Contrastive Self-Supervised Learning

Zili Wang

What is Self-Supervised Learning?

Generative / Predictive

Loss measured in the output space

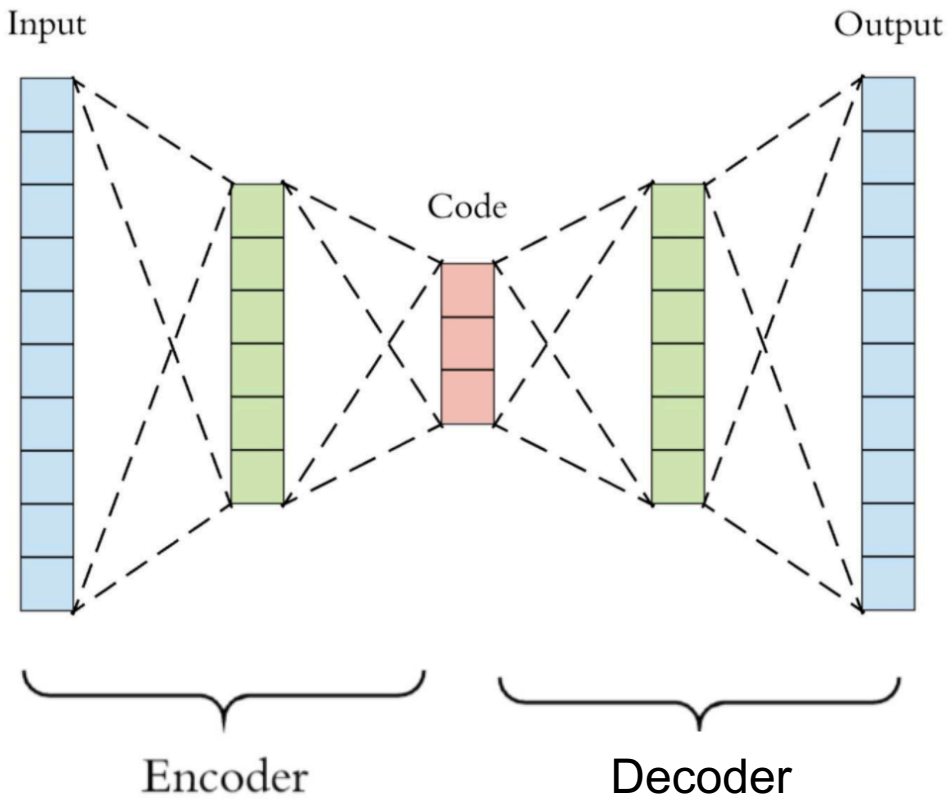
Examples: Colorization, Auto-Encoders

Contrastive

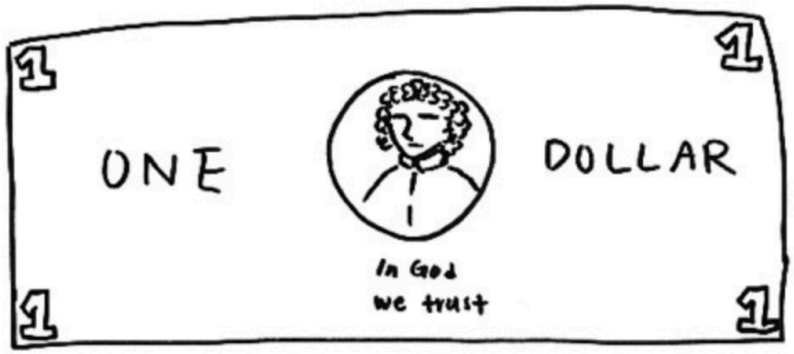
Loss measured in the representation space

Examples: TCN, CPC, Deep-InfoMax

Generative Methods



Contrastive Methods



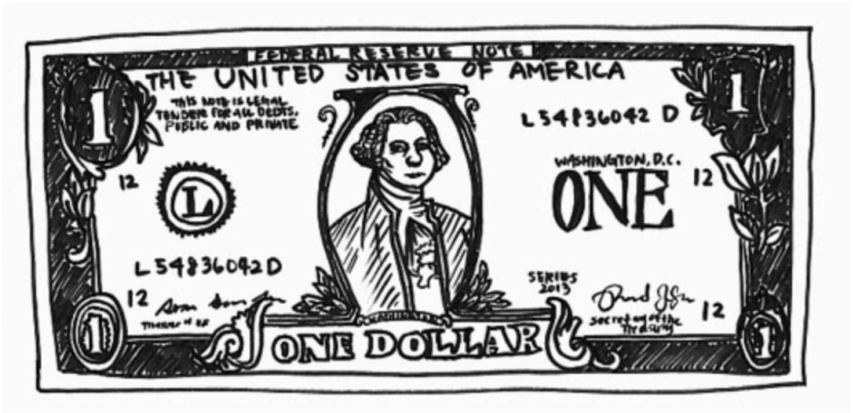
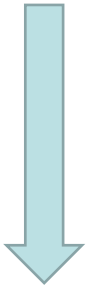


Fig. Left: Drawing of a dollar bill from memory.
Fig. Right: Drawing subsequently made with a dollar bill present.

How do contrastive methods work?

$$\text{score}(f(x), f(x^+)) >> \text{score}(f(x), f(x^-))$$

- here x^+ is data point similar or congruent to x, referred to as a *positive* sample.
- x^- is a data point dissimilar to x, referred to as a *negative* sample.
- the score function is a metric that measures the similarity between two features.



