \{1, 2, 3\}$). For partial label learning algorithm $\mathcal{A} \in \{$PL-KNN, PL-SVM, PL-ECOC, IPAL, SURE$\}$, the performance of $\mathcal{A}$-CENDA is compared against that of $\mathcal{A}$-DELIN and $\mathcal{A}$ where the best performance on each data set is shown in boldface.

Comparing Algorithms	mediamill	tmc2007	slashdot	amazon	DeliciousMIL	bookmark	sports	sector
				$r = 1$ (*one false positive label*)				
PL-KNN	0.640±0.032	0.401±0.012	0.162±0.021	0.027±0.029	0.033±0.029	0.192±0.023	0.307±0.019	0.013±0.007
PL-KNN-DELIN	0.693±0.014	0.686±0.011	0.637±0.023	0.644±0.014	0.441±0.019	**0.487±0.022**	0.873±0.015	0.525±0.028
PL-KNN-CENDA	**0.703±0.013**	**0.745±0.013**	**0.748±0.014**	**0.651±0.013**	**0.471±0.017**	0.462±0.014	**0.914±0.009**	**0.573±0.032**
PL-SVM	0.495±0.032	0.643±0.016	0.595±0.018	0.119±0.016	0.036±0.007	0.283±0.022	0.666±0.009	0.073±0.012
PL-SVM-DELIN	0.594±0.016	0.698±0.012	0.690±0.019	0.645±0.011	0.444±0.013	**0.530±0.038**	0.835±0.012	0.520±0.023
PL-SVM-CENDA	**0.614±0.012**	**0.740±0.014**	**0.763±0.014**	**0.652±0.014**	**0.456±0.012**	0.504±0.044	**0.870±0.013**	**0.561±0.036**
PL-ECOC	0.592±0.023	0.635±0.014	0.523±0.013	0.063±0.021	0.072±0.033	0.366±0.018	0.697±0.020	0.062±0.021
PL-ECOC-DELIN	0.594±0.027	0.707±0.013	0.665±0.022	0.644±0.022	0.441±0.022	**0.510±0.049**	0.874±0.016	0.529±0.046
PL-ECOC-CENDA	**0.687±0.011**	**0.761±0.011**	**0.762±0.011**	**0.651±0.021**	**0.494±0.011**	0.488±0.027	**0.916±0.006**	**0.563±0.018**
IPAL	0.635±0.021	0.588±0.014	0.413±0.013	0.105±0.011	0.072±0.021	0.311±0.022	0.905±0.009	0.152±0.027
IPAL-DELIN	0.650±0.023	0.649±0.012	0.634±0.017	0.644±0.018	0.443±0.008	**0.509±0.043**	0.890±0.011	0.530±0.022
IPAL-CENDA	**0.671±0.012**	**0.705±0.010**	**0.752±0.011**	**0.652±0.014**	**0.495±0.011**	0.477±0.021	**0.916±0.005**	**0.564±0.017**
SURE	0.689±0.018	0.647±0.016	0.577±0.026	0.153±0.021	0.116±0.028	0.390±0.013	0.755±0.031	0.120±0.029
SURE-DELIN	0.708±0.017	0.708±0.013	0.673±0.007	0.643±0.016	0.441±0.032	**0.524±0.022**	0.877±0.013	0.527±0.019
SURE-CENDA	**0.718±0.010**	**0.758±0.013**	**0.763±0.011**	**0.652±0.015**	**0.504±0.021**	0.498±0.031	**0.916±0.007**	**0.563±0.024**
				$r = 2$ (*two false positive labels*)				
PL-KNN	0.633±0.024	0.408±0.021	0.160±0.020	0.021±0.009	0.033±0.012	0.164±0.032	0.296±0.013	0.015±0.007
PL-KNN-DELIN	0.668±0.016	0.678±0.012	0.606±0.018	0.481±0.022	0.216±0.022	**0.463±0.012**	0.847±0.020	**0.388±0.223**
PL-KNN-CENDA	**0.684±0.009**	**0.725±0.007**	**0.668±0.021**	**0.493±0.020**	**0.273±0.013**	0.400±0.021	**0.870±0.015**	0.387±0.162
PL-SVM	0.494±0.031	0.626±0.023	0.575±0.022	0.073±0.012	0.033±0.011	0.259±0.019	0.646±0.016	0.054±0.011
PL-SVM-DELIN	0.596±0.017	0.694±0.021	0.656±0.022	0.483±0.032	0.219±0.028	**0.501±0.032**	0.808±0.024	0.376±0.021
PL-SVM-CENDA	**0.610±0.016**	**0.727±0.013**	**0.687±0.013**	**0.490±0.014**	**0.261±0.009**	0.440±0.022	**0.839±0.016**	**0.384±0.016**
