IWPR2018
May 26-28
Jinan, China

Binary Relevance for Multi-Label Learning

Min-Ling Zhang

PALM Group, School of Computer Science and Engineering,

MOE Key Laboratory of Computer Network & Information Integration

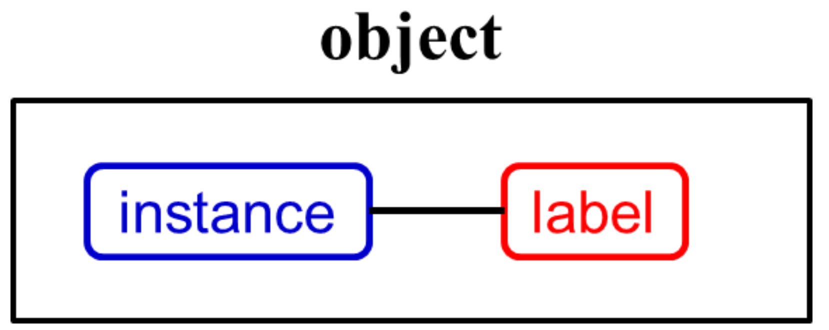
Southeast University, China

May 27, Jinan

Outline

- Multi-Label Learning (MLL)
- Binary Relevance for MLL
- Our Recent Studies
  - Towards Class-Imbalance Aware MLL
  - Leverage Relative Labeling-Importance for MLL
- Conclusion

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

real-world object has unique labeling

Multi-Label Objects

Sunset

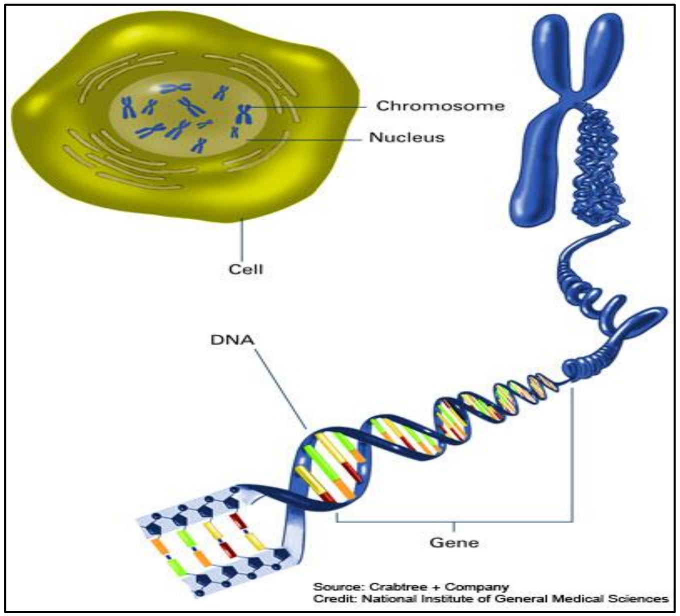
Clouds

Trees

Countryside

……

Multi-Label Objects



Metabolism

Transcription

Protein synthesis

......

Multi-Label Objects

Piano

Classical music

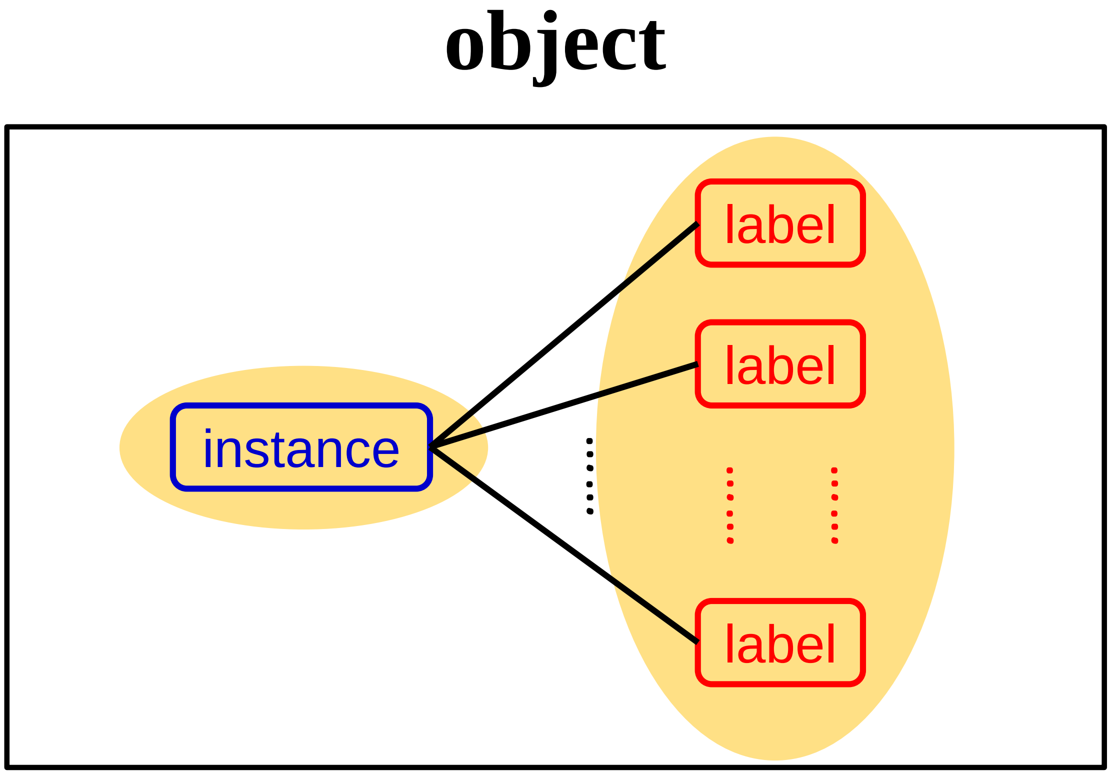
Mozart

Austria

……

Multi-label objects are ubiquitous !

Multi-Label Learning (MLL)



Multi-Label Learning (MLL)

Formal Definition of MLL

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 $\{(\boldsymbol{x}_i, Y_i) \mid 1 \leq i \leq m\}$

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

$Y_i \subseteq \mathcal{Y}$ is the label set associated with $\boldsymbol{x}_i$

Outputs

h : multi-label predictor $\mathcal{X} \rightarrow 2^{\mathcal{Y}}$

Outline

- Multi-Label Learning (MLL)
- Binary Relevance for MLL
- Our Recent Studies
  - Towards Class-Imbalance Aware MLL
  - Leverage Relative Labeling-Importance for MLL
- Conclusion

Binary Relevance (BR) [Boutell et al., PRJ04]

The most intuitive solution to MLL

decompose MLL into q independent binary problems

for j=1 to q **do**

Generate the *binary training set* $\mathcal{D}_j$ from $\mathcal{D}$;

Train binary classifier: $g_j \leftarrow \mathcal{B}(\mathcal{D}_j)$

end

$h(\boldsymbol{x}^*) = \{y_j \mid g_j(\boldsymbol{x}^*) > 0,\ 1 \leq j \leq q\}$

$\mathcal{D} = \{(\boldsymbol{x}_i, Y_i) \mid 1 \leq i \leq m\}$

(MLL training set)

$$\mathcal{D}_j = \{(\boldsymbol{x}_i, \phi(Y_i, y_j)) \mid 1 \leq i \leq m\}$$

$$\text{where } \phi(Y_i, y_j) = \begin{cases} 1, & \text{if } y_j \in Y_i \\ 0, & \text{otherwise} \end{cases}$$

Binary Relevance (BR) [Boutell et al., PRJ04]

The most intuitive solution to MLL

decompose MLL into q independent binary problems

for

er

$h($

Pros:

conceptually simple, efficient and easy to implement

Cons:

label correlations totally ignored

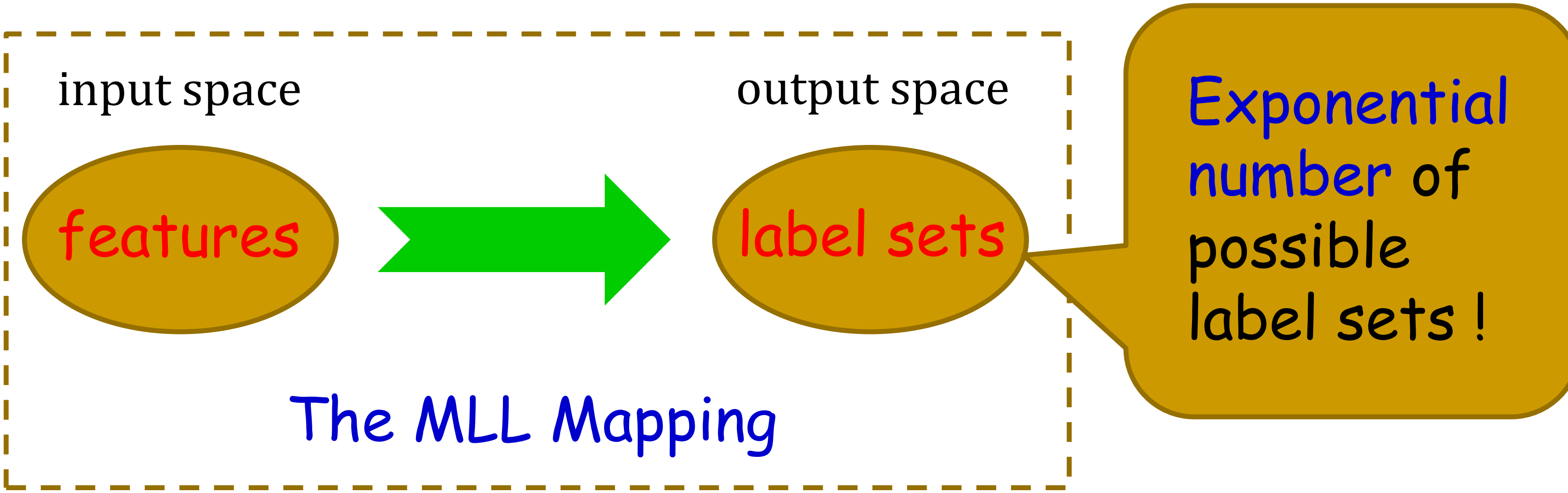
$$\mathcal{D} = \{(\boldsymbol{x}_i, Y_i) \mid 1 \leq i \leq m\}$$

(MLL training set)

$$\mathcal{D}_j = \{(\boldsymbol{x}_i, \phi(Y_i, y_j)) \mid 1 \leq i \leq m\}$$

$$\text{where } \phi(Y_i, y_j) = \begin{cases} 1, & \text{if } y_j \in Y_i \\ 0, & \text{otherwise} \end{cases}$$

Major Challenge – **Huge Output Space**



q=5 ➔ 32 label sets

q=10 ➔ ~1k label sets

q=20 ➔ ~1M label sets

……

Common Strategy

Exploiting Label Correlations

e.g.: *An image labeled as lions and grassland would be likely annotated with label Africa*

Endow BR with Label Correlations (1)

Chaining-style methods

[Read et al., ECML PKDD'09/MLJ11; Dembczyński et al., ICML'10/ECAI'12; Kumar et al., ECML PKDD'12; Senge et al., Gfkl'13; Li & Zhou, MCS'13; Mena et al., MLJ17]

Step I: Specify a chaining order over all the class labels

$$\tau: \ \{1, \ldots, q\} \rightarrow \{1, \ldots, q\}$$

Step II: Induce one binary classifier for each label along the chain, by treating preceding labels as extra features

$$f_j : \mathcal{X} \times \mathcal{Y}_{\tau(1)} \times \cdots \mathcal{Y}_{\tau(j-1)} \rightarrow \mathcal{Y}_{\tau(j)} \ (1 \leq j \leq q)$$

Random correlations among labels

Endow BR with Label Correlations (2)

Stacking-style methods

[Godbole & Sarawagi, PAKDD'04; Tsoumakas et al., MLD'09; Zhang & Zhou, KDD'10; Montañes et al., PRJ14; Loza Mencía et al., MLJ16]

Step I: Invoke the standard BR procedure

$$g_j : \mathcal{X} \rightarrow y_j \;\; (1 \leq j \leq q)$$

Step II: Induce one binary classifier for each label, by treating BR classifiers' outputs as extra features

$$f_j : \mathcal{X} \times g_1(\cdot) \times \cdots g_q(\cdot) \rightarrow y_j \;\; (1 \leq j \leq q)$$

Full-order correlations among labels

To Enhance BR…

Exploiting
Label Correlations

Necessary

But Not
Sufficient

Two Inherent Properties

- **Class-imbalance**
- **Relative Labeling-importance**

Outline

- Multi-Label Learning (MLL)
- Binary Relevance for MLL
- Our Recent Studies
  - Towards Class-Imbalance Aware MLL
  - Leverage Relative Labeling-Importance for MLL
- Conclusion

Towards Class-Imbalance Aware Multi-Label Learning

Class-Imbalance for MLL

An inherent property for MLL: class-imbalance

For each class label $y_j \in \mathcal{Y}$:

$\mathcal{D}_j^+$: the set of *positive* training examples w.r.t. y_j

$\mathcal{D}_j^-$: the set of *negative* training examples w.r.t. y_j

$$ImR_j = \frac{\max(|\mathcal{D}_j^+|, |\mathcal{D}_j^-|)}{\min(|\mathcal{D}_j^+|, |\mathcal{D}_j^-|)}$$ (*imbalance ratio*)

For the *rcv1* data set (with 42 class labels), we have:

minimum ImR_j ($\min_{1\leq j\leq q} ImR_j$): >3

average ImR_j ($\frac{1}{q}\sum_{j=1}^{q} ImR_j$): >15

maximum ImR_j ($\max_{1\leq j\leq q} ImR_j$): >50

Existing Approaches Towards Class-Imbalance MLL

Binary Decomposition

Decompose MLL into q independent binary learning problems

\+

- ✓ **Over-sampling/Under-sampling**
  apply over-sampling/under-sampling techniques [Spyromitros-Xioufis et al., IJCAI'11] [Tahir et al., PRJ12] [Charte et al., KBS15]

- ✓ **Parameter tuning**
  optimizing the classification threshold [Fan & Lin, TechReport07] [Quevedo et al., PRJ12] [Pillai et al., PRJ13]

- ✓ **Optimizing imbalance-specific metric**
  optimizing the F-measure [Petterson & Caetano, NIPS'10] [Dembczyński et al., ICML'13]

Existing Approaches Towards Class-Imbalance MLL

Ignoring Label Correlations!

ampling

r-sampling

s-Xioufis et al., IJCAI'11]

RJ12] [Charte et al., KBS15]

Binary Decomposition

Decompose MLL into q independent binary learning problems

+

✓ **Parameter tuning**

optimizing the classification threshold [Fan & Lin, TechReport07] [Quevedo et al., PRJ12] [Pillai et al., PRJ13]

✓ **Optimizing imbalance-specific metric**

optimizing the F-measure [Petterson & Caetano, NIPS'10] [Dembczyński et al., ICML'13]

The COCOA Approach

Basic Strategy

Cross-coupling + class-imbalance learner aggregation

Training Phase

Cross-coupling each label with other labels → Generate multi-class imbalance classifier

Testing Phase

Aggregate classifiers' outputs for each label → Predict by querying aggregation results

Training Phase

For each class label $y_j \in \mathcal{Y}$, induce a real-valued function $f_j : \mathcal{X} \to \mathbb{R}$ by cross-coupling with other class labels

suppose y_k $(k \neq j)$ is chosen to couple with y_j, a four-class training set $\mathcal{D}_{jk}$ can be derived from $\mathcal{D}$ as follows:

$$\mathcal{D}_{jk} = \{(\boldsymbol{x}_i, \psi(Y_i, y_j, y_k)) \mid 1 \leq i \leq m\}$$

$$\text{where} \quad \psi(Y_i, y_j, y_k) = \begin{cases} 0, & \text{if } y_j \notin Y_i \text{ and } y_k \notin Y_i \\ +1, & \text{if } y_j \notin Y_i \text{ and } y_k \in Y_i \\ +2, & \text{if } y_j \in Y_i \text{ and } y_k \notin Y_i \\ +3, & \text{if } y_j \in Y_i \text{ and } y_k \in Y_i \end{cases}$$

$\psi(Y_i, y_j, y_k)$ is determined by the joint assignment of y_j and y_k w.r.t. Y_i

Training Phase – Cont'd

$$\mathcal{D}_{jk} = \{(\boldsymbol{x}_i, \psi(Y_i, y_j, y_k)) \mid 1 \leq i \leq m\}$$

$$\text{where} \quad \psi(Y_i, y_j, y_k) = \begin{cases} 0, & \text{if } y_j \notin Y_i \text{ and } y_k \notin Y_i \\ +1, & \text{if } y_j \notin Y_i \text{ and } y_k \in Y_i \\ +2, & \text{if } y_j \in Y_i \text{ and } y_k \notin Y_i \\ +3, & \text{if } y_j \in Y_i \text{ and } y_k \in Y_i \end{cases}$$

Merge

WLOG, suppose positive examples correspond to the *minority* class

For $\mathcal{D}_{jk}$, the first class $(\psi(Y_i, y_j, y_k) = 0)$ would be largest and the fourth class $(\psi(Y_i, y_j, y_k) = +3)$ would be smallest

The worst imbalance ratio would roughly turn into $ImR_j * ImR_k$

Training Phase – Cont'd

$$\mathcal{D}_{jk} = \{(\boldsymbol{x}_i, \psi(Y_i, y_j, y_k)) \mid 1 \le i \le N\}$$

$$\text{where} \quad \psi(Y_i, y_j, y_k) = \begin{cases} 0, & \text{if } y_j \notin Y_i \text{ and } y_k \notin Y_i \\ +1, & \text{if } y_j \notin Y_i \text{ and } y_k \in Y_i \\ +2, & \text{if } y_j \in Y_i \text{ and } y_k \notin Y_i \\ +3, & \text{if } y_j \in Y_i \text{ and } y_k \in Y_i \end{cases}$$

