Debiased Contrastive Learning

Ching-Yao Chuang, Joshua Robinson, Yin Yen-Chen, Antonio Torralba, Stepanie Jegelka

Presented by Meilu Yuan

Outline

- Premier - contrastive learning (concepts and example work - SimCLR);
- Sampling Bias;
- Debiased Contrastive Loss Function;
- Experiment Results;
- Generalization Bound;

Premier – Contrastive Learning

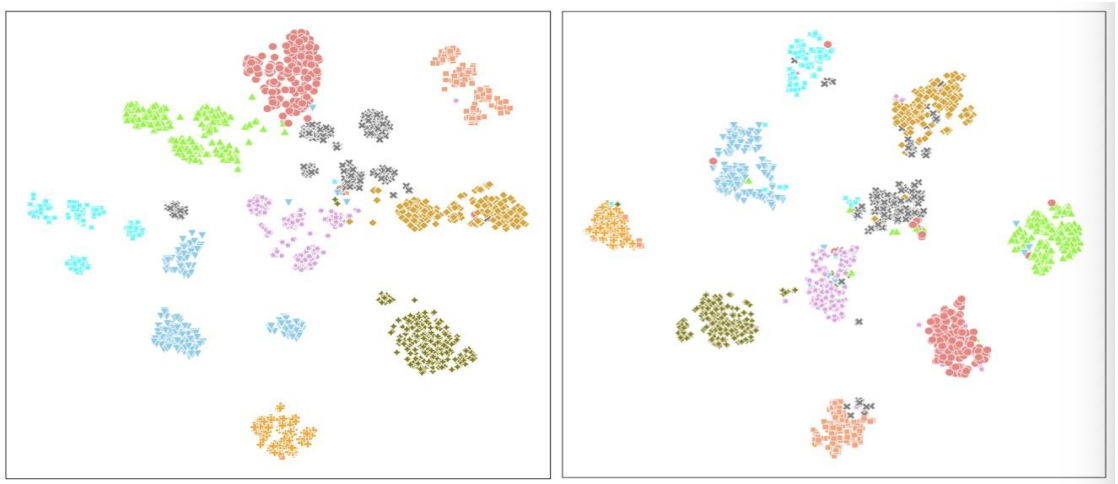
Contrastive Representation Learning -

- Key Idea - contrast semantically similar (positive) and dissimilar (negative) pairs of data points, encouraging the representations of similar pairs (x, x+) to be closer and those of dissimilar pairs (x,x-) to be far apart/more orthogonal;
- Can be applied in supervised and unsupervised tasks [recently, one of most powerful approaches in self-supervised learning];
- 2 Key Components - 1) Data pairs as input; 2) Loss functions e.g. contrastive loss, triplet loss, NCE, infoNEC, Soft-Nearest Neighbors Loss;
- Example (will be covered) - SimCLR(SSL) ;

Contrastive Learning – Core Idea/Key Ingredients

Core Idea (shown in visualization)

Supervised Learning Task (color = class)



without contrastive learning

with contrastive learning

PS - a random t-SNE plot to show the core idea, pls do not take it too serious.

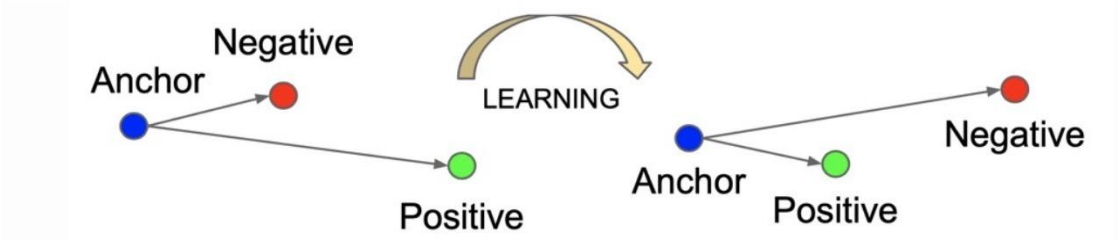
Contrastive Learning – Training Objectives

Contrastive Loss Function is based on the similarity between the representations of input data points.

