Learning Discrete Representations via Information Maximizing Self-Augmented Training

Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, Masashi Sugiyama

Weihua Hu	Takeru Miyato	Seiya Tokui	Eiichi Matsumoto	Masashi Sugiyama
University of Tokyo	Preferred Networks	Preferred Networks	Preferred Networks	RIKEN
RIKEN	ATR	University of Tokyo	University of Tokyo	University of Tokyo

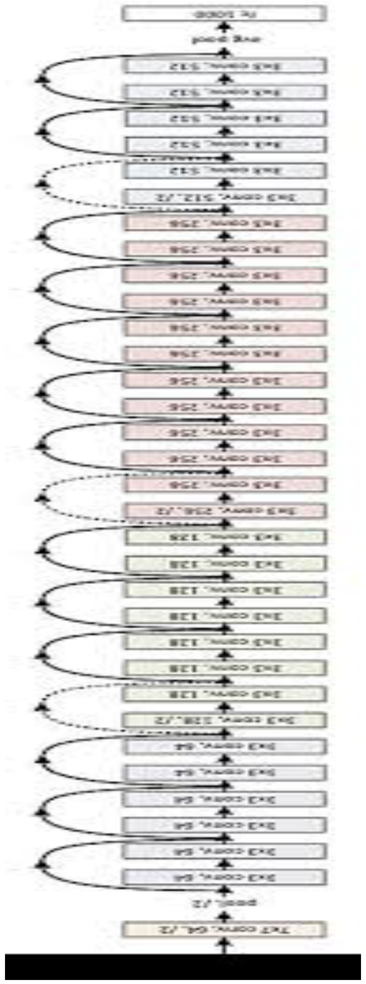
*Based on the work performed at Preferred Networks

Big success of supervised Deep Learning

(Very) deep neural networks

Huge number of Labeled data

+

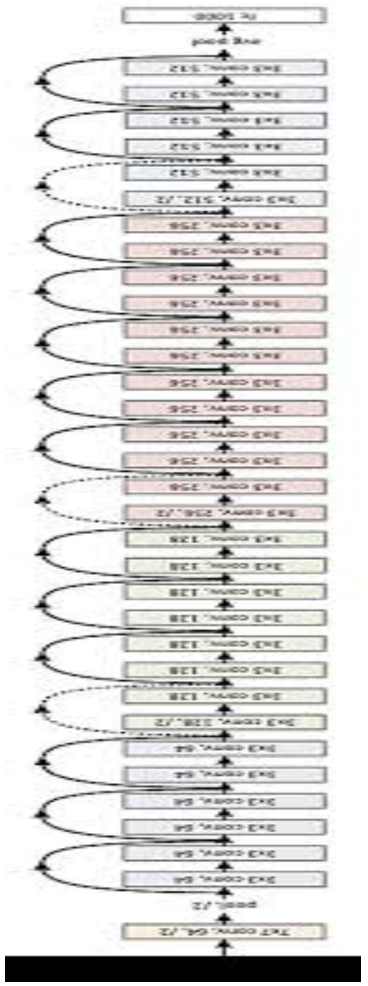


Big success of supervised Deep Learning

(Very) deep neural networks

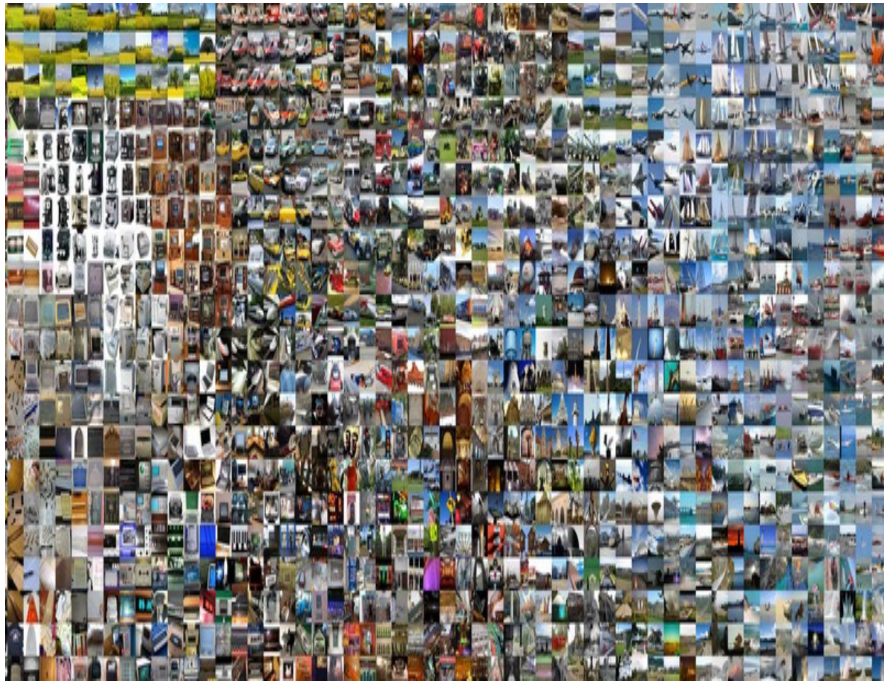
Huge number of Labeled data

+



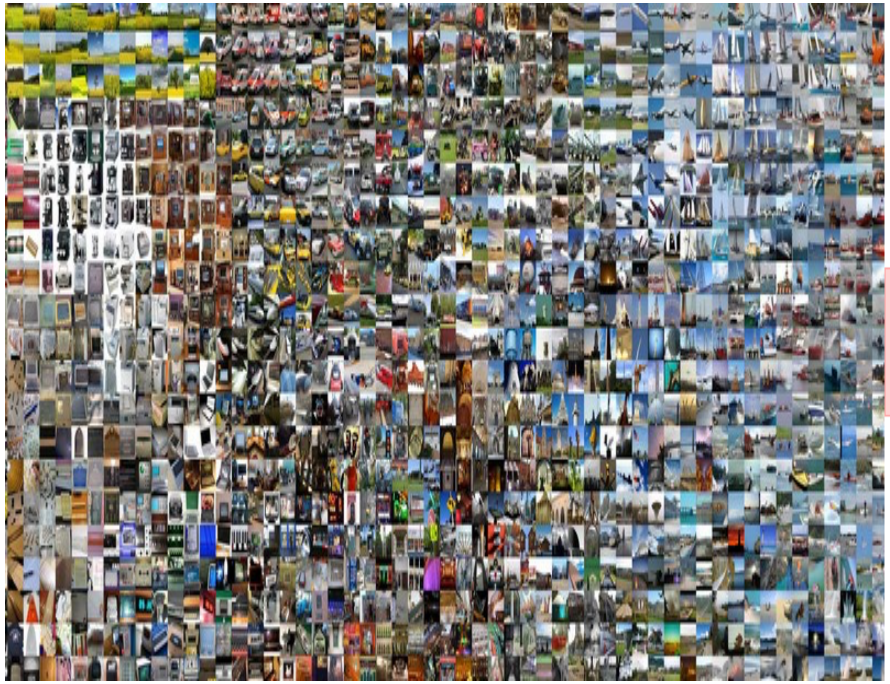
Unsupervised Discrete Representation Learning