Merge

four-class to tri-class

$$\mathcal{D}_{jk}^{\mathbf{tri}} = \left\{\left(\boldsymbol{x}_i, \psi^{\mathbf{tri}}(Y_i, y_j, y_k)\right) \mid 1 \le i \le N\right\}$$

$$\text{where} \quad \psi^{\mathbf{tri}}(Y_i, y_j, y_k) = \begin{cases} 0, & \text{if } y_j \notin Y_i \text{ and } y_k \notin Y_i \\ +1, & \text{if } y_j \notin Y_i \text{ and } y_k \in Y_i \\ +2, & \text{if } y_j \in Y_i \end{cases}$$

Training Phase – Cont'd

for j=1 to q **do**

Draw a *random* subset $\mathcal{I}_K \subset \mathcal{Y} \setminus \{y_j\}$ containing K class labels;

for $y_k \in \mathcal{I}_K$ **do**

Form the tri-class training set $\mathcal{D}_{jk}^{\mathbf{tri}}$;

$g_{jk} \leftarrow \boxed{\mathcal{M}}(\mathcal{D}_{jk}^{\mathbf{tri}});$

end

end

Multi-class imbalance learner

Testing Phase

for j=1 to q **do**

Draw a *random* subset $\mathcal{I}_K \subset \mathcal{Y} \setminus \{y_j\}$ containing K class labels;

for $y_k \in \mathcal{I}_K$ **do**

Form the tri-class training set $\mathcal{D}_{jk}^{\textbf{tri}}$;

$g_{jk} \leftarrow \boxed{\mathcal{M}}(\mathcal{D}_{jk}^{\textbf{tri}})$;

end

end

Multi-class imbalance learner

for j=1 to q **do**

$f_j(\boldsymbol{x}) = \sum_{y_k \in \mathcal{I}_K} \boxed{g_{jk}(+2 \mid \boldsymbol{x})}$

end

predictive confidence that $\boldsymbol{x}$ has positive assignment w.r.t. y_j

$Y = \{y_j \mid \forall 1 \leq j \leq q: \ f_j(\boldsymbol{x}) > \boxed{t_j}\}$

threshold obtained by optimizing empirical F-measure w.r.t. y_j

Testing Phase

for j=1 to q **do**

Draw a *random* subset $\mathcal{I}_K \subset$

for $y_k \in \mathcal{I}_K$ **do**

Form the tri-class traini

$g_{jk} \leftarrow \boxed{\mathcal{M}}(\mathcal{D}_{jk}^{\mathbf{tri}});$

end

end

Multi-class imbalance learner

Label Cross-Coupling

for j=1 to q **do**

$$f_j(\boldsymbol{x}) = \sum\nolimits_{y_k \in \mathcal{I}_K} \boxed{g_{jk}(+2 \mid \boldsymbol{x})}$$

end

$$Y = \{y_j \mid \forall 1 \leq j \leq q: \; f_j(\boldsymbol{x}) > \boxed{t_j}\}$$

Class-Imbalance Learner Aggregation

as

imizing

w.r.t. y_j

Experimental Setup – **Data Sets**

Table 2: Characteristics of the benchmark multi-label data sets.

Data set	$\|\mathcal{S}\|$	$dim(\mathcal{S})$	$L(\mathcal{S})$	$F(\mathcal{S})$	$LCard(\mathcal{S})$	$LDen(\mathcal{S})$	$DL(\mathcal{S})$	$PDL(\mathcal{S})$	Imbalance Ratio		
									min	max	avg
CAL500	502	68	124	numeric	25.058	0.202	502	1.000	1.040	24.390	3.846
Emotions	593	72	6	numeric	1.869	0.311	27	0.046	1.247	3.003	2.146
Medical	978	144	14	numeric	1.075	0.077	42	0.043	2.674	43.478	11.236
Enron	1702	50	24	nominal	3.113	0.130	547	0.321	1.000	43.478	5.348
Scene	2407	294	6	numeric	1.074	0.179	15	0.006	3.521	5.618	4.566
Yeast	2417	103	13	numeric	4.233	0.325	189	0.078	1.328	12.500	2.778
Slashdot	3782	53	14	nominal	1.134	0.081	118	0.031	5.464	35.714	10.989
Corel5k	5000	499	44	nominal	2.214	0.050	1037	0.207	3.460	50.000	17.857
Rcv1 (subset 1)	6000	472	42	numeric	2.458	0.059	574	0.096	3.344	50.000	15.152
Rcv1 (subset 2)	6000	472	39	numeric	2.170	0.056	489	0.082	3.215	47.619	15.873
Eurlex-sm	19348	250	27	numeric	1.492	0.055	497	0.026	3.509	47.619	16.393
Tmc2007	28596	500	15	nominal	2.100	0.140	637	0.022	1.447	34.483	5.848
Mediamill	43907	120	29	numeric	4.010	0.138	3540	0.079	1.748	45.455	7.092

Thirteen benchmark multi-label data sets

- ✓ average imbalance ratio ranges from 2.146 to 17.857
- ✓ ten times of random train/test splits (50%/50%) + pairwise *t*-test
- ✓ imbalance-specific metrics: (macro-averaging) *F-measure* and *AUC*

Experimental Setup – **Comparing Algorithms**

COCOA : $K = \min(q-1, 10)$

First Series Binary decomposition + imbalance learning techniques

USAM: under-sampling

SMOTE: over-sampling

USAM-EN, SMOTE-EN: ensemble version

RML: optimizing F-measure [Petterson & Caetano, NIPS'10]

Second Series Well-established MLL learning algorithms

ML-KNN: First-order approach [Zhang & Zhou, PRJ07]

CLR: Second-order approach [Fürnkranz et al., MLJ08]

ECC: High-order approach [Read et al., MLJ11]

RAKEL: High-order approach [Tsoumakas et al., TKDE11]

Experimental Results – **F-measure**

Table 3: Performance of each comparing algorithm (mean±std. deviation) in terms of *macro-averaging F-measure* (MACRO-F). In addition, •/◦ indicates whether COCOA is statistically superior/inferior to the comparing algorithm on each data set (pairwise *t*-test at 1% significance level).

Algorithm	Data Set: CAL500	Emotions	Medical	Enron	Scene	Yeast	Slashdot
COCOA	0.207±0.009	0.662±0.013	0.690±0.015	0.324±0.009	0.732±0.013	0.457±0.015	0.327±0.009
USAM	0.217±0.006◦	0.591±0.016•	0.670±0.012•	0.266±0.011•	0.624±0.008•	0.432±0.010•	0.259±0.010•
USAM-EN	0.246±0.004◦	0.590±0.018•	0.665±0.025•	0.274±0.010•	0.620±0.011•	0.437±0.012	0.296±0.007•
SMOTE	0.237±0.006◦	0.584±0.020•	0.672±0.022	0.266±0.006•	0.619±0.007•	0.430±0.006•	0.326±0.005
SMOTE-EN	0.239±0.004◦	0.582±0.017•	0.672±0.022	0.275±0.004•	0.624±0.007•	0.431±0.005•	0.315±0.007
RML	0.209±0.008	0.645±0.016	0.666±0.018	0.309±0.010•	0.684±0.013•	0.471±0.014	0.311±0.009•
ML-KNN	0.074±0.002•	0.592±0.026•	0.474±0.031•	0.174±0.011•	0.715±0.011	0.380±0.008•	0.198±0.014•
CLR	0.081±0.007•	0.595±0.017•	0.650±0.012•	0.229±0.006•	0.631±0.013•	0.413±0.010•	0.233±0.007•
ECC	0.102±0.004•	0.642±0.014•	0.647±0.021•	0.241±0.006•	0.716±0.009	0.394±0.008•	0.250±0.007•
RAKEL	0.193±0.003•	0.613±0.018•	0.576±0.014•	0.256±0.006•	0.686±0.008•	0.420±0.005•	0.248±0.006•

