Lecture 19: Spectral Representation Learning

Lecturer: Bo Dai Scribes: Sandilya Sai Garimella & Vidhya Kewale

19.1 Recap

SimCLR (**Sim**ple **C**ontrastive **L**earning of Visual **R**epresentations)

- self-supervised contrastive learning method for visual representations
- works with a single modality: images
- trains the model to bring augmented views of the same image closer together in the embedding space, while pushing apart representations of different images [4]

CLIP (**C**ontrastive **L**anguage-**I**mage **P**re-training)

- extends contrastive learning to multiple modalities using image-text pairs
- learns aligned representations across the visual and textual domains
- trains the model to maximize the cosine similarity between embeddings of matching image-text pairs, while minimizing it for non-matching pairs [3]

Energy-Based Models (EBM) and Noise-Contrastive Estimation (NCE)

1. conditional probability in the same modality

$$p(x'|x) = p(x')exp(\varphi(x)^T\varphi(x^T))$$

2. cross-modal conditional probabilities

$$p(y|x) = p(y)exp(\varphi(x)^T\nu(y))$$
$$p(x|y) = p(x)exp(\varphi(x)^T\nu(y))$$

19.2 SimCLR

All of them use ranking-based NCE to estimate a special EBM. SIMCLR is as follows:

$$p\left(x' \mid x\right)=p\left(x'\right) \exp \left(\varphi(x)^{\top} \varphi\left(x'\right)\right)$$

We formulate the loss function using the follows:

$$\max_{\varphi} \sum_{i=1}^{n}\left[\varphi\left(x_i\right)^{\top} \varphi\left(x_i'\right)-\log \sum_{j=1}^{k} \exp \left(\varphi\left(x_i\right)^{\top} \varphi\left(x_j'\right)\right)\right]$$

$$\max_{\varphi} f(\theta); \quad D_\theta l(\theta)=\sum_{i=1}^{n} \varphi\left(x_i\right)^{\top} \varphi\left(x_i'\right) \cdot\left(\nabla_\theta \varphi\left(x_i\right)+\nabla_\theta \varphi\left(x_i'\right)\right)$$

$$\sum_{i=1}^{n}\left(\sum_{r=1}^{k} \frac{\exp \left(\varphi\left(x_i\right)^{\top} \varphi\left(x_i'\right)\right) \cdot\left(\nabla_\theta \varphi\left(x_i\right)+\nabla_\theta \varphi\left(x_i'\right)\right)}{\sum_{r=1}^{k} \exp \left(\varphi\left(x_r\right)^{\top} \varphi\left(x_i\right)\right)}\right)=O(nk)$$

computation cost is O(nk). We typically also use k=n, therefore the computation cost becomes O(n^2)

The explanation with data is we have:

$$\left\{x_i\right\}_{i=1}^{B}$$

$$\sim\left\{x_i'\right\}_{i=1}^{B}$$

Therefore the computation cost will be:

$$i,\{x-i\} \quad (B-1) \Rightarrow \sim O\left(B^2\right)$$

To circumvent this quadratic computation cost, we can use a binary-based NCE instead of a ranking-based NCE. With this, instead of $O(nk)$, we can get $O(2B) \sim O(B)$. This was presented in the paper cited here [5].

Coming back to this expression to derive spectral learning and Bootstrap your own latent (BYOL) [2]:

$$p\left(x^* \mid x\right)=p\left(x'\right) \exp \left(\varphi(x)^{\top} \varphi(x)\right)$$

We remove the exponential because it makes the gradient calculation harder:

$$p\left(x' \mid x\right)=p\left(x'\right) \varphi\left(x'\right)^{\top} \varphi(x)$$