Softmax

$$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{\exp\left(f(x)^T f\left(x^+\right)\right)}{\exp\left(f(x)^T f\left(x^+\right)\right) + \sum_{j=1}^{N-1} \exp\left(f(x)^T f\left(x_j\right)\right)} \right]$$

Contrastive methods using margin (2006)

Data: $\mathcal{I} = \{X_1, \ldots, X_P\}, X_i \in \mathbb{R}^D$ Goal: G_W

Euclidean distance : $D_W(\vec{X}_1, \vec{X}_2) = \|G_W(\vec{X}_1) - G_W(\vec{X}_2)\|_2$

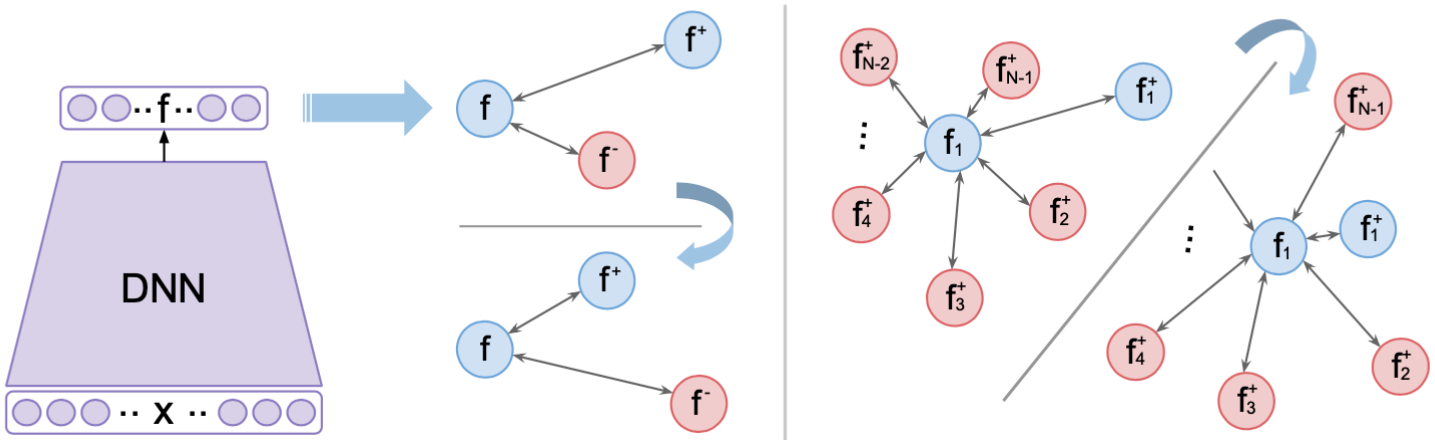
Loss function: $$\mathcal{L} = \sum_{i=1}^{P} (1 - Y) L_S(D_W^i) + Y L_D(D_W^i) \ ,$$

$$L(W, Y, \vec{X}_1, \vec{X}_2) = (1 - Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m - D_W)\}^2$$

Where the m is the margin

Dimensionality Reduction by Learning an Invariant Mapping
Raia Hadsell, Sumit Chopra, Yann LeCun (CVPR 2006)

N-pair loss（2016）

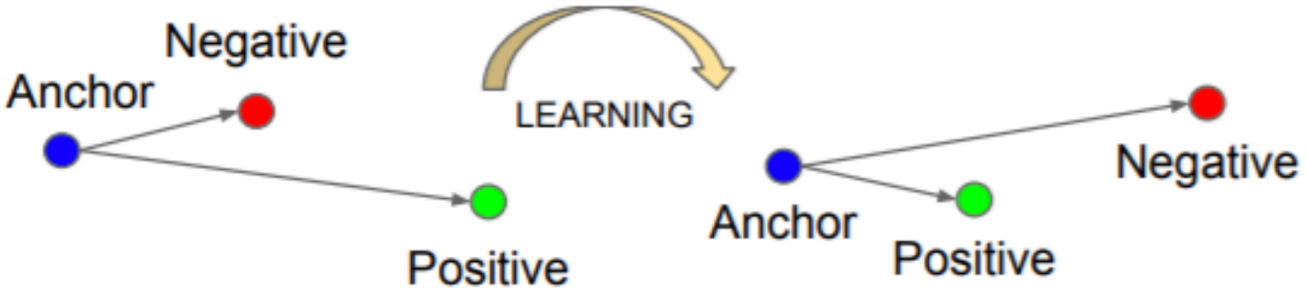


$$\mathcal{L}_{\text{cont}}^{m}\left(x_i, x_j; f\right) = 1\left\{y_i = y_j\right\}\left\|f_i - f_j\right\|_2^2 + 1\left\{y_i \neq y_j\right\}\max\left(0, m - \left\|f_i - f_j\right\|_2\right)^2$$

$$\mathcal{L}_{\text{tri}}^{m}\left(x, x^+, x^-; f\right) = \max\left(0, \left\|f - f^+\right\|_2^2 - \left\|f - f^-\right\|_2^2 + m\right)$$

Improved Deep Metric Learning with Multi-class N-pair Loss Objective
Kihyuk Sohn (NeurIPS 2016)

What is Triplet loss?



$$L = max(d(a,p) - d(a,n) + margin, 0)$$

FaceNet: A Unified Embedding for Face Recognition and Clustering
Florian Schroff, Dmitry Kalenichenko, James Philbin citation:5739 2015

N-pair loss（2016）

$$\mathcal{L}_{\text{cont}}^{m}\left(x_i, x_j; f\right) = 1\left\{y_i = y_j\right\}\left\|f_i - f_j\right\|_2^2 + 1\left\{y_i \neq y_j\right\}\max\left(0, m - \left\|f_i - f_j\right\|_2\right)^2$$