Algorithm	Data Set: Corel5k	Rcv1 (subset 1)	Rcv1 (subset 2)	Eurlex-sm	Tmc2007	Mediamill	win/tie/loss counts for COCOA
COCOA	0.195±0.004	0.363±0.008	0.337±0.009	0.703±0.007	0.669±0.002	0.459±0.004	/
USAM	0.141±0.004•	0.318±0.005•	0.306±0.005•	0.562±0.007•	0.607±0.002•	0.337±0.003•	12/0/1
USAM-EN	0.150±0.002•	0.317±0.005•	0.303±0.005•	0.563±0.004•	0.608±0.002•	0.337±0.003•	11/1/1
SMOTE	0.125±0.003•	0.314±0.006•	0.305±0.004•	0.552±0.003•	0.566±0.003•	0.338±0.001•	10/2/1
SMOTE-EN	0.126±0.002•	0.313±0.004•	0.304±0.004•	0.553±0.003•	0.567±0.003•	0.341±0.001•	10/2/1
RML	0.215±0.009◦	0.387±0.020◦	0.363±0.029◦	0.059±0.003•	0.568±0.039•	0.268±0.019•	6/4/3
ML-KNN	0.028±0.004•	0.122±0.008•	0.103±0.008•	0.525±0.012•	0.479±0.008•	0.245±0.004•	12/1/0
CLR	0.049±0.004•	0.227±0.007•	0.226±0.006•	0.599±0.006•	0.623±0.003•	0.268±0.004•	13/0/0
ECC	0.064±0.004•	0.216±0.007•	0.199±0.004•	0.619±0.009•	0.642±0.003•	0.277±0.002•	12/1/0
RAKEL	0.084±0.005•	0.272±0.007•	0.263±0.005•	0.632±0.008•	0.643±0.004•	0.378±0.002•	13/0/0

Experimental Results – **F-measure**

Table 3: Performance of each comparing algorithm (mean±std. deviation) in terms of *macro-averaging F-measure* (MACRO-F). In addition, •/○ indicates whether COCOA is statistically superior/inferior to the comparing algorithm on each data set (pairwise *t*-test at 1% significance level).

Algorithm	CAL500	...	east	Slashdot
COCOA	0.207±0.00		±0.015	0.327±0.009
USAM	0.217±0.00		±0.010•	0.259±0.010•
USAM-EN	0.246±0.00		±0.012	0.296±0.007•
SMOTE	0.237±0.00		±0.006•	0.326±0.005
SMOTE-EN	0.239±0.00		±0.005•	0.315±0.007
RML	0.209±0.00		±0.014	0.311±0.009•
ML-KNN	0.074±0.00		±0.008•	0.198±0.014•
CLR	0.081±0.00		±0.010•	0.233±0.007•
ECC	0.102±0.00		±0.008•	0.250±0.007•
RAKEL	0.193±0.00		±0.005•	0.248±0.006•

Algorithm	Corel5k	Rcv1 (subset 1)	Rcv1 (subset 2)	Eurlex-sm	Tmc2007	Mediamill	win/tie/loss counts for COCOA
COCOA	0.195±0.004	0.363±0.008	0.337±0.009	0.703±0.007	0.669±0.002	0.459±0.004	/
USAM	0.141±					0.337±0.003•	12/0/1
USAM-EN	0.150					337±0.003•	11/1/1
SMOTE	0.125					338±0.001•	10/2/1
SMOTE-EN	0.126					341±0.001•	10/2/1
RML	0.215					268±0.019•	6/4/3
ML-KNN	0.028					245±0.004•	12/1/0
CLR	0.049						13/0/0
ECC	0.064					0.002•	12/1/0
RAKEL	0.084					378±0.002•	13/0/0

COCOA significantly outperforms the first series of comparing algorithms in 46.2% (RML), 76.9% (SMOTE, SMOTE-EN), 84.6% (USAM-EN) and 92.3% (USAM) cases

COCOA significantly outperforms or is at least comparable to the second series of comparing algorithms

Experimental Results – **AUC**

Table 4: Performance of each comparing algorithm (mean±std. deviation) in terms of *macro-averaging AUC* (MACRO-AUC). In addition, •/∘ indicates whether COCOA is statistically superior/inferior to the comparing algorithm on each data set (pairwise t-test at 1% significance level).

Algorithm	Data Set						
	CAL500	Emotions	Medical	Enron	Scene	Yeast	Slashdot
COCOA	0.557±0.005	0.843±0.010	0.958±0.006	0.731±0.006	0.943±0.003	0.710±0.006	0.736±0.005
USAM	0.514±0.005•	0.708±0.019•	0.855±0.012•	0.606±0.010•	0.790±0.009•	0.578±0.006•	0.617±0.004•
USAM-EN	0.513±0.004•	0.708±0.015•	0.860±0.024•	0.600±0.004•	0.788±0.009•	0.583±0.006•	0.618±0.004•
SMOTE	0.513±0.005•	0.703±0.019•	0.874±0.019•	0.617±0.007•	0.776±0.008•	0.579±0.006•	0.688±0.008•
SMOTE-EN	0.513±0.004•	0.704±0.013•	0.874±0.019•	0.617±0.007•	0.777±0.011•	0.581±0.007•	0.686±0.008•
RML	–	–	–	–	–	–	–
ML-KNN	0.516±0.007•	0.806±0.015•	0.909±0.008•	0.663±0.006•	0.926±0.005•	0.679±0.004•	0.676±0.006•
CLR	0.561±0.004∘	0.796±0.010•	0.948±0.008•	0.709±0.007•	0.894±0.005•	0.650±0.004•	0.698±0.009•
ECC	0.549±0.007•	0.841±0.009	0.925±0.009•	0.723±0.006•	0.938±0.003•	0.689±0.006•	0.706±0.009•
RAKEL	0.528±0.005•	0.797±0.015•	0.828±0.006•	0.640±0.003•	0.892±0.004•	0.640±0.004•	0.612±0.002•

Algorithm	Data Set						win/tie/loss counts for COCOA
	Corel5k	Rcv1 (subset 1)	Rcv1 (subset 2)	Eurlex-sm	Tmc2007	Mediamill	
COCOA	0.719±0.004	0.889±0.003	0.882±0.002	0.957±0.002	0.930±0.001	0.843±0.001	/
USAM	0.572±0.003•	0.674±0.010•	0.672±0.009•	0.788±0.009•	0.801±0.003•	0.655±0.004•	13/0/0
USAM-EN	0.574±0.002•	0.676±0.010•	0.671±0.010•	0.789±0.006•	0.800±0.003•	0.654±0.006•	13/0/0
SMOTE	0.597±0.004•	0.625±0.009•	0.620±0.008•	0.795±0.005•	0.793±0.003•	0.669±0.002•	13/0/0
SMOTE-EN	0.596±0.004•	0.626±0.006•	0.620±0.009•	0.795±0.004•	0.793±0.003•	0.670±0.002•	13/0/0
RML	–	–	–	–	–	–	–
ML-KNN	0.590±0.005•	0.718±0.009•	0.710±0.009•	0.887±0.004•	0.849±0.003•	0.767±0.001•	13/0/0
CLR	0.740±0.002∘	0.891±0.003	0.882±0.002	0.944±0.001	0.906±0.001•	0.805±0.001•	8/3/2
ECC	0.697±0.006•	0.864±0.002•	0.855±0.003•	0.945±0.002•	0.921±0.001•	0.826±0.001•	12/1/0
RAKEL	0.552±0.002•	0.728±0.003•	0.721±0.003•	0.872±0.005•	0.859±0.002•	0.737±0.001•	13/0/0

* MACRO-AUC not applicable to RML, which does not yield real-valued outputs on each class label [Petterson and Caetano, 2010].

Experimental Results – **AUC**

Table 4: Performance of each comparing algorithm (mean±std. deviation) in terms of *macro-averaging AUC* (MACRO-AUC). In addition, •/∘ indicates whether COCOA is statistically superior/inferior to the comparing algorithm on each data set (pairwise t-test at 1% significance level).

Algorithm	CAL500	Emotions	Medical	Enron	Scene	Yeast	Slashdot
COCOA	0.557±0.005	0.843±0.010	0.958±0.006	0.731±0.006	0.943±0.003	0.710±0.006	0.736±0.005
USAM	0.514±0.005•	0.708±0.019•	0.855±0.012•	0.606±0.010•	0.790±0.009•	0.578±0.006•	0.617±0.004•
USAM-EN	0.513±0.004•	0.708±0.015•	0.860±0.024•	0.600±0.004•	0.788±0.009•	0.583±0.006•	0.618±0.004•
SMOTE	0.513±0.005					006•	0.688±0.008•
SMOTE-EN	0.513±0.00					07•	0.686±0.008•
RML	–						–
ML-KNN	0.516±0.00					04•	0.676±0.006•
CLR	0.561±0.00					04•	0.698±0.009•
ECC	0.549±0.00					06•	0.706±0.009•
RAKEL	0.528±0.00					4•	0.612±0.002•

Algorithm	Corel5k					ll	win/tie/loss counts for COCOA
COCOA	0.719±0.004	0.889±0.003	0.882±0.002	0.957±0.002	0.930±0.001	0.843±0.001	/
USAM	0.5					0.655±0.004•	13/0/0
USAM-EN	0.					.654±0.006•	13/0/0
SMOTE	0.					.669±0.002•	13/0/0
SMOTE-EN	0.					.670±0.002•	13/0/0
RML						–	–
ML-KNN	0.					.767±0.001•	13/0/0
CLR	0.						8/3/2
ECC	0.					0.001•	12/1/0
RAKEL	0.					.737±0.001•	13/0/0

* MACRO-AUC not applicable to RML, which does not yield real-valued outputs on each class label [Petterson and Caetano, 2010].

