

# Learning Discrete Representations via Information Maximizing Self-Augmented Training

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

University of Tokyo

RIKEN

Preferred Networks

ATR

Preferred Networks

University of Tokyo

Preferred Networks

University of Tokyo

RIKEN

University of Tokyo

\*Based on the work  
performed at  
Preferred Networks



東京大学  
THE UNIVERSITY OF TOKYO



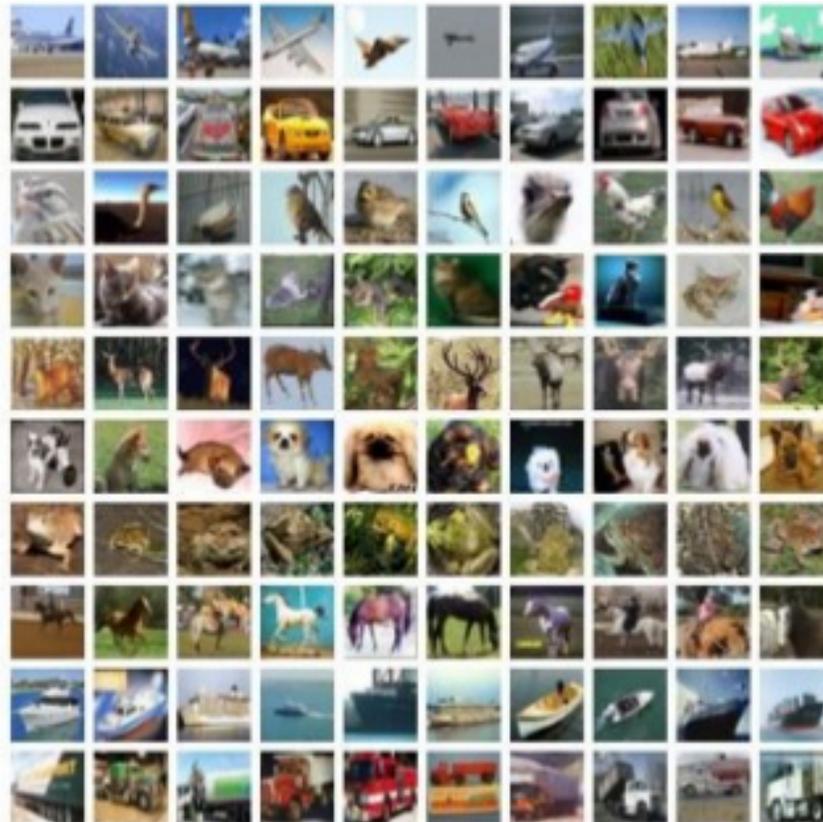


# Big success of supervised Deep Learning

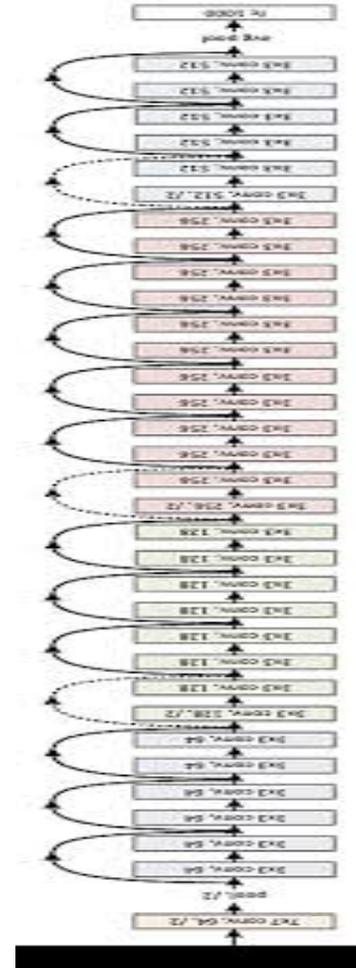
(Very) deep neural networks

Huge number of Labeled data

airplane  
automobile  
bird  
cat  
deer  
dog  
frog  
horse  
ship  
truck

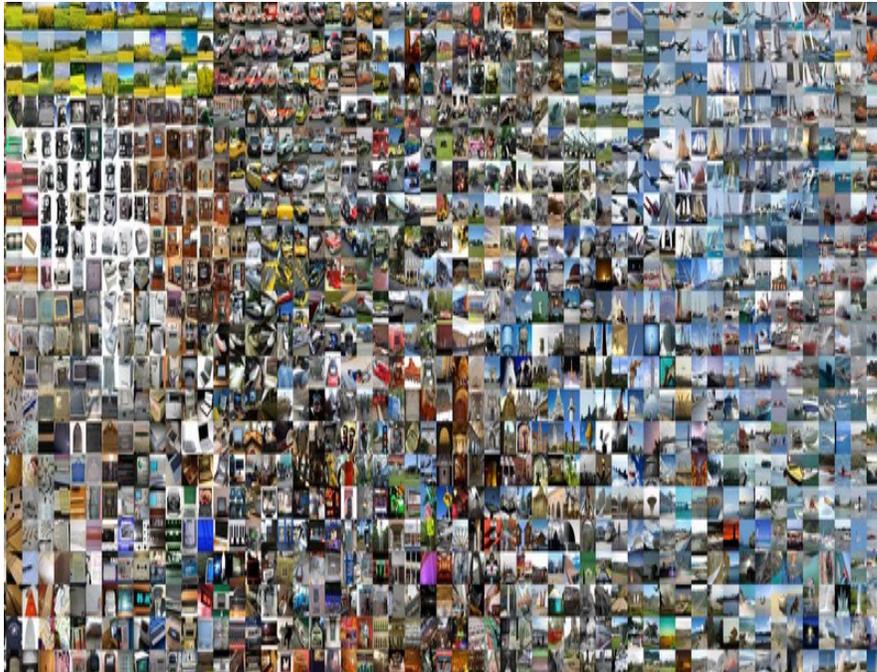


+



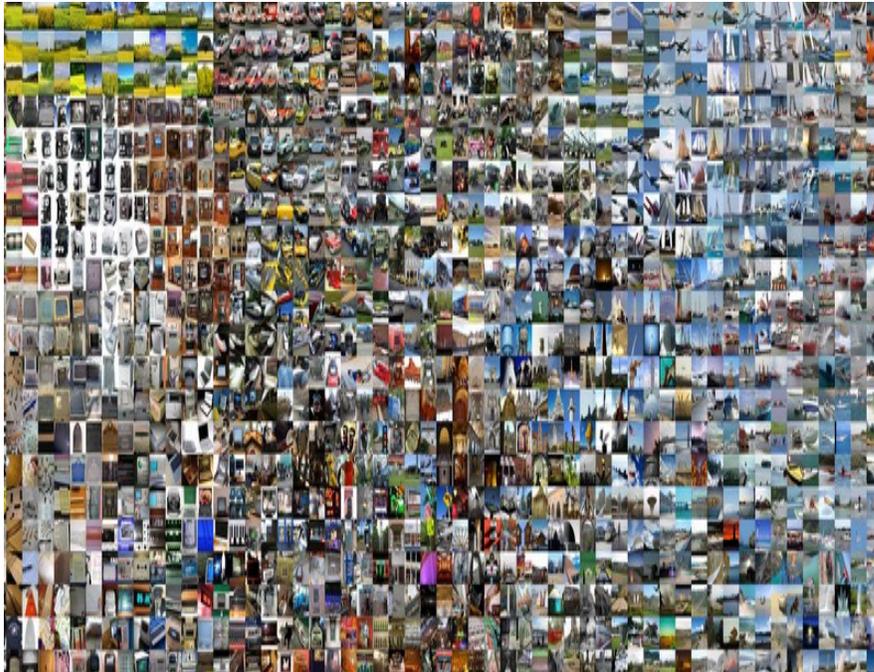
# Unsupervised Discrete Representation Learning