Unlabeled data



Unsupervised Discrete Representation Learning

Unlabeled data

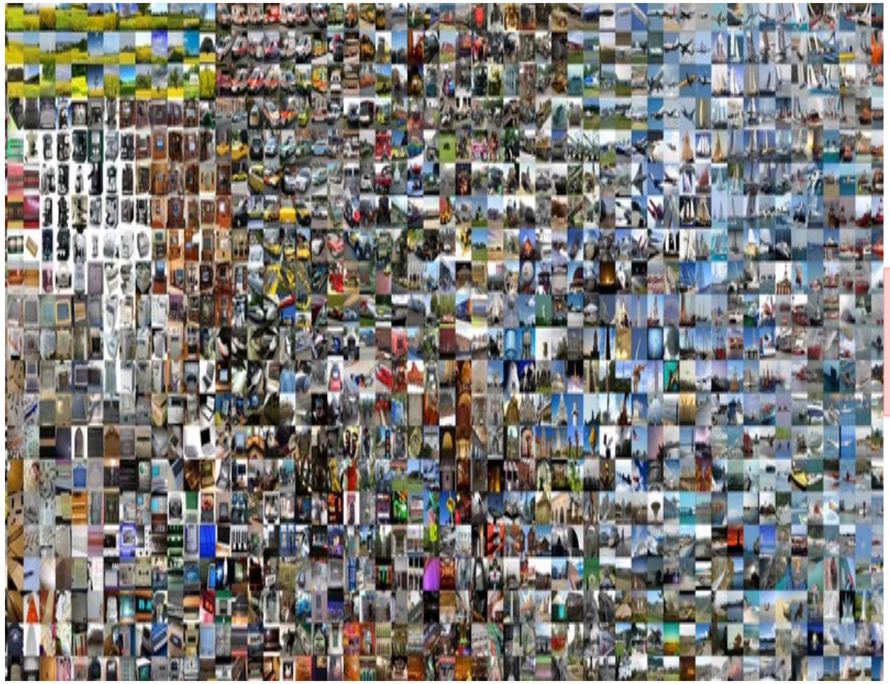


Learn to map

Discrete representations

Unsupervised Discrete Representation Learning

Unlabeled data



Learn to map

Discrete representations

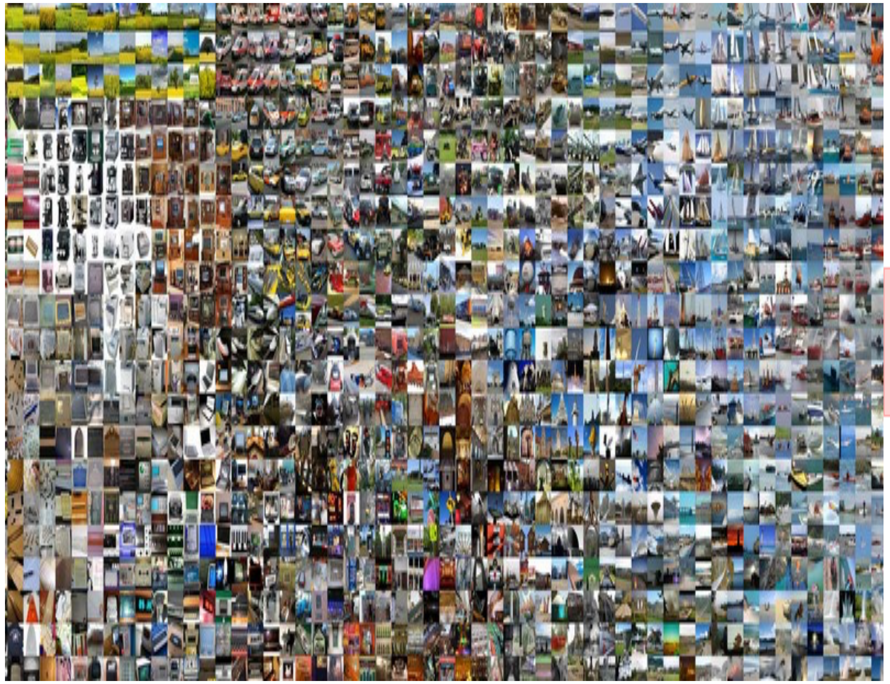
Clustering

Map to cluster assignments

0, 1, 5, 8, 9, 1, 3, 2, 4, 3,
9, 3, 2, 0, 2, 1, 4, 3, 1, 3

Unsupervised Discrete Representation Learning

Unlabeled data



Learn to map

Discrete representations

Clustering

Map to cluster assignments

0, 1, 5, 8, 9, 1, 3, 2, 4, 3,
9, 3, 2, 0, 2, 1, 4, 3, 1, 3

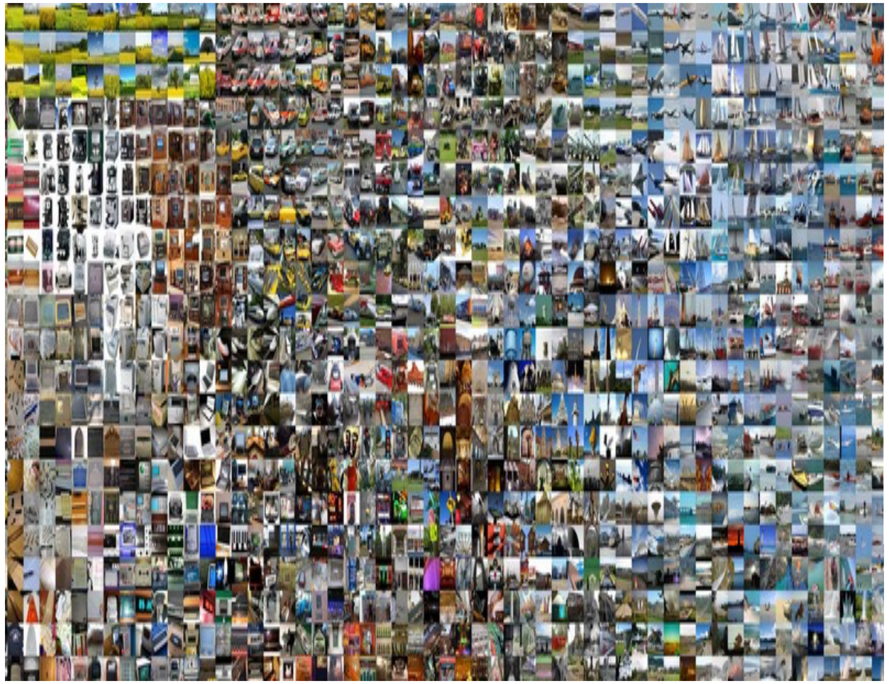
Hash Learning

Map to binary codes

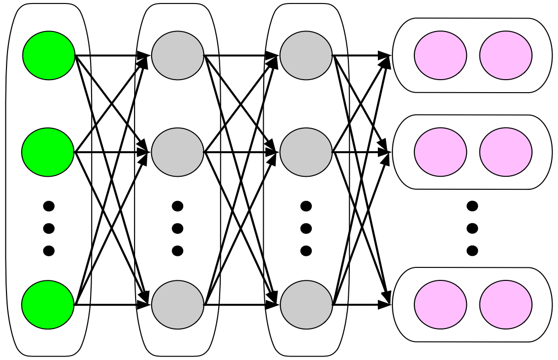
0001, 0101, 1110, 1111,
0000, 0111, 0000, 1011

Deep Neural Networks (DNN) are Promising

Unlabeled data



DNN



✓ **Flexible**

✓ **Scalable**

Discrete representations

Clustering

Map to cluster assignments

0, 1, 5, 8, 9, 1, 3, 2, 4, 3,
9, 3, 2, 0, 2, 1, 4, 3, 1, 3

Hash Learning

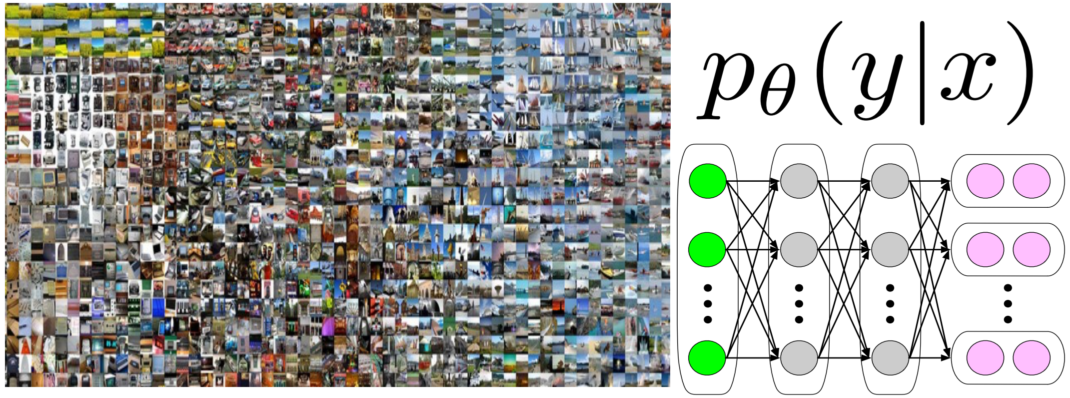
Map to binary codes

0001, 0101, 1110, 1111,
0000, 0111, 0000, 1011

Problem setting

- **Goal:** learn deep probabilistic classifier.

$p_\theta(y|x)$



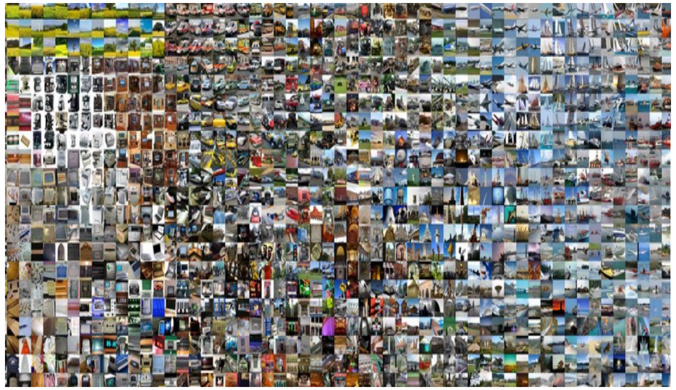
Discrete representations

1. Cluster assignment
2. Binary codes

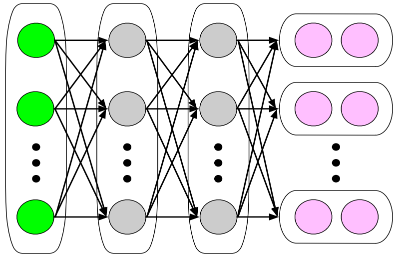
X Y

Problem setting

- **Goal:** learn deep probabilistic classifier.