Original Contrastive Loss => A Pair of Data as Input (Chopra et al. 2005)

$$\mathcal{L}_{\text{cont}}(\mathbf{x}_i, \mathbf{x}_j, \theta) = \mathbb{1}[y_i = y_j] \|f_\theta(\mathbf{x}_i) - f_\theta(\mathbf{x}_j)\|_2^2 + \mathbb{1}[y_i \neq y_j] \max(0, \epsilon - \|f_\theta(\mathbf{x}_i) - f_\theta(\mathbf{x}_j)\|_2)^2$$

Triplet Loss => Data Trio as Input (Schroff et al. 2015)



$$\mathcal{L}_{\text{triplet}}(\mathbf{x}, \mathbf{x}^+, \mathbf{x}^-) = \sum_{\mathbf{x} \in \mathcal{X}} \max\left(0, \|f(\mathbf{x}) - f(\mathbf{x}^+)\|_2^2 - \|f(\mathbf{x}) - f(\mathbf{x}^-)\|_2^2 + \epsilon\right)$$

Contrastive Learning – Training Objectives

……

Soft-Nearest Neighbors Loss => extends to include multiple pos and neg data as input

(Frosst et al. 2019)

Given a batch of data $\{x_i, y_i\}_1^B$, for each data point x_i, calculate the loss value with $L_{snn\_x_i}$, then take the expectation over a L_{snn} of data as .

Where function $sim(\cdot,\cdot)$ measures the similarity between representations.

$$L_{snn\_x_i} = -log \frac{\sum_{i \neq j,\ y_i = y_j,\ j=1..B} \exp\left(sim\left(f(x_i),\ f(x_j)\right)/\tau \right)}{\sum_{i \neq k, k=1...B} \exp\left(sim\left(f(x_i),\ f(x_j)\right)/\tau \right)}$$

Numerator - all positive pairs!

Denominator - all pairs!

$$L_{snn} = -\frac{1}{B}\sum_{i=0}^{B} log \frac{\sum_{i \neq j,\ y_i = y_j,\ j=1..B} \exp\left(sim\left(f(x_i),\ f(x_j)\right)/\tau \right)}{\sum_{i \neq k, k=1...B} \exp\left(sim\left(f(x_i),\ f(x_j)\right)/\tau \right)}$$

We can have a temperature **τ** to tuning how concentrated the features will be, in representation space

E.g. Small **τ** => loss will be dominated by small distances so widely separated representations can't contribute to loss too much;

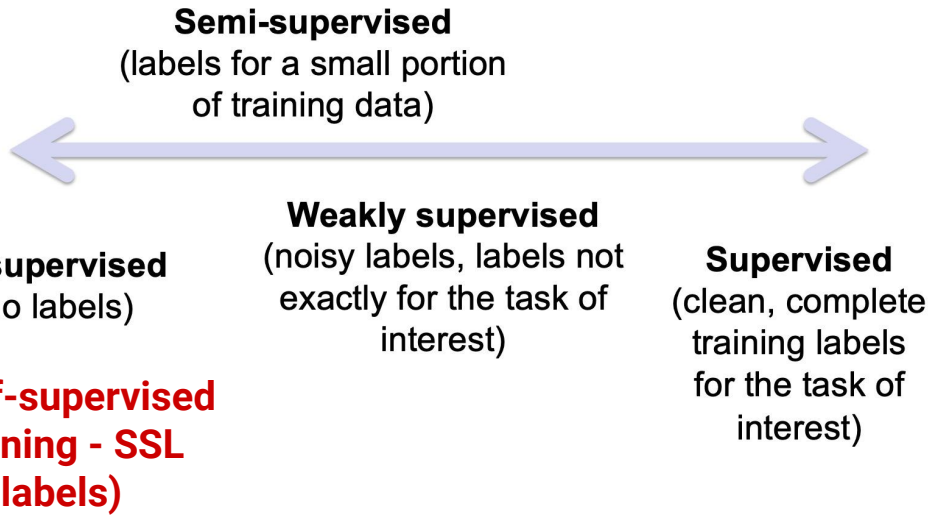
Interlude – Contrastive Learning for Self-supervised Learning