Unlabeled data



# Unsupervised Discrete Representation Learning

Unlabeled data



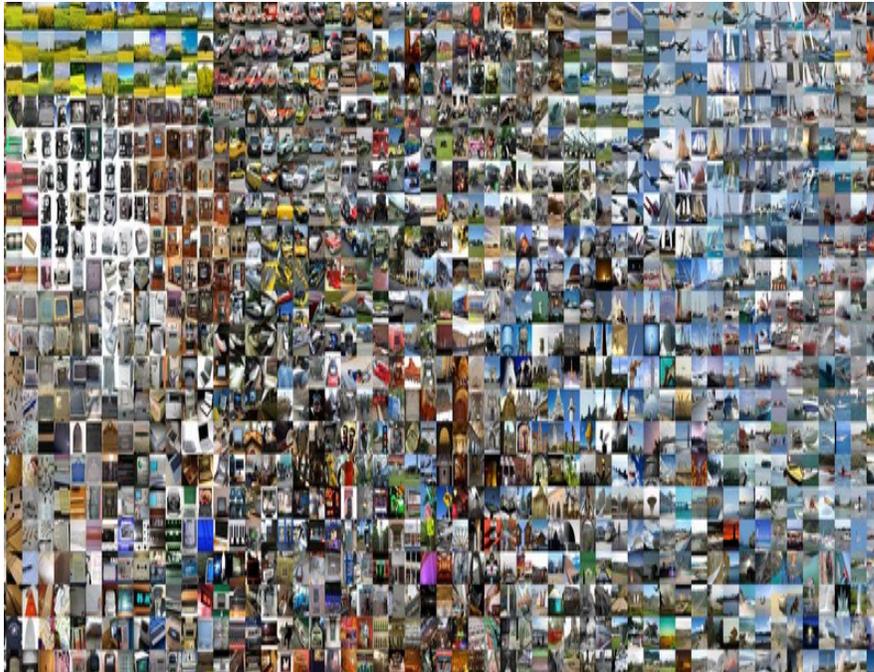
Learn to  
map

Discrete representations

```
00001110 01111101 00111111 00100111 11000000 11110000 00011011 100000011
00011100 10011000 00000001 00000111 11101110 11111000 00111100 111110100
11111101 00110110 01100011 00100011 00011100 00000111 00110000 111100100
00010101 11100111 11011000 01110010 00111110 01100011 11010110 000110010
11100110 11110111 00100111 11000000 11101110 00000011 10010000 111101111
10001110 10011011 10000001 00000110 10110011 11001100 11110001 110000000
11001101 00000000 10011110 00000000 11000001 11000010 10001100 011111110
01001110 11111011 10110011 00001111 10011011 10000111 11110001 000000110
01001100 10000100 00011001 10011001 11111111 00110011 11000000 110100100
00000111 11111111 00101001 00101011 10000011 01011111 11000011 010010100
00000010 10111111 11110000 11001110 11011111 11011000 11111111 110001100
11111110 00000101 11011110 10011111 11001001 01101101 10010001 100111111
00011011 01100100 11000100 11100011 00011111 00001000 00001100 000111100
00111100 11100011 00011001 00100100 10011011 10000011 10111001 110001011
01000011 00000111 11111010 00000111 11100001 11001000 10000011 000001101
10011100 11111001 11000000 10110100 11001000 10011000 11001111 001001100
11101111 11011111 01001110 00110011 01111001 00101101 10011100 011111000
01000111 01101110 00001100 00001111 11000011 11001101 01101111 011111100
10010011 10000111 10110111 00011100 00111111 00000011 01011010 110000011
```

