CAIIAM “知识智能及其产业应用论坛”

偏标记学习的研究
(Research on Partial Label Learning)

Min-Ling Zhang (张敏灵)

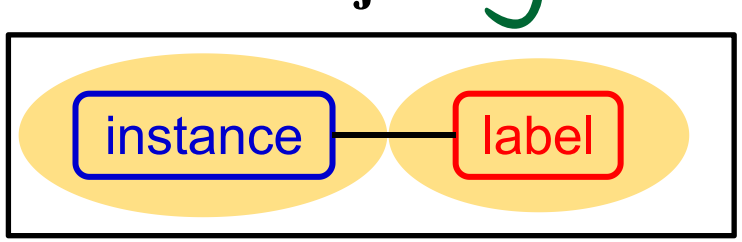
School of Computer Science and Engineering,

MOE Key Lab. of Computer Network & Information Integration

Southeast University, China

Dec. 1, Suzhou

Traditional Supervised Learning

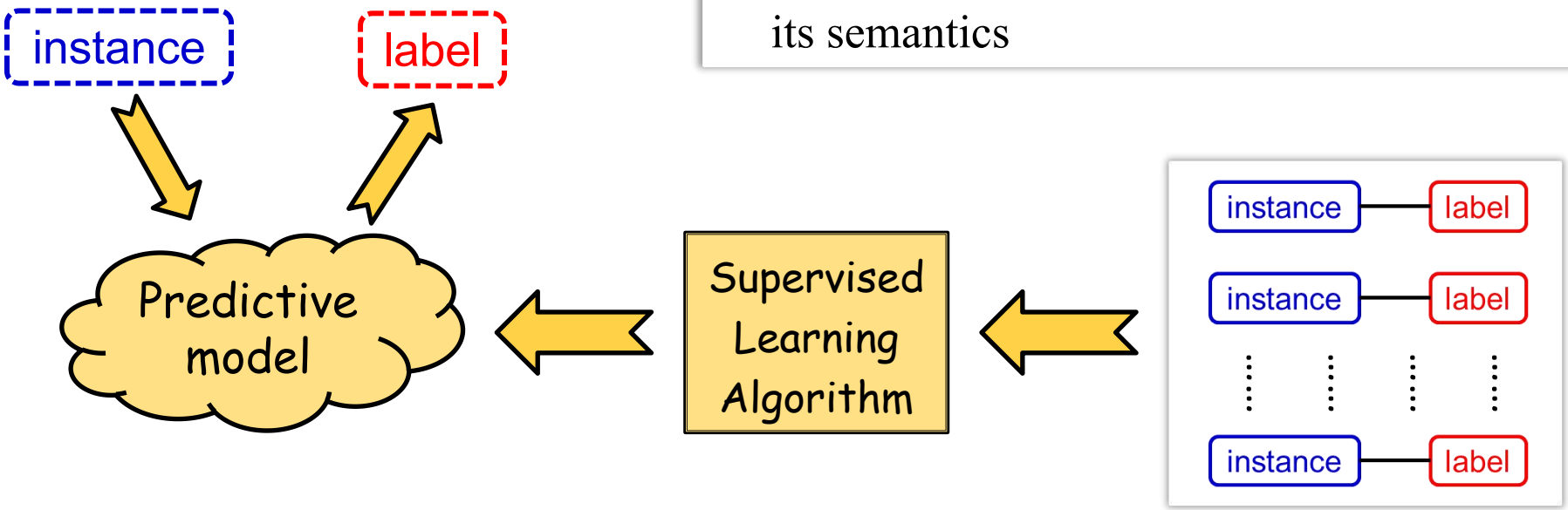


Input Space

represented by a single instance (feature vector) characterizing its properties

Output Space

associated with a single label characterizing its semantics



Basic Assumption: Strong Supervision

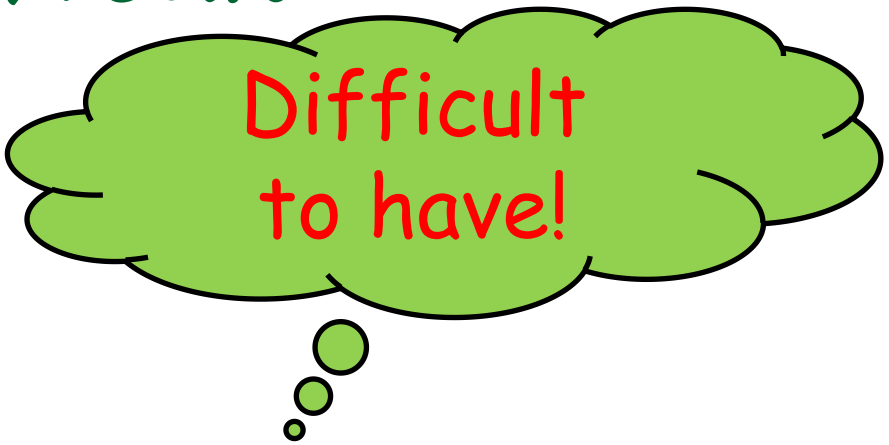
label ⟺ supervision information

Key factor for successful learning
(encoding *semantics* and *regularities* for the learning problem)

Strong supervision assumption

- Sufficient labeling
  abundant labeled training data are available
- Explicit labeling
  object labeling is unique and unambiguous

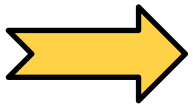
But, Supervision Is Usually Weak



Constrained by:

- Limited resources
- Physical environment
- Problem properties
- ……

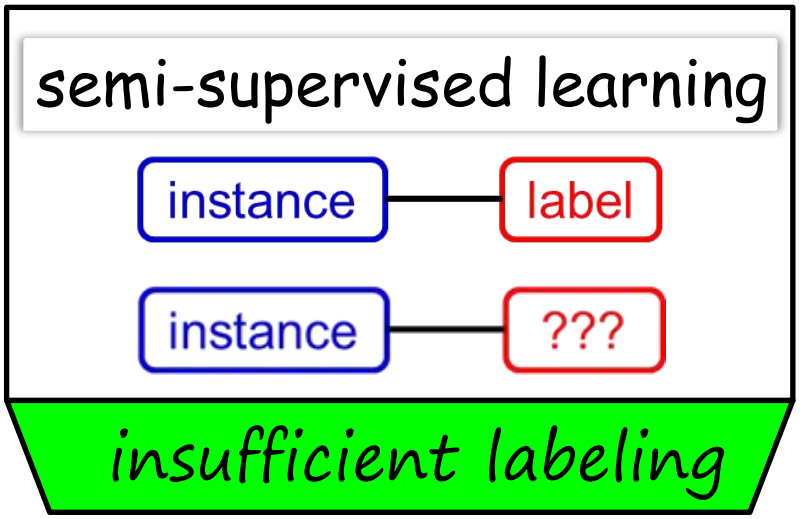
Strong supervision (*sufficient & explicit*)