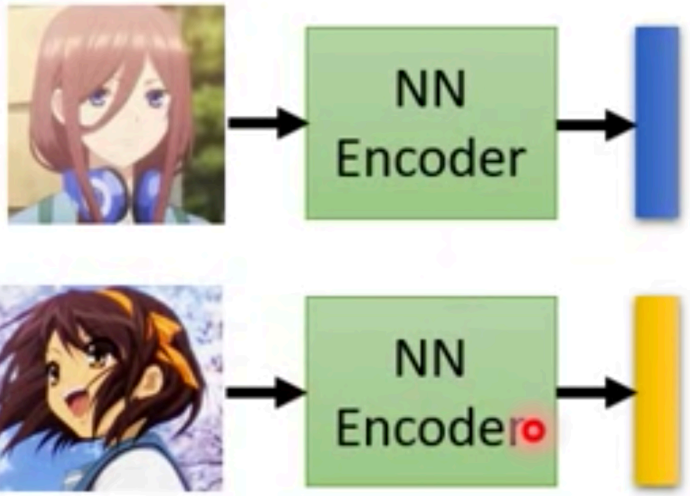
$$\mathcal{L}_{\text{tri}}^{m}\left(x, x^+, x^-; f\right) = \max\left(0, \left\|f - f^+\right\|_2^2 - \left\|f - f^-\right\|_2^2 + m\right)$$

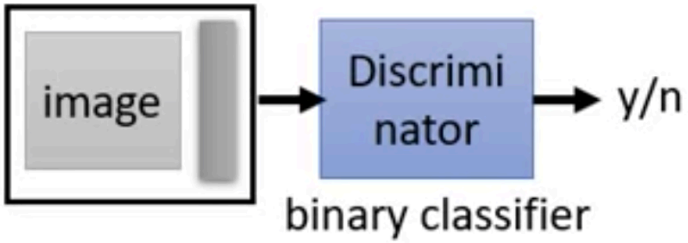
$$\mathcal{L}_{(2+1)\text{-tuplet}}\left(\left\{x, x^+, x_i\right\}; f\right) = \mathcal{L}_{\text{triplet}}\left(\left\{x, x^+, x_i\right\}; f\right) = \log\left(1 + \exp\left(f^\top f_i - f^\top f^+\right)\right)$$

$$\log\left(1 + \sum_{i=1}^{L-1}\exp\left(f^\top f_i - f^\top f^+\right)\right) = -\log\frac{\exp\left(f^\top f^+\right)}{\exp\left(f^\top f^+\right) + \sum_{i=1}^{L-1}\exp\left(f^\top f_i\right)}$$

Improved Deep Metric Learning with Multi-class N-pair Loss Objective
Kihyuk Sohn (NeurIPS 2016)

How to evaluate an encoder?





Train ϕ to minimize L_D

$$L_D^* = \min_{\phi} L_D$$

Small L_D^* ➡ The embeddings are representative.

Large L_D^* ➡ Not representative

More about VAE from Hung-yi Lee

Contrastive Predictive Coding(2019)

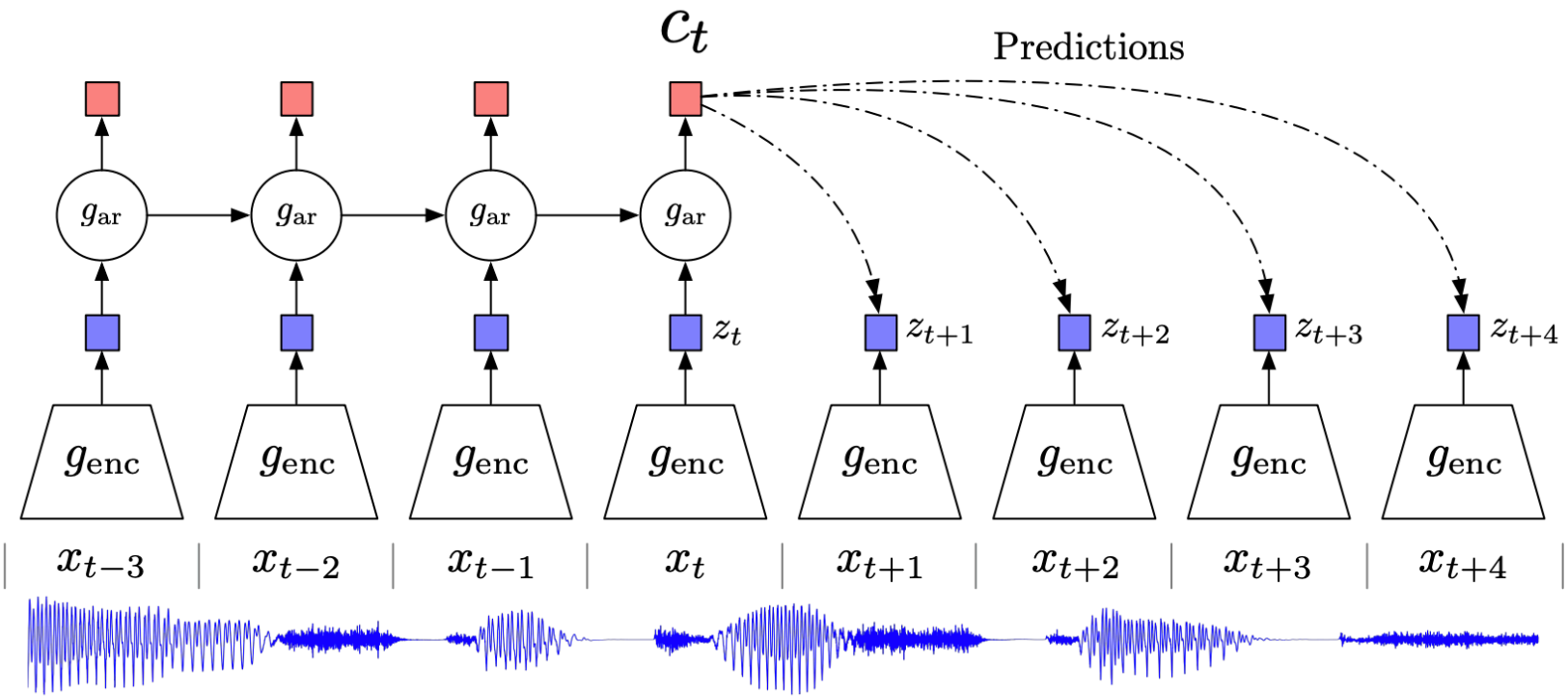


Figure 1: Overview of Contrastive Predictive Coding, the proposed representation learning approach. Although this figure shows audio as input, we use the same setup for images, text and reinforcement learning.