# Unsupervised Discrete Representation Learning

Unlabeled data



Learn to  
map



Discrete representations

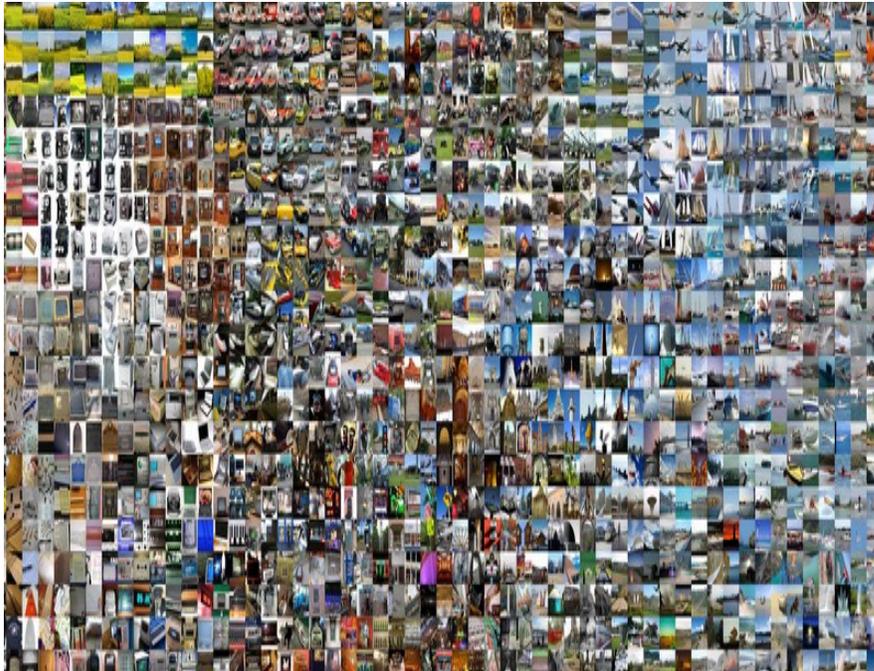
**Clustering**

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

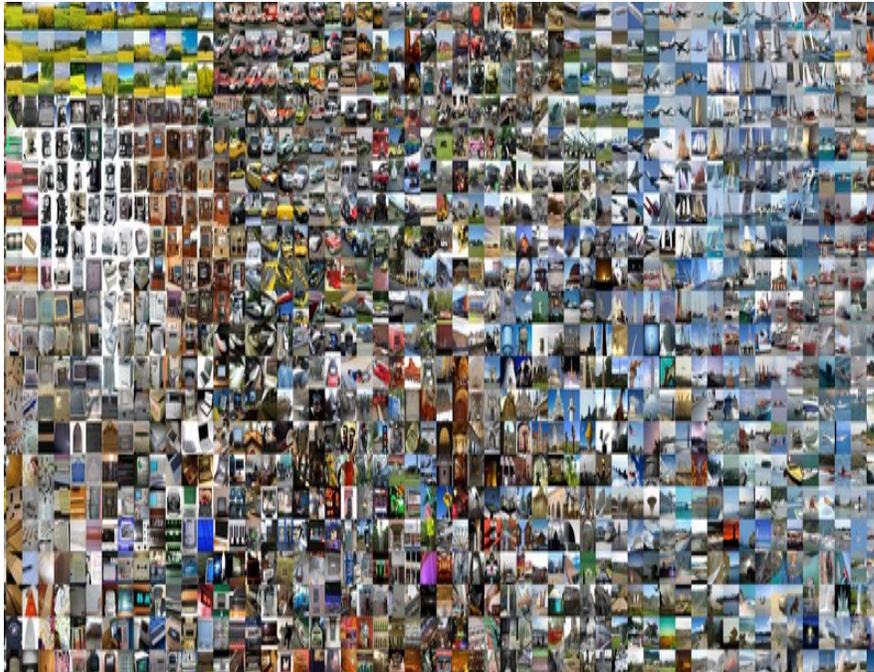
**Hash Learning**

Map to binary codes

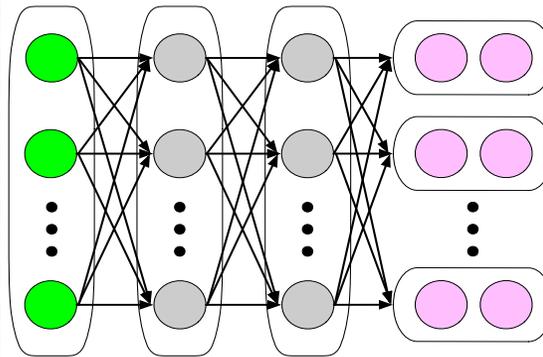
0001, 0101, 1110, 1111,  
0000, 0111, 0000, 1011