$p_\theta(y|x)$



Discrete representations

1. Cluster assignment
2. Binary codes

X Y

Regularized Information Maximization (RIM) [Gomes+ 2010]

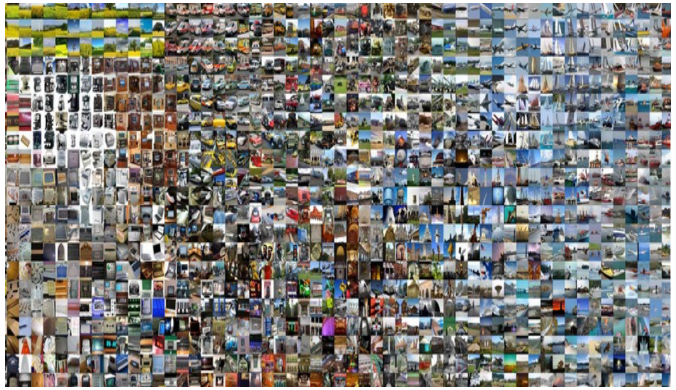
$$\max_{\theta} \boxed{I(X;Y)} - \boxed{\mathcal{R}(\theta)}$$

Regularization (weight-decay)

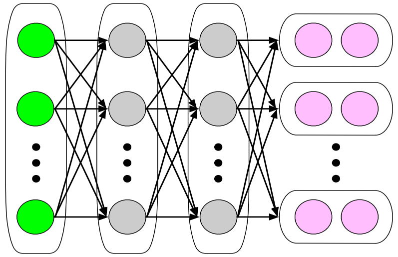
Mutual Information

Problem setting

- **Goal:** learn deep probabilistic classifier.



$p_\theta(y|x)$



X Y

Discrete representations

1. Cluster assignment
2. Binary codes

Regularized Information Maximization (RIM) [Gomes+ 2010]

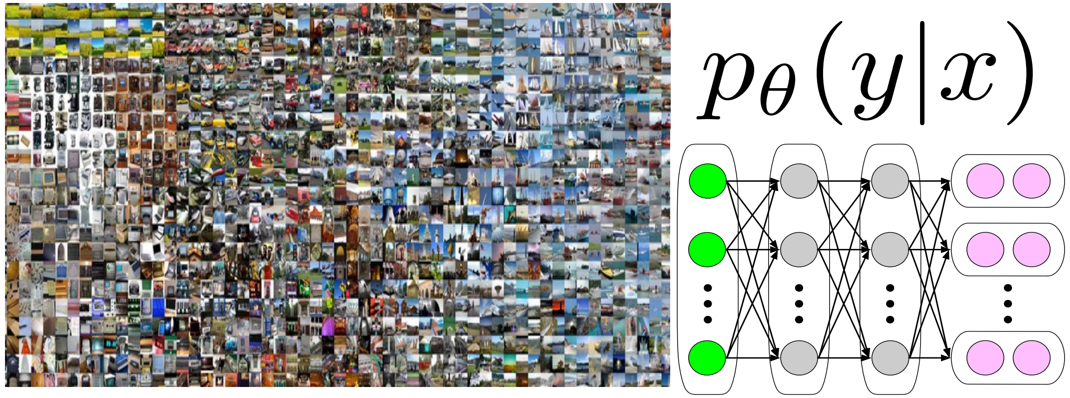
$$\max_\theta \boxed{I(X;Y)} - \boxed{\mathcal{R}(\theta)}$$

Regularization (weight-decay)

Only applicable to clustering.

Problem setting

- **Goal:** learn deep probabilistic classifier.



$p_\theta(y|x)$

Discrete representations

1. Cluster assignment
2. Binary codes

X Y

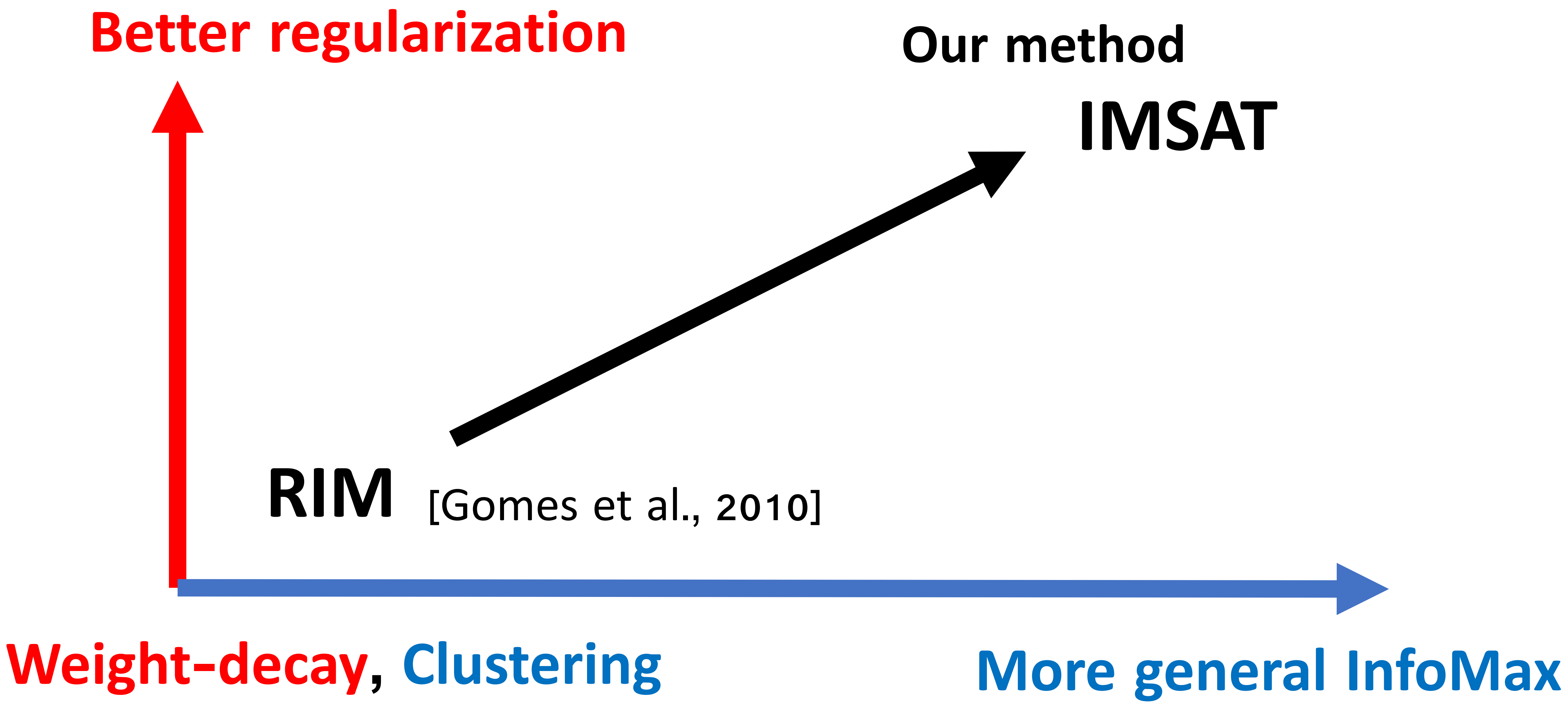
Regularized Information Maximization (RIM) [Gomes+ 2010]

$$\max_\theta \boxed{I(X;Y)} - \boxed{\mathcal{R}(\theta)}$$

Weight-decay is restrictive.

Only applicable to clustering.