The L2 loss function is now defined as:

$$\begin{aligned} & l_2 \int\left\|p\left(x' \mid x\right)-p\left(x'\right) \varphi\left(x'\right)^{\top} \varphi(x)\right\|^2 d x d x' \\ & =\int p\left(x' \mid x\right)^2 d x d x'-2 \int p\left(x' \mid x\right) p\left(x'\right) \quad p\left(x'\right)^{\top} \varphi(x) d x d x'\Big) \end{aligned}$$

$$p\left(x' \mid x\right) p(x) = p\left(x'\right) p(x) p\left(x'\right)^{\top} p(x) \text{ from}$$
$$p\left(x' \mid x\right) = p\left(x'\right) \varphi\left(x'\right)^{\top} \varphi(x)$$
$$\int \left\| \frac{p\left(x', x\right)}{\sqrt{p(x)}\sqrt{p(x)}} \sqrt{p\left(x'\right)} \sqrt{p(x)^2} \varphi\left(x'\right)' \varphi(x) \right\|^2 dx dx'$$
$$= \int \left(\frac{p\left(x', x\right)}{\sqrt{p(x)}\sqrt{p(x)}}\right)^2 dx dx' - 2 \int \left(p\left(x', x\right) p(x')^T \varphi(x)\right) dx dx' +$$
$$\int p(x') p(x) (\varphi(x')^T \varphi(x))^2 dx dx'$$

We observe that the terms in the integrals can be simplified using the definition of expectation; therefore we can apply sampling here. The above simplifies to:

$$= -2 E_{p(x,x')} \left[\varphi\left(x'\right)^{\top} \varphi(x)\right] + E_{p(xp(x)} \left[\left[\varphi(x')^{\top} (\varphi(x)\right)^2\right].$$

From above, we can see that we sample only once but can use it for computing both expectation terms.

$$p\left(x', x\right) = p(x) \varphi(x)^{\top} \quad p\left(x'\right) \varphi\left(x'\right)$$

but we write this as

$$p\left(x', x\right) = \Psi(x)^{\top} \Psi(x')$$

This is called the Eigen decomposition spectral perspective of representation.

19.2.1 BYOL w/o ν

The loss function is, using similar reason to above:

$$\min_{\varphi, \nu} \int || \left(\rho(x', x) \frac{\varphi(x')^T \varphi(x)}{\sqrt{\rho(x')\rho(x)}} \sqrt{\rho(x')} \sqrt{\rho(x)}\right) ||^2 \, dx' dx$$

Alternative Optimization

Add a constraint such that $\nu = \varphi$.

(min problem above) $\propto 2 E_{p(x',x)} \left[\nu(x)^T \varphi(x)\right] - E_{p(x',x)} \left[\varphi(x')^T \varphi(x) \varphi(x)^{\top} \varphi(x')\right]$

With the above expanded, we can do separate sampling.

$$\Lambda_t = E_{p(x)} \left[\nu_{\Psi}(x) \nu_{\Psi}(x)^T\right]$$

$$-2 E_{p(x,x')} \left[\varphi(x') \nu(x)^T\right] + E_{p(x)} \varphi(x)^T \Lambda_t \varphi(x)$$

19.3 PCA

Finding the maximal eigenspace while matching the y's are different.

We have the following, noting that the trace operator is invariant under cyclic permutations:

$$\hat{\rho} = (x, x') \in R^{n \times n}, \quad n \text{ samples}$$

$$\Psi(x) \in R^{n \times d}$$

$$E_{p(x,x')}\left[\Psi(x)\Psi(x')^T\right]$$

$$E_{p(x)}\left[\Psi(x)T\Psi(x)\right] = I_{d \times d}$$

Penalty method:

$$\max_{\Psi} E_{p(x,x')}\left[\Psi(x)\Psi(x)^T\right] - \lambda trace(E_{p(x)}\left(\Psi(x)\Psi(x)^T\right) - I)^2$$

The above is a variant of VICreg [1].

References

[1] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[2] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[3] Alexey Kovalev. Clip from openai: what is it and how you can try it out yourself, 2021.

[4] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training, 2021.

[5] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.