# 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

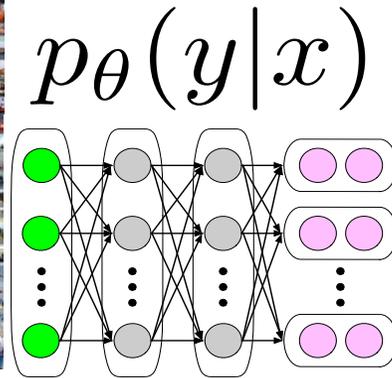
**Hash Learning**

**Map to binary codes**

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

# Problem setting

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

 $X$ 

## Discrete representations

1. Cluster assignment
2. Binary codes

 $Y$

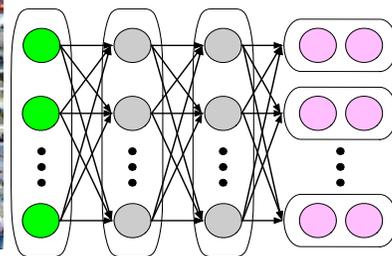
# Problem setting

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



$X$

$p_{\theta}(y|x)$



**Discrete representations**

1. Cluster assignment
2. Binary codes

$Y$

**Regularized Information Maximization (RIM)** [Gomes+ 2010]

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

**Mutual Information**

**Regularization (weight-decay)**

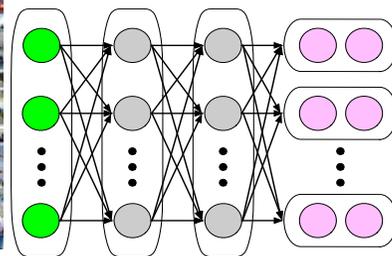
# Problem setting

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



$X$

$$p_{\theta}(y|x)$$



**Discrete representations**

1. Cluster assignment
2. Binary codes

$Y$

**Regularized Information Maximization (RIM)** [Gomes+ 2010]

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

**Regularization (weight-decay)**



**Only applicable to clustering.**

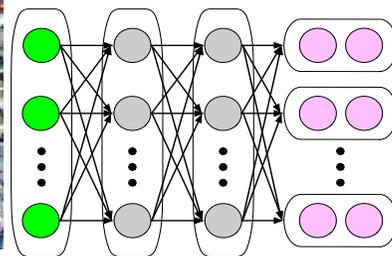
# Problem setting

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



$X$

$$p_{\theta}(y|x)$$



**Discrete representations**

1. Cluster assignment
2. Binary codes

$Y$

**Regularized Information Maximization (RIM)** [Gomes+ 2010]

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

☹️ **Weight-decay is restrictive.**

☹️ **Only applicable to clustering.**

# Our Contributions in a nutshell

**Better regularization**

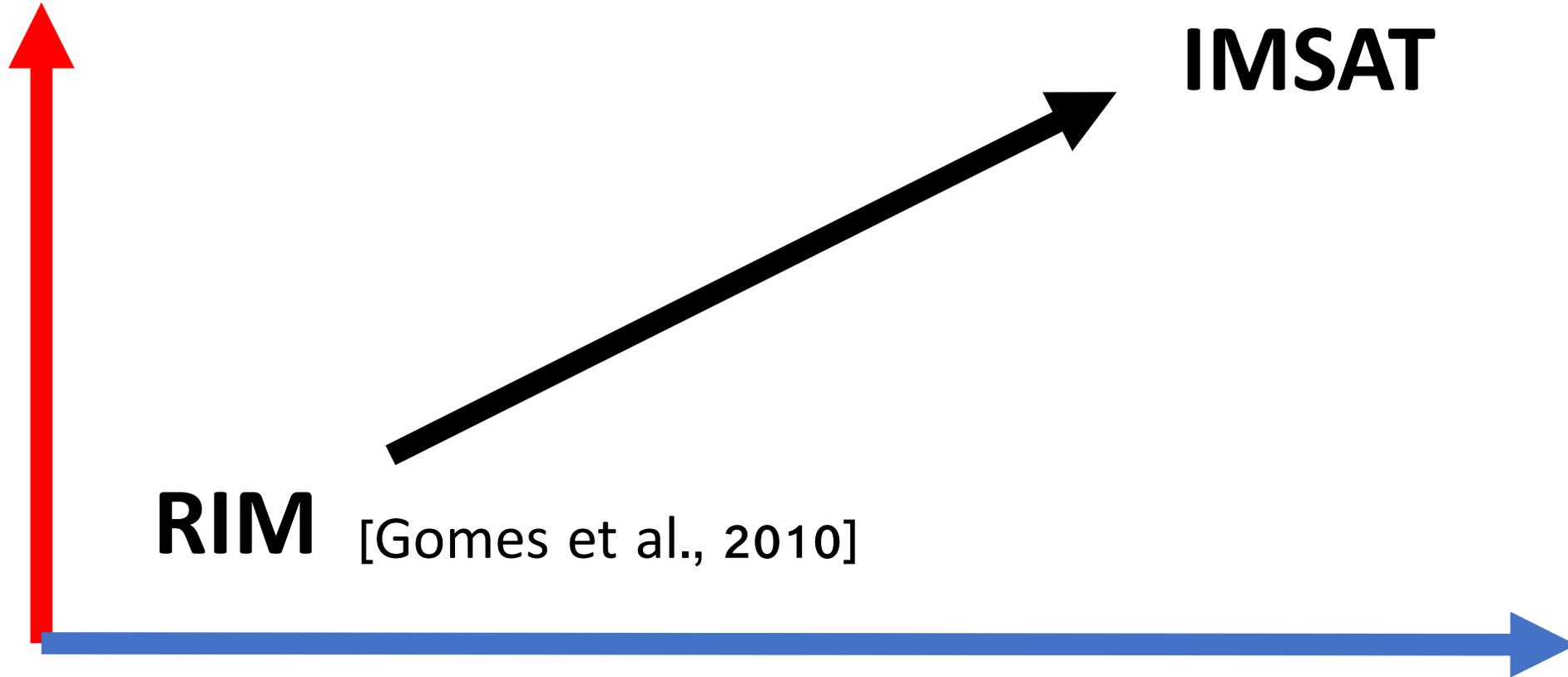
**Our method**

**IMSAT**

**RIM** [Gomes et al., 2010]

**Weight-decay, Clustering**

**More general InfoMax**

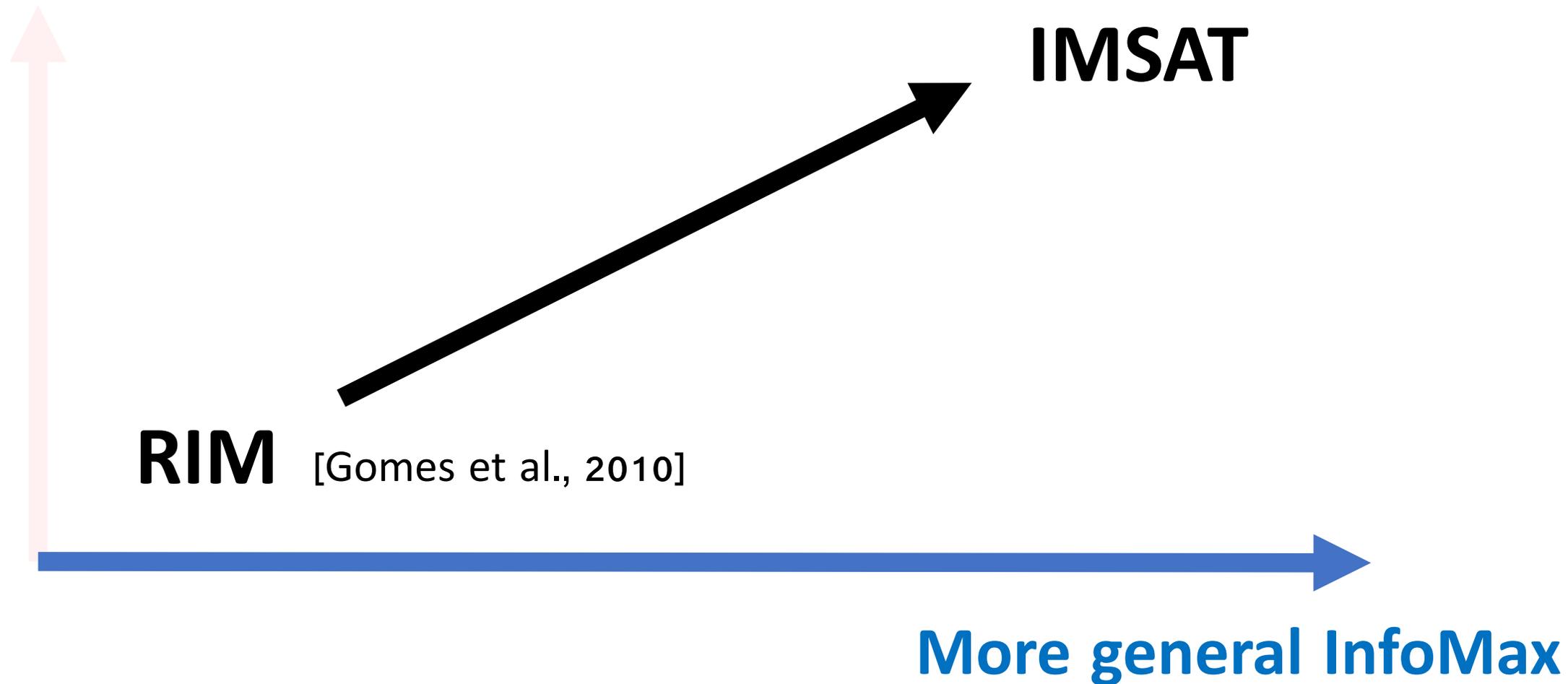


# Outline

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

# Information Maximization

Better regularization



# 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  $p_{\theta}(y_1, \dots, y_D | x)$  via  $\max_{\theta} I(X; Y_1, \dots, Y_D)$

# Information Maximization

For discrete representation learning:

Our proposal:

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

• **Challenge:** Combinatorial summation

→ We need approximation!

$$\sum_{y_1} \sum_{y_2} \cdots \sum_{y_D}$$

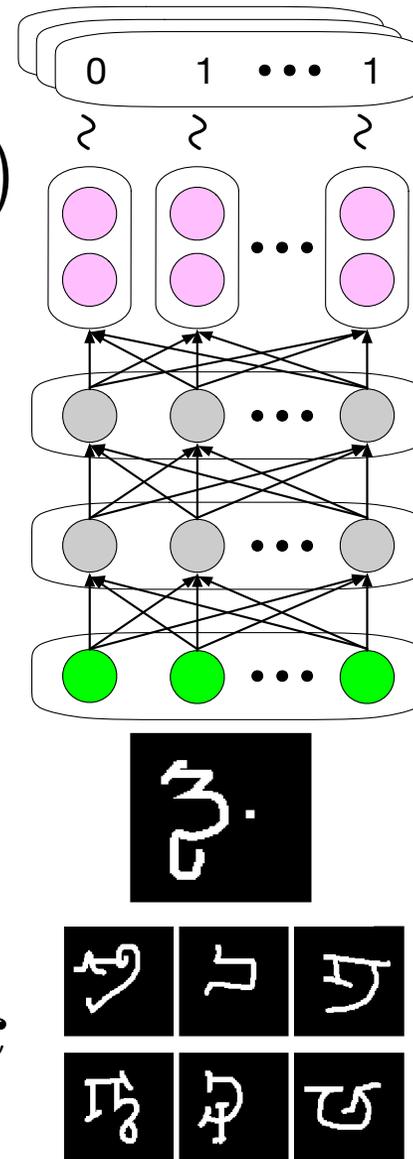
# Information Maximization

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

$$I(X; Y_1, \dots, Y_D) \approx \sum_{d=1}^D I(X; Y_d) - \sum_{1 \leq d \neq d' \leq D} I(Y_d; Y_{d'})$$