COCOA significantly outperforms the first series of comparing algorithms in all cases

COCOA is outperformed by CLR in only two cases, and achieves superior or at least comparable performance in the rest cases

Experimental Results – **Further Analysis**

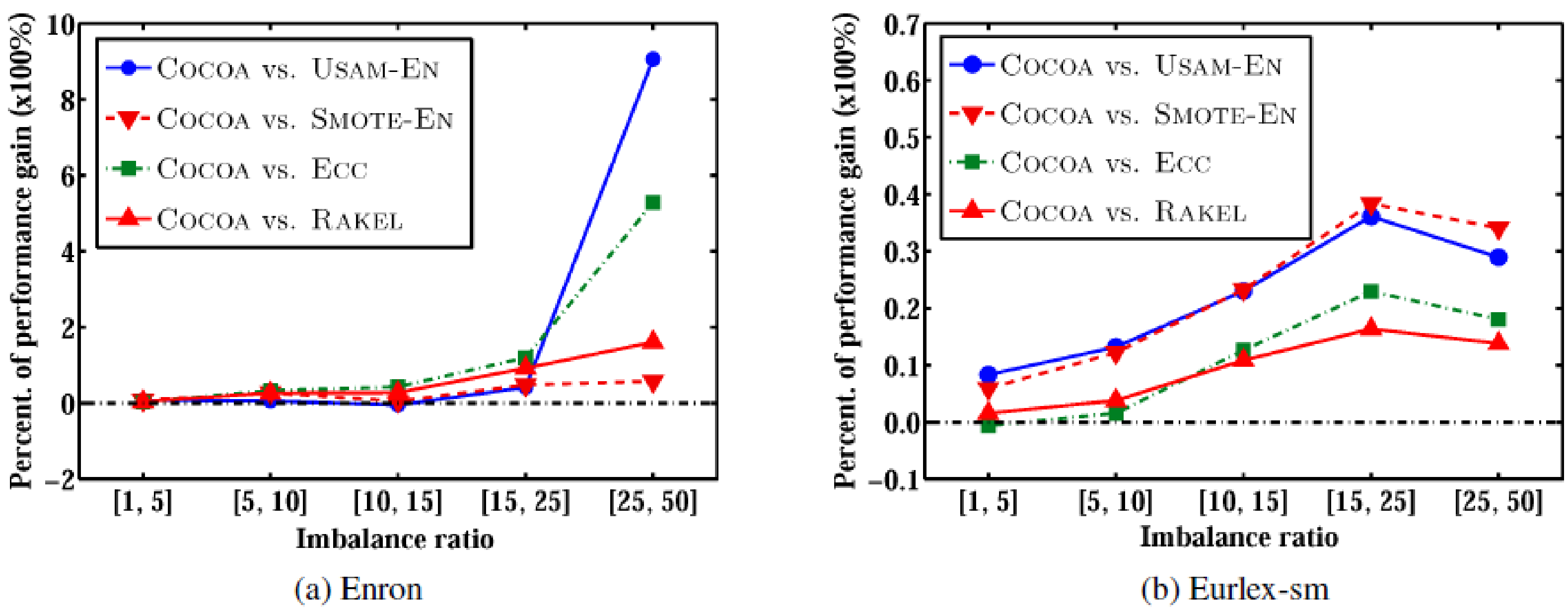


Figure 1: Performance gain between Cocoa and the comparing algorithm (PG_k) changes as the level of imbalance ratio (I_k) increases. On either data set, the performance of each algorithm is evaluated based on F-measure.

H-axis: Level of imbalance ratio

V-axis: Performance gain between Cocoa and the comparing algorithm

Experimental Results – **Further Analysis**

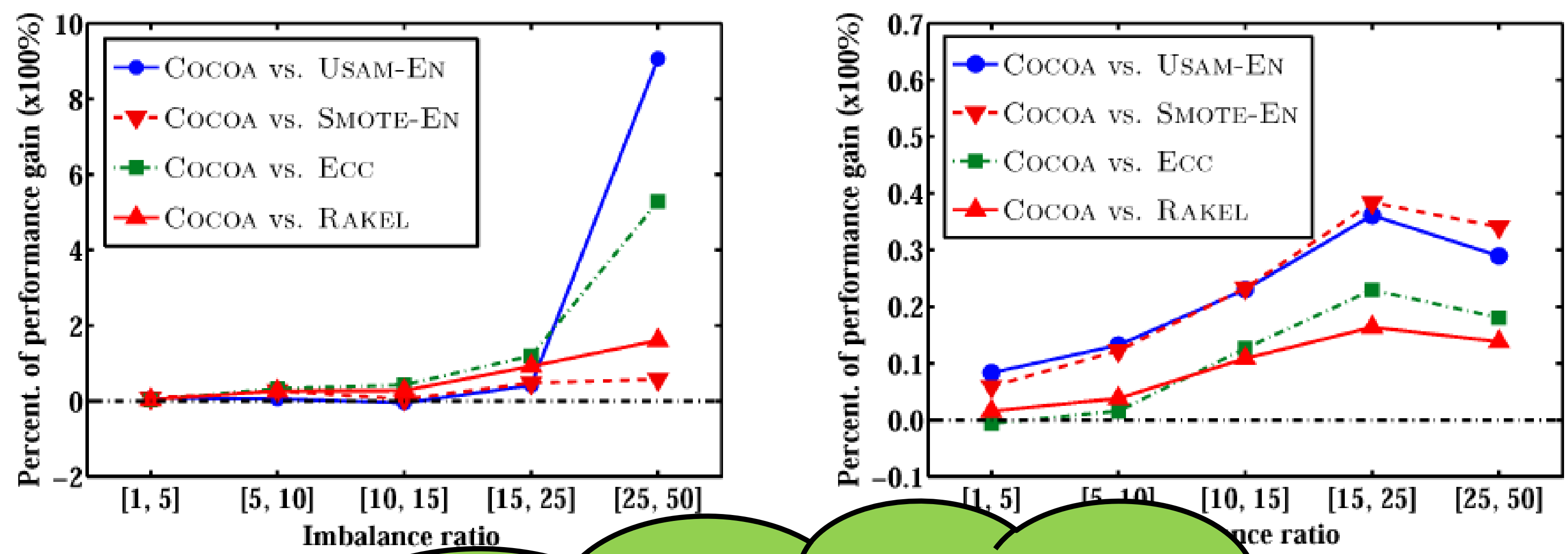


Figure 1: Performance g… of imbalance ratio (I_k) increases. On either…

performance advantage of Cocoa is more pronounced as the imbalance ratio increases