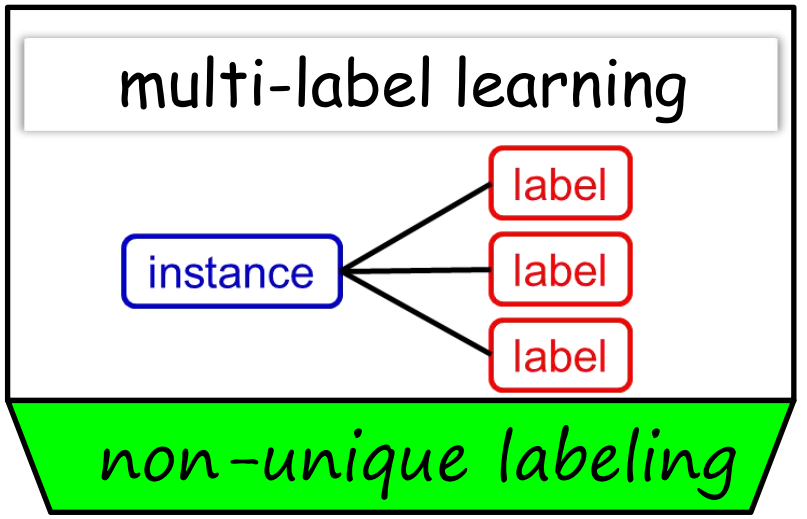


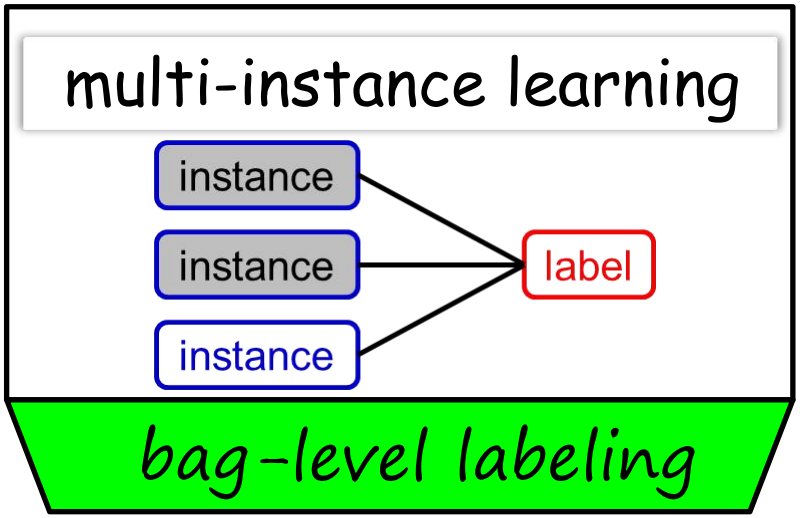
Strong generalization ability

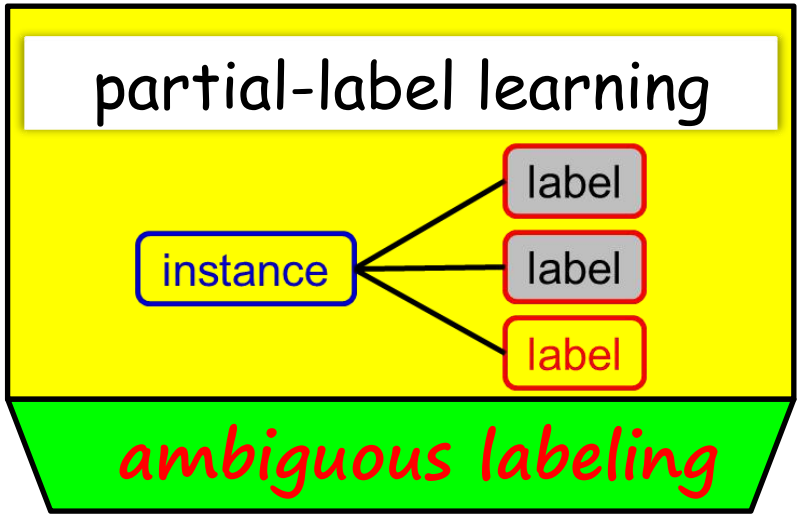
In practice, we usually have to learn with weak supervision [Zhou, NSR18]

For Example…


semi-supervised learning
instance
label
instance
???
insufficient labeling


multi-label learning
instance
label
label
label
non-unique labeling


multi-instance learning
instance
instance
instance
label
bag-level labeling


partial-label learning
instance
label
label
label
ambiguous labeling

Partial Label Learning

- The Framework

Partial Label

Appreciator A ------>  -------> Picasso style ×

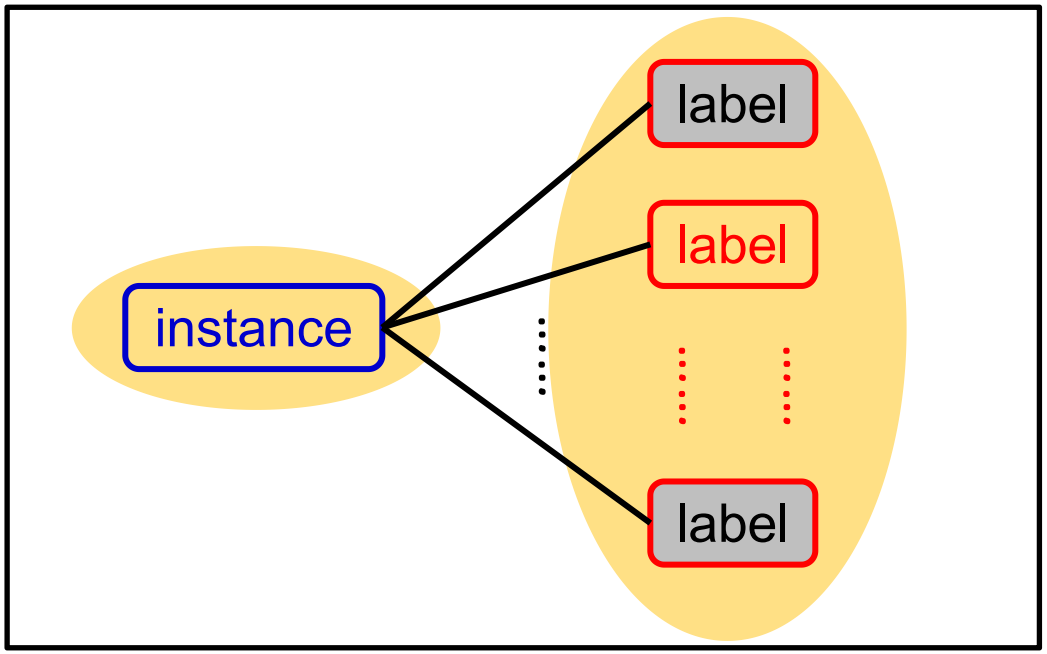
Appreciator B ------> -------> Monet style ×

Appreciator C ------> -------> van Gogh style √

Widely exist in real-world applications

- Image classification [Cour et al., JMLR11] [Chen et al., TPAMI18] [Sun et al., AAAI'19]
- Learning from crowds [Raykar et al., JMLR10] [Yu & Zhang, MLJ17]
- Eco-/Bio-informatics [Briggs et al., KDD'12] [Tang & Zhang, AAAI'17] [Yu et al., ICDM'18]
- NLP [Zhou et al., TALLIP18]
- ……