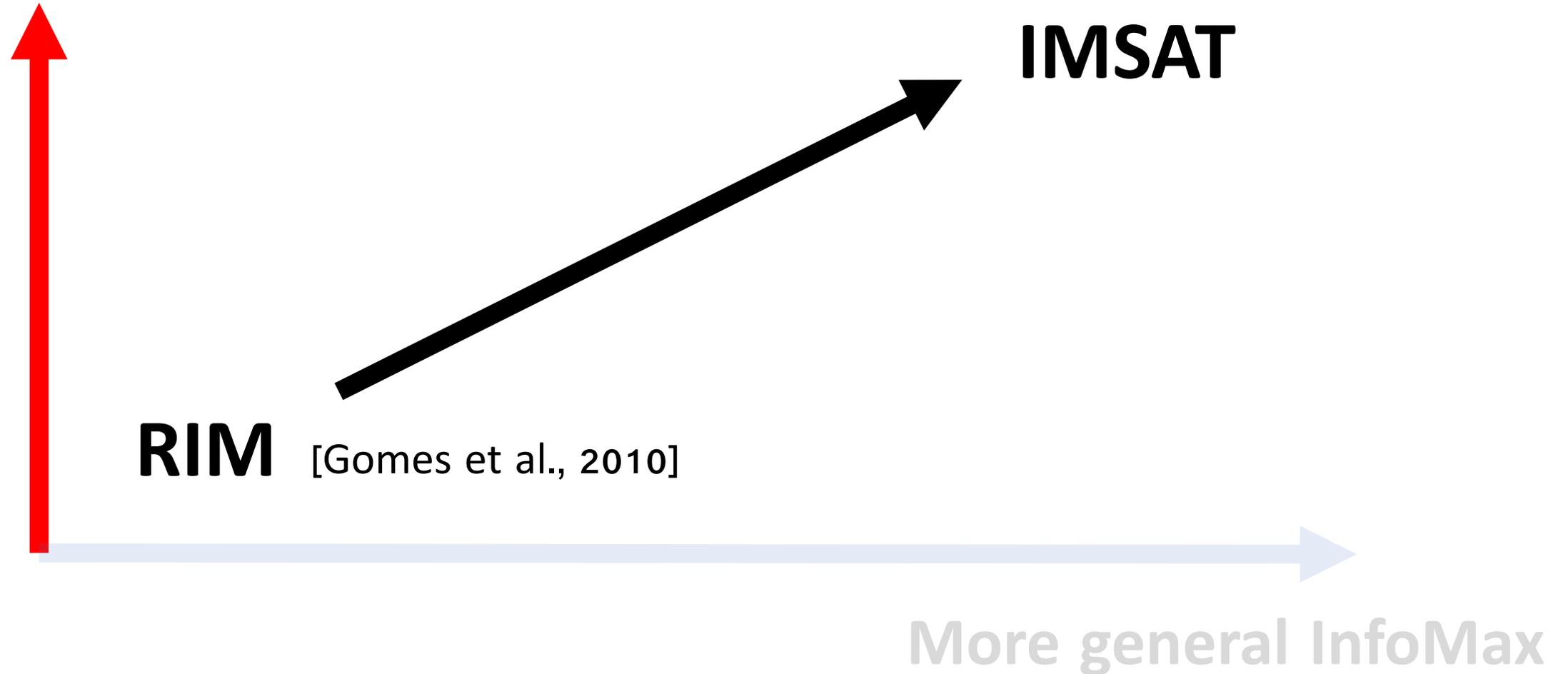
$$y = (y_1, \dots, y_D)$$

$$p_{\theta}(y|x)$$



# Self-Augmented Training (SAT)

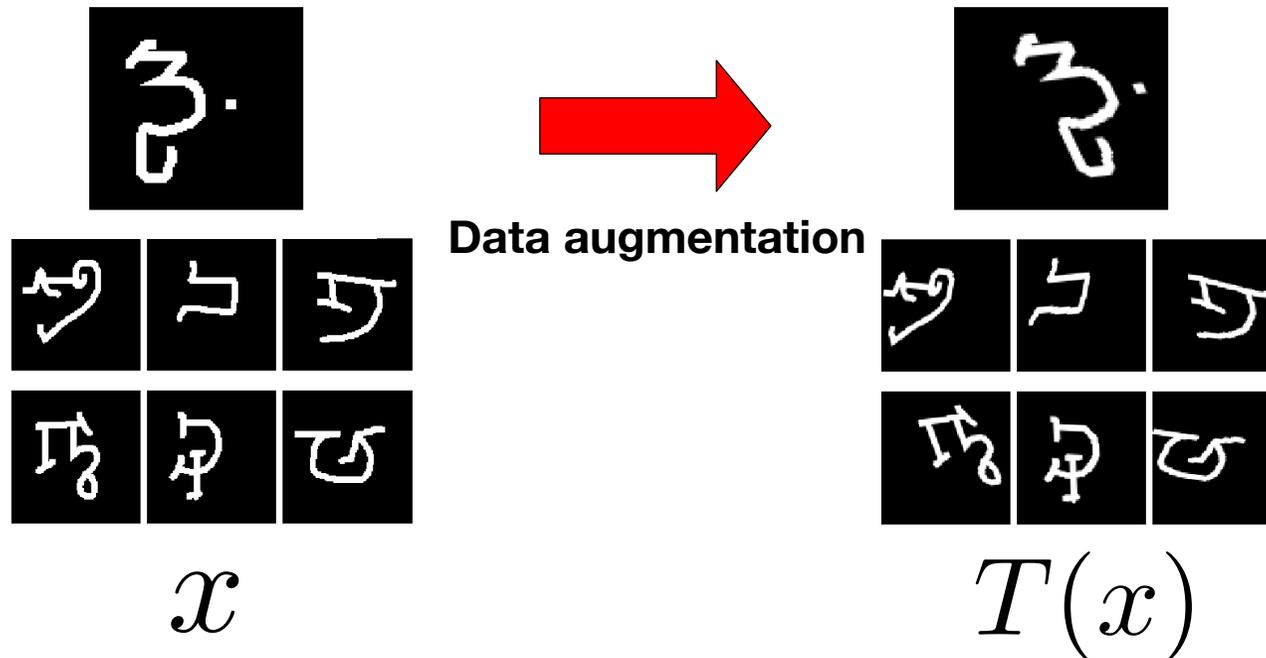
**Better regularization**



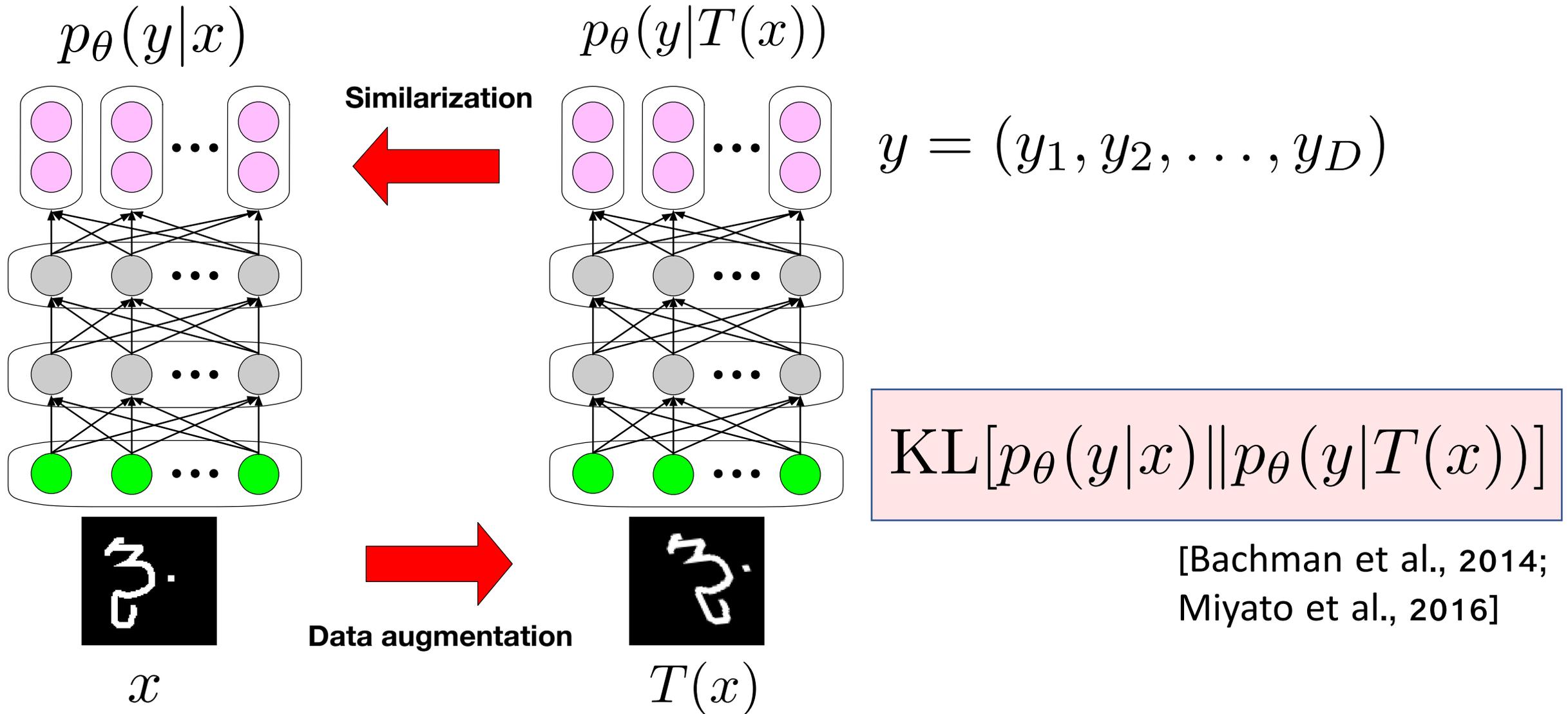
# Self-Augmented Training (SAT)