- **Self-Supervised Learning -**

A form of unsupervised learning where the data provides the supervision;

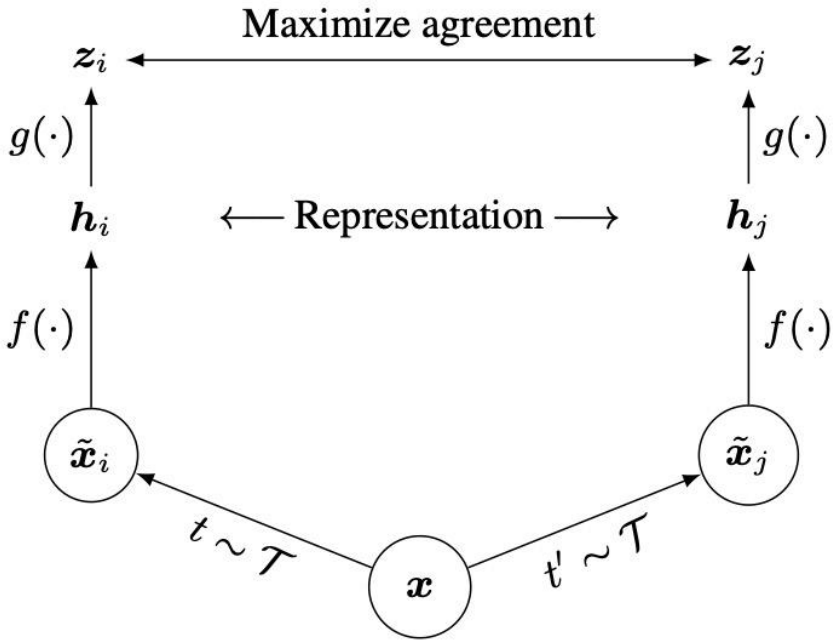
In general,

1) Withhold some part of the data point (a patch of image) and task the network with predicting unknowns (image);
2) Generate labels from data itself;



Contrastive Learning (example) – SimCLR

(SimCLR was designed for self-supervised ImageNet Classification task.)



No access to the true labels!

SimCLR learns representations by maximizing agreement between differently augmented views of the same data example via a contrastive loss in the latent space.

1. Two Different Random transformations on the anchor image;
2. NN-based encoder f(·) to extract representation vectors from augmented data (No Constraint!);
3. Small NN projector g(·) that maps reps to latent space where contrastive loss is applied;
4. Contrastive loss calculated on whole batch;

T. Chen et al. A Simple Framework for Contrastive Learning of Visual Representation. ICML'20

Contrastive Learning (example) – SimCLR

Loss Function of SimCLR

- **No explicit negative data points!**
- **For a pair of distorted images, all 2(N-1) distorted images will be treated negative data point.**

1. Loss calculation for a pair of distorted images;

$$l(i,j) = -\log \frac{\exp\left(sim(z_i, z_j)/\tau\right)}{\sum_{k=1,\ k\neq i}^{2N} \exp\left(sim(z_i, z_j)/\tau\right)}$$