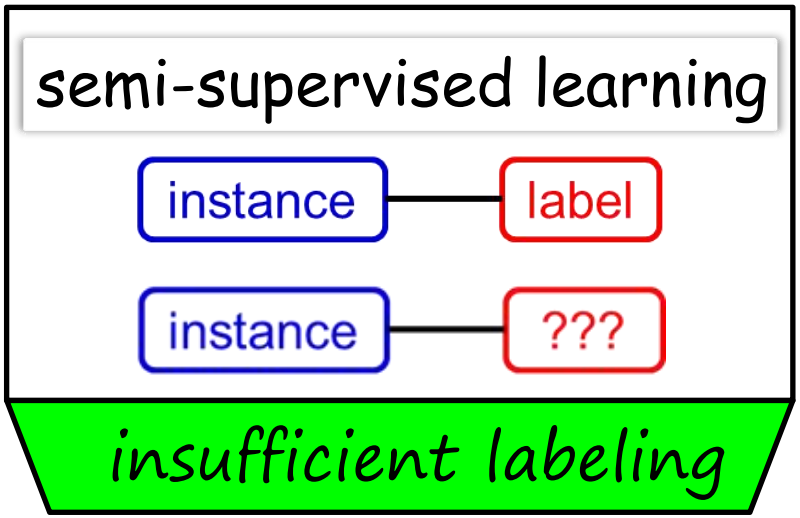
Partial-Label Learning (PLL)

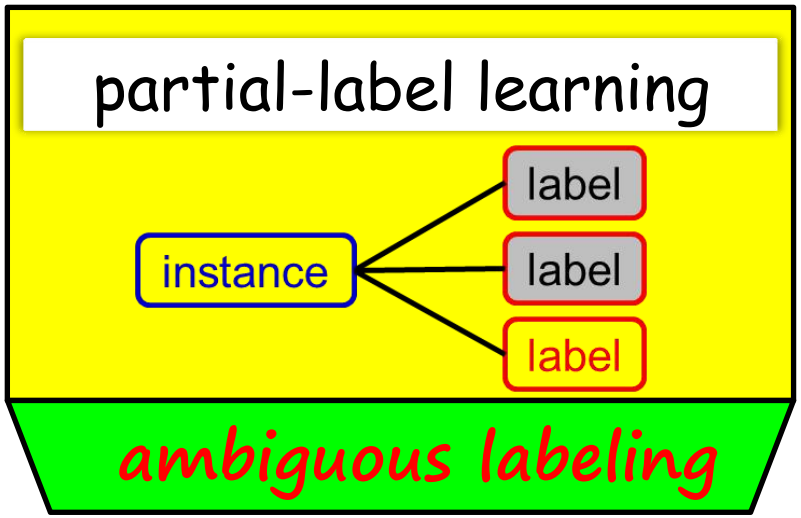


- Each object is associated with multiple candidate labels
- Only one of the candidate label is the unknown ground-truth label

Partial-Label Learning (PLL)