Augmentation function  $T(\cdot) : \mathcal{X} \rightarrow \mathcal{X}$

→ User-specified transformation that does not change the meaning of data



# Self-Augmented Training (SAT)



# Self-Augmented Training (SAT)

- **Local perturbation**

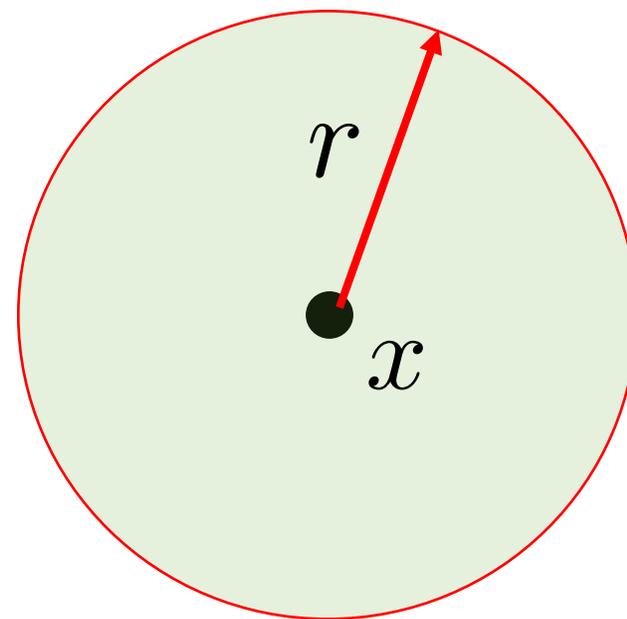
$$T(x) = x + r, \quad \|r\|_2 = \epsilon$$

- **Random Perturbation Training (RPT)**

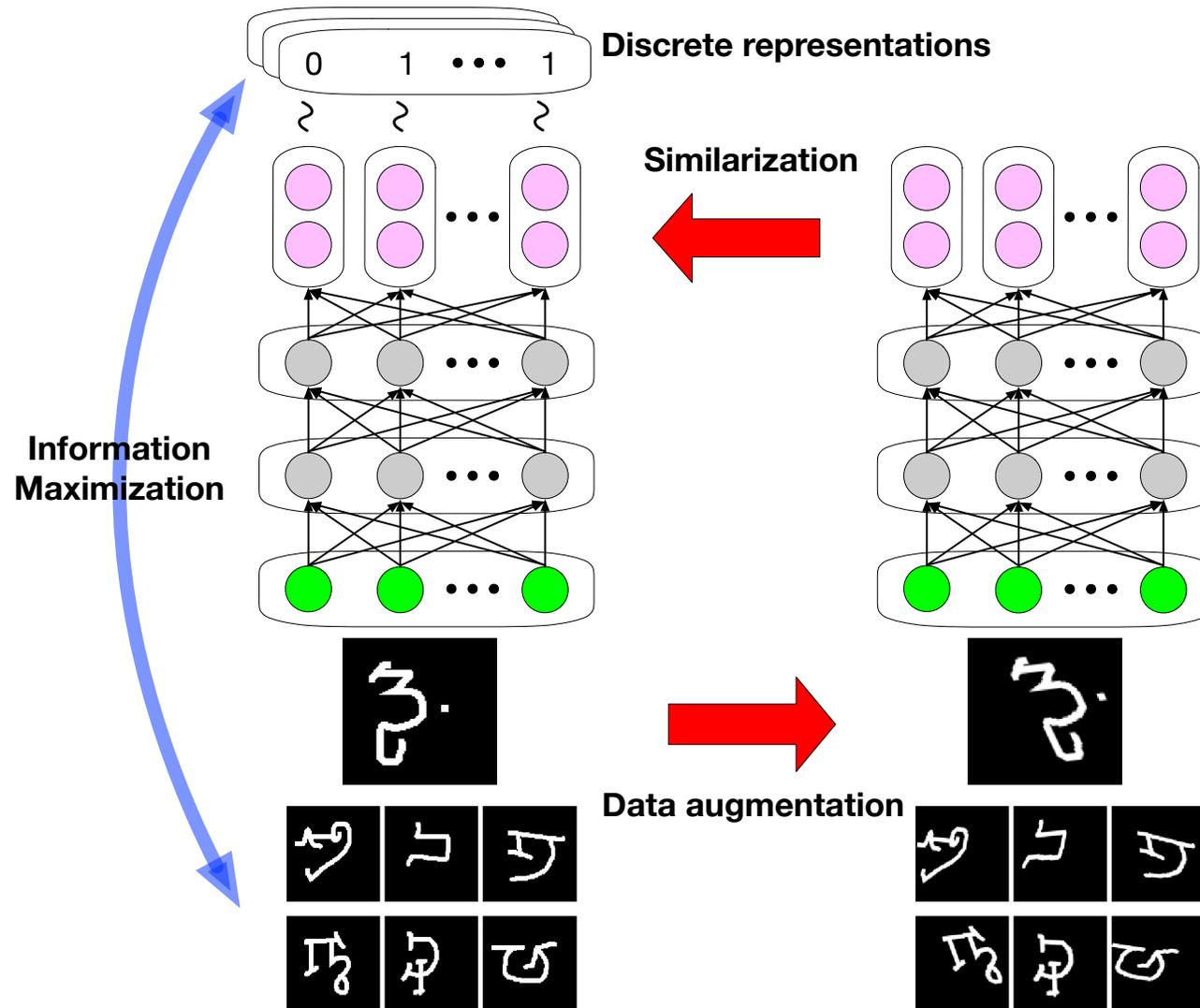
[Bachman et al., 2014]