PL-ECOC	0.522±0.026	0.574±0.017	0.433±0.027	0.049±0.013	0.058±0.022	0.288±0.035	0.601±0.037	0.033±0.013
PL-ECOC-DELIN	0.589±0.019	0.688±0.020	0.625±0.023	0.481±0.012	0.218±0.017	**0.490±0.015**	0.842±0.021	**0.386±0.022**
PL-ECOC-CENDA	**0.630±0.013**	**0.737±0.015**	**0.683±0.012**	**0.492±0.007**	**0.254±0.024**	0.421±0.017	**0.869±0.017**	0.385±0.016
IPAL	0.593±0.018	0.588±0.013	0.400±0.015	0.086±0.010	0.047±0.021	0.310±0.018	**0.902±0.010**	0.137±0.009
IPAL-DELIN	0.622±0.003	0.642±0.015	0.604±0.032	0.483±0.012	0.216±0.012	**0.490±0.033**	0.862±0.007	0.388±0.026
IPAL-CENDA	**0.634±0.012**	**0.690±0.012**	**0.675±0.012**	**0.491±0.011**	**0.296±0.013**	0.411±0.019	0.888±0.010	**0.391±0.015**
SURE	0.688±0.024	0.640±0.027	0.574±0.025	0.102±0.014	0.115±0.032	0.374±0.011	0.711±0.022	0.107±0.038
SURE-DELIN	0.695±0.013	0.721±0.009	0.645±0.021	0.483±0.016	0.216±0.021	**0.504±0.041**	0.845±0.016	**0.388±0.023**
SURE-CENDA	**0.711±0.023**	**0.742±0.013**	**0.686±0.012**	**0.493±0.013**	**0.272±0.014**	0.435±0.023	**0.873±0.007**	0.387±0.017
				$r = 3$ (*three false positive labels*)				
PL-KNN	0.598±0.012	0.366±0.014	0.168±0.018	0.025±0.006	0.041±0.012	0.141±0.022	0.290±0.016	0.018±0.014
PL-KNN-DELIN	**0.648±0.023**	0.655±0.012	0.576±0.016	0.365±0.012	0.156±0.015	**0.421±0.023**	0.822±0.011	0.296±0.015
PL-KNN-CENDA	0.641±0.016	**0.691±0.008**	**0.592±0.007**	**0.375±0.013**	**0.224±0.013**	0.371±0.012	**0.858±0.011**	**0.306±0.008**
PL-SVM	0.471±0.039	0.619±0.015	0.562±0.038	0.065±0.025	0.038±0.010	0.247±0.015	0.603±0.017	0.050±0.026
PL-SVM-DELIN	0.600±0.011	0.682±0.021	**0.623±0.035**	0.364±0.027	0.156±0.012	**0.465±0.017**	0.789±0.020	0.295±0.028
PL-SVM-CENDA	**0.601±0.003**	**0.708±0.015**	0.605±0.028	**0.375±0.018**	**0.198±0.020**	0.409±0.012	**0.828±0.015**	**0.302±0.027**
PL-ECOC	0.101±0.024	0.569±0.013	0.374±0.026	0.029±0.013	0.066±0.023	0.203±0.032	0.491±0.023	0.019±0.012
PL-ECOC-DELIN	**0.223±0.113**	0.622±0.023	0.582±0.015	0.367±0.016	0.157±0.015	**0.428±0.026**	0.802±0.021	0.296±0.018
PL-ECOC-CENDA	0.170±0.026	**0.669±0.012**	**0.598±0.012**	**0.375±0.013**	**0.202±0.013**	0.389±0.028	**0.843±0.012**	**0.306±0.011**
IPAL	0.525±0.012	0.555±0.019	0.373±0.010	0.084±0.014	0.041±0.012	0.293±0.017	0.862±0.019	0.142±0.010
IPAL-DELIN	0.571±0.012	0.631±0.013	0.569±0.012	0.364±0.011	0.156±0.013	**0.428±0.012**	0.839±0.012	0.298±0.011
IPAL-CENDA	**0.579±0.011**	**0.670±0.015**	**0.582±0.014**	**0.375±0.012**	**0.248±0.010**	0.372±0.018	**0.876±0.012**	**0.309±0.012**
SURE	0.668±0.024	0.628±0.016	0.541±0.020	0.072±0.017	0.671±0.017	0.370±0.023	0.671±0.009	0.095±0.023
SURE-DELIN	0.702±0.011	0.690±0.013	**0.608±0.038**	0.364±0.012	0.822±0.012	**0.463±0.029**	0.822±0.009	0.303±0.033
SURE-CENDA	**0.705±0.012**	**0.726±0.015**	0.606±0.034	**0.375±0.011**	**0.858±0.013**	0.406±0.024	**0.858±0.011**	**0.306±0.046**