PLL vs. SSL

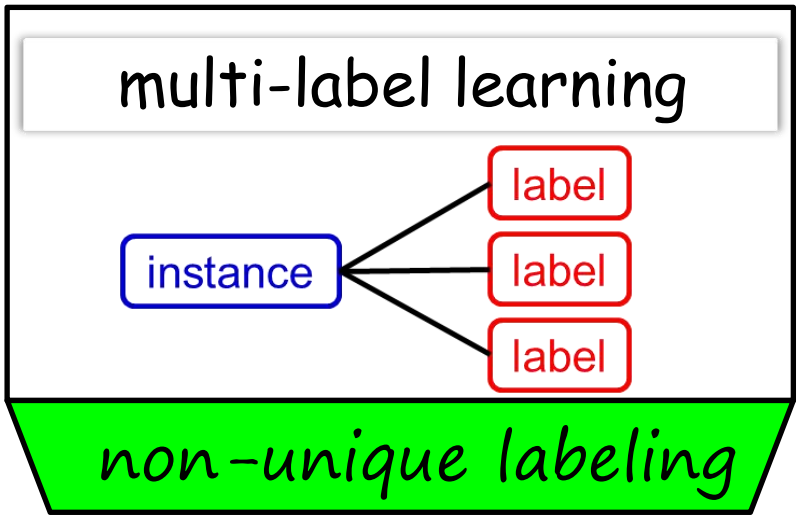


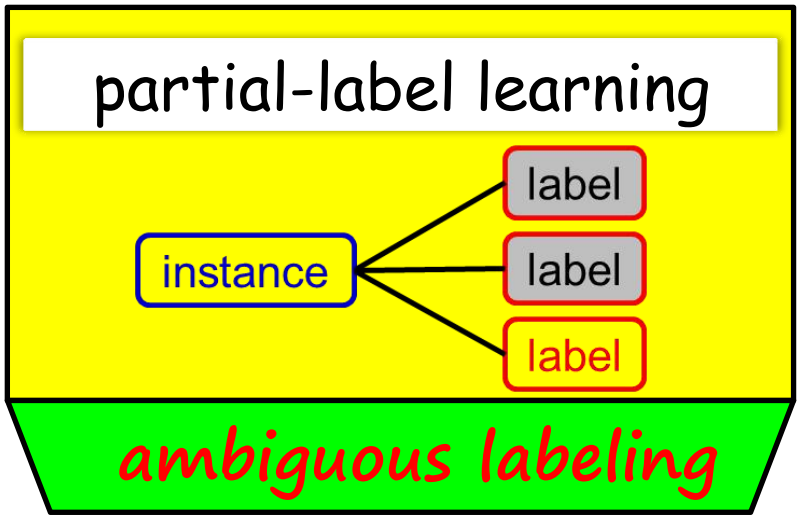


Unlabel: ground-truth label assumes the whole label space

Partial label: ground-truth label is confined within the candidate label set

PLL vs. MLL

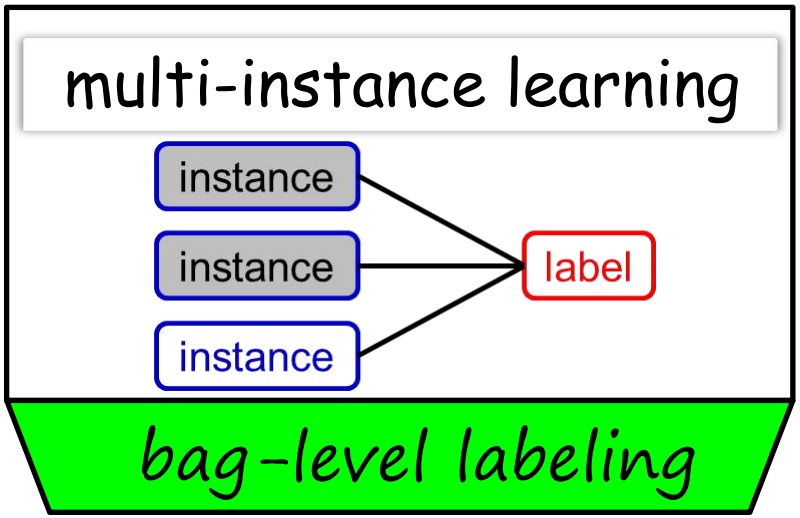


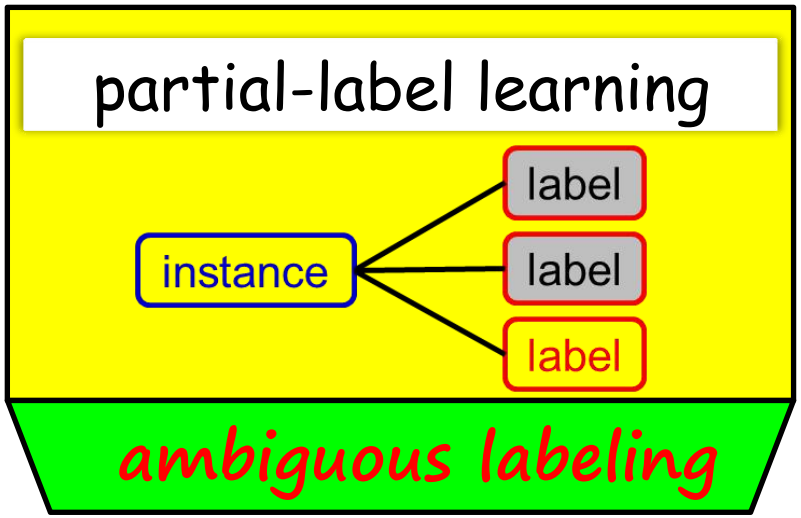


Multi-label: all the associated labels are valid ones

Partial label: only one of the associated label is valid

PLL vs. MIL





Multi-instance: one label assigned to a bag of instances, with ambiguity in the input space

Partial label: multiple labels assigned to a single instance, with ambiguity in the output space

Partial Label Learning

- Existing Approaches

Formal Definition of PLL

Settings

$\mathcal{X}$: d-dimensional feature space $\mathbb{R}^d$

$\mathcal{Y}$: label space with q labels $\{y_1, y_2, \cdots, y_q\}$

Inputs

$\mathcal{D}$: training set with m examples $\{(\boldsymbol{x}_i, S_i) \mid 1 \leq i \leq m\}$

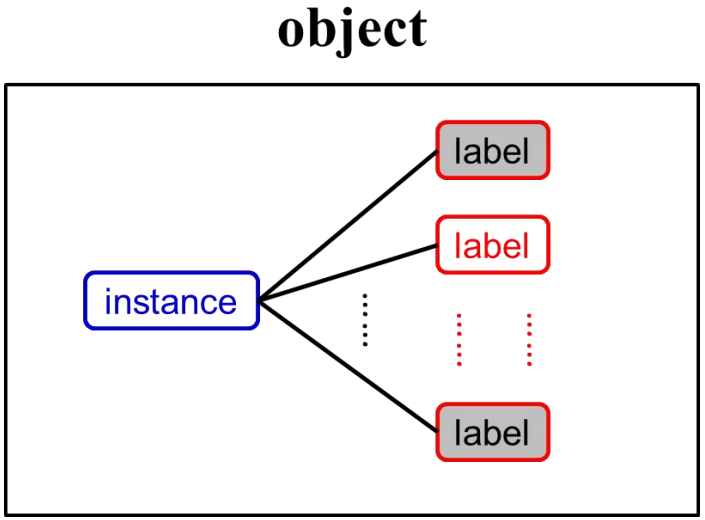
$\boldsymbol{x}_i \in \mathcal{X}$ is a d-dimensional feature vector $(\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \cdots, \boldsymbol{x}_{id})^{\mathrm{T}}$

$S_i \subseteq \mathcal{Y}$ is the candidate label set for $\boldsymbol{x}_i$, with its (unknown) ground-truth label $y_i \in S_i$

Outputs

h : multi-class predictor $\mathcal{X} \to \mathcal{Y}$

Key Challenge



Ambiguous labeling

ground-truth label not accessible by the learning algorithm

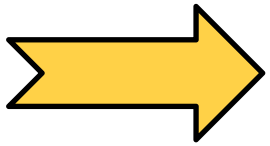
Common strategy: Disambiguation

- Disambiguation by ground-truth label identification
- Disambiguation by candidate label averaging

Disambiguation by Identification

Basic strategy

treating the ground-truth label as latent variable



identified via iterative refining procedure such as EM

$$\hat{y}_i = \arg\max_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta})$$

Latent ground-truth label

Parametric classification model

[Nguyen & Caruana, KDD'08] [Liu & Dietterich, NIPS'12] [Chen et al., CVPR'13] [Zhang et al., KDD'16] [Yu & Zhang, MLJ17] [Chen et al., TPAMI18]

Disambiguation by Identification (Cont.)

Maximum-likelihood formulation:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{m} \log\left(\sum_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta})\right)$$

Maximum margin formulation:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{m} \log\left(\max_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta}) - \max_{y \notin S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta}))\right)$$

……

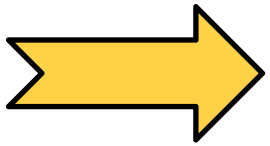
Potential weakness:

the identified label may turn out to be the false positive label

Disambiguation by Averaging

Basic strategy

treating all the candidate labels in an equal manner



make final prediction by averaging their modeling outputs

$$\frac{1}{|S_i|}\sum_{y\in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta}) \quad \oplus \quad F(\boldsymbol{x}_i, y; \boldsymbol{\theta})\ (y \notin S_i)$$

Average output over candidate labels

Output over non-candidate labels

[Hullermeier & Beringer, IDA06] [Cour et al., CVPR'09] [Cour et al., JMLR11] [Zhang & Yu, IJCAI'15] [Gong et al., TCYB18]

Disambiguation by Averaging (Cont.)

Convex formulation:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{m} \Psi\left(\frac{1}{|S_i|}\sum_{y\in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta})\right) + \sum_{y\notin S_i} \Psi\left(-F(\boldsymbol{x}_i, y; \boldsymbol{\theta})\right)$$

Instance-based formulation:

$$f(\boldsymbol{x}^*) = \arg\max_{y\in\mathcal{Y}} \sum\nolimits_{j\in\mathcal{N}(\boldsymbol{x}^*)} [\![y \in S_j]\!]$$

……

Potential weakness:

ground-truth output overwhelmed by false positive outputs

Partial Label Learning

- Recent Work I

Feature-Aware Disambiguation

Common strategy: **Disambiguation**

- Disambiguation by identification

$$\hat{y}_i = \arg\max_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta})$$

- Disambiguation by averaging

$$\frac{1}{|S_i|} \sum_{y \in S_i} F(\boldsymbol{x}_i, y; \boldsymbol{\theta}) \quad \oplus \quad F(\boldsymbol{x}_i, y; \boldsymbol{\theta}) \ (y \notin S_i)$$

Distinguishing the modeling outputs of a single instance over all labels

An intuitive assumption

Information from feature (instance) space may help the disambiguation process

The PL-LEAF Approach [KDD'16; KDD'19]