Representation Learning with Contrastive Predictive Coding (Aaron van den Oord, Yazhe Li, Oriol Vinyals) 2018 citation: 385

Contrastive Predictive Coding(2019)

Motivation: Learning Representation

$$z_t = g_{enc}(x_t)$$

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$$

Predicating the representation using LSTM

$$f_k(x_{t+k}, c_t) = exp(z_{t+k}^T W_k c_t)$$

Contrastive Predictive Coding(2019)

Predicating the representation using LSTM

$$f_k(x_{t+k}, c_t) = exp(z_{t+k}^T W_k c_t)$$

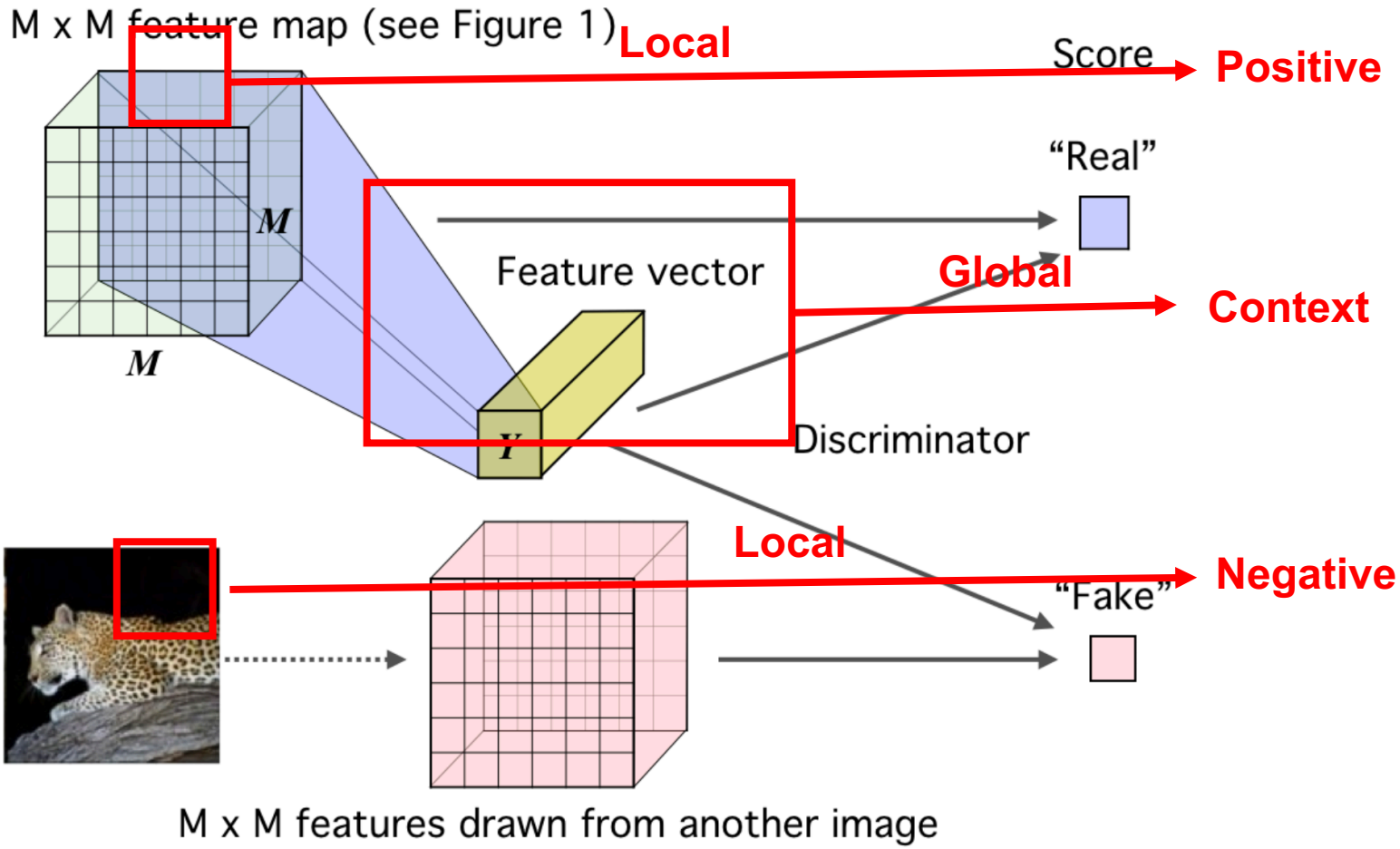
Loss function

$$L_N = -E[log\frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}]$$

$$\frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)} \Rightarrow \sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for } j = 1, \ldots, K.$$

希望这个式子可以接近于1，这样子的话就是loss越来越小，预测的效果越来越好

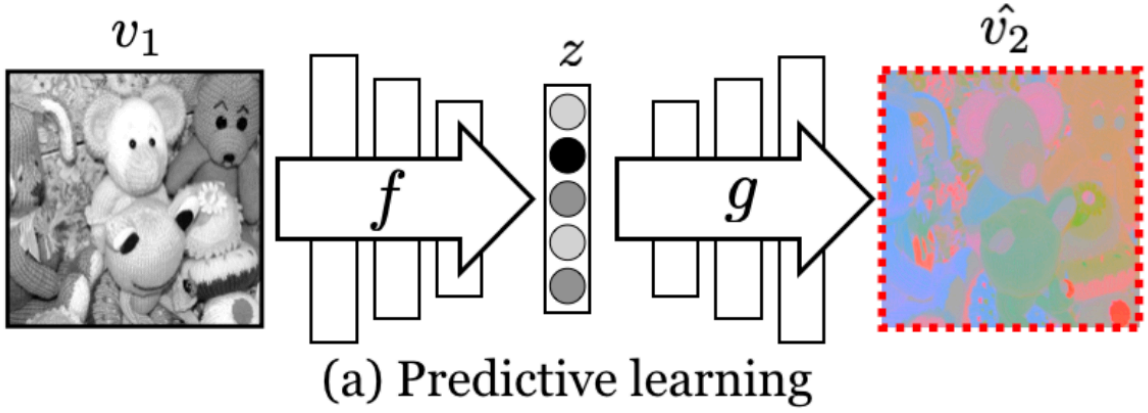
Learning deep representations by mutual information (2019)