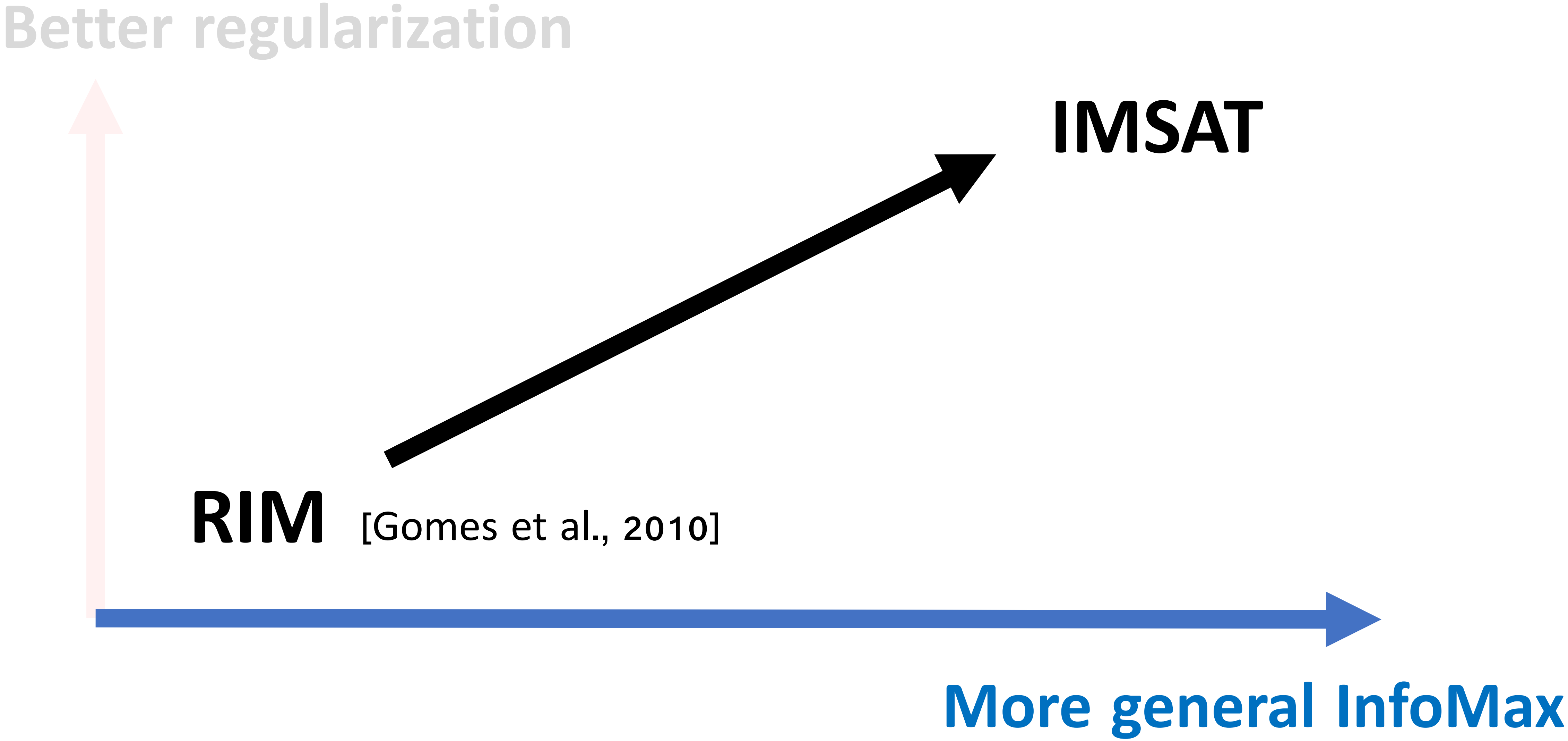
Our Contributions in a nutshell



Outline

1. Introduction
2. Proposed Method: IMSAT = IM + SAT
   - Information Maximization (IM)
   - Self-Augmented Training (SAT)
3. Experiments
4. Conclusions

Information Maximization



Information Maximization

Previous approach:

InfoMax clustering [Bridle et al., 1991, Gomes et al., 2010]:

Learn $p_\theta(y|x)$ via $\max_\theta I(X;Y)$

Information Maximization

For discrete representation learning?

Information Maximization

For discrete representation learning:

Our proposal:

Learn $p_\theta(y_1, \ldots, y_D|x)$ via $\max_\theta I(X; Y_1, \ldots, Y_D)$

Information Maximization

For discrete representation learning:

Our proposal:

Learn $p_\theta(y_1, \ldots, y_D | x)$ via $\max_\theta I(X; Y_1, \ldots, Y_D)$

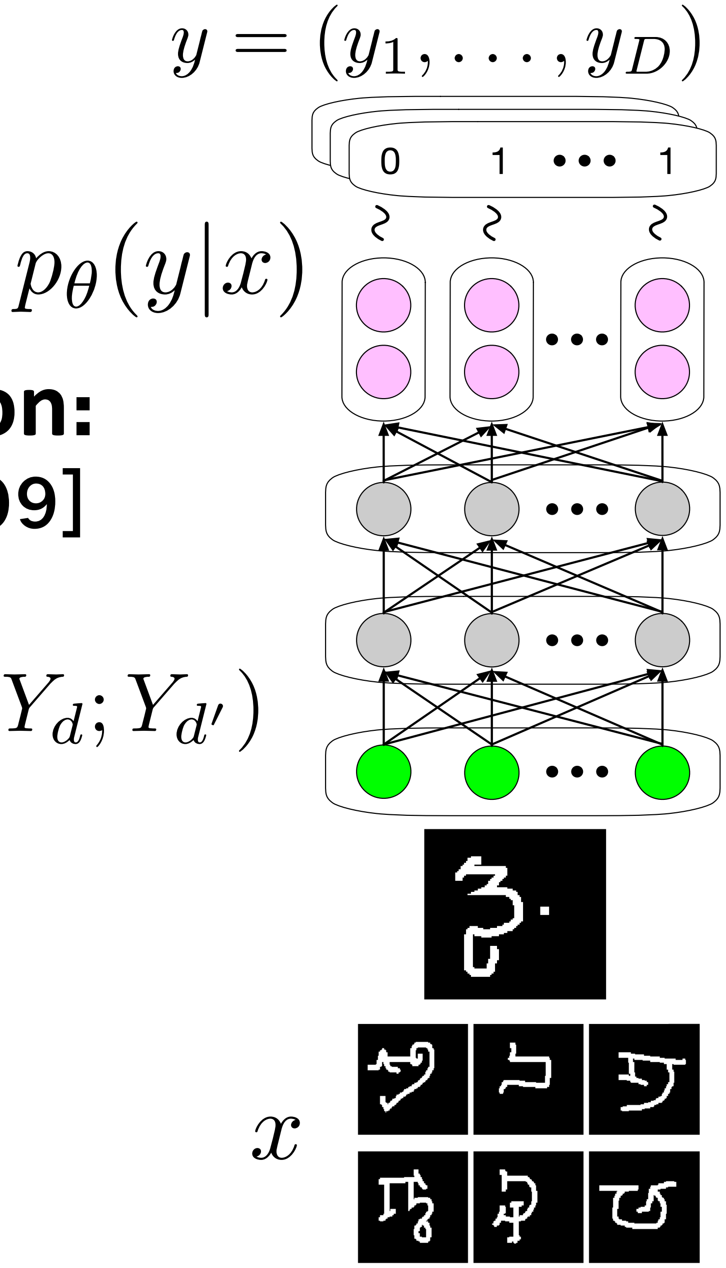
- **Challenge:** Combinatorial summation $\sum_{y_1} \sum_{y_2} \cdots \sum_{y_D}$

  → We need approximation!