The usefulness of feature space information

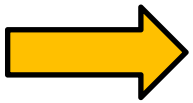
$(\boldsymbol{x}, \{y_2, y_3\})$

$(\boldsymbol{x}', \{y_1, y_2\})$

$(\boldsymbol{x}'', \{y_3, y_4\})$

feature space

$d(\boldsymbol{x}, \boldsymbol{x}') < d(\boldsymbol{x}, \boldsymbol{x}'')$



label space

$\mathbb{P}(y_2 \mid \boldsymbol{x}) > \mathbb{P}(y_3 \mid \boldsymbol{x})$

Feature-aware disambiguation

- Structural relationships among training examples in the feature space would be retained in the label space
- Induce predictive model by exploiting the disambiguated labeling information

Graph-based Feature-Aware Disambiguation

kNN weighted graph

$$\mathcal{G} = (V, E, \mathbf{W})$$

$$V = \{\boldsymbol{x}_i \mid 1 \le i \le m\} \qquad E = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid \boldsymbol{x}_i \in \mathrm{kNN}(\boldsymbol{x}_j),\ i \neq j\}$$

$$\mathbf{W} = [W_{ij}]_{m \times m}$$

$$\begin{aligned} \min_{\mathbf{W}_{\cdot j}} \quad & \left\| \boldsymbol{x}_j - \textstyle\sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in E} W_{ij} \cdot \boldsymbol{x}_i \right\|^2 \\ \text{s.t.}: \quad & \textstyle\sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in E} W_{ij} = 1 \\ & W_{ij} \ge 0 \ \ (\forall (\boldsymbol{x}_i, \boldsymbol{x}_j) \in E) \end{aligned}$$

Labeling confidences

$$\boldsymbol{\Lambda} = [\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_m]$$

labeling confidences over candidate labels are generated by referring to the structural relationships

$$\begin{aligned} \min_{\boldsymbol{\Lambda}} \quad & \textstyle\sum_{j=1}^{m} \left\| \boldsymbol{\lambda}_j - \sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in E} W_{ij} \cdot \boldsymbol{\lambda}_i \right\|^2 \\ \text{s.t.}: \quad & \lambda_{jk} = 0 \ \ (\forall 1 \le j \le m,\ y_k \notin S_j) \\ & \lambda_{jk} \ge 0 \ \ (\forall 1 \le j \le m,\ y_k \in S_j) \\ & \textstyle\sum_{y_k \in S_j} \lambda_{jk} = 1 \ \ (\forall 1 \le j \le m) \end{aligned}$$

Predictive Model Induction

Transform $\mathcal{D}$ into its disambiguated counterpart $\mathcal{D}_{\text{dis}}$

$(\boldsymbol{x}_i, S_i) \Longrightarrow (\boldsymbol{x}_i, \boldsymbol{\lambda}_i)$

candidate label set ⟹ (real-valued) labeling confidence

Induce predictive model via *multi-regression SVR (MSVR)*

$$\{\boldsymbol{\Theta}, \boldsymbol{b}\} = \{(\boldsymbol{\theta}_k, b_k) \mid 1 \le k \le q\}$$

$$L(\boldsymbol{\Theta}, \boldsymbol{b}) = \frac{1}{2}\sum_{k=1}^{q} ||\boldsymbol{\theta}_k||^2 + C_1 \sum_{i=1}^{m} L_1(u_i) + C_2 \sum_{i=1}^{m} v_i$$

$L_1(u_i)$: ε-insensitive loss

v_i: PL empirical loss

iterative gradient-based optimization with closed-form solution in each iteration

Experimental Setup

Comparing Algorithms

PL-LEAF : k=10 for kNN graph construction; kernelized MSVR

averaging-based disambiguatio

CLPL: Base learner: SVM with squared hinge loss

PL-kNN: # nearest neighbors = 10

identification-based disambiguation

PL-SVM: Regularization parameter pool $\{10^{-3},\ldots,10^{3}\}$

LSB-CMM: # mixture components = q

Experimental Protocol

Ten-times random train/test split + Pairwise t-test

Controlled UCI Data Sets

Controlled UCI Data Sets

Data set	# Examples	# Features	# Class Labels
vehicle	846	18	4
segment	2,310	18	7
abalone	4,177	7	29
satimage	6,345	36	7
usps	9,298	256	10
pendigits	10,992	16	10

Generating an **artificial** PL data set from an UCI data set with three controlling parameters p, r, ϵ

Controlled UCI Data Sets

Controlled UCI Data Sets			
Data set	# Examples	# Features	# Class Labels
vehicle	846	18	4
segment	2,310	18	7
abalone	4,177	7	29
satimage	6,345	36	7
usps	9,298	256	10
pendigits	10,992	16	10

Generating an **artificial** PL data set from an UCI data set with three controlling parameters p, r, ϵ

p : Proportion of examples which are partially labeled $(|S_i| \neq 1)$

r : # false positive labels in candidate label set $(|S_i| = r + 1)$

ϵ : Co-occurring probability for one extra candidate label

Fix r (=1, 2, 3), vary $p, \epsilon \in \{0.1, \ldots, 0.7\}$

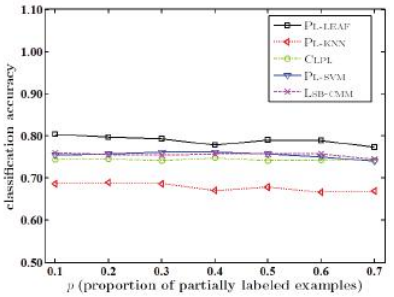
Fix r (=1), p (=1), vary $\epsilon \in \{0.1, \ldots, 0.7\}$

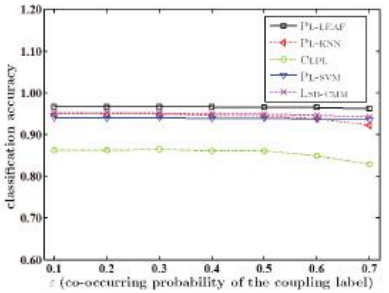
28 configurations per UCI data set

Controlled UCI Data Sets (Cont.)

Table 3: Win/tie/loss counts (pairwise t-test at 0.05 significance level) on the classification performance of PL-LEAF against each comparing algorithm.

	PL-LEAF against			
	PL-KNN	CLPL	PL-SVM	LSB-CMM
[Figure 1]	26/7/9	31/11/0	27/13/2	20/16/6
[Figure 2]	28/7/7	42/0/0	35/7/0	23/16/3
[Figure 3]	28/7/7	40/2/0	33/9/0	23/12/7
[Figure 4]	29/6/7	39/3/0	32/10/0	26/12/4
In Total	**111/27/30**	**152/16/0**	**127/39/2**	**92/56/20**





(e) usps

Out of 168 statistical tests (28 configurations x 6 UCI data sets)

- PL-LEAF outperforms PL-KNN and CLPL in 66.0% and 90.4% cases
- PL-LEAF outperforms PL-SVM and LSB-CMM in 75.5% and 54.7% cases

Real-World Data Sets

Real-World Data Sets					
Data set	# Examples	# Features	# Class Labels	Avg. # CLs	Task Domain
FG-NET	1,002	262	78	7.48	*facial age estimation* [20]
Lost	1,122	108	16	2.23	*automatic face naming* [8]
MSRCv2	1,758	48	23	3.16	*object classification* [16]
BirdSong	4,998	38	13	2.18	*bird song classification* [3]
Soccer Player	17,472	279	171	2.09	*automatic face naming* [25]
Yahoo! News	22,991	163	219	1.91	*automatic face naming* [12]

automatic face naming

instance: face cropped from image/video

candidate labels: names extracted from associated captions/subtitles

object classification

instance: image segmentation

candidate labels: objects appearing within the same image

bird song classification

instance: singing syllable of the bird

candidate labels: bird species jointly singing within 10-seconds period

URL: http://palm.seu.edu.cn/zhangml/Resources.htm#partial data

Real-World Data Sets (Cont.)