Learning deep representations by mutual information estimation and maximization

Kaiming He and Yoshua Bengio et al (ICLR 2019)

Multi-view representation learning (2019)



(a) Predictive learning

Lab 空间：将图片分解到两个view：L(光度)，AB(色度)

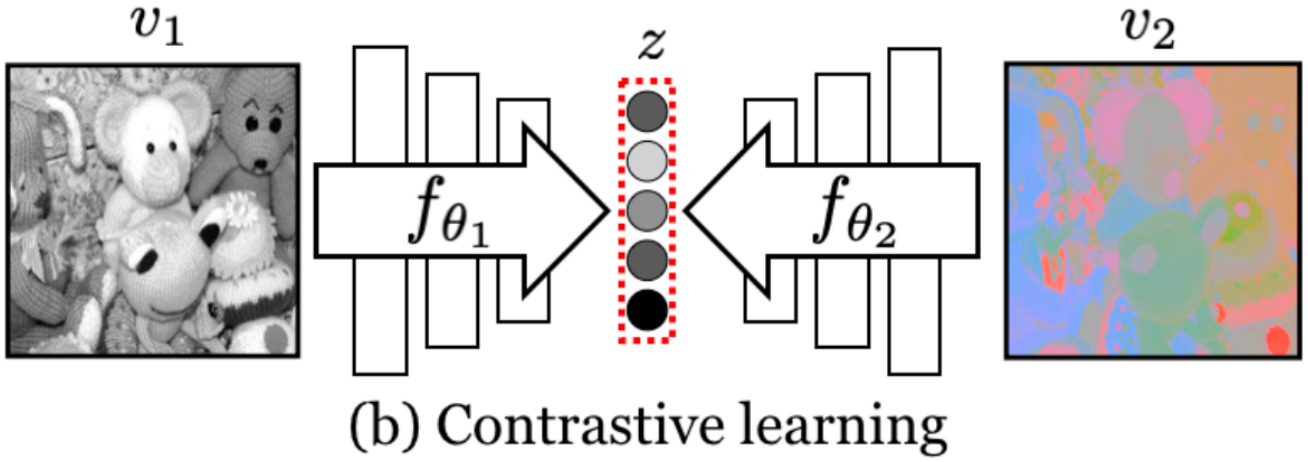
v_1 代表光照， v_2 代表色度　　$z = f(v_1)$ 和 $\hat{v}_2 = g(z)$

目标：迫使 $\hat{v}_2$ 接近 v_2

缺点：丢失建模关联和复杂结构的能力 ⟹ Contrastive Learning

Contrastive Multiview Coding
Yonglong Tian, Dilip Krishnan, Phillip Isola (2019)

Multi-view representation learning



(b) Contrastive learning

Positive: $x \sim p(v_1, v_2)$ or $x = \{v_1^i, v_2^i\}$

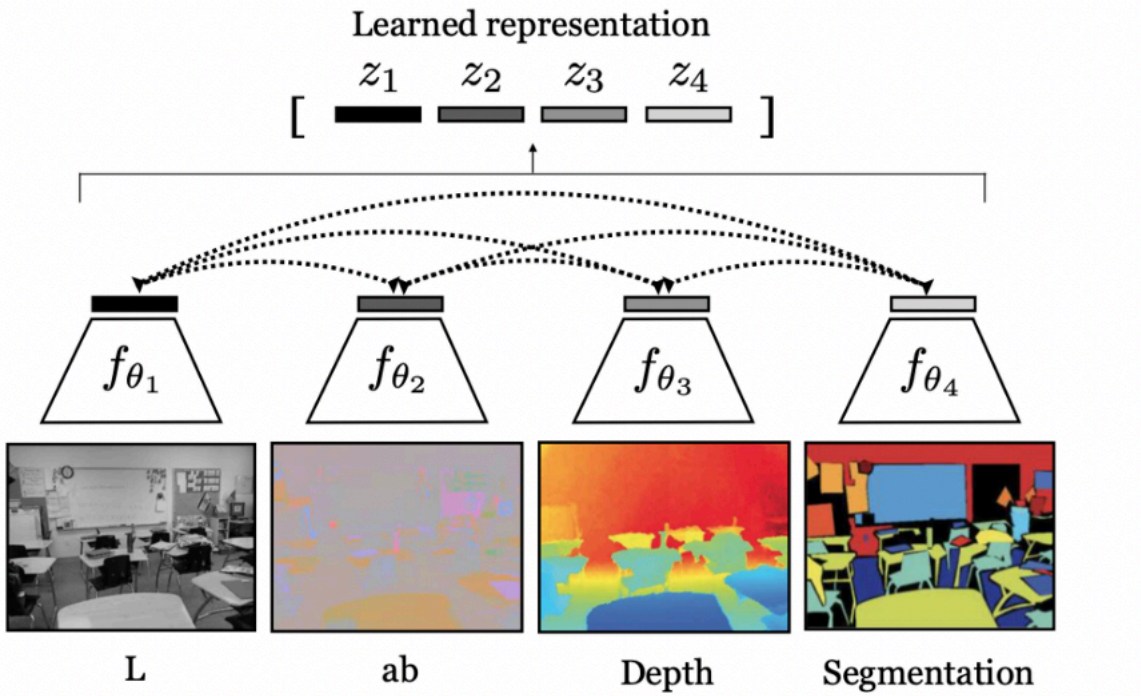
Negative: $y \sim p(v_1)\,p(v_2)$ or $y = \{v_1^i, v_2^j\}$