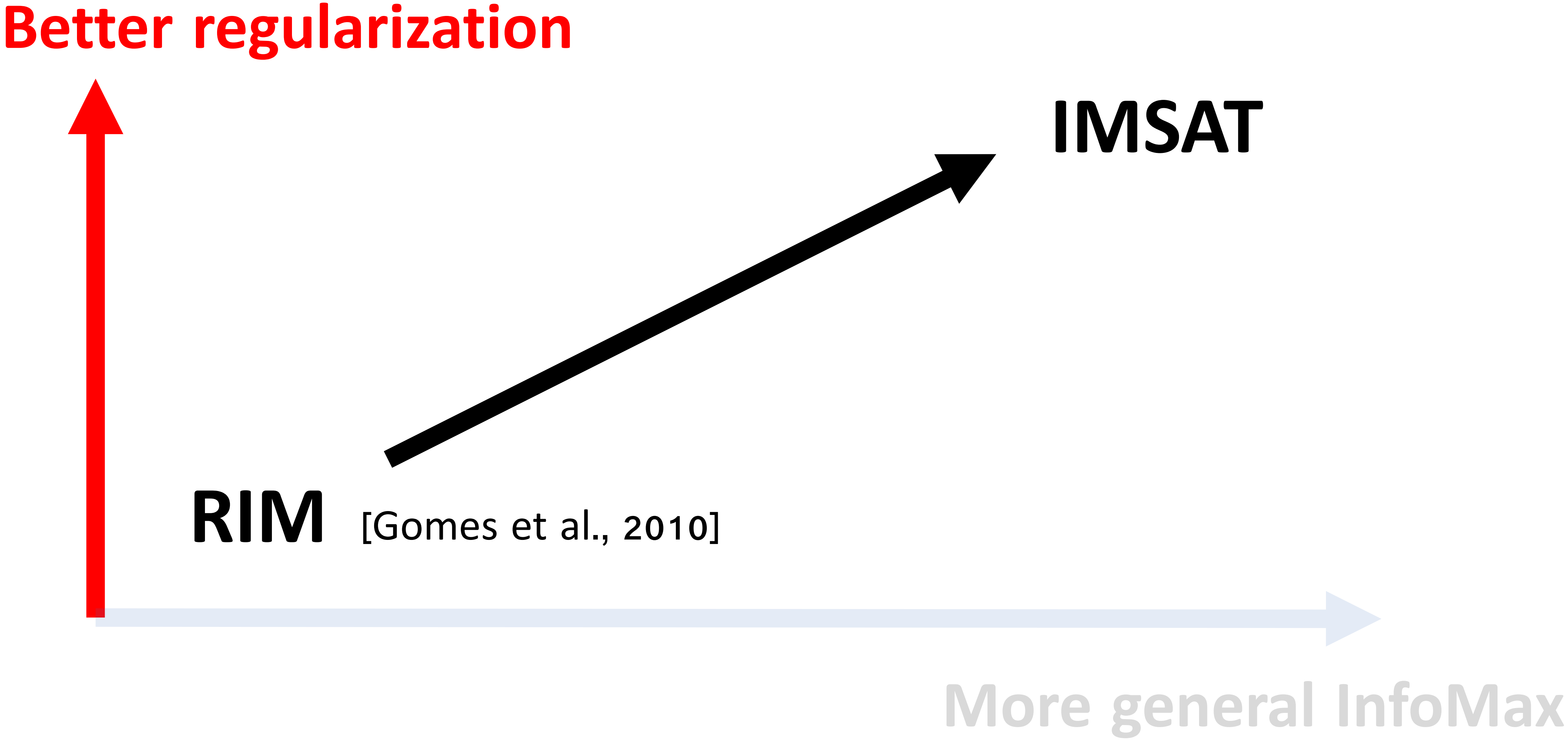
Information Maximization

Approximate **up to second order interaction:**
[Brown 2009]

$$I(X; Y_1, \ldots, Y_D) \approx \sum_{d=1}^{D} I(X; Y_d) - \sum_{1 \le d \ne d' \le D} I(Y_d; Y_{d'})$$

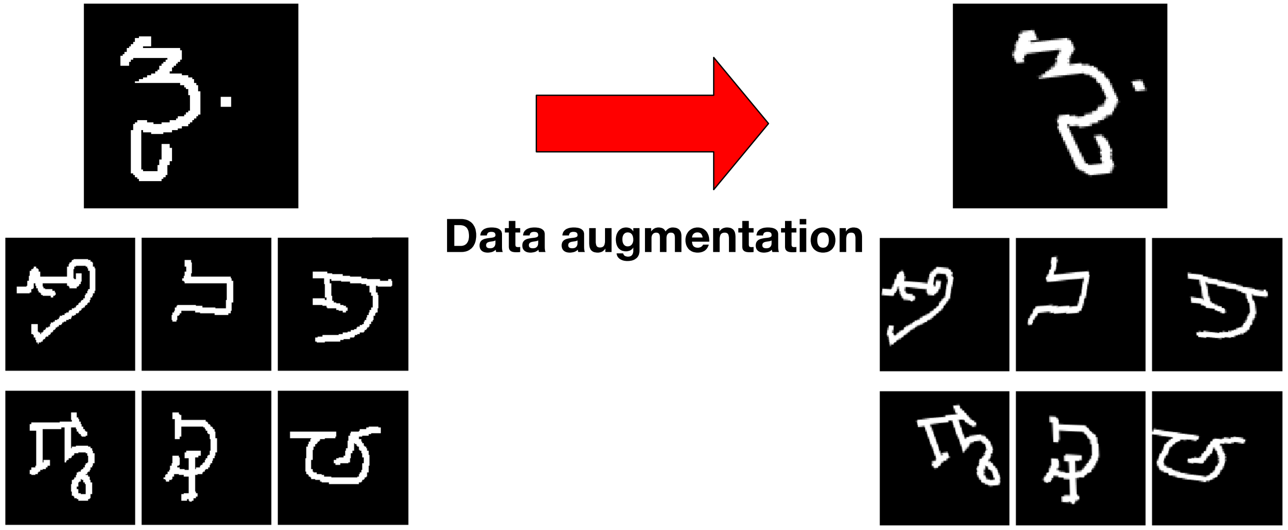


Self-Augmented Training (SAT)



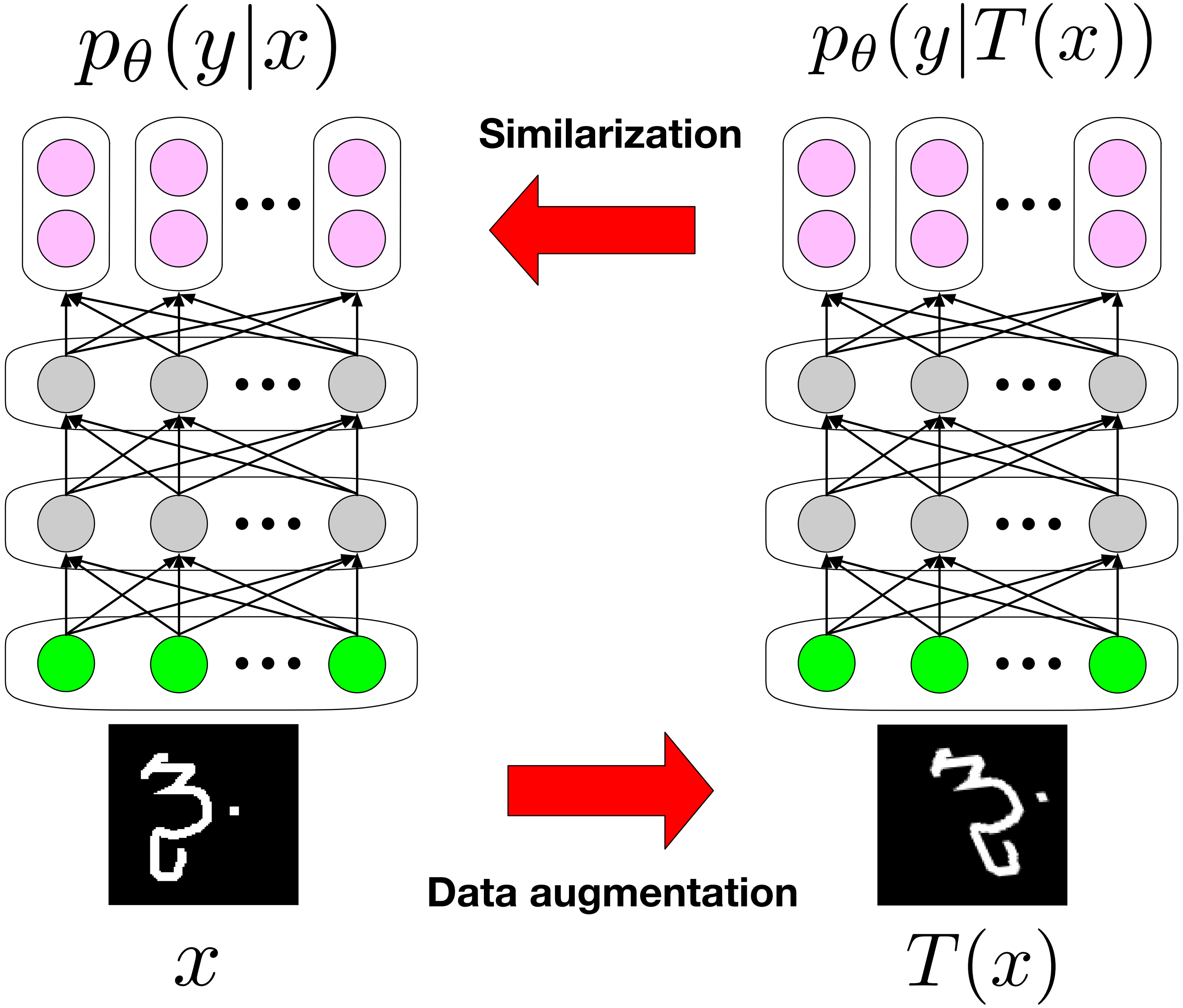
Self-Augmented Training (SAT)