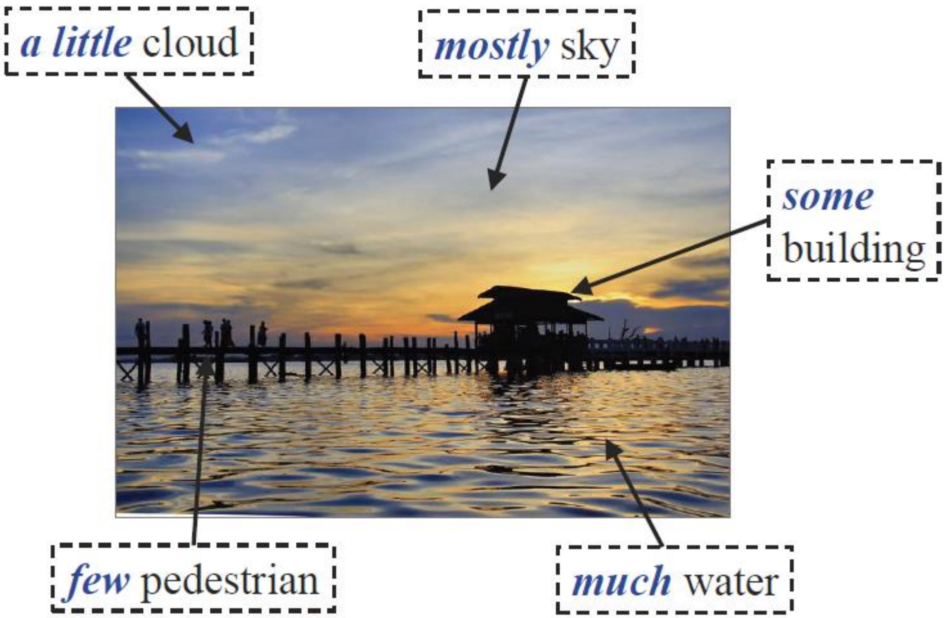
H-axis: Leve…

V-axis: Performa… …paring algorithm

Leverage Relative Labeling-Importance for Multi-Label Learning

Labeling-Importance for MLL

Labeling-importance is relative by nature



An image annotated with multiple labels *sky*, *water*, *building* and *cloud*

(Implicit) relative importance

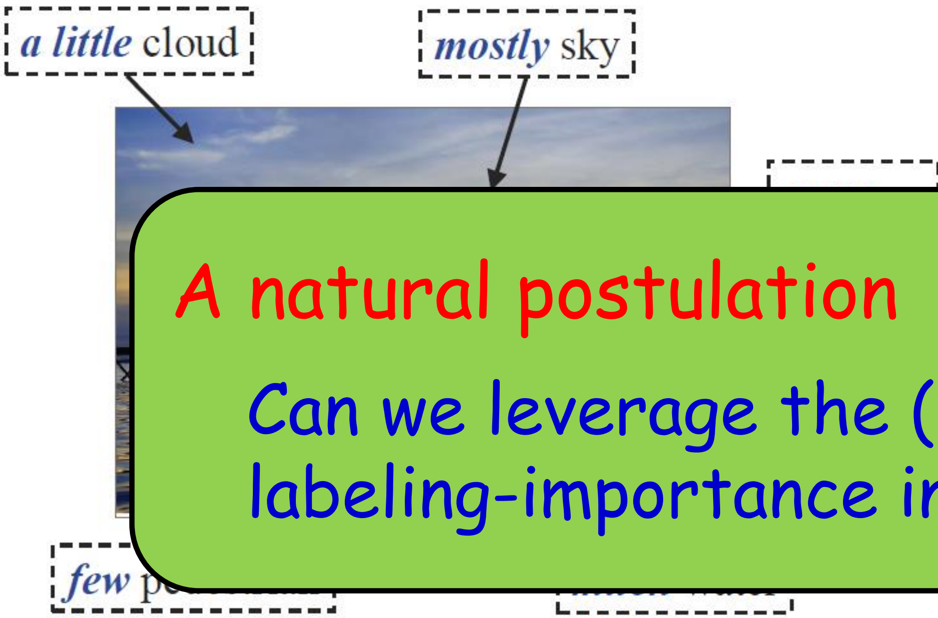
$$sky \succ water \succ building \succ cloud$$

Multi-category document ⇒ *Different topical importance*

Multi-functionality gene ⇒ *Different expression level*

Labeling-Importance for MLL

Labeling-importance is relative by nature



An image annotated with multiple labels *sky*, *water*, *building* and *cloud*

A natural postulation

Can we leverage the (implicit) relative labeling-importance information for MLL?

Multi-category document ⇨ *Different topical importance*

Multi-functionality gene ⇨ *Different expression level*

Relative Labeling-Importance (RLI)

Definition: *Relative Labeling-Importance (RLI) Degree*

Given any instance $\boldsymbol{x} \in \mathcal{X}$, the RLI degree of label $y_l \in \mathcal{Y}$ for $\boldsymbol{x}$ is denoted as $\mu_{\boldsymbol{x}}^{y_l}$, which satisfies the following constraints:

(i) ***non-negativity***: $\mu_{\boldsymbol{x}}^{y_l} \geq 0$

(ii) ***normalization***: $\sum_{y_l \in \mathcal{Y}} \mu_{\boldsymbol{x}}^{y_l} = 1$

The Reliab Approach

Implicit RLI degree estimation	+	Prediction Model Induction

Relative Labeling-Importance (RLI)

Definition: *Relative Labeling-Importance (RLI) Degree*

Given any instance $x \in \mathcal{X}$, *the RLI degree of label* $y_l \in \mathcal{Y}$ *for* … *h satisfies* … :

≥ 0

$\sum_{y_l \in \mathcal{Y}} \mu$ …

Iterative label propagation on weighted graph

Regularized maximum entropy with RLI degree

The RELIAB Approach

Implicit RLI degree estimation + Prediction Model Induction

Implicit RLI Degree Estimation

Weighted Graph Construction

$$G = (V, E, W)$$

- $V = \{\boldsymbol{x}_i \mid 1 \leq i \leq m\}$
- $E = \{(\boldsymbol{x}_i, \boldsymbol{x}_j) \mid j \neq i\}$

fully-connected graph over all the training examples

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

$$\forall_{i,j=1}^{m}: \quad w_{ij} = \begin{cases} \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2\sigma^2}\right), & \text{if } i \neq j \\ 0, & \text{if } i = j \end{cases}$$

$$\mathbf{D} = \mathrm{diag}[d_1, d_2, \ldots, d_m] \qquad d_i = \sum_{j=1}^{m} w_{ij}$$

Implicit RLI Degree Estimation (Cont.)

Iterative Label Propagation

Set the label propagation matrix: $\mathbf{P} = \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$

Assume a matrix $\mathbf{F} = [f_{il}]_{m\times q}$ with non-negative entries

$f_{il} \geq 0:$ proportional to the labeling-importance $\mu_{\boldsymbol{x}_i}^{y_l}$ $(1 \leq i \leq q;\ \ y_l \in \mathcal{Y})$

initialize $\mathbf{F}^{(0)} = \mathbf{\Phi} = [\phi_{il}]_{m\times q}:\ \ \phi_{il} = \begin{cases} 1, & \text{if } y_l \in Y_i \\ 0, & \text{otherwise} \end{cases}$

Update **F** iteratively by propagating labeling-importance information

$$\mathbf{F}^{(t)} = \alpha\mathbf{P}\mathbf{F}^{(t-1)} + (1-\alpha)\mathbf{\Phi}$$

Converges to:

$$\mathbf{F}^{*} = (1-\alpha)(\mathbf{I} - \alpha\mathbf{P})^{-1}\mathbf{\Phi}$$

Prediction Model Induction

Estimated RLI Information

$$\mathcal{U} = \{\mu_{\boldsymbol{x}_i}^{y_l} \mid 1 \leq i \leq q;\ y_l \in \mathcal{Y}\}$$

$$\mathbf{F}^* \Longrightarrow \mu_{\boldsymbol{x}_i}^{y_l} = \frac{f_{il}^*}{\sum_{k=1}^{q} f_{ik}^*}$$

Maximum Entropy Classification Model

$$f(y_l \mid \boldsymbol{x}, \boldsymbol{\Theta}) = \frac{1}{Z(\boldsymbol{x})} \exp\left(\boldsymbol{\theta}_l^\top \boldsymbol{x}\right)$$

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_q]$$

$$Z(\boldsymbol{x}) = \sum_{k=1}^{q} \exp\left(\boldsymbol{\theta}_l^\top \boldsymbol{x}\right)$$

Prediction Model Induction (Cont.)

Objective Function

$$V(f,\mathcal{U},\mathcal{D}) = \boxed{V_{dis}(f,\mathcal{U})} + \beta \cdot \boxed{V_{emp}(f,\mathcal{D})}$$

How f fits the estimated RLI information

How f classifies training samples

KL divergence

$$V_{dis}(f,\mathcal{U}) = \sum_{i=1}^{m}\sum_{l=1}^{q}\left(\mu_{\boldsymbol{x}_i}^{y_l} \ln \frac{\mu_{\boldsymbol{x}_i}^{y_l}}{f(y_l \mid \boldsymbol{x}_i, \boldsymbol{\Theta})}\right)$$

Empirical ranking loss

Minimized by the quasi-newton L-BFGS algorithm

Experimental Setup – **Data Sets**

Table III
CHARACTERISTICS OF THE BENCHMARK MULTI-LABEL DATA SETS.

Data set	$\|\mathcal{S}\|$	$dim(\mathcal{S})$	$L(\mathcal{S})$	$F(\mathcal{S})$	$LCard(\mathcal{S})$	$LDen(\mathcal{S})$	$DL(\mathcal{S})$	$PDL(\mathcal{S})$	Domain
cal500	502	68	174	numeric	26.044	0.150	502	1.000	audio
emotions	593	72	6	numeric	1.868	0.311	27	0.046	audio
medical	978	1,449	45	nominal	1.245	0.028	94	0.096	text
llog	1,460	1,004	75	nominal	1.180	0.016	304	0.208	text
msra	1,868	898	19	numeric	6.315	0.332	947	0.507	image
image	2,000	294	5	numeric	1.236	0.247	20	0.010	image
scene	2.407	294	5	numeric	1.074	0.179	15	0.006	image
yeast	2.417	103	14	numeric	4.237	0.303	198	0.082	biology
slashdot	3,782	1,079	22	nominal	1.181	0.054	156	0.041	text
corel5k	5,000	499	374	nominal	3.522	0.009	3,175	0.635	image
rcv1-s1	6,000	500	101	nominal	2.880	0.029	1,028	0.171	text
rcv1-s2	6,000	500	101	nominal	2.634	0.026	954	0.159	text
rcv1-s3	6,000	500	101	nominal	2.614	0.026	939	0.156	text
rcv1-s4	6,000	500	101	nominal	2.484	0.025	816	0.136	text
rcv1-s5	6,000	500	101	nominal	2.642	0.026	946	0.158	text
bibtex	7,395	1836	159	nominal	2.402	0.015	2,856	0.386	text
mediamill	43,907	120	101	numeric	4.376	0.043	6,555	0.149	video

Seventeen benchmark multi-label data sets

regular-scale: 9; # large-scale: 8

Experimental Setup – **Algorithms & Evaluation**

Comparing Algorithms

RELIAB **versus**

- BR (*first-order*)
- CLR (*second-order*)
- ECC, RAKEL (*high-order*)

Evaluation Metrics

Example-based: *one-error, coverage, ranking loss, average*

Label-based: *macro-averaging F1, micro-averaging F1*

Evaluation Protocol

N-fold cross-validation + **Friedman test**

Experimental Results – **Regular-Scale**

Table IV
PREDICTIVE PERFORMANCE OF EACH COMPARING ALGORITHM (MEAN±STD. DEVIATION) ON THE NINE REGULAR-SCALE DATA SETS.