4.2 Synthetic Data Sets

Synthetic partial label data sets are generated from multi-class data sets with controlling parameter r according to the widely-used strategy [8, 10, 11, 16, 27, 45, 49]. Specifically, r controls the number of *false positive* labels in the candidate label set of PL examples. Given a multi-class example $(\boldsymbol{x}_i, y_i)$, one PL example $(\boldsymbol{x}_i, S_i)$ can be generated by adding r false positive labels $\Delta_r \subseteq \mathcal{Y} \setminus \{y_i\}, |\Delta_r| = r$ randomly into S_i along with the ground-truth label y_i, i.e. $S_i = \Delta_r \bigcup \{y_i\}$.

Table 2 summarizes characteristics of the synthetic data sets ($r \in \{1, 2, 3\}$) which are roughly ordered according to the number of features in the feature space.[3] Accordingly, detailed experimental results of each comparing algorithm over various synthetic

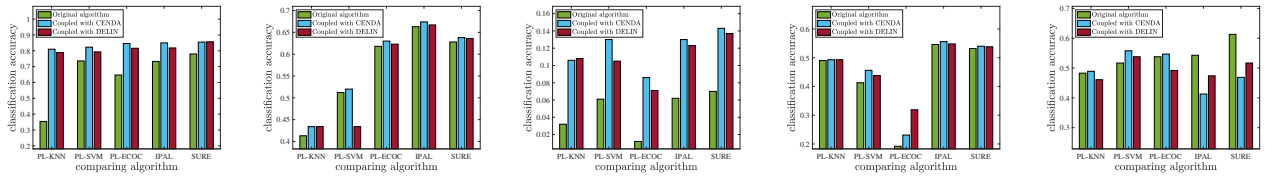
[3] In Table 2, most multi-class data sets are derived from multi-label benchmark data sets [57] by retaining examples with only one relevant label.

Table 4: Win/tie/loss counts (pairwise t-test at 0.05 significance level) between $\mathcal{A}$-CENDA and $\mathcal{A}$, $\mathcal{A}$-DELIN in terms of different number of false positive labels ($r = 1, 2, 3$).

	$\mathcal{A}$-CENDA against $\mathcal{A}$					$\mathcal{A}$-CENDA against $\mathcal{A}$-DELIN				
	$\mathcal{A}$=PL-KNN	$\mathcal{A}$= PL-SVM	$\mathcal{A}$=PL-ECOC	$\mathcal{A}$=IPAL	$\mathcal{A}$=SURE	$\mathcal{A}$=PL-KNN	$\mathcal{A}$= PL-SVM	$\mathcal{A}$= PL-ECOC	$\mathcal{A}$=IPAL	$\mathcal{A}$=SURE
$r = 1$	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	7/0/1	7/0/1	7/0/1	7/0/1	7/0/1
$r = 2$	8/0/0	8/0/0	8/0/0	7/0/1	8/0/0	6/1/1	7/0/1	6/1/1	7/0/1	6/1/1
$r = 3$	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	6/0/2	5/1/2	6/0/2	7/0/1	4/3/1
In Total	**24/0/0**	**24/0/0**	**24/0/0**	**23/0/1**	**24/0/0**	**19/1/4**	**19/1/4**	**19/1/4**	**21/0/3**	**17/4/3**

Table 5: Characteristics of the real-world experimental data sets.

Data Set	# Examples	# Features	# Class Labels	average # Candidate Labels	Task Domain
Lost	1,122	108	16	2.23	*automatic face naming* [11]
Yahoo! News	22,991	163	219	1.91	*automatic face naming* [19]
FG-NET	1,002	262	78	7.48	*facial age estimation* [31]
Soccer Player	17,472	279	171	2.09	*automatic face naming* [47]
Mirflickr	2,780	1,536	14	2.76	*web image classification* [20]



(a) Lost (b) Yahoo! News (c) FG-NET (d) Soccer Player (e) Mirflickr

Figure 1: Classification accuracy of each partial label learning algorithm on real-world data sets (*green bar*: original algorithm; *blue bar*: coupled with CENDA; *red bar*: coupled with DELIN).

Table 6: Win/tie/loss statistics (pairwise t-test at 0.05 significance level) between $\mathcal{A}$-CENDA and $\mathcal{A}$, $\mathcal{A}$-DELIN on real-world data sets.

Data Set	$\mathcal{A}$-CENDA against $\mathcal{A}$					$\mathcal{A}$-CENDA against $\mathcal{A}$-DELIN				
	$\mathcal{A}$=PL-KNN	$\mathcal{A}$= PL-SVM	$\mathcal{A}$=PL-ECOC	$\mathcal{A}$=IPAL	$\mathcal{A}$=SURE	$\mathcal{A}$=PL-KNN	$\mathcal{A}$= PL-SVM	$\mathcal{A}$= PL-ECOC	$\mathcal{A}$=IPAL	$\mathcal{A}$=SURE
Lost	win	win	win	win	win	win	win	win	win	tie
Yahoo! News	win	win	win	win	win	tie	win	win	win	tie
FG-NET	win	win	win	win	win	tie	win	win	win	win
Soccer Player	tie	win	win	win	win	tie	win	loss	win	tie
Mirflickr	win	win	win	loss	loss	win	win	win	loss	loss
In Total	**4/1/0**	**5/0/0**	**5/0/0**	**4/0/1**	**4/0/1**	**2/3/0**	**5/0/0**	**4/0/1**	**4/0/1**	**1/3/1**

data sets are reported in Table 3. For partial label learning algorithm $\mathcal{A} \in$ {PL-KNN, PL-SVM, PL-ECOC, IPAL, SURE}, $\mathcal{A}$-CENDA is compared against both $\mathcal{A}$ and $\mathcal{A}$-DELIN where the best classification accuracy is shown in boldface. Furthermore, pairwise t-test at 0.05 significance level is conducted to show whether the performance difference between $\mathcal{A}$-CENDA and $\mathcal{A}$-DELIN or between $\mathcal{A}$-CENDA and $\mathcal{A}$ is significant, where the resulting win/tie/loss counts are reported in Table 4. Based on the above experimental results, we can draw the following conclusions:

- Among all the 120 cases (8 data sets $\times$ 3 settings of $r \times 5$ PL learning algorithms; Table 3), $\mathcal{A}$-CENDA achieves better performance than $\mathcal{A}$ in 99.2% cases after coupling with the proposed dimensionality reduction approach. The only exception is on `sports` ($r = 2$) which corresponds to the synthetic data set with largest number of examples. Furthermore, $\mathcal{A}$-CENDA achieves better performance than $\mathcal{A}$-DELIN in 95 out of 120 cases.
- For PL-KNN, PL-SVM and PL-ECOC, their performance have been significantly improved by CENDA in all cases (Table 4). On the five data sets with more than 1,300 features (i.e.

amazon, DeliciousMIL, bookmark, sports and sector), out of the 45 statistical comparisons (3 PL learning algorithms x 5 data sets x 3 settings of r), the classification accuracy has been improved with CENDA by more than **0.20** in 35 cases. These results indicate that the benefits brought by CENDA are rather noticeable under the challenging circumstances of high dimensionality.

- On the two data sets amazon and DeliciousMIL with least number of examples, the classification accuracy for IPAL and SURE has been improved with CENDA by more than **0.35**, **0.15** and **0.20** for r=1, 2 and 3 respectively. These results indicate that the benefits brought by CENDA are rather noticeable under the challenging circumstances of insufficient training examples.

4.3 Real-World Data Sets

In addition to synthetic data sets, a number of real-world partial label data sets have been collected from several task domains including FG-NET [31] for facial age estimation, Lost [11], Soccer Player [47] and Yahoo! News [19] for automatic face naming from images or videos, Mirflickr [20] for web image classification.[4]

In the data set of *facial age estimation*, an example consists of a human face with landmarks and a set of candidate ages annotated by crowdsourced labelers. In the data sets of *automatic face naming*, an example consists of face image cropped from an image or video frame and a set of candidate names extracted from the associated captions or subtitles. In the data set of *web image classification*, an example consists of an image and a set of candidate annotations extracted from the web environment. Table 5 summarizes characteristics of the real-world partial label data sets.

Fig. 1 illustrates the classification accuracy of each partial label learning algorithm before and after employing dimensionality reduction techniques (CENDA or DELIN) on each real-world data set. In addition, we conduct pairwise t-test at 0.05 significance level to validate whether the performance differences between $\mathcal{A}$-CENDA and $\mathcal{A}$, $\mathcal{A}$-DELIN are significant. The win/tie/loss statistics are reported in Table 6.

From the reported results on real-world data sets, it's impressive to observe that:

- Out of the 25 statistical comparisons (5 algorithms $\times$ 5 data sets), the predictive performance of $\mathcal{A}$-CENDA is significantly superior to that of $\mathcal{A}$ in 22 cases. There are only two losses of $\mathcal{A}$-CENDA against $\mathcal{A}$ on Mirflickr with $\mathcal{A} \in$ {IPAL, SURE}.
- As shown in Fig. 1(c), the performance improvement of $\mathcal{A}$-CENDA against $\mathcal{A}$ is rather pronounced on the FG-NET data set, which is challenging to handle with least number of examples but large average number of candidate labels. It is worth noting that the classification accuracy of each partial label learning algorithm has at least been doubled on FG-NET by coupling with CENDA. These impressive results indicate that CENDA could bring rather noticeable improvements even under the challenging circumstances of insufficient training examples and high rate of false positive labels.
- In most tasks, $\mathcal{A}$-CENDA achieves superior or at least statistically comparable performance against $\mathcal{A}$-DELIN. The three losses of $\mathcal{A}$-CENDA against $\mathcal{A}$-DELIN take place on Soccer Player with $\mathcal{A}$=PL-ECOC and Mirflickr with $\mathcal{A} \in$ {IPAL, SURE} .