Table 4: Classification accuracy (mean±std) of each comparing algorithm on the real-world partial label data sets. In addition, •/○ indicates whether the performance of PL-LEAF is statistically superior/inferior to the comparing algorithm on each data set (pairwise t-test at 0.05 significance level).

	PL-LEAF	PL-KNN	CLPL	PL-SVM	LSB-CMM
FG-NET	0.072±0.010	0.037±0.008•	0.047±0.017•	0.058±0.010•	0.056±0.008•
FG-NET (MAE3)	0.411±0.012	0.284±0.035•	0.240±0.045•	0.343±0.022•	0.344±0.026•
FG-NET (MAE5)	0.550±0.018	0.438±0.033•	0.343±0.055•	0.473±0.016•	0.478±0.025•
Lost	0.664±0.020	0.332±0.030•	0.670±0.024	0.639±0.056	0.591±0.019•
MSRCv2	0.459±0.013	0.417±0.012•	0.375±0.020•	0.417±0.027•	0.431±0.008•
BirdSong	0.706±0.012	0.637±0.009•	0.624±0.009•	0.671±0.018•	0.692±0.015•
Soccer Player	0.515±0.004	0.494±0.004•	0.347±0.004•	0.430±0.004•	0.506±0.006•
Yahoo! News	0.597±0.004	0.403±0.004•	0.457±0.005•	0.615±0.002○	0.594±0.007

- On ***FG-NET***, ***MSRCv2, BirdSong*** and ***Soccer Player***, PL-LEAF is superior to all the comparing algorithms
- On ***Lost***, PL-LEAF is superior or at least comparable to all the comparing algorithms
- On ***Yahoo! News***, PL-LEAF is only inferior to PL-SVM

Partial Label Learning

- Recent Work II

Disambiguation-free PLL

Goal of PLL Induce a multi-class predictor $h : \mathcal{X} \to \mathcal{Y}$

Popular Binary Decomposition

- One-vs-Rest (#classifiers: q)
- One-vs-One (#classifiers: $q(q-1)/2$)

Not applicable due to the unknown ground-truth label

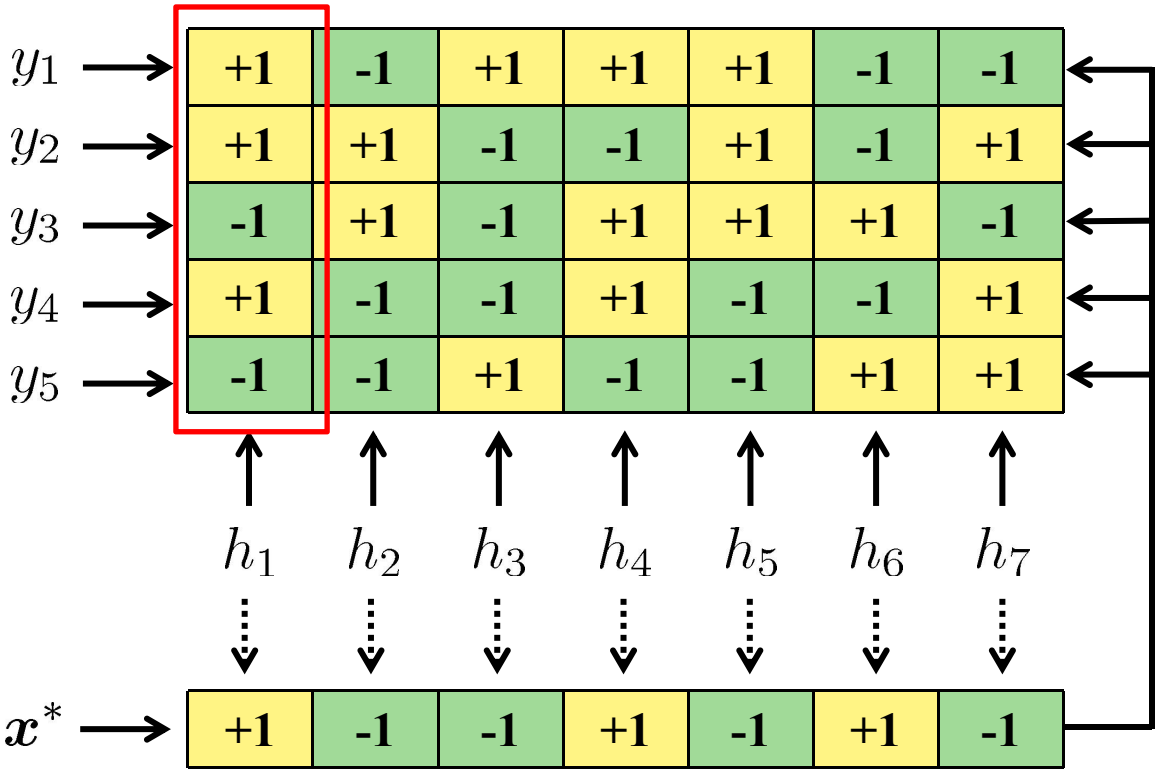
PL-ECOC (Partial-label Learning with Error-Correcting Output Codes)

Two major advantages

- Naturally enable binary decomposition
- Disambiguation-free

The PL-ECOC Approach [TKDE17]