Score function: $h_\theta(\cdot)$

$$\mathcal{L}_{\text{contrast}} = -\mathop{\mathbb{E}}_{S}\left[\log \frac{h_\theta(x)}{h_\theta(x) + \sum_{i=1}^{k} h_\theta(y_i)}\right] \xrightarrow{\text{固定一个view}} \mathcal{L}_{\text{contrast}}^{V_1,V_2} = -\mathop{\mathbb{E}}_{\{v_1^i, v_2^i\}}\left[\log \frac{h_\theta\left(\{v_1^i, v_2^i\}\right)}{\sum_{j=1}^{N} h_\theta\left(\left\{v_1^i, v_2^j\right\}\right)}\right]$$

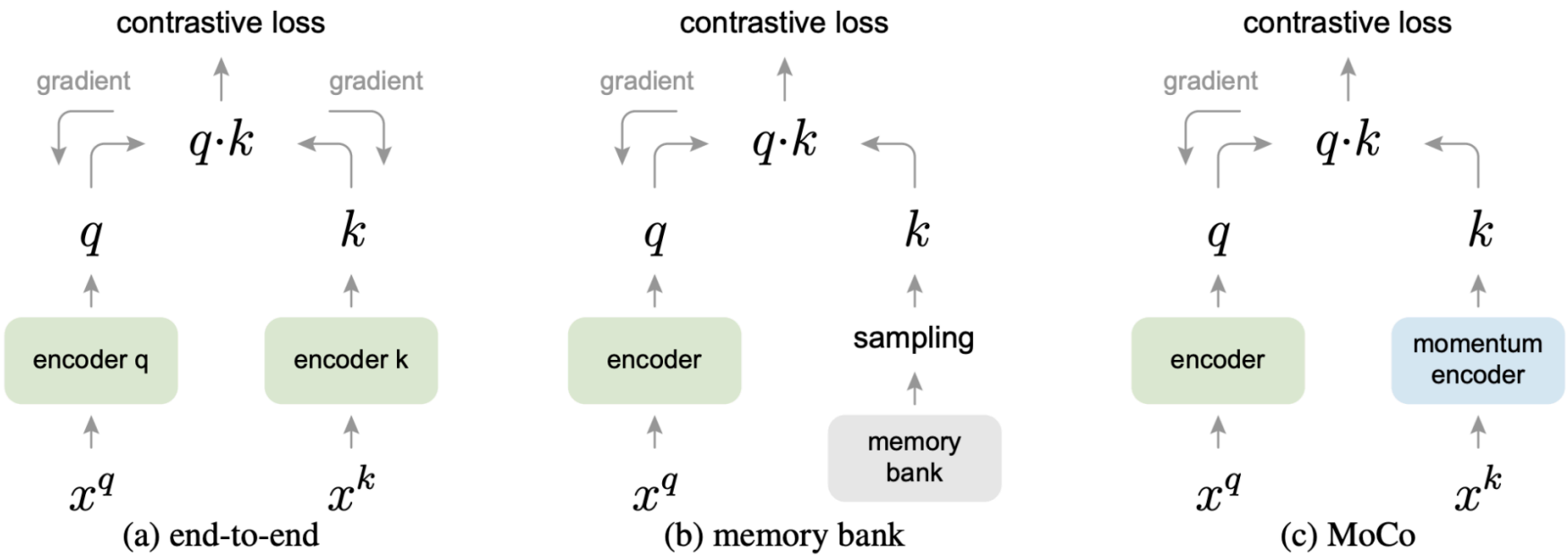
Contrastive Multiview Coding
Yonglong Tian, Dilip Krishnan, Phillip Isola (2019)

Multi-view representation learning



Contrastive Multiview Coding
Yonglong Tian, Dilip Krishnan, Phillip Isola (2019)

Moco



梯度回传
只有当前mini-batch的数据

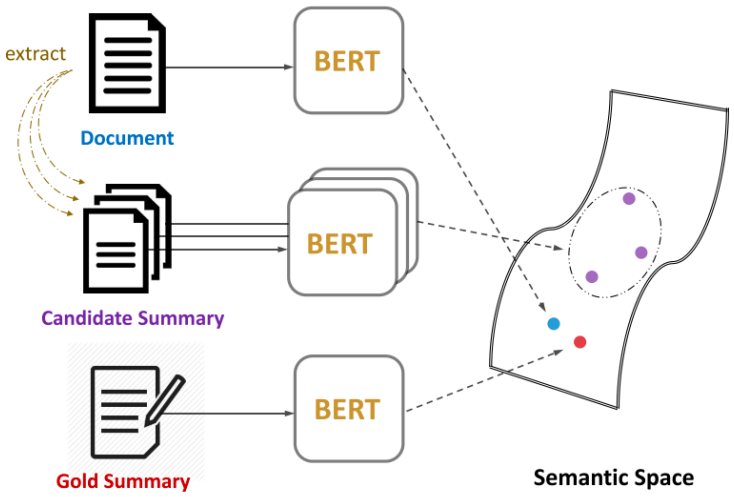
1.需要很大的内存
2.不能及时更新

1.维护一个queue
更新负样本

Momentum Contrast for Unsupervised Visual Representation Learning

Kaiming He Haoqi Fan Yuxin Wu Saining Xie Ross Girshick CVPR 2020

Application on Text Summarization



$$\mathcal{L}_1 = \max(0, f(D, C) - f(D, C^*) + \gamma_1), \quad (7)$$

$$\mathcal{L}_2 = \max(0, f(D, C_j) - f(D, C_i) + (j - i) * \gamma_2) \quad (i < j), \quad (8)$$