[4]Data available at: http://palm.seu.edu.cn/zhangml/Resources. htm#partial_data

4.4 Sensitivity Analysis

As shown in Table 1, thr serves as a crucial parameter for CENDA which controls the number of retained features after dimensionality reduction. Table 7 shows the trend of predictive accuracy of partial label algorithms coupled with CENDA as the parameter thr varies in $\{0.9, 0.99, 0.999\}$. For comparison, DELIN is employed to reduce the partial label data to the same dimensionality to compare the effects of different methods. The experimental results are reported in Table 7 where the better result between $\mathcal{A}$-CENDA and $\mathcal{A}$-DELIN is shown in boldface. As shown in Table 7, the performance of each partial label learning algorithm coupled with CENDA fluctuates moderately as the value of thr changes. Specifically, there is no single value of thr which can consistently lead to the best performance. Therefore, further performance improvement can be achieved by fine-tuning the value of thr for different data sets and partial label learning algorithms, although 0.999 is a reasonable default setting for thr in this paper.

In addition to thr, μ (trade-off parameter in Eq.(9)) and k (# nearest neighbors) also serve as important parameters for CENDA. Fig. 2 illustrates how the performance of each partial label learning algorithm coupled with CENDA changes respectively as μ increases from 0.2 to 0.8 with step-size 0.1 and k increases from 3 to 10 with step-size 1 on four data sets. As shown in Fig. 2, the performance of each partial label learning algorithm coupled with CENDA is relatively stable as the value of μ or k varies. Therefore, the value of μ and k is fixed to be 0.5 and 8 respectively for comparative studies in this paper.

5 CONCLUSION

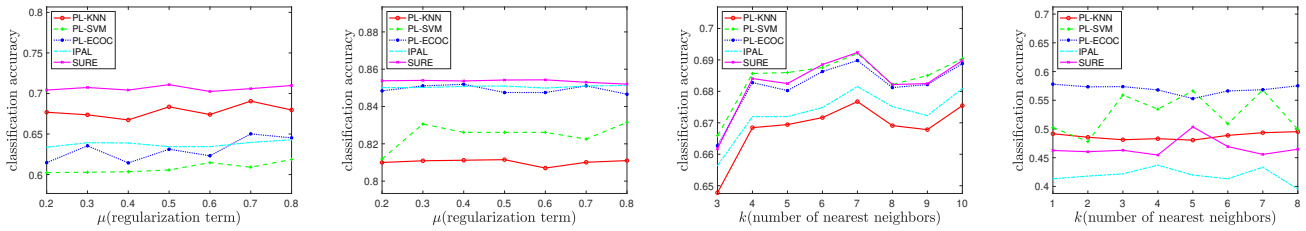
In this paper, we propose a novel method to enhance partial label learning algorithms via manipulating the feature space by dimensionality reduction. The proposed method performs dimensionality reduction by maximizing the dependence between projected feature information and confidence-based labeling information iteratively. In each iteration, the projection matrix is identified by solving an generalized eigenvalue problem derived from the adapted Hilbert-Schmidt Independence Criterion and the confidences of candidate labels are updated by conducting kNN aggregation in the projected feature space. Comprehensive experimental studies over synthetic and real-world data sets show that CENDA is an effective preprocessing method to improve the performance of well-established partial label learning algorithms.

It is worth mentioning that the labeling confidence matrix $\mathbf{Y}$ derived from CENDA which is simply ignored in the follow-up partial label training procedure. Proper utilization of this derived side information may bring further improvement of predictive performance for specific partial label learning algorithms. Apart from Hilbert-Schmidt Independence Criterion, other sophisticated dimensionality reduction methods are expected to be appropriately introduced to facilitate partial label learning.

Table 7: Classification accuracy of $\mathcal{A}$-CENDA ($\mathcal{A}$ ∈{PL-KNN, PL-SVM, PL-ECOC, IPAL, SURE}) changes as the number of retained features varies with *thr* taking value within {0.9, 0.99, 0.999}). On each data set, the better performance between $\mathcal{A}$-CENDA and $\mathcal{A}$-DELIN is shown in boldface. For reference purpose, the classification accuracy of $\mathcal{A}$ on the original feature space is also shown in the lower part of the table. (Yahoo! News and Soccer Player are abbreviated as LNY and Spd respectively)