Comparing algorithm	cal500	emotions	medical	llog	msra	image	scene	yeast	slashdot
					One-error ↓				
RELIAB	**0.129±0.019**	**0.273±0.019**	**0.160±0.012**	**0.745±0.007**	**0.066±0.014**	**0.348±0.016**	**0.248±0.007**	**0.223±0.011**	0.509±0.014
BR	0.906±0.025	0.375±0.027	0.306±0.031	0.885±0.013	0.362±0.013	0.527±0.011	0.472±0.016	0.284±0.010	0.731±0.014
CLR	0.375±0.118	0.356±0.030	0.706±0.149	0.883±0.023	0.152±0.009	0.502±0.016	0.367±0.017	0.272±0.012	0.978±0.003
ECC	0.255±0.028	0.353±0.040	0.187±0.016	0.794±0.011	0.211±0.011	0.475±0.011	0.378±0.015	0.261±0.010	**0.476±0.015**
RAKEL	0.672±0.029	0.394±0.027	0.252±0.025	0.876±0.015	0.288±0.014	0.498±0.013	0.440±0.016	0.297±0.012	0.596±0.011
					Coverage ↓				
RELIAB	**0.744±0.008**	**0.304±0.014**	**0.045±0.007**	**0.156±0.005**	**0.545±0.012**	**0.204±0.005**	**0.099±0.003**	**0.453±0.007**	**0.138±0.002**
BR	0.877±0.009	0.364±0.015	0.117±0.018	0.380±0.006	0.716±0.004	0.297±0.009	0.209±0.010	0.479±0.007	0.261±0.009
CLR	0.792±0.014	0.351±0.016	0.134±0.026	0.234±0.019	0.636±0.004	0.285±0.009	0.119±0.004	0.496±0.006	0.271±0.004
ECC	0.796±0.008	0.356±0.013	0.052±0.007	0.195±0.006	0.665±0.004	0.271±0.008	0.144±0.008	0.479±0.006	**0.138±0.006**
RAKEL	0.958±0.003	0.386±0.016	0.113±0.012	0.360±0.007	0.698±0.006	0.293±0.008	0.190±0.009	0.573±0.008	0.219±0.005
					Ranking loss ↓				
RELIAB	**0.179±0.003**	**0.165±0.011**	**0.030±0.006**	**0.121±0.004**	**0.134±0.008**	**0.185±0.006**	**0.081±0.002**	**0.171±0.006**	0.122±0.002
BR	0.266±0.005	0.233±0.016	0.089±0.013	0.329±0.005	0.287±0.004	0.309±0.010	0.230±0.012	0.191±0.005	0.242±0.009
CLR	0.248±0.029	0.222±0.014	0.114±0.024	0.197±0.017	0.207±0.003	0.291±0.010	0.125±0.005	0.200±0.005	0.258±0.005
ECC	0.218±0.004	0.227±0.017	0.036±0.006	0.156±0.005	0.238±0.004	0.273±0.010	0.154±0.008	0.193±0.005	**0.121±0.006**
RAKEL	0.342±0.003	0.260±0.016	0.087±0.009	0.309±0.006	0.260±0.004	0.303±0.009	0.209±0.010	0.254±0.006	0.198±0.005
					Average precision ↑				
RELIAB	**0.503±0.007**	**0.796±0.011**	**0.876±0.010**	**0.394±0.009**	**0.816±0.012**	**0.774±0.008**	**0.853±0.004**	**0.760±0.007**	0.613±0.010
BR	0.301±0.006	0.730±0.015	0.756±0.025	0.214±0.014	0.626±0.005	0.656±0.007	0.692±0.012	0.733±0.007	0.427±0.013
CLR	0.383±0.048	0.742±0.016	0.403±0.051	0.209±0.019	0.722±0.003	0.672±0.010	0.781±0.008	0.729±0.008	0.251±0.007
ECC	0.431±0.005	0.740±0.021	0.856±0.011	0.335±0.009	0.684±0.004	0.690±0.008	0.763±0.010	0.738±0.007	**0.631±0.012**
RAKEL	0.323±0.006	0.713±0.017	0.782±0.017	0.228±0.012	0.661±0.005	0.670±0.008	0.713±0.011	0.697±0.006	0.529±0.009
					Macro-averaging F1 ↑				
RELIAB	0.171±0.007	**0.642±0.009**	0.419±0.049	**0.128±0.032**	**0.565±0.015**	**0.586±0.014**	**0.664±0.031**	0.409±0.013	0.324±0.047
BR	0.172±0.003	0.564±0.022	0.422±0.032	0.110±0.022	0.454±0.005	0.473±0.006	0.541±0.011	0.392±0.006	0.290±0.011
CLR	0.108±0.037	0.575±0.018	0.175±0.048	0.105±0.032	0.481±0.007	0.472±0.007	0.581±0.008	0.398±0.008	0.104±0.003
ECC	0.116±0.005	0.557±0.022	**0.464±0.039**	0.121±0.026	0.455±0.007	0.473±0.012	0.575±0.015	0.393±0.006	**0.399±0.012**
RAKEL	**0.174±0.004**	0.569±0.021	0.443±0.040	0.119±0.020	0.435±0.010	0.486±0.011	0.556±0.014	**0.420±0.006**	0.346±0.009
					Micro-averaging F1 ↑				
RELIAB	**0.468±0.006**	**0.642±0.008**	0.695±0.013	**0.182±0.014**	**0.683±0.012**	**0.577±0.016**	**0.644±0.029**	**0.637±0.004**	0.430±0.010
BR	0.331±0.004	0.574±0.023	0.643±0.028	0.130±0.007	0.546±0.005	0.474±0.006	0.536±0.010	0.613±0.006	0.281±0.012
CLR	0.286±0.084	0.581±0.018	0.270±0.136	0.101±0.043	0.604±0.006	0.472±0.007	0.568±0.007	0.610±0.006	0.011±0.002
ECC	0.353±0.005	0.566±0.024	**0.751±0.017**	0.149±0.015	0.575±0.003	0.472±0.012	0.568±0.014	0.617±0.006	**0.480±0.015**
RAKEL	0.353±0.007	0.576±0.020	0.689±0.022	0.148±0.010	0.576±0.006	0.486±0.012	0.546±0.012	0.613±0.007	0.378±0.012

Across all evaluation metrics

RELIAB

ranks 1st in 83.3% cases

ranks 2nd in 11.1% cases

Experimental Results – **Large-Scale**

Table V
PREDICTIVE PERFORMANCE OF EACH COMPARING ALGORITHM (MEAN±STD. DEVIATION) ON THE EIGHT LARGE-SCALE DATA SETS.