Model	Multi-News		
	R-1	R-2	R-L
LEAD	43.08	14.27	38.97
ORACLE	49.06	21.54	44.27
MATCH-ORACLE	47.45	17.41	43.14
BERTEXT	45.80	16.42	41.53
+ 3gram-Blocking	44.94	15.47	40.63
+ 4gram-Blocking	45.86	16.23	41.57
MATCHSUM (BERT-base)	**46.20**	**16.51**	**41.89**

Extractive Summarization as Text Matching
Ming Zhong and Xipeng Qiu et al (ACL 2020)

Application on Machine Translation

Algorithm 1 Contrastive Learning for NMT

Input: $D = \{\langle \mathbf{x}^{(s)}, \mathbf{y}^{(s)} \rangle\}_{s=1}^{S}$

Output: $\hat{\boldsymbol{\theta}}_{\mathrm{CL}}$

1: Obtain $\hat{\boldsymbol{\theta}}_{\mathrm{MLE}}$ using maximum likelihood estimation on D with random initialization;

2: Construct $\tilde{D} = \{\langle \mathbf{x}^{(s)}, \tilde{\mathbf{y}}^{(s)} \rangle\}_{s=1}^{S}$ based on D automatically;

3: Obtain $\hat{\boldsymbol{\theta}}_{\mathrm{CL}}$ using contrastive learning on $\tilde{D}$ with $\hat{\boldsymbol{\theta}}_{\mathrm{MLE}}$ as a starting point.

$$\hat{\boldsymbol{\theta}}_{\mathrm{MLE}} = \underset{\boldsymbol{\theta}}{\mathrm{argmax}} \Big\{ L(\boldsymbol{\theta}) \Big\},$$

$$L(\boldsymbol{\theta}) = \sum_{s=1}^{S} \log P(\mathbf{y}^{(s)} | \mathbf{x}^{(s)}; \boldsymbol{\theta})$$

$$\hat{\boldsymbol{\theta}}_{\mathrm{CL}} = \underset{\boldsymbol{\theta}}{\mathrm{argmin}} \Big\{ J(\boldsymbol{\theta}) \Big\},$$

where the max-margin loss is defined as

$$J(\boldsymbol{\theta}) = \sum_{s=1}^{S} \max \Bigg\{ \sum_{n=1}^{N} \log P(\tilde{\mathbf{y}}_n^{(s)} | \mathbf{x}^{(s)}; \boldsymbol{\theta}) + \eta - N \log P(\mathbf{y}^{(s)} | \mathbf{x}^{(s)}; \boldsymbol{\theta}), 0 \Bigg\}.$$

Reducing Word Omission Errors in Neural Machine Translation: A Contrastive Learning Approach

Zonghan Yang, Yong Cheng, Yang Liu, Maosong Sun(ACL 2019)

Application on Machine Translation

Algorithm 1 Contrastive Learning for NMT

Input: $D = \{\langle \mathbf{x}^{(s)}, \mathbf{y}^{(s)} \rangle\}_{s=1}^{S}$

Output: $\hat{\boldsymbol{\theta}}_{\mathrm{CL}}$

1: Obtain $\hat{\boldsymbol{\theta}}_{\mathrm{MLE}}$ using maximum likelihood estimation on D with random initialization;

2: Construct $\tilde{D} = \{\langle \mathbf{x}^{(s)}, \tilde{\mathbf{y}}^{(s)} \rangle\}_{s=1}^{S}$ based on D automatically;

3: Obtain $\hat{\boldsymbol{\theta}}_{\mathrm{CL}}$ using contrastive learning on $\tilde{D}$ with $\hat{\boldsymbol{\theta}}_{\mathrm{MLE}}$ as a starting point.

$$\hat{\boldsymbol{\theta}}_{\mathrm{CL}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \Big\{ J(\boldsymbol{\theta}) \Big\},$$

where the max-margin loss is defined as

$$J(\boldsymbol{\theta}) = \sum_{s=1}^{S} \max \Bigg\{ \sum_{n=1}^{N} \log P(\tilde{\mathbf{y}}_n^{(s)} | \mathbf{x}^{(s)}; \boldsymbol{\theta}) + \eta - N \log P(\mathbf{y}^{(s)} | \mathbf{x}^{(s)}; \boldsymbol{\theta}), 0 \Bigg\}.$$

For each ground-truth sentence pair $\langle \mathbf{x}^{(s)}, \mathbf{y}^{(s)} \rangle$, it is possible to sample N negative examples $\langle \mathbf{x}^{(s)}, \tilde{\mathbf{y}}_1^{(s)} \rangle, \ldots, \langle \mathbf{x}^{(s)}, \tilde{\mathbf{y}}_N^{(s)} \rangle$. For simplicity, we set $N = 1$ and use $\tilde{D} = \{\langle \mathbf{x}^{(s)}, \tilde{\mathbf{y}}^{(s)} \rangle\}_{s=1}^{S}$ in our experiments.

Reducing Word Omission Errors in Neural Machine Translation: A Contrastive Learning Approach

Zonghan Yang, Yong Cheng, Yang Liu, Maosong Sun(ACL 2019)