Augmentation function $T(\cdot) : \mathcal{X} \to \mathcal{X}$
→ User-specified transformation that does not change the meaning of data



x $\qquad$ $T(x)$

Self-Augmented Training (SAT)



$$y = (y_1, y_2, \ldots, y_D)$$

$$\boxed{\mathrm{KL}[p_\theta(y|x) \| p_\theta(y|T(x))]}$$

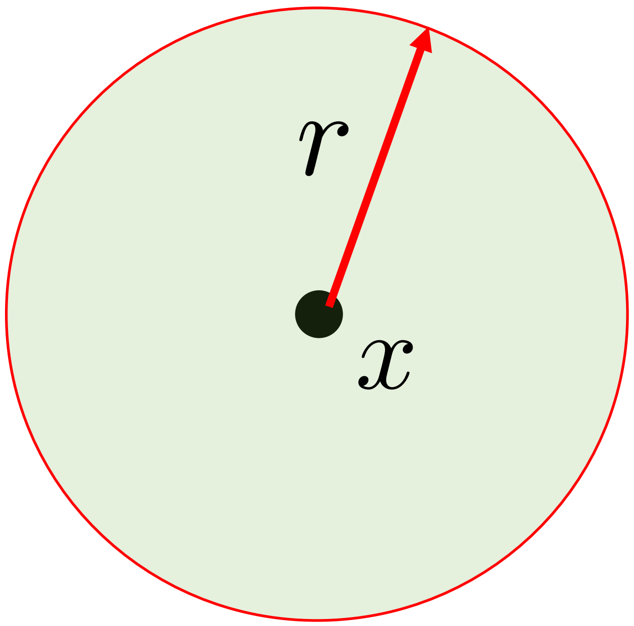
[Bachman et al., 2014; Miyato et al., 2016]

Self-Augmented Training (SAT)

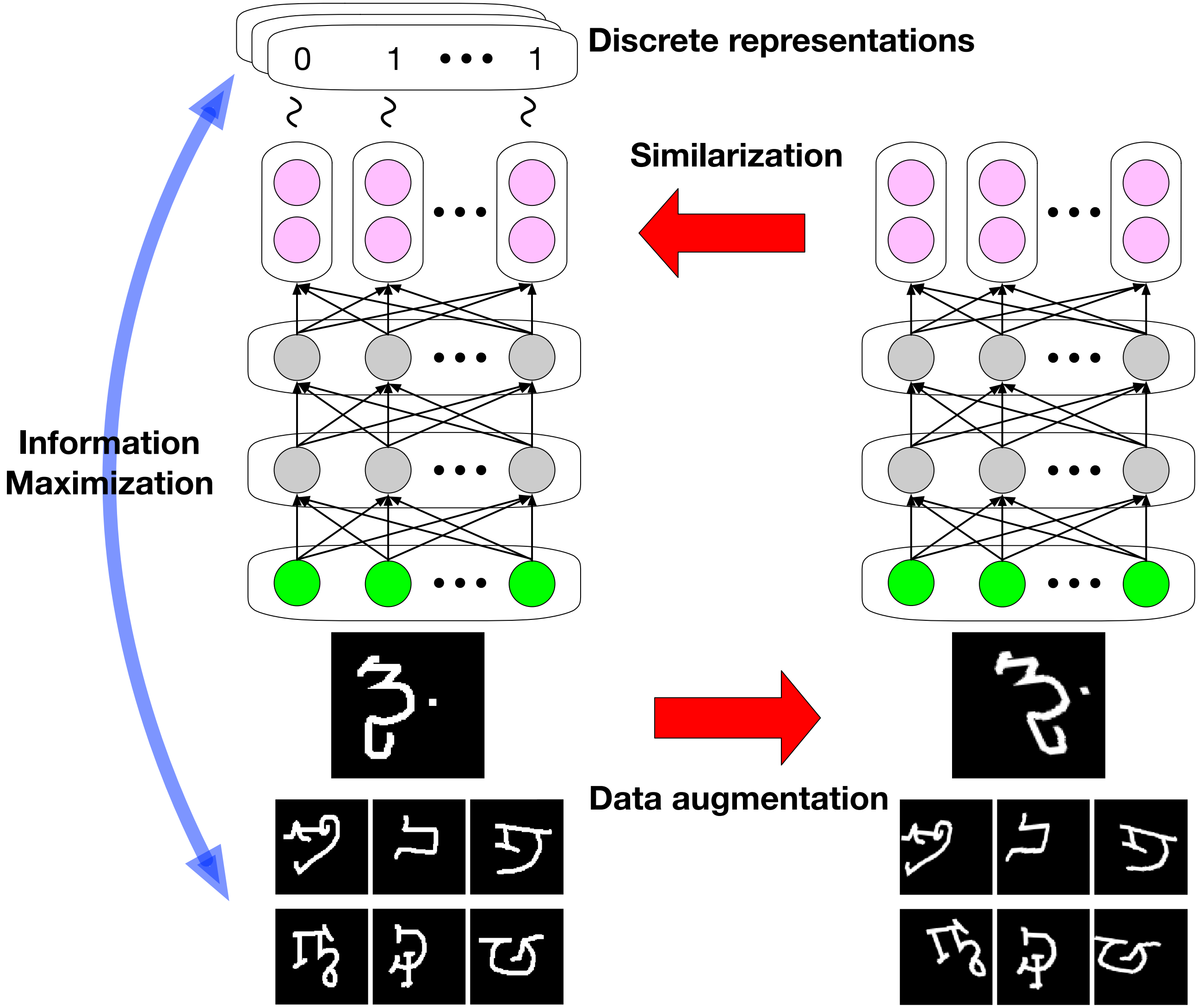
- **Local perturbation**

$$T(x) = x + r, \ \|r\|_2 = \epsilon$$

  - **Random Perturbation Training (RPT)**
    [Bachman et al., 2014]
  - **Virtual Adversarial Training (VAT)**
    [Miyato et al., 2016]



IMSAT = Information Maximizing + SAT



Outline

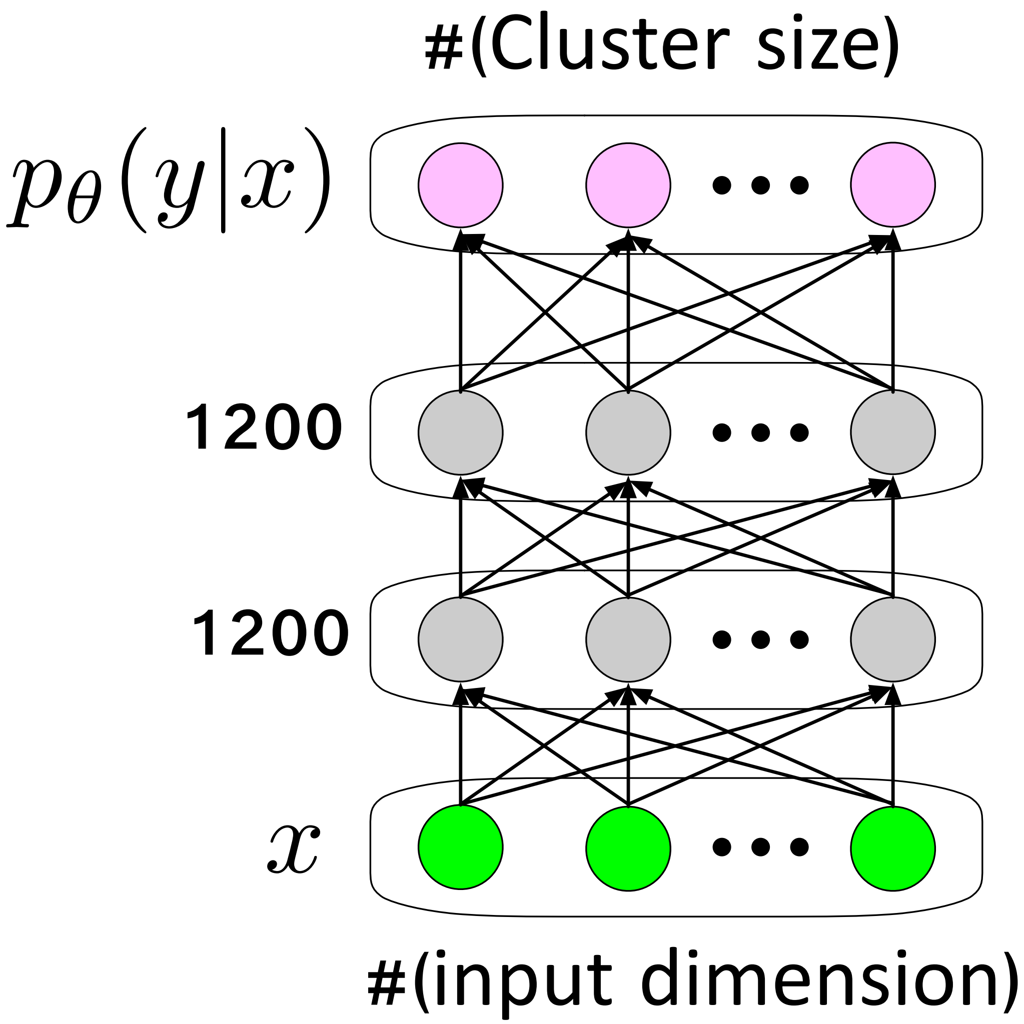
1. Introduction
2. Proposed Method: IMSAT
   - Information Maximization (IM)
   - Self-Augmented Training (SAT)
3. Experiments
4. Conclusions