- **Virtual Adversarial Training (VAT)**

[Miyato et al., 2016]



# IMSAT = Information Maximizing + SAT



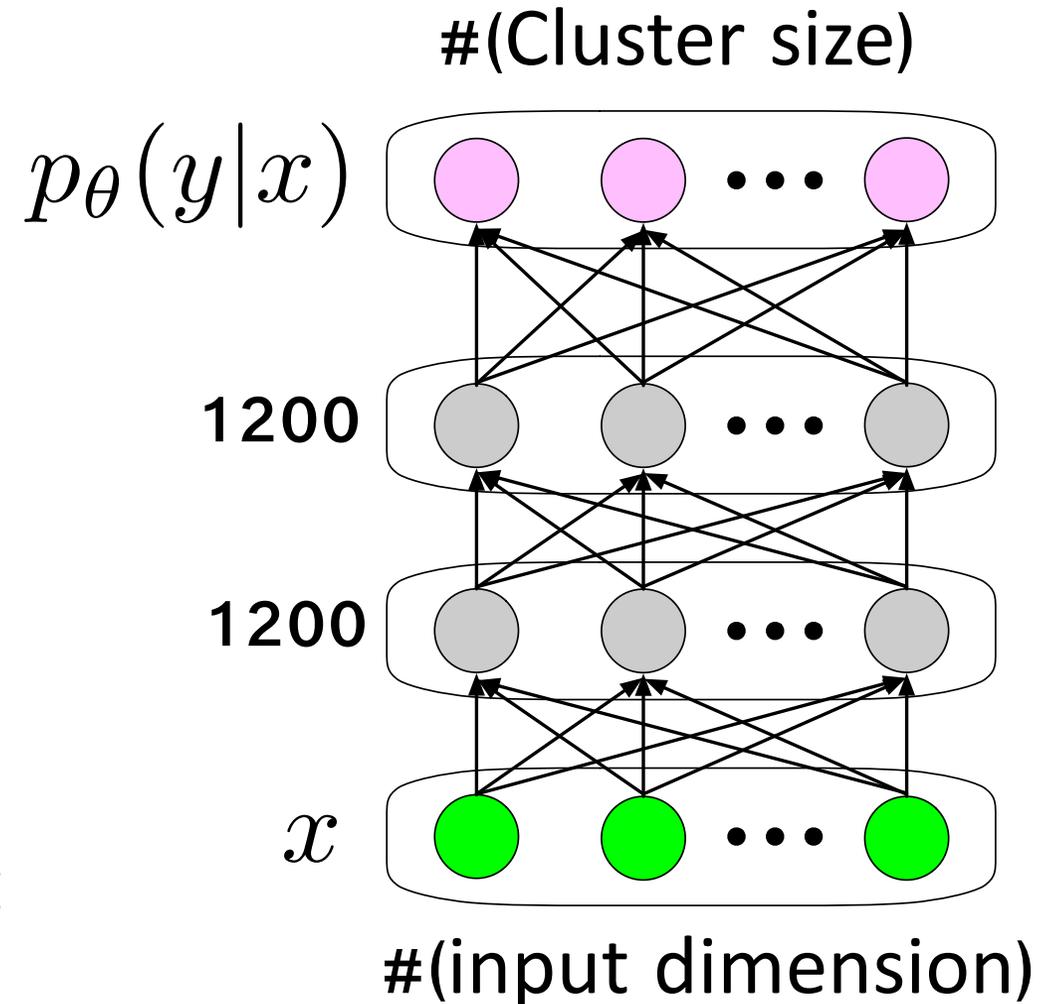
# Outline

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)

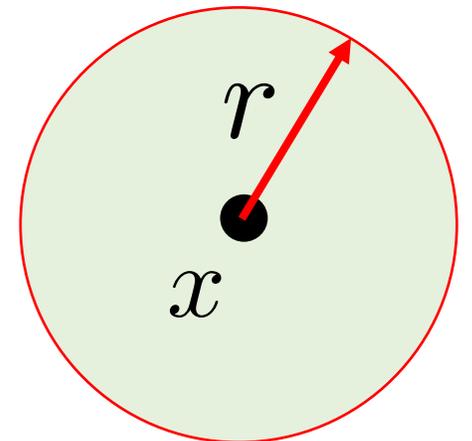
Method	MNIST	Omniglot	STL	CIFAR10	CIFAR100	SVHN	Reuters	20news
<i>K</i> -means	53.2	12.0	85.6	34.4	21.5	17.9	54.1	15.5
dAE+ <i>K</i> -means	79.8 <sup>†</sup>	14.1	72.2	44.2	20.8	17.4	67.2	22.1
DEC [Xie et al., 2014]	84.3 <sup>†</sup>	5.7 (0.3)	78.1 (0.1)	<b>46.9 (0.9)</b>	14.3 (0.6)	11.9 (0.4)	67.3 (0.2)	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)	<b>50.9 (3.1)</b>
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)
<b>IMSAT (RPT)</b>	89.6 (5.4)	16.4 (3.1)	92.8 (2.5)	45.5 (2.9)	24.7 (0.5)	35.9 (4.3)	<b>71.9 (6.5)</b>	24.4 (4.7)
<b>IMSAT (VAT)</b>	<b>98.4 (0.4)</b>	<b>24.0 (0.9)</b>	<b>94.1 (0.4)</b>	45.6 (0.8)	<b>27.5 (0.4)</b>	<b>57.3 (3.9)</b>	<b>71.0 (4.9)</b>	31.1 (1.9)