Data Set	*thr*	Retained Features	PL-KNN-DELIN		PL-SVM-DELIN		PL-ECOC-DELIN		IPAL-DELIN		SURE-DELIN	
			CENDA	DELIN	CENDA	DELIN	CENDA	DELIN	CENDA	DELIN	CENDA	DELIN
Lost	0.9	6	**0.718±0.033**	0.656±0.021	**0.719±0.024**	0.614±0.019	**0.739±0.039**	0.658±0.016	**0.718±0.012**	0.634±0.017	**0.732±0.019**	0.665±0.021
	0.99	11	**0.804±0.027**	0.782±0.015	**0.815±0.022**	0.777±0.026	**0.828±0.023**	0.816±0.039	**0.839±0.023**	0.822±0.021	**0.842±0.027**	0.803±0.026
	0.999	13	**0.810±0.012**	0.789±0.006	**0.823±0.013**	0.793±0.021	**0.846±0.016**	0.816±0.029	**0.850±0.014**	0.818±0.011	0.855±0.013	**0.857±0.012**
LYN	0.9	5	**0.423±0.016**	0.306±0.008	**0.425±0.006**	0.244±0.007	**0.435±0.009**	0.263±0.008	**0.379±0.010**	0.257±0.008	**0.476±0.010**	0.342±0.002
	0.99	63	**0.506±0.010**	**0.506±0.012**	**0.506±0.007**	0.503±0.009	0.645±0.012	**0.646±0.007**	**0.667±0.006**	0.664±0.008	**0.646±0.008**	**0.646±0.004**
	0.999	120	**0.434±0.008**	**0.434±0.011**	**0.520±0.009**	0.434±0.006	**0.630±0.006**	0.623±0.003	**0.674±0.003**	0.667±0.006	**0.638±0.007**	0.636±0.007
FG-NET	0.9	15	0.133±0.022	**0.150±0.030**	0.094±0.028	**0.119±0.036**	**0.101±0.035**	0.094±0.015	**0.133±0.024**	0.126±0.023	**0.135±0.022**	0.124±0.017
	0.99	34	**0.118±0.014**	**0.118±0.027**	**0.110±0.035**	0.104±0.037	**0.064±0.016**	0.071±0.017	0.125±0.017	**0.133±0.012**	0.129±0.012	**0.146±0.022**
	0.999	40	0.106±0.021	**0.108±0.036**	**0.130±0.029**	0.105±0.040	**0.086±0.023**	0.071±0.022	**0.130±0.012**	0.123±0.014	**0.143±0.014**	0.137±0.012
Spd	0.9	24	**0.504±0.011**	0.499±0.022	**0.379±0.062**	0.371±0.016	**0.079±0.013**	0.075±0.043	**0.529±0.023**	0.527±0.020	**0.527±0.022**	0.519±0.020
	0.99	108	**0.497±0.012**	0.495±0.023	**0.457±0.054**	0.437±0.022	0.285±0.032	**0.295±0.031**	**0.558±0.012**	0.553±0.014	**0.542±0.009**	0.537±0.020
	0.999	151	**0.493±0.016**	**0.493±0.034**	**0.456±0.072**	0.438±0.023	0.231±0.012	**0.319±0.022**	**0.556±0.032**	0.548±0.023	**0.540±0.013**	0.538±0.020
Mirflickr	0.9	5	**0.580±0.017**	0.578±0.029	**0.513±0.032**	0.439±0.019	**0.479±0.022**	0.468±0.013	0.472±0.018	**0.523±0.031**	**0.602±0.013**	0.588±0.033
	0.99	11	**0.494±0.011**	0.493±0.033	**0.545±0.017**	0.524±0.015	**0.576±0.013**	0.497±0.023	0.405±0.020	**0.497±0.022**	0.500±0.017	**0.577±0.023**
	0.999	13	**0.489±0.009**	0.461±0.034	**0.558±0.012**	0.538±0.029	**0.547±0.013**	0.492±0.016	0.413±0.016	**0.474±0.026**	0.469±0.011	**0.517±0.011**

		Original Features	PL-KNN	PL-SVM	PL-ECOC	IPAL	SURE
Lost	-	108	0.354±0.012	0.736±0.014	0.647±0.039	0.733±0.031	0.780±0.021
LYN	-	163	0.413±0.023	0.512±0.011	0.618±0.017	0.663±0.012	0.628±0.007
FG-NET	-	262	0.032±0.017	0.061±0.013	0.012±0.012	0.062±0.026	0.070±0.014
Spd	-	279	0.490±0.016	0.413±0.025	0.192±0.033	0.546±0.018	0.532±0.016
Mirflickr	-	1,536	0.479±0.012	0.517±0.062	0.538±0.016	0.543±0.066	0.613±0.022



(a) mediamill(r = 2) (b) Lost (c) slashdot(r = 2) (d) Mirflickr

Figure 2: Trend of classification accuracy of $\mathcal{A}$-CENDA ($\mathcal{A}$ ∈{PL-KNN, PL-SVM, PL-ECOC, IPAL SURE}). The regularization term (i.e. μ) increases from 0.2 to 0.8 with step-size 0.1 in (a) synthetic data set `mediamill` (r = 2) and (b) real-world data set `Lost`; the number of nearest neighbors used for updating confidences of candidate labels (i.e. k) increases from 3 to 10 with step-size 1 in (c) synthetic data set `slashdot` (r = 2) and (d) real-world data set `Mirflickr`.

REFERENCES

[1] K. Altun and B. Barshan. 2010. Human activity recognition using inertial/magnetic sensor units. In *Proceedings of the 1st International Conference on Human Behavior Understanding*. Istanbul, Turkey, 38–51.

[2] J. Amores. 2013. Multiple instance classification: Review, taxonomy and comparative study. *Artificial intelligence* 201 (2013), 81–105.

[3] S. Boyd, S. P. Boyd, and L. Vandenberghe. 2004. *Convex optimization*. New York: Cambridge University Press.

[4] F. Briggs, X. Z. Fern, and R. Raich. 2012. Rank-loss support instance machines for MIML instance annotation. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Beijing, China, 534–542.