Experiments (Clustering)

- Measure clustering accuracy

- Batch normalization
- ReLU activation
- Softmax output

※ Implementation available online
https://github.com/weihua916/imsat



Experiments (Clustering)

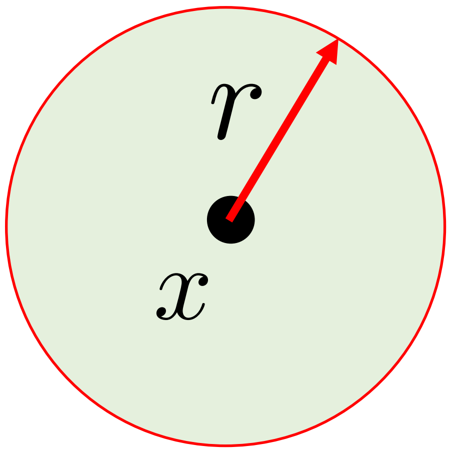
Method	MNIST	Omniglot	STL	CIFAR10	CIFAR100	SVHN	Reuters	20news
K-means	53.2	12.0	85.6	34.4	21.5	17.9	54.1	15.5
dAE+K-means	79.8 †	14.1	72.2	44.2	20.8	17.4	67.2	22.1
DEC [Xie et al., 2014]	84.3 †	5.7 (0.3)	78.1 (0.1)	**46.9 (0.9)**	14.3 (0.6)	11.9 (0.4)	67.3 (0.2)	30.8 (1.8)
Linear RIM	59.6 (2.3)	11.1 (0.2)	73.5 (6.5)	40.3 (2.1)	23.7 (0.8)	20.2 (1.4)	62.8 (7.8)	**50.9 (3.1)**
Linear IMSAT (VAT)	61.1 (1.9)	12.3 (0.2)	91.7 (0.5)	40.7 (0.6)	23.9 (0.4)	18.2 (1.9)	42.9 (0.8)	43.9 (3.3)
Deep RIM	58.5 (3.5)	5.8 (2.2)	92.5 (2.2)	40.3 (3.5)	13.4 (1.2)	26.8 (3.2)	62.3 (3.9)	25.1 (2.8)
IMSAT (RPT)	89.6 (5.4)	16.4 (3.1)	92.8 (2.5)	45.5 (2.9)	24.7 (0.5)	35.9 (4.3)	**71.9 (6.5)**	24.4 (4.7)
IMSAT (VAT)	**98.4 (0.4)**	**24.0 (0.9)**	**94.1 (0.4)**	45.6 (0.8)	**27.5 (0.4)**	**57.3 (3.9)**	**71.0 (4.9)**	31.1 (1.9)

- Tested on 8 benchmark datasets.
- Hyper-parameters are fixed throughout the datasets.

Experiments (Clustering)

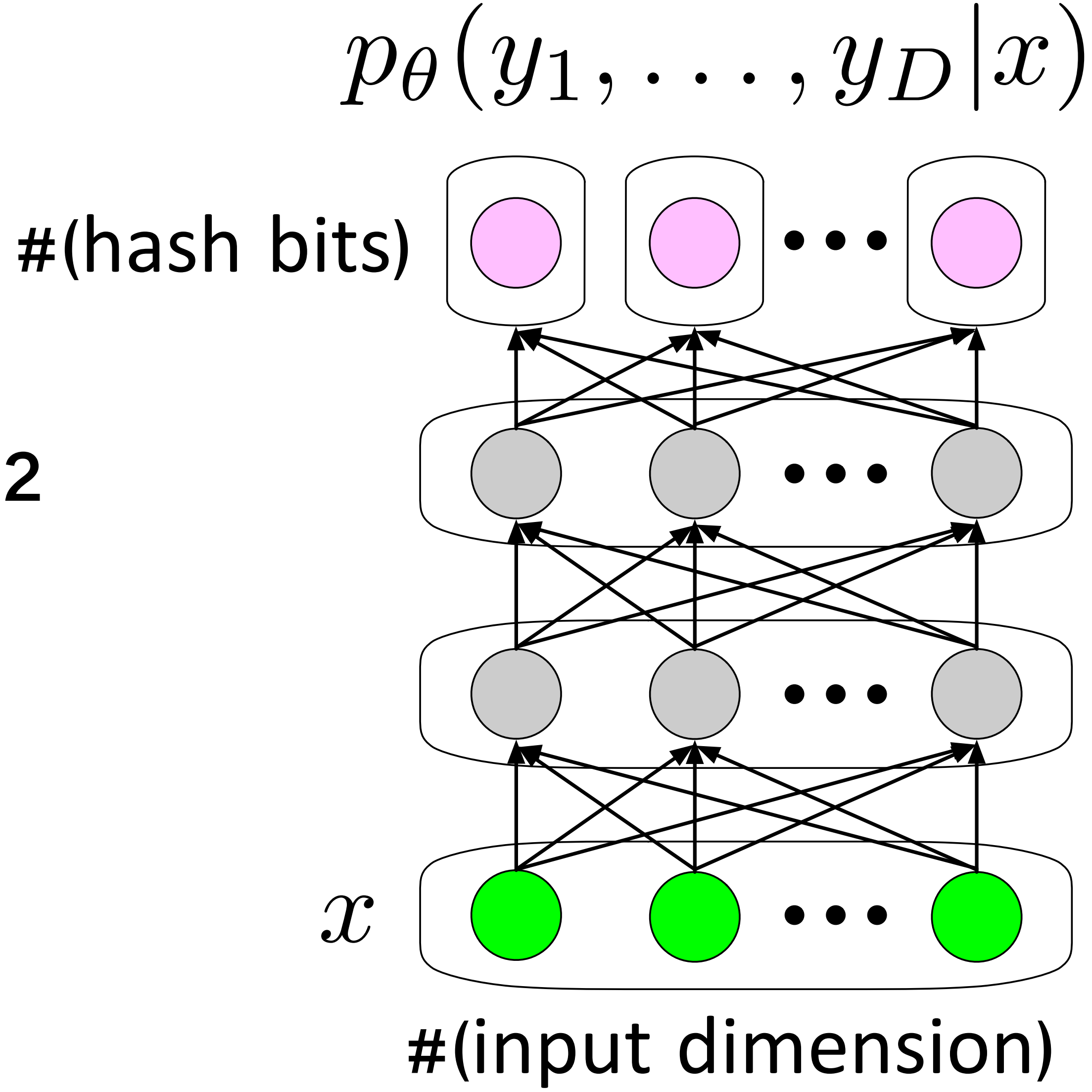
Method	MNIST	Omniglot	STL	CIFAR10	CIFAR100	SVHN	Reuters	20news
K-means	53.2	12.0	85.6	34.4	21.5	17.9	54.1	15.5
dAE+K-means	79.8 †	14.1	72.2	44.2	20.8	17.4	67.2	22.1
DEC	84.3 †	5.7 (0.3)	78.1 (0.1)	**46.9 (0.9)**	14.3 (0.6)	11.9 (0.4)	67.3 (0.2)	30.8 (1.8)
Linear RIM	59.6 (2.3)	11.1 (0.2)	73.5 (6.5)	40.3 (2.1)	23.7 (0.8)	20.2 (1.4)	62.8 (7.8)	**50.9 (3.1)**
Linear IMSAT (VAT)	61.1 (1.9)	12.3 (0.2)	91.7 (0.5)	40.7 (0.6)	23.9 (0.4)	18.2 (1.9)	42.9 (0.8)	43.9 (3.3)
Deep RIM	58.5 (3.5)	5.8 (2.2)	92.5 (2.2)	40.3 (3.5)	13.4 (1.2)	26.8 (3.2)	62.3 (3.9)	25.1 (2.8)
IMSAT (RPT)	89.6 (5.4)	16.4 (3.1)	92.8 (2.5)	45.5 (2.9)	24.7 (0.5)	35.9 (4.3)	**71.9 (6.5)**	24.4 (4.7)
IMSAT (VAT)	**98.4 (0.4)**	**24.0 (0.9)**	**94.1 (0.4)**	45.6 (0.8)	**27.5 (0.4)**	**57.3 (3.9)**	**71.0 (4.9)**	31.1 (1.9)

- Used perturbation as augmentation function.
- IMSAT (VAT) achieved state-of-the-art performance.