Comparing algorithm	corel5k	rcv1-s1	rcv1-s2	rcv1-s3	rcv1-s4	rcv1-s5	bibtex	mediamill
				One-error ↓				
RELIAB	0.795±0.009	0.510±0.005	**0.479±0.006**	**0.487±0.007**	**0.466±0.008**	**0.467±0.012**	**0.418±0.007**	0.192±0.007
BR	0.921±0.004	0.736±0.006	0.758±0.008	0.755±0.003	0.737±0.010	0.763±0.008	0.880±0.004	0.185±0.004
CLR	**0.748±0.011**	0.503±0.006	0.549±0.006	0.549±0.025	0.584±0.076	0.678±0.092	0.514±0.003	**0.147±0.002**
ECC	0.911±0.004	**0.490±0.005**	0.515±0.007	0.512±0.006	0.485±0.004	0.495±0.005	0.907±0.003	0.158±0.002
RAKEL	0.867±0.004	0.626±0.008	0.622±0.008	0.637±0.008	0.618±0.010	0.614±0.013	0.779±0.015	0.200±0.003
				Coverage ↓				
RELIAB	0.342±0.008	0.158±0.002	0.128±0.004	**0.130±0.004**	**0.118±0.005**	**0.123±0.004**	**0.113±0.003**	0.198±0.002
BR	0.757±0.007	0.411±0.004	0.377±0.006	0.366±0.003	0.314±0.005	0.366±0.004	0.434±0.007	0.136±0.001
CLR	**0.311±0.011**	**0.123±0.002**	**0.122±0.004**	**0.130±0.018**	0.152±0.044	0.204±0.041	0.136±0.002	**0.127±0.001**
ECC	0.889±0.004	0.176±0.002	0.168±0.006	0.166±0.003	0.148±0.003	0.160±0.004	0.460±0.006	0.132±0.001
RAKEL	0.855±0.005	0.457±0.011	0.387±0.009	0.370±0.005	0.354±0.009	0.380±0.010	0.401±0.008	0.503±0.001
				Ranking loss ↓				
RELIAB	0.152±0.005	0.069±0.001	**0.054±0.002**	**0.055±0.002**	**0.050±0.002**	**0.051±0.001**	**0.063±0.002**	0.058±0.001
BR	0.416±0.006	0.214±0.002	0.213±0.004	0.207±0.002	0.169±0.004	0.204±0.004	0.280±0.002	0.036±0.001
CLR	**0.147±0.007**	**0.052±0.001**	0.055±0.002	0.063±0.015	0.083±0.037	0.125±0.035	0.080±0.002	**0.033±0.001**
ECC	0.600±0.005	0.079±0.00	0.079±0.003	0.078±0.002	0.070±0.001	0.074±0.002	0.307±0.006	0.036±0.001
RAKEL	0.547±0.004	0.245±0.008	0.225±0.007	0.216±0.003	0.204±0.007	0.220±0.005	0.250±0.006	0.190±0.001
				Average precision ↑				
RELIAB	0.221±0.007	0.532±0.003	**0.583±0.006**	**0.583±0.005**	**0.607±0.002**	**0.589±0.007**	**0.562±0.003**	0.676±0.003
BR	0.122±0.003	0.334±0.003	0.340±0.008	0.340±0.002	0.372±0.007	0.342±0.007	0.186±0.005	0.738±0.001
CLR	**0.222±0.007**	**0.555±0.004**	0.542±0.004	0.527±0.040	0.459±0.013	0.312±0.014	0.469±0.002	**0.758±0.001**
ECC	0.093±0.004	0.528±0.004	0.536±0.004	0.538±0.005	0.565±0.001	0.547±0.004	0.151±0.004	0.750±0.001
RAKEL	0.125±0.002	0.371±0.005	0.401±0.006	0.398±0.004	0.425±0.006	0.405±0.003	0.249±0.007	0.573±0.001
				Macro-averaging F1 ↑				
RELIAB	**0.089±0.008**	**0.253±0.003**	**0.260±0.009**	**0.266±0.021**	**0.258±0.015**	**0.271±0.006**	**0.300±0.009**	0.053±0.001
BR	0.073±0.006	0.187±0.004	0.167±0.006	0.171±0.008	0.170±0.006	0.167±0.004	0.127±0.003	0.197±0.003
CLR	0.074±0.012	0.233±0.008	0.221±0.006	0.213±0.032	0.157±0.073	0.088±0.079	0.247±0.003	0.171±0.002
ECC	0.062±0.009	0.198±0.009	0.174±0.004	0.174±0.015	0.185±0.013	0.184±0.009	0.101±0.002	0.163±0.002
RAKEL	0.079±0.007	0.194±0.007	0.174±0.005	0.174±0.005	0.180±0.009	0.188±0.003	0.177±0.007	**0.206±0.002**
				Micro-averaging F1 ↑				
RELIAB	**0.178±0.008**	**0.428±0.012**	**0.459±0.007**	**0.449±0.010**	**0.472±0.005**	**0.462±0.007**	**0.378±0.015**	0.502±0.005
BR	0.120±0.002	0.291±0.002	0.282±0.005	0.279±0.002	0.298±0.002	0.289±0.005	0.128±0.003	0.576±0.001
CLR	0.113±0.023	0.392±0.005	0.365±0.004	0.358±0.027	0.305±0.010	0.182±0.121	0.260±0.003	**0.585±0.001**
ECC	0.102±0.005	0.359±0.005	0.338±0.006	0.337±0.006	0.368±0.002	0.364±0.009	0.102±0.003	0.568±0.001
RAKEL	0.134±0.003	0.311±0.002	0.309±0.003	0.306±0.005	0.326±0.004	0.320±0.005	0.174±0.007	0.576±0.001

Across all evaluation metrics

RELIAB

ranks 1st in 68.7% cases

ranks 2nd in 16.7% cases

Experimental Results – **Friedman Test**

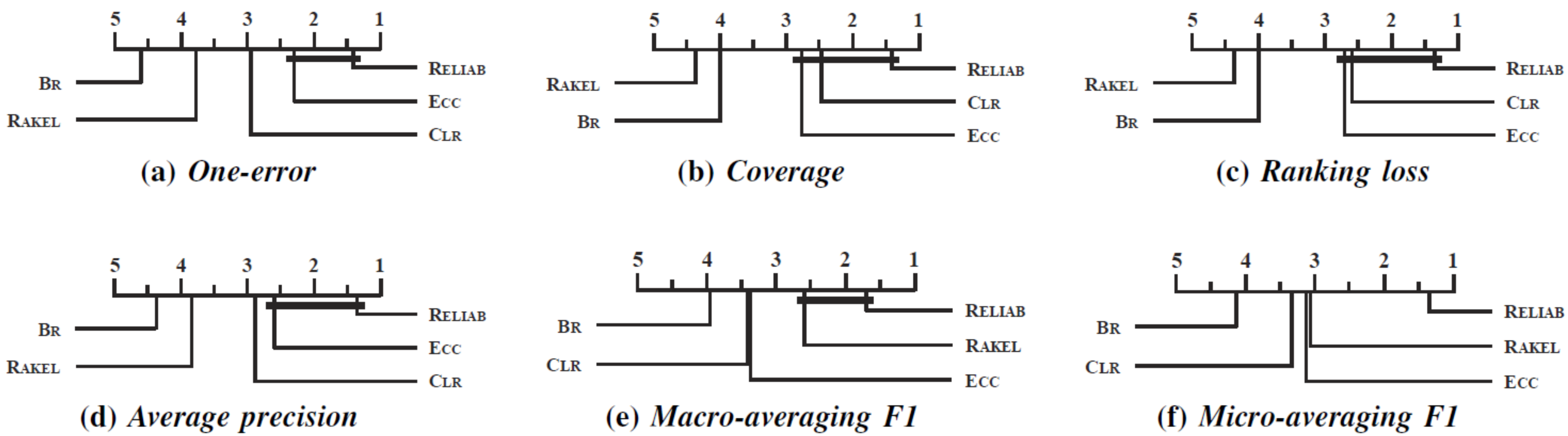


Figure 2. Comparison of RELIAB (control algorithm) against other comparing algorithms with the *Bonferroni-Dunn test*. Algorithms not connected with RELIAB in the CD diagram are considered to have significantly different performance from the control algorithm (CD=1.3547 at 0.05 significance level).

RELIAB

- ✓ *achieves **optimal (lowest) rank** in terms of each metric*
- ✓ *significantly **outperforms BR on all metrics***
- ✓ *significantly **outperforms CLR, ECC and RAKEL on 4, 2 and 5 metrics** respectively*

Outline

- Multi-Label Learning (MLL)
- Binary Relevance for MLL
- Our Recent Studies
  - Towards Class-Imbalance Aware MLL
  - Leverage Relative Labeling-Importance for MLL
- Conclusion

Conclusion

BR is arguably the most popular approach towards MLL

seminal papers on:

BR: ***1370+ citations***

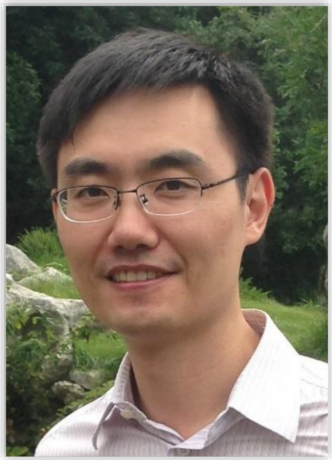
chaining-style BR: ***1200+ citations***

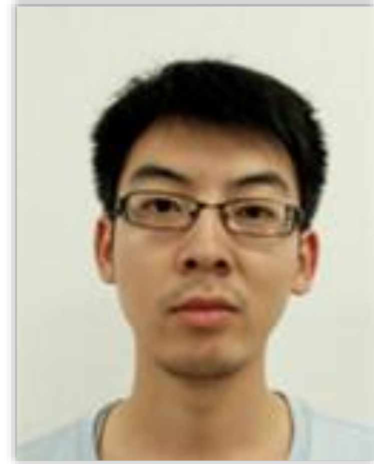
stacking-style BR: ***540+ citations & PAKDD 10-Year BPA***

To make BR work effectively, one should…

Exploiting Label Correlations	+	Exploring Inherent Properties

Joint Work With…





Prof. Xin Geng
Southeast University

Dr. Xu-Ying Liu
Southeast University

Mr. Yu-Kun Li
Southeast University
Baidu Inc.

Thanks!