Positive Pairs

Positive+Negative Pairs

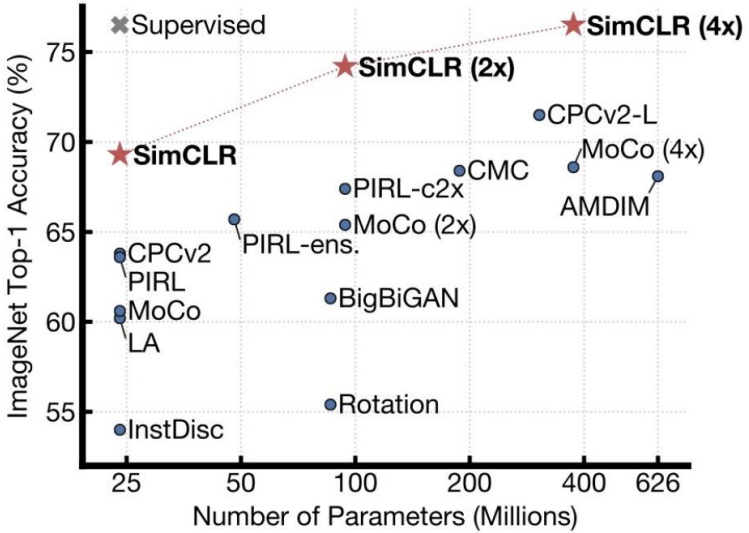
2. Loss function on a whole batch;

$$L_{SimCLR} = \frac{1}{2N} \sum_{k=1}^{N} [\, l(2k-1, 2k) + l(2k, 2k-1) \,]$$

T. Chen et al. A Simple Framework for Contrastive Learning of Visual Representation. ICML'20

Contrastive Learning (example) – SimCLR

Experiments and Performance

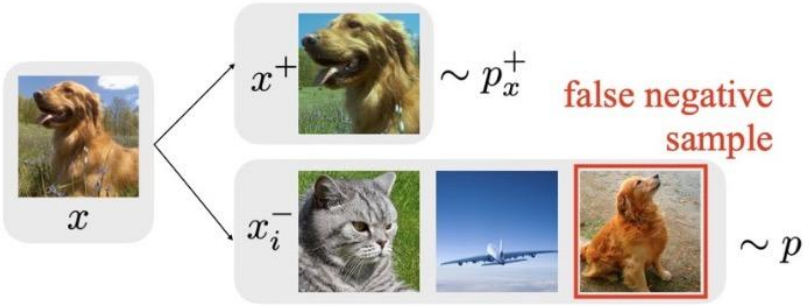


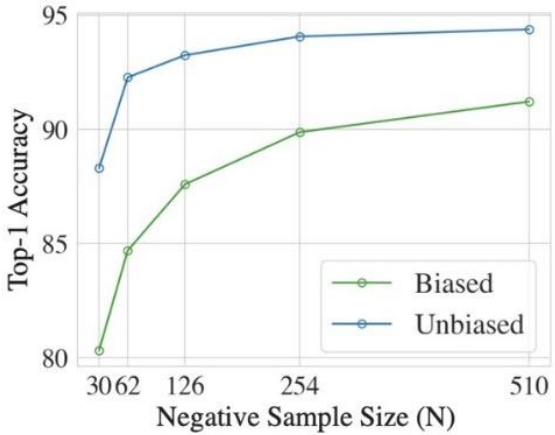
On ImageNet classification, when evaluated by top-1 accuracy, SimCLR outperforms all other self-supervised learning methods with a large gap.

T. Chen et al. A Simple Framework for Contrastive Learning of Visual Representation. ICML'20

Sampling Bias

- Unsupervised learning, don't have access to the true labels, in reality, negative data points might have the same class label as anchor x;





On cifar10

Sampling bias - False negative samples can result in a big performance drop

Debiased Contrastive Learning

“It develops a debiased contrastive objective that corrects for the sampling of the same-label datapoints without knowing the true labels;

Empirically, proposed contrastive objective consistently outperforms SOTA of SSL representation learning on vision, language , and RL benchmarks;

Theoretically, established generalization bound for downstream classification tasks;
”

Debiased Contrastive Loss

Problem Setup - biased and unbiased contrastive losses for one (anchor) data point.