Application on Machine Translation

Dataset: WMT2017

Systems	EN-DE BLEU	EN-DE #Speed	EN-DE #Param	ZH-EN BLEU	ZH-EN #Param
Existing NMT systems					
Trans.base (Vaswani et al., 2017)	27.3	N/A	65.0M	N/A	N/A
+Context-Aware SANs (Yang et al., 2019a)	28.26	N/A	106.9M	24.67	126.8M
+Convolutional SANs (Yang et al., 2019b)	28.18	N/A	88.0M	24.80	N/A
+BIARN (Hao et al., 2019)	28.21	N/A	97.4M	24.70	107.3M
Trans.big (Vaswani et al., 2017)	28.4	N/A	213.0M	N/A	N/A
+Context-Aware SANs (Yang et al., 2019a)	28.89	N/A	339.6M	24.56	379.4M
+Convolutional SANs (Yang et al., 2019b)	28.74	N/A	339.6M	25.01	N/A
+BIARN (Hao et al., 2019)	28.98	N/A	333.5M	25.10	373.3M
Our NMT systems					
Trans.base	27.48	13.2K	66.5M	24.28	74.7M
+SCWAContext	28.28+	12.1K	72.8M	24.79+	81.0M
+TCWALoss	27.94+	14.3K	66.5M	24.65	74.7M
+BCWAContLoss	28.51+	13.1K	72.8M	24.94+	81.0M
Trans.big	28.45	11.2K	221.1M	24.55	237.5M
+BCWAContLoss	29.14+	10.1K	246.3M	25.12+	262.7M

Fig. [2]

Method	Zh-En	De-En	Ru-En
MLE	23.90	34.88	31.24
MLE + CP	24.04	34.93	31.36
WordDropout	23.73	34.63	31.05
CL_{one}	24.92 $^{++**\dagger\dagger}$	35.74 $^{++**\dagger\dagger}$	32.04 $^{++**\dagger\dagger}$
CL_{two}	24.76 $^{++**\dagger\dagger}$	35.54 $^{++**\dagger\dagger}$	31.94 $^{++*\dagger\dagger}$
CL_{three}	24.52 $^{+*\dagger\dagger}$	35.44 $^{++*\dagger\dagger}$	32.20 $^{++**\dagger\dagger}$
CL_{low}	24.13 †	34.96 †	31.47 $^{++\dagger}$
CL_{high}	24.77 $^{++**\dagger\dagger}$	35.24 $^{++\dagger\dagger}$	31.70 $^{++\dagger\dagger}$
CL_{V}	24.12 †	35.02 ††	31.73 $^{++*\dagger\dagger}$
CL_{IN}	24.71 $^{++**\dagger\dagger}$	35.26 $^{+*\dagger\dagger}$	31.76 $^{++*\dagger\dagger}$

Fig. [1]

[1]Reducing Word Omission Errors in Neural Machine Translation:
A Contrastive Learning Approach
Zonghan Yang, Yong Cheng, Yang Liu, Maosong Sun(ACL 2019)
[2]Content Word Aware Neural Machine Translation
Kehai Chen, Rui Wang∗, Masao Utiyama, and Eiichiro
Sumita(ACL 2020)

Application on Commonsense Reasoning

Task: Pronoun Disambiguation and Winograd Schema Challenge problems.

Sentence-1: *The trophy doesn't fit in the suitcase because* ***it*** *is too small.*
Answers: A) the trophy ***B)*** the suitcase

Sentence-2: *The trophy doesn't fit in the suitcase because* ***it*** *is too big.*
Answers: A) the trophy ***B)*** the suitcase

Contrastive Self-Supervised Learning for Commonsense Reasoning
Tassilo Klein, Moin Nabi(ACL 2020)

Application on Commonsense Reasoning

PDP-60 (sup.) (Davis et al., 2016)	
Patric Dhondt (WS Challenge 2016)	45.0 %
Nicos Issak (WS Challenge 2016)	48.3 %
Quan Liu (WS Challenge 2016-*winner*)	58.3 %
USSM + Supervised DeepNet	53.3 %
USSM + Supervised DeepNet + 3 KB	66.7 %
BERT-ft (Kocijan et al., 2019)	78.3 %
PDP-60 (unsupervised)	
Unsupervised Sem. Similarity (USSM)	55.0 %
Transformer LM (Vaswani et al., 2017)	58.3 %
BERT LM (Trinh and Le, 2018)	60.0 %
MAS (Klein and Nabi, 2019)	68.3 %
DSSM (Wang et al., 2019)	75.0 %
CSS (Proposed Method)	**90.0 %**

WSC-273 (sup.) (Levesque et al., 2012)	
USSM + KB	52.0%
USSM + Supervised DeepNet + KB	52.8 %
Transformer (Vaswani et al., 2017)	54.1 %
Know. Hunter (Emami et al., 2018)	57.1 %
GPT-ft (Kocijan et al., 2019)	67.4 %
BERT-ft (Kocijan et al., 2019)	71.4 %
WSC-273 (unsupervised)	
Single LMs (Trinh and Le, 2018)	54.5 %
MAS (Klein and Nabi, 2019)	60.3 %
DSSM (Wang et al., 2019)	63.0 %
Ensemble LMs (Trinh and Le, 2018)	63.8 %
CSS (Proposed Method)	**69.6 %**
DPR (Rahman and Ng, 2012)	
(Rahman and Ng, 2012)	73.0%
(Peng et al., 2015)	76.4 %
CSS (Proposed Method)	**80.1 %**
KnowRef (Emami et al., 2019)	
E2E (Emami et al., 2019)	58.0 %
BERT-ft (Emami et al., 2019)	61.0 %
CSS (Proposed Method)	**65.5 %**

Table 1: Results on different tasks. From Top to bottom: PDP, WSC, DPR, KnowRef. The first two task performances are subdivided into two parts. Upper part: supervised, lower part: unsupervised.

Contrastive Self-Supervised Learning for
Tassilo Klein, Moin Nabi(ACL 2020)

Other reference

[1]CONTRASTIVE LEARNING OF STRUCTURED WORLD MODELS
Thomas Kipf, Elise van der Pol, Max Welling (ICLR 2020)
[2] DEEP GRAPH INFOMAX
Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò,
Yoshua Bengio, R Devon Hjelm (ICLR 2019)
[3]Adversarial Contrastive Estimation.
Avishek Joey Bose, Huan Ling, Yanshuai Cao (ACL 2018)
[4] Contrastive Language Adaptation for Cross-Lingual Stance Detection
Mitra Mohtarami, James R. Glass, Preslav Nakov (EMNLP 2019)