[5] M.-A. Carbonneaua, V. Cheplyginabc, E. Granger, and G. Gagnon. 2018. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition* 77 (2018), 329–353.

[6] J. Chai, I. W. Tsang, and W. Chen. 2020. Large margin partial label machine. *IEEE Transactions on Neural Networks and Learning Systems* 31, 7 (2020), 2594–2608.

[7] B. Chang, U. Kruger, R. Kustra, and J. Zhang. 2013. Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment. In *Proceedings of the 30th International Conference on Machine Learning*. Atlanta, Georgia, 316–324.

[8] C.-H. Chen, V. M. Patel, and R. Chellappa. 2018. Learning from ambiguously labeled face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 7 (2018), 1653–1667.

[9] J.-H. Chen, S.-W. Ji, B Ceran, Q Li, M.-R. Wu, and J.-P. Ye. 2008. Learning subspace kernels for classification. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada*. 106–114.

[10] Y.-C. Chen, V. M. Patel, R. Chellappa, and P. J. Phillips. 2014. Ambiguously labeled learning using dictionaries. *IEEE Transactios on Information Forensics and Security* 9, 12 (2014), 2076–2088.

[11] T. Cour, B. Sapp, and B. Taskar. 2011. Learning from partial labels. *Journal of Machine Learning Research* 12 (2011), 1501–1536.

[12] D. Dheeru and E. Karra Taniskidou. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[13] L. Feng and B. An. 2019. Partial label learning with self-guided retraining. In *Proceedings of the 13th AAAI Conference on Artificial Intelligence. Honolulu, Hawaii.* 3542–3549.

[14] M. J. Gangeh, H. Zarkoob, and A. Ghodsi. 2017. Fast and scalable feature selection for gene expression data using hilbert-schmidt independence criterion. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 14, 1 (2017), 167–181.

[15] B. Ghojogh, F. Karray, and M. Crowley. 2019. Eigenvalue and generalized eigenvalue problems: Tutorial. *ArXiv:1903.11240* (2019).

[16] C. Gong, T. Liu, Y. Tang, J. Yang, J. Yang, and D. Tao. 2018. A regularization approach for instance-based superset label learning. *IEEE Transactions on Cybernetics* 48, 3 (2018), 967–978.

[17] D. Greenfeld and U. Shalit. 2020. Robust learning with the hilbert-schmidt independence criterion. In *Proceedings of the 37th International Conference on Machine Learning, Virtual Event*. 3759–3768.

[18] A. Gretton, O.r Bousquet, A. Smola, and B. Schölkopf. 2005. Measuring statistical dependence with Hilbert-Schmidt norms. In *Proceedings of the 16th International Conference on Algorithmic Learning Theory*. Berlin, Heidelberg, 63–77.

[19] M. Guillaumin, J. Verbeek, and C. Schmid. 2010. Multiple instance metric learning from automatically labeled bags of faces. In *Lecture Notes in Computer Science 6311*, K. Daniilidis, P. Maragos, and N. Paragios (Eds.). Springer, Berlin, 634–647.

[20] M. J. Huiskes and M. S. Lew. 2008. The MIR Flickr retrieval evaluation. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*. Vancouver, Canada, 39–43.

[21] E. Hüllermeier and J. Beringer. 2006. Learning from ambiguously labeled examples. *Intelligent Data Analysis* 10, 5 (2006), 419–439.

[22] S.-W. Ji and J.-P. Ye. 2009. Linear dimensionality reduction for multi-label classification. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA*.

[23] L. Jie and F. Orabona. 2010. Learning from candidate labeling sets. In *Advances in Neural Information Processing Systems 23*, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta (Eds.). MIT Press, Cambridge, MA, 1504–1512.

[24] R. Jin and Z. Ghahramani. 2003. Learning with multiple labels. In *Advances in Neural Information Processing Systems 15*, S. Becker, S. Thrun, and K. Obermayer (Eds.). MIT Press, Cambridge, MA, 897–904.

[25] I. Katakis, G. Tsoumakas, and I. Vlahavas. 2008. Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD 2008 Discovery Challenge*. Antwerp, Belgium.

[26] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney. 2009. Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics* 6, 1 (2009), 29–123.

[27] L. Liu and T. Dietterich. 2012. A conditional multinomial mixture model for superset label learning. In *Advances in Neural Information Processing Systems 25*, P. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). MIT Press, Cambridge, MA, 557–565.

[28] J. Lv, M. Xu, L. Feng, G. Niu, X. Geng, and M. Sugiyama. 2020. Progressive identification of true labels for partial-label learning. In *Proceedings of the 37th International Conference on Machine Learning*. Virtual Conference, 6500–6510.

[29] G. Lyu, S. Feng, T. Wang, and C. Lang. 2021, in press. A self-paced regularization framework for partial-label learning. *IEEE Transactions on Cybernetics* (2021, in press).

[30] N. Nguyen and R. Caruana. 2008. Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Las Vegas, NV, 381–389.