Experiments (Hash Learning)

- 3 evaluation metrics:
  - mean average precision
  - precision @ sample=500
  - precision @ hamming dist=2
- 16-bit (D = 16)



Experiments (Hash Learning)

Method (Dimensions of hidden layers)	Hamming ranking (mAP)		precision @ sample = 500		precision @ r = 2	
	MNIST	CIFAR10	MNIST	CIFAR10	MNIST	CIFAR10
Spectral hash (Weiss et al., 2009)	26.6	12.6	56.3	18.8	57.5	18.5
PCA-ITQ (Gong et al., 2013)	41.2	15.7	66.4	22.5	65.7	22.6
Deep Hash (60-30)	43.1	16.2	67.9	23.8	66.1	23.3
Linear RIM	35.9 (0.6)	**24.0 (3.5)**	68.9 (1.1)	15.9 (0.5)	71.3 (0.9)	14.2 (0.3)
Deep RIM (60-30)	42.7 (2.8)	15.2 (0.5)	67.9 (2.7)	21.8 (0.9)	65.9 (2.7)	21.2 (0.9)
Deep RIM (200-200)	43.7 (3.7)	15.6 (0.6)	68.7 (4.9)	21.6 (1.2)	67.0 (4.9)	21.1 (1.1)
Deep RIM (400-400)	43.9 (2.7)	15.4 (0.2)	69.0 (3.2)	21.5 (0.4)	66.7 (3.2)	20.9 (0.3)
IMSAT (VAT) (60-30)	61.2 (2.5)	19.8 (1.2)	78.6 (2.1)	21.0 (1.8)	76.5 (2.3)	19.3 (1.6)
IMSAT (VAT) (200-200)	80.7 (2.2)	21.2 (0.8)	95.8 (1.0)	**27.3 (1.3)**	94.6 (1.4)	26.1 (1.3)
IMSAT (VAT) (400-400)	**83.9 (2.3)**	21.4 (0.5)	**97.0 (0.8)**	**27.3 (1.1)**	**96.2 (1.1)**	**26.4 (1.0)**

- Tested on 2 benchmark datasets.
- Hyper-parameters are fixed throughout the datasets.

Experiments (Hash Learning)

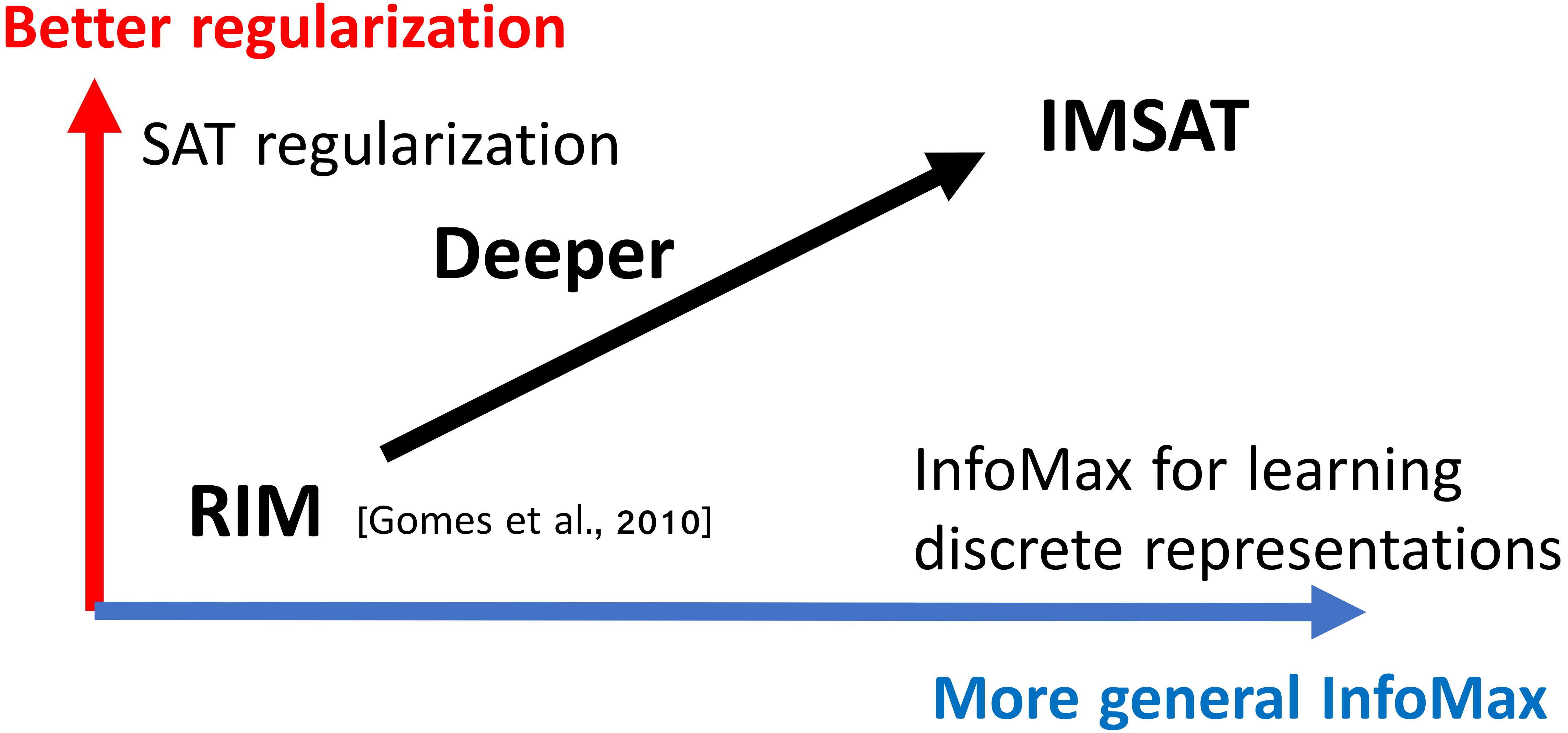
Method (Dimensions of hidden layers)	Hamming ranking (mAP)		precision @ sample = 500		precision @ r = 2	
	MNIST	CIFAR10	MNIST	CIFAR10	MNIST	CIFAR10
Spectral hash (Weiss et al., 2009)	26.6	12.6	56.3	18.8	57.5	18.5
PCA-ITQ (Gong et al., 2013)	41.2	15.7	66.4	22.5	65.7	22.6
Deep Hash (60-30)	43.1	16.2	67.9	23.8	66.1	23.3
Linear RIM	35.9 (0.6)	**24.0 (3.5)**	68.9 (1.1)	15.9 (0.5)	71.3 (0.9)	14.2 (0.3)
Deep RIM (60-30)	42.7 (2.8)	15.2 (0.5)	67.9 (2.7)	21.8 (0.9)	65.9 (2.7)	21.2 (0.9)
Deep RIM (200-200)	43.7 (3.7)	15.6 (0.6)	68.7 (4.9)	21.6 (1.2)	67.0 (4.9)	21.1 (1.1)
Deep RIM (400-400)	43.9 (2.7)	15.4 (0.2)	69.0 (3.2)	21.5 (0.4)	66.7 (3.2)	20.9 (0.3)
IMSAT (VAT) (60-30)	61.2 (2.5)	19.8 (1.2)	78.6 (2.1)	21.0 (1.8)	76.5 (2.3)	19.3 (1.6)
IMSAT (VAT) (200-200)	80.7 (2.2)	21.2 (0.8)	95.8 (1.0)	**27.3 (1.3)**	94.6 (1.4)	26.1 (1.3)
IMSAT (VAT) (400-400)	**83.9 (2.3)**	21.4 (0.5)	**97.0 (0.8)**	**27.3 (1.1)**	**96.2 (1.1)**	**26.4 (1.0)**

- IMSAT (VAT) outperformed the previous methods.

Outline

1. Introduction
2. Background (Regularized Information Maximization [Gomes et al., 2010])
3. Proposed Method (Information Maximizing Self-Augmented Training)
4. Experiments
5. Conclusions

Conclusions



Conclusions