Illustrative procedure of ECOC



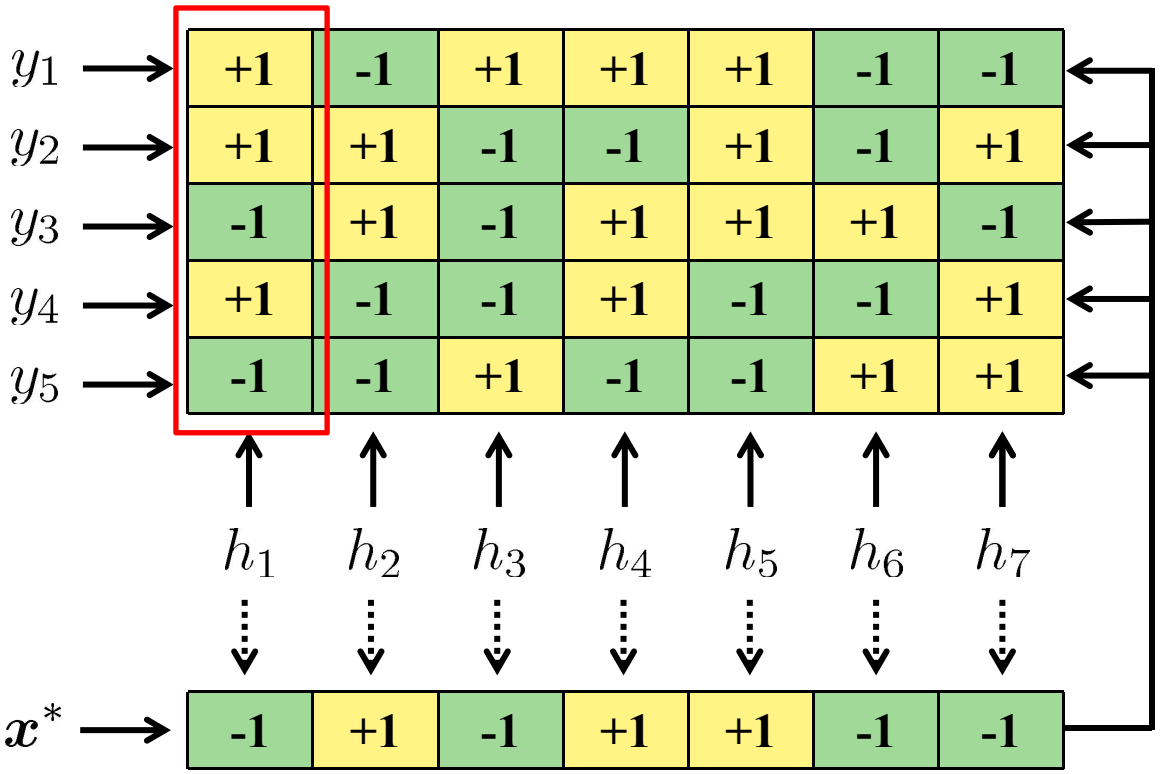
For each multi-class example $(\boldsymbol{x}_i, y_i)$

- $h_1(\boldsymbol{x}_i) = +1$ if $y_i \in \{y_1, y_2, y_4\}$
- $h_1(\boldsymbol{x}_i) = -1$ if $y_i \in \{y_3, y_5\}$

Identify the class with closest codeword to test instance x^*

The PL-ECOC Approach (Cont.)

Illustrative procedure of PL-ECOC



For each partial-label example $(\boldsymbol{x}_i, S_i)$

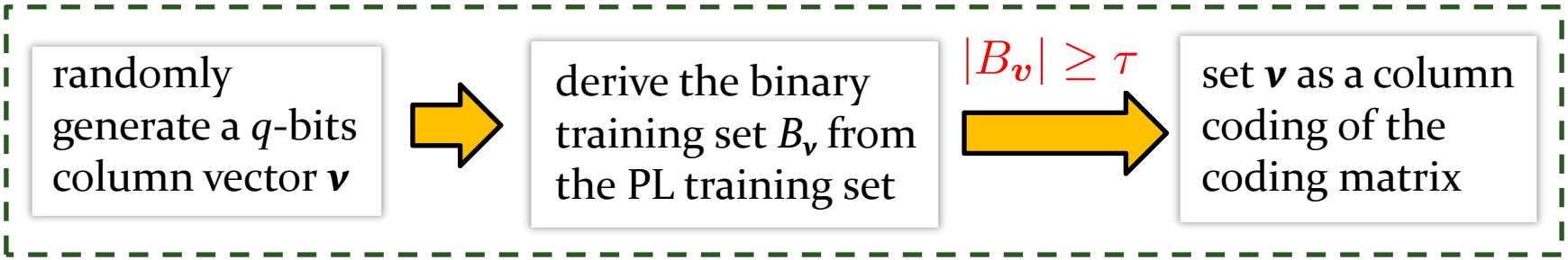
- $h_1(\boldsymbol{x}_i) = +1$ if $S_i \subseteq \{y_1, y_2, y_4\}$
- $h_1(\boldsymbol{x}_i) = -1$ if $S_i \subseteq \{y_3, y_5\}$
- ignored w.r.t. h_1 otherwise

make prediction in the same way as ECOC

The PL-ECOC Approach

Complete Pipeline of PL-ECOC

- Coding matrix generation



Repeat until reaching the ECOC coding length L

- Binary classifier induction

  induce a total of L binary classifiers, one for each column coding

- Make prediction for unseen instance

  identify the class whose codeword is closest to the classifiers' outputs on unseen instance

Experimental Setup

Comparing Algorithms

PL-ECOC : Coding length $=\lceil 10 \cdot \log_2(q) \rceil$; Base learner: Libsvm

averaging-based disambiguatio

CLPL: Base learner: SVM with squared hinge loss

PL-*k*NN: # nearest neighbors = 5

identification-based disambiguation

PL-SVM: Regularization parameter pool $\{10^{-3},\ldots,10^{3}\}$

LSB-CMM: # mixture components = q

Experimental Protocol

Ten-fold cross-validation + Pairwise *t*-test

Controlled UCI Data Sets

TABLE 3
Win/tie/loss counts (pairwise t-test at 0.05 significance level) on the classification performance of PL-ECOC against each comparing algorithm on the controlled UCI data sets.

PL-ECOC against		Data Sets (names in abbreivation) Eco.	Der.	Veh.	Seg.	Aba.	Sat.	Usp.	Pen.	Let.	Subtotal	In Total
PL-KNN	[Figure 1]	0/7/0	1/6/0	7/0/0	3/4/0	7/0/0	0/7/0	7/0/0	5/2/0	7/0/0	37/26/0	156/96/0
	[Figure 2]	0/7/0	3/4/0	7/0/0	2/5/0	7/0/0	0/7/0	7/0/0	7/0/0	5/2/0	38/25/0	
	[Figure 3]	0/7/0	2/5/0	7/0/0	4/3/0	7/0/0	1/6/0	7/0/0	7/0/0	5/2/0	40/23/0	
	[Figure 4]	2/5/0	3/4/0	7/0/0	2/5/0	7/0/0	3/4/0	7/0/0	6/1/0	4/3/0	41/22/0	
CLPL	[Figure 1]	0/7/0	0/7/0	6/1/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	48/15/0	181/71/0
	[Figure 2]	0/7/0	0/7/0	3/4/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	45/18/0	
	[Figure 3]	0/7/0	0/7/0	3/4/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	45/18/0	
	[Figure 4]	0/7/0	1/6/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	43/20/0	
PL-SVM	[Figure 1]	0/7/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	49/14/0	195/57/0
	[Figure 2]	0/7/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	49/14/0	
	[Figure 3]	0/7/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	49/14/0	
	[Figure 4]	0/7/0	0/7/0	6/1/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	7/0/0	48/15/0	
LSB-CMM	[Figure 1]	7/0/0	0/7/0	1/6/0	7/0/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	43/20/0	179/73/0
	[Figure 2]	7/0/0	0/7/0	1/6/0	7/0/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	43/20/0	
	[Figure 3]	7/0/0	0/7/0	4/3/0	7/0/0	0/7/0	7/0/0	7/0/0	7/0/0	7/0/0	46/17/0	
	[Figure 4]	7/0/0	2/5/0	1/6/0	7/0/0	2/5/0	7/0/0	7/0/0	7/0/0	7/0/0	47/16/0	

Out of 252 statistical tests (28 configurations x 9 UCI data sets)

- None of the comparing algorithms significantly outperformed PL-ECOC
- PL-ECOC outperforms PL-KNN and CLPL in 61.9% and 71.8% cases respectively
- PL-ECOC outperforms PL-SVM and LSB-CMM in 77.3% and 71.0% cases respectively

Real-World Data Sets (Cont.)