[31] G. Panis and A. Lanitis. 2015. An overview of research activities in facial age estimation using the FG-NET aging database. In *Lecture Notes in Computer Science 8926*, C. Rother L. Agapito, M. M. Bronstein (Ed.). Springer, Berlin, 737–750.

[32] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. Merschmann. 2018. Categorizing feature selection methods for multi-label classification. *Artificial Intelligence Review* 49, 1 (2018), 57–78.

[33] B.-Y. Qian and I. Davidson. 2010. Semi-supervised dimension reduction for multi-label classification. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. Atlanta, GA, 569–574.

[34] X. Ren, W. He, M. Qu, C. R. Voss, H. Ji, and J. Han. 2016. Label noise reduction in entity typing by heterogeneous partial-label embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco, CA, 1825–1834.

[35] J. D. M. Rennie and R. Rifkin. 2001. *Improving multiclass text classification with the support vector machines*. Technical Report AIM-2001-026. Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[36] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, and A. W. M. Smeulders. 2006. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proceedings of the 14th ACM International Conference on Multimedia*. Santa Barbara, CA, 421–430.

[37] H. Soleimani and D. J. Miller. 2016. Semi-supervised multi-label topic models for document classification and sentence labeling. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. Indianapolis, IN, 105–114.

[38] A. N. Srivastava and B. Zane-Ulman. 2005. Discovering recurring anomalies in text reports regarding complex space systems. In *Proceedings of the 26th IEEE Aerospace Conference*. Big Sky, MT, 3853–3862.

[39] K. Sun, Z. Min, and J. Wang. 2019. PP-PLL: Probability propagation for partial label learning. In *Lecture Notes in Computer Science 11907*, U. Brefeld, E. Fromont, A. Hotho, A. Knobbe, M. Maathuis, and C. Robardet (Eds.). Springer, Berlin, 123–137.

[40] L. Sun, S.-W. Ji, and J.-P. Ye. 2013. *Multi-label Dimensionality Reduction*. CRC Press.

[41] C.-Z. Tang and M.-L. Zhang. 2017. Confidence-rated discriminative partial label learning. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*. San Francisco, CA, 2611–2617.

[42] D.-B. Wang, L. Li, and M.-L. Zhang. 2019. Adaptive graph guided disambiguation for partial label learning. In *Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Anchorage, AK, 83–91.

[43] J.-H. Wu and M.-L. Zhang. 2019. Disambiguation enabled linear discriminant analysis for partial label dimensionality reduction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, 416–424.

[44] M. Xu and L.-Z. Guo. 2021. Learning from group supervision: the impact of supervision deficiency on multi-label learning. *Science China Information Sciences* 64, 3 (2021), 1–13.

[45] F. Yu and M.-L. Zhang. 2017. Maximum margin partial label learning. *Machine Learning* 106, 4 (2017), 573–593.

[46] K. Yu, S. Yu, and V. Tresp. 2005. Multi-label informed latent semantic indexing. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Salvador, Brazil, 258–265.

[47] Z. Zeng, S. Xiao, K. Jia, T.-H. Chan, S. Gao, D. Xu, and Y. Ma. 2013. Learning by associating ambiguously labeled images. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Portland, OR, 708–715.

[48] M.-L. Zhang and F. Yu. 2015. Solving the partial label learning problem: An instance-based approach. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*. Buenos Aires, Argentina, 4048–4054.

[49] M.-L. Zhang, F. Yu, and C.-Z. Tang. 2017. Disambiguation-free partial label learning. *IEEE Transactions on Knowledge and Data Engineering* 29, 10 (2017), 2155–2167.

[50] M.-L. Zhang and Z.-H. Zhou. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition* 40, 7 (2007), 2038–2048.

[51] M.-L. Zhang and Z.-H. Zhou. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2014), 1819–1837.

[52] Y. Zhang and Z.-H. Zhou. 2010. Multilabel dimensionality reduction via dependence maximization. *ACM Transactions on Knowledge Discovery from Data* 4, 3 (2010), 1–21.

[53] S. Zhao, P. Ni, H. Chen, C. Li, and Z. Dai. 2021, in press. Partial label learning via conditional-label-aware disambiguation. *Journal of Computer Science and Technology* (2021, in press).

[54] D. Zhou, Z. Zhang, M.-L. Zhang, and Y. He. 2018. Weakly supervised POS tagging without disambiguation. *ACM Transactions on Asian and Low-Resource Language Information Processing* 17, 4 (2018), Article 35.

[55] X.-Y. Zhou and M. Belkin. 2014. Semi-supervised learning. In *Academic Press Library in Signal Processing*. Vol. 1. Elsevier, 1239–1269.

[56] Z.-H. Zhou. 2018. A brief introduction to weakly supervised learning. *National Science Review* 5, 1 (2018), 44–53.

[57] Z.-H. Zhou and M.-L. Zhang. 2017. Multi-label learning. In *Encyclopedia of Machine Learning and Data Mining*, C. Sammut and G. I. Webb (Eds.). Springer, Berlin, 875–881.

[58] X.-J. Zhu and A. B. Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 3, 1 (2009), 1–130.