Biased contrastive loss
(calculated based on real sample)

$$\mathbb{E}_{x,x^+,\{x_i^-\}_{i=1}^N}\left[-\log\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)}+\sum_{i=1}^N e^{f(x)^T f(x_i^-)}}\right]. \quad (1)$$

$$L_{\text{Unbiased}}^N(f)=\mathbb{E}_{\substack{x\sim p,x^+\sim p_x^+\\ x_i^-\sim p_x^-}}\left[-\log\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)}+\frac{Q}{N}\sum_{i=1}^N e^{f(x)^T f(x_i^-)}}\right], \quad (2)$$

Q - weighting factor

p(x), p(x-) is unknown, p(x+) is known
[because positive examples x+ are generated e.g. by applying image transformation];

Debiased Contrastive Loss

the class probabilities $\rho(c) = \tau^+$ are uniform, and let $\tau^- = 1 - \tau^+$ be the probability of observing any different class.

Debiased Contrastive Loss Function -

$$e^{-1/t} \leq \mathbb{E}_{x^- \sim p_x^-} e^{f(x)^T f(x_i^-)}$$

Neg (false neg) pos re-scale To cancel out false neg

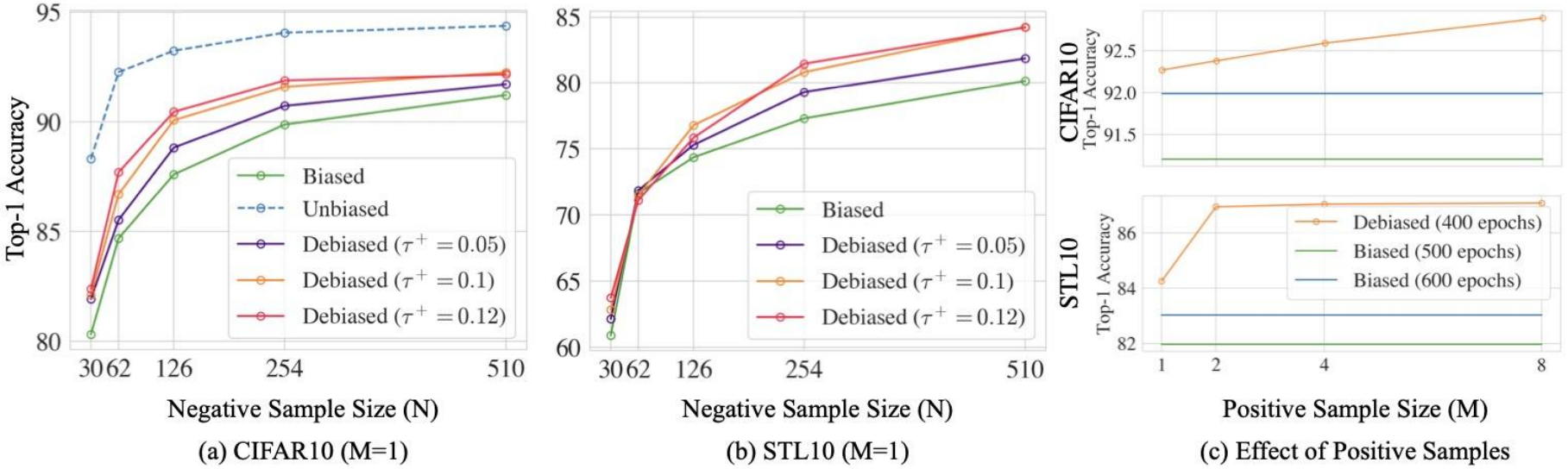
$$g(x, \{u_i\}_{i=1}^N, \{v_i\}_{i=1}^M) = \max\left\{\frac{1}{\tau^-}\left(\frac{1}{N}\sum_{i=1}^{N} e^{f(x)^T f(u_i)} - \tau^+ \frac{1}{M}\sum_{i=1}^{M} e^{f(x)^T f(v_i)}\right),\ e^{-1/t}\right\}. \quad (7)$$