TABLE 4
Predictive accuracy (mean±std) of each comparing algorithm on the real-world PL data sets. In addition, •/○ indicates whether the performance of PL-ECOC is statistically superior/inferior to the comparing algorithm on each data set (pairwise t-test at 0.05 significate level).

	PL-ECOC	PL-KNN	CLPL	PL-SVM	LSB-CMM
Lost	0.703±0.052	0.424±0.041•	0.742±0.038○	0.729±0.040	0.707±0.055
MSRCv2	0.505±0.027	0.448±0.037•	0.413±0.039•	0.482±0.043	0.456±0.031•
BirdSong	0.740±0.016	0.614±0.024•	0.632±0.017•	0.663±0.032•	0.717±0.024•
Soccer Player	0.537±0.020	0.497±0.014•	0.368±0.010•	0.443±0.014•	0.525±0.015
LYN 10	0.694±0.010	0.460±0.012•	0.605±0.013•	0.692±0.009	0.703±0.010○
LYN 20	0.697±.0.012	0.469±0.015•	0.585±0.010•	0.686±0.011•	0.702±0.011
LYN 50	0.694±0.008	0.472±0.014•	0.540±0.012•	0.666±0.002•	0.679±0.007•
LYN 100	0.680±0.012	0.459±0.010•	0.507±0.011•	0.655±0.010•	0.673±0.010
LYN 200	0.662±0.010	0.457±0.014•	0.462±0.009•	0.636±0.010•	0.648±0.007•

- On ***BirdSong***, ***LYN 50*** and ***LYN 200***, PL-ECOC is superior to all the comparing algorithms
- On ***Soccer Player***, ***LYN 20***, ***LYN 100*** and ***MSRCv2***, PL-ECOC is superior or at least comparable to all the comparing algorithms
- On ***Lost*** and ***LYN 10***, PL-ECOC is inferior to the comparing algorithms in only two cases (CLPL on *Lost*; LSB-CMM on *LYN 10*)

Sensitivity Analysis for Coding Length

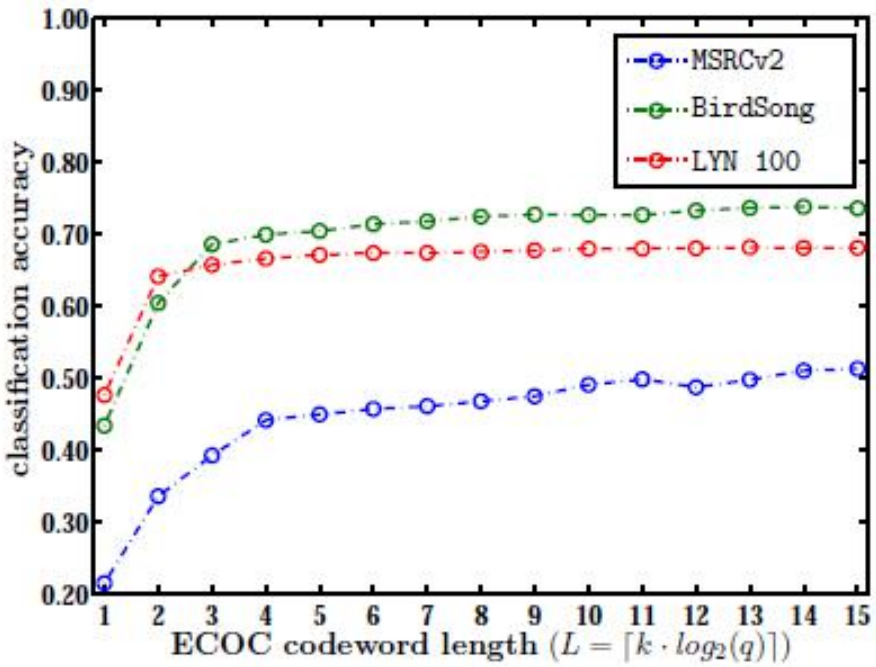


Fig. 5. Classification accuracy of PL-ECOC changes as the codeword length L increases from $\lceil log_2(q) \rceil$ to $\lceil 15 \cdot log_2(q) \rceil$ with step-size $\lceil log_2(q) \rceil$.

- Accuracy improves as the coding length increases
- Becomes stable as coding length approaches $\lceil 10 \cdot \log_2(q) \rceil$

Partial Label Learning

- Related Resources

Introductory Papers

- Cour T, Sapp B, Taskar B. Learning from partial labels. **Journal of Machine Learning Research**, 2011, 12(May): 1501-1536.
- Zhang M-L, Yu F, Tang C-Z. Disambiguation-free partial label learning. **IEEE Transactions on Knowledge and Data Engineering**, 2017, 29(10): 2155-2167.
- Zhang M-L, Zhou Z-H. A review on multi-label learning algorithms. **IEEE Transactions on Knowledge and Data Engineering**, 2014, 26(8): 1819-1837.
- Amores J. Multiple instance classification: Review, taxonomy, and comparative study. **Artificial Intelligence**, 201, 81-105.
- Zhou Z-H, Zhang M-L, Huang S-J, Li Y-F. Multi-instance multi-label learning. **Artificial Intelligence**, 2012, 176(1): 2291-2320.
- Geng X. Label Distribution Learning. **IEEE Transactions on Knowledge and Data Engineering**, 2016, 28(7): 1734-1748.
- ……

Data Sets

- Partial label learning (PLL)
  - http://palm.seu.edu.cn/zhangml/Resources.htm#partial_data
  - http://www.timotheecour.com/tv_data/tv_data.html
  - http://web.engr.oregonstate.edu/~briggsf/
  - http://research.microsoft.com/en-us/projects/objectclassrecognition/
- Multi-label learning (MLL)
  - http://mulan.sourceforge.net/datasets.html
  - http://meka.sourceforge.net/#datasets
  - http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/multilabel.html
- Multi-instance multi-label learning (MIML)
  - http://lamda.nju.edu.cn/data_MIMLimage.ashx
  - http://lamda.nju.edu.cn/data_MItext.ashx
  - http://lamda.nju.edu.cn/data_MIMLprotein.ashx

Codes

- Partial label learning (PLL)
  - http://www.timotheecour.com/tv_data/partial_label_learning_toolbox.html
  - http://web.engr.oregonstate.edu/~liuli/files/LSB-CMM_1.0.tar.gz
  - http://cse.seu.edu.cn/PersonalPage/zhangml/Resources.htm#codes
- Multi-label learning (MLL)
  - http://mulan.sourceforge.net/index.html
  - http://meka.sourceforge.net/
  - http://palm.seu.edu.cn/zhangml/Resources.htm#codes_mll
- Multi-instance multi-label learning (MIML)
  - http://lamda.nju.edu.cn/code_MIML.ashx
- Label distribution learning (LDL)
  - http://cse.seu.edu.cn/PersonalPage/xgeng/LDL

Thanks!