Debiased Contrastive Learning
(2020)

Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, Stefanie Jegelka

Summary

Contributions

In contrastive learning the objective is to contrast, semantically similar pairs, i.e., positives (x, x^+), and dissimilar pairs, i.e., negatives (x, x^-). During training, the network f is encouraged to embed the positives similarly and negatives in a more orthogonal manner, by optimizing the following a contrastive loss:

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

However, in the loss above, since in an unsupervised scenario, we don not have access to the labels, the positives are replaced with a given image, and its augmented version, and the N negatives counterparts x^- are uniformly drawn from the training data. In such a case, we might have a sampling bias, where we sample a false negative x^- that is similar to x as illustrated in the Figure below.

Figure 1: **"Sampling bias":** The common practice of drawing negative examples x_i^- from the data distribution $p(x)$ may result in x_i^- that are actually similar to x.

The authors of this work propose an unbiased contrastive loss while assuming access to only unlabeled data by directly approximating the negative examples.

Method

To derive the unbiased contrastive loss, let's start by defining some notations:

- The Observations x are drawn from a data distribution $p(x)$ over $\mathcal{X}$.
- An embedding function $f : \mathcal{X} \rightarrow \mathbb{R}^d$, mapping the input space to a hypersphere (i.e., we L^2 normalize the outputs to compute the cosine similarity, projecting the output vector into the hypersphere) with radius $1/t$ with t as the scaling temperature.
- Discrete latent classes $\mathcal{C}$, where positive pairs (x, x^+) have the same latent class.

- The distribution over classes is $\rho(c)$, and the joint probability is $p_{x,c}(x,c) = p(x \mid c)\rho(c)$.
- With $h : \mathcal{X} \rightarrow \mathcal{C}$ as the function assigning the latent class labels, the probability of a given sample x' being a positive for x is: $p_x^+(x') = p(x' \mid h(x') = h(x))$. Similarity, for x' being a negative: $p_x^-(x') = p(x' \mid h(x') \neq h(x))$.
- The class probabilities are assumed to be uniform $\rho(c) = \tau^+$, with $\tau^- = 1-\tau^+$ as the probability of observing a different class.

The objective is to get closer to the ideal unbiased contrastive loss without having access to the true negatives p_x^-:

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

By decomposing the data distribution as $p(x') = \tau^+ p_x^+(x') + \tau^- p_x^-(x')$, a possible approach is to replace p_x^- with $p_x^-(x') = (p(x') - \tau^+ p_x^+(x'))/\tau^-$ and use the empirical counterparts for p and p_x^+.

$$\frac{1}{(\tau^-)^N}\sum_{k=0}^N \binom{N}{k}(-\tau^+)^k \mathbb{E}_{x\sim p, x^+\sim p_x^+, \{x_i^-\}_{i=1}^k\sim p_x^+, \{x_i^-\}_{i=k+1}^N\sim p}\left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N e^{f(x)^T f(x_i^-)}}\right]$$

But computing this loss is quite expensive for large N, and also requires N positives to estimate p_x^+. To overcome this, the authors propose an asymptotic form as the number of negatives N goes to infinity.

$$\mathbb{E}_{x\sim p_x, x^+\sim p_x^+, x_i^-\sim p_x^-}\left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \frac{Q}{N}\sum_{i=1}^N e^{f(x)^T f(x_i^-)}}\right]$$
$$\longrightarrow \mathbb{E}_{x\sim p, x^+\sim p_x^+}\left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \frac{Q}{\tau^-}\left(\mathbb{E}_{x^-\sim p}\left[e^{f(x)^T f(x^-)}\right] - \tau^+\mathbb{E}_{v\sim p_x^+}\left[e^{f(x)^T f(v)}\right]\right)}\right]$$

And by leveraging N negative samples from p and M positives from p_x^+, the second term in the denominator can be estimated as follows:

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

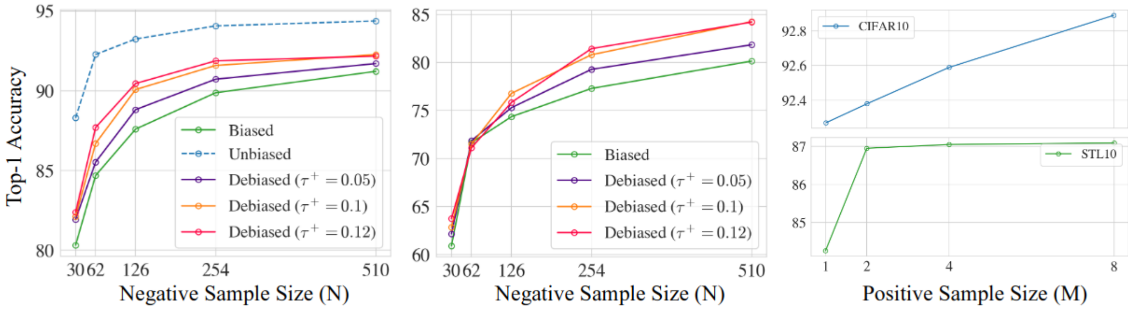
Resulting in the debiased contrastive loss:

$$L_{\text{Debiased}}^{N,M}(f) = \mathbb{E}_{x\sim p; x^+\sim p_x^+, \{u_i\}_{i=1}^N\sim p^N, v_i\}_{i=1}^N\sim p_x^{+M}}\left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + Ng\left(x, \{u_i\}_{i=1'}^N \{v_i\}_{i=1}^M\right)}\right]$$

```
# pos: exponential for positive example
# neg: sum of exponentials for negative examples
# N  : number of negative examples
# t  : temperature scaling
# tau_plus: class probability

standard_loss = -log(pos / (pos + neg))
Ng = max((-N * tau_plus * pos + neg) / (1-tau_plus), N * e**(-1/t))
debiased_loss = -log(pos / (pos + Ng))
```

Results



(a) CIFAR10 (M=1) (b) STL10 (M=1) (c) Effect of Positive Samples

Figure 4: **Classification accuracy on CIFAR10 and STL10.** (a,b) Biased and Debiased ($M = 1$) SimCLR with different negative sample size N where $N = 2(BatchSize - 1)$. (c) Increasing the positive sample size M improves the performance of debiased SimCLR.

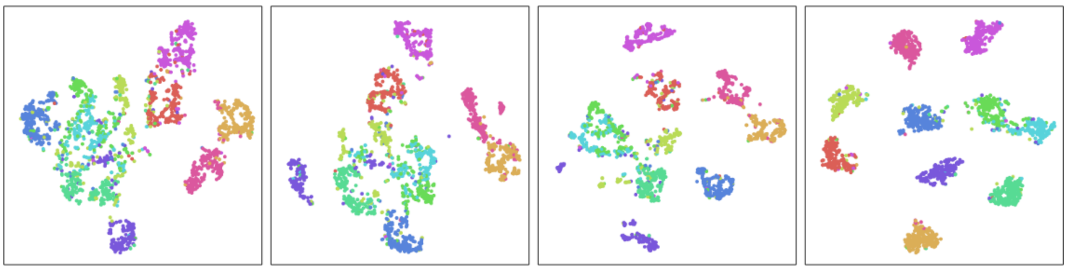

Biased Debiased M=1 Debiased M=8 Unbiased

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.

Objective	**MR**	**CR**	**SUBJ**	**MPQA**	**TREC**	**MSRP (Acc)**	**MSRP (F1)**
Biased (QT)	**76.8**	81.3	86.6	93.4	**89.8**	73.6	81.8
Debiased ($\tau^+ = 0.005$)	76.5	81.5	86.6	93.6	89.1	74.2	82.3
Debiased ($\tau^+ = 0.01$)	76.2	**82.9**	**86.9**	**93.7**	89.1	**74.7**	**82.7**

Table 2: **Classification accuracy on downstream tasks.** We compare sentence representations on six classification tasks. 10-fold cross validation is used in testing the performance for binary classification tasks (MR, CR, SUBJ, MPQA)