$$L_{\text{Debiased}}^{N,M}(f) = \mathbb{E}_{\substack{x\sim p;\, x^+\sim p_x^+ \\ \{u_i\}_{i=1}^N \sim p^N \\ \{v_i\}_{i=1}^N \sim p_x^{+M}}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + N g\left(x, \{u_i\}_{i=1}^N, \{v_i\}_{i=1}^M\right)}\right], \quad (8)$$

Intuitively,
This debiased loss tries to include some positive samples (which can be generated) in the denominator to balance out false negative samples then rescale the term of negative pairs.

Experiment Results

Cifar10&STL10 - SimCLR + ResNet50(encoder), adam, lr=0.001, temperature t =0.5, dim of latent space=128



Applying Debiased Contrastive Loss Improves performance, larger N,M lead to better performance (Theorem3)

Experiment Results

Cifar10&STL10 - SimCLR + ResNet50(encoder), adam, lr=0.001, temperature t =0.5, dim of latent space=128

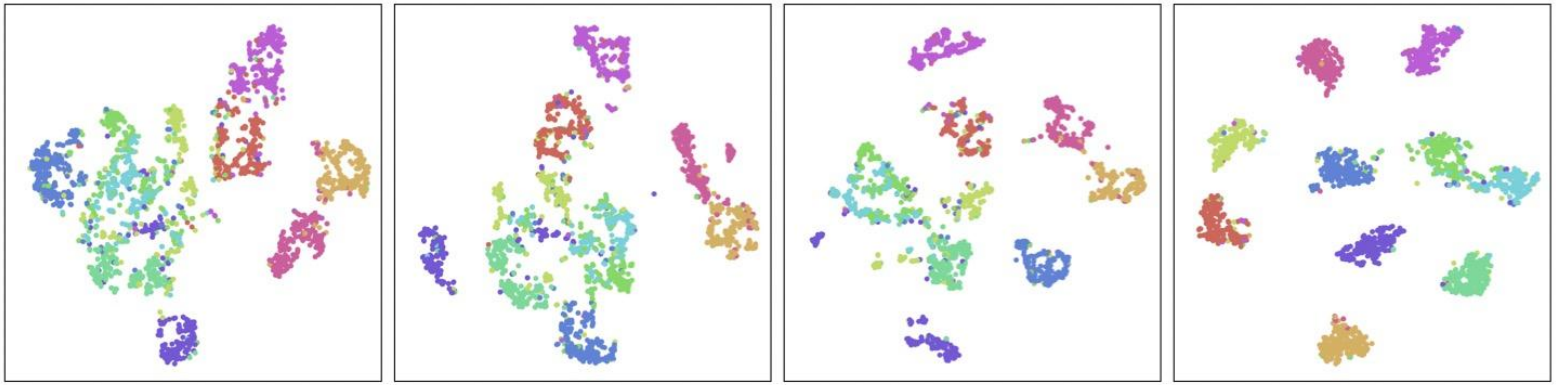


Figure 5: **t-SNE visualization of learned representations on CIFAR10.** Classes are indicated by colors. The debiased objective ($\tau^+ = 0.1$) leads to better data clustering than the (standard) biased loss; its effect is closer to the supervised unbiased objective.

Experiment Results

ImageNet100 (randomly sampled 100 classes from ImageNet)

Objective	Top-1	Top-5
Biased (CMC)	73.58	92.06
Debiased ($\tau^+ = 0.005$)	73.86	91.86
Debiased ($\tau^+ = 0.01$)	**74.6**	**92.08**

Table 1: **ImageNet-100** Top-1 and Top-5 classification results.

PS - Changed baseline as CMC (performs better than SimCLR)

Thank You!