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

# Experiments (Clustering)

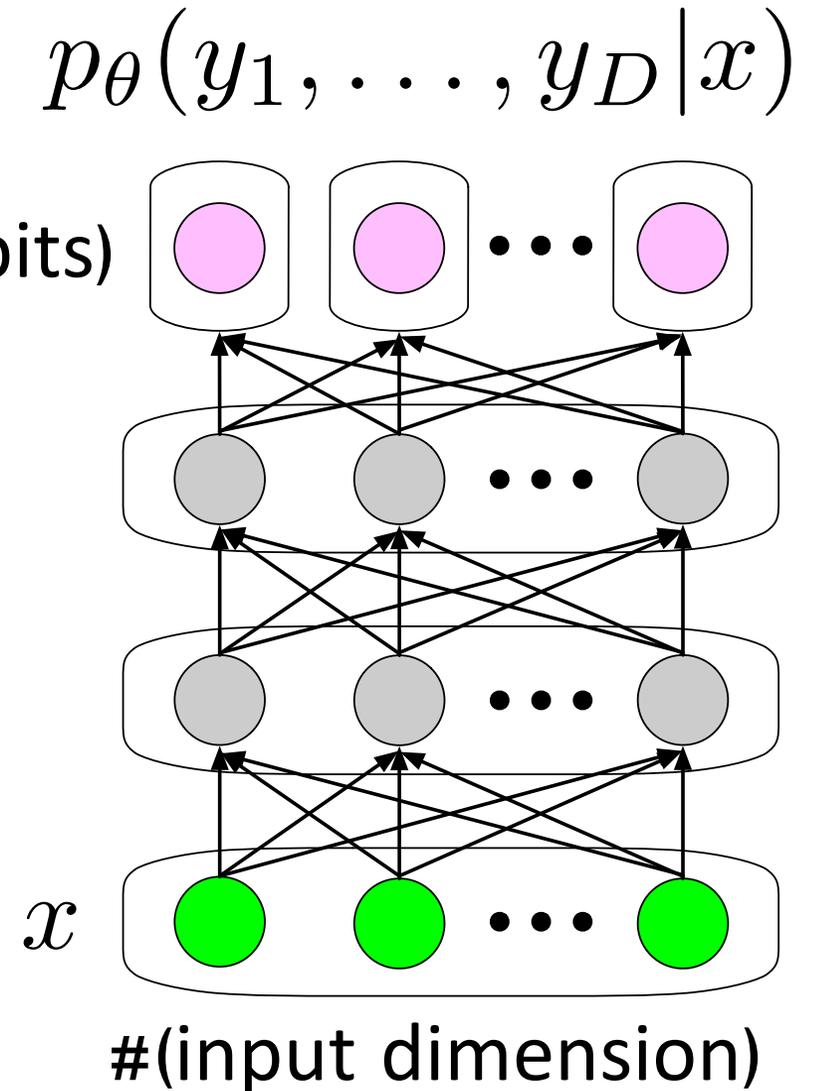
Method	MNIST	Omniglot	STL	CIFAR10	CIFAR100	SVHN	Reuters	20news
$K$ -means	53.2	12.0	85.6	34.4	21.5	17.9	54.1	15.5
dAE+ $K$ -means	79.8 <sup>†</sup>	14.1	72.2	44.2	20.8	17.4	67.2	22.1
DEC	84.3 <sup>†</sup>	5.7 (0.3)	78.1 (0.1)	<b>46.9 (0.9)</b>	14.3 (0.6)	11.9 (0.4)	67.3 (0.2)	30.8 (1.8)
Linear RIM	59.6 (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)	<b>50.9 (3.1)</b>
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)
<b>IMSAT (RPT)</b>	89.6 (5.4)	16.4 (3.1)	92.8 (2.5)	45.5 (2.9)	24.7 (0.5)	35.9 (4.3)	<b>71.9 (6.5)</b>	24.4 (4.7)
<b>IMSAT (VAT)</b>	<b>98.4 (0.4)</b>	<b>24.0 (0.9)</b>	<b>94.1 (0.4)</b>	45.6 (0.8)	<b>27.5 (0.4)</b>	<b>57.3 (3.9)</b>	<b>71.0 (4.9)</b>	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)	<b>24.0 (3.5)</b>	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)
<b>IMSAT (VAT) (60-30)</b>	61.2 (2.5)	19.8 (1.2)	78.6 (2.1)	21.0 (1.8)	76.5 (2.3)	19.3 (1.6)
<b>IMSAT (VAT) (200-200)</b>	80.7 (2.2)	21.2 (0.8)	95.8 (1.0)	<b>27.3 (1.3)</b>	94.6 (1.4)	26.1 (1.3)
<b>IMSAT (VAT) (400-400)</b>	<b>83.9 (2.3)</b>	21.4 (0.5)	<b>97.0 (0.8)</b>	<b>27.3 (1.1)</b>	<b>96.2 (1.1)</b>	<b>26.4 (1.0)</b>

- 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)	<b>24.0 (3.5)</b>	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)
<b>IMSAT (VAT) (60-30)</b>	61.2 (2.5)	19.8 (1.2)	78.6 (2.1)	21.0 (1.8)	76.5 (2.3)	19.3 (1.6)
<b>IMSAT (VAT) (200-200)</b>	80.7 (2.2)	21.2 (0.8)	95.8 (1.0)	<b>27.3 (1.3)</b>	94.6 (1.4)	26.1 (1.3)
<b>IMSAT (VAT) (400-400)</b>	<b>83.9 (2.3)</b>	21.4 (0.5)	<b>97.0 (0.8)</b>	<b>27.3 (1.1)</b>	<b>96.2 (1.1)</b>	<b>26.4 (1.0)</b>

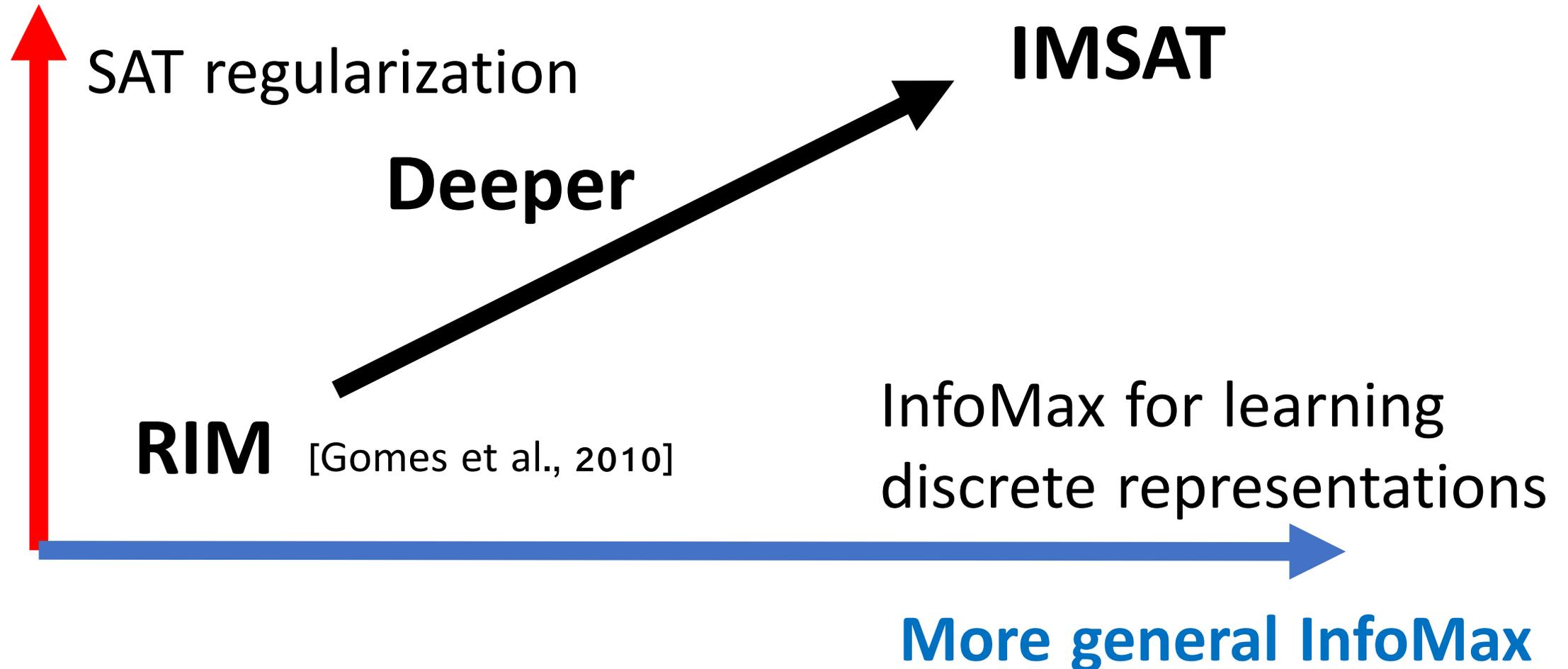
- IMSAT (VAT) outperformed the previous methods.

# 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

## Better regularization



# Conclusions

**Better regularization**

