Towards Good Practices in Self-supervised Representation Learning

Srikar Appalaraju, Yi Zhu, Yusheng Xie, István Fehérvári
Amazon Inc.
{srikara, yzaws, yushx, istvanfe} @ amazon.com

Abstract

Self-supervised representation learning has seen remarkable progress in the last few years. More recently, contrastive instance learning has shown impressive results compared to its supervised learning counterparts. However, even with the ever increased interest in contrastive instance learning, it is still largely unclear why these methods work so well. In this paper, we aim to unravel some of the mysteries behind their success, which are the *good practices*. Through an extensive empirical analysis, we hope to not only provide insights but also lay out a set of best practices that led to the success of recent work in self-supervised representation learning.

1 Introduction

Self-supervised representation learning (SSL) has been a hot area of research in the last several years [1–11]. The allure of SSL is the promise of annotation-free ground-truth to ultimately learn a superior data representation (when compared to supervised learning). More recently, a type of contrastive learning method based on instance discrimination as pretext task [12, 13] has taken-off as it has been consistently demonstrated to outperform its supervised counterparts on downstream tasks like image classification and object detection [14–19, 11].

Putting the progress in contrastive instance learning aside for a moment, many recent observations seem to contradict what we have known from supervised learning. For example, [20, 19] have shown that simply adding a nonlinear projection head, i.e., one fully-connected (fc) layer and one activation layer, can significantly improve the quality of learned representations. Quantitatively, the nonlinear projection head can help to improve top-1 classification accuracy on ImageNet by over 10% in [19] and 5.6% in [21]. However, adding such a shallow multilayer perceptron (MLP) head is often not effective in supervised learning. Take another example, recent methods [17, 19] adopt aggressive and strong data augmentation during contrastive pre-training. Although data augmentation has been proven to be useful, overly aggressive augmentations often lead to worse results in supervised or other self-supervised learning methods. Then why contrastive instance learning does not suffer from strong data augmentation? At this moment, there is no concrete evidence to answer these questions.

After closely observing recent contrastive instance learning work, it becomes apparent that what seem to be design choices are in-fact good practices which are largely responsible to their disruptive success. Furthermore, some of these good practices can effectively be transferred to other non-contrastive, unsupervised learning methods [10]. Hence in this work, we focus on the importance of these design choices using extensive experimental evidence. We hope to provide insights to the self-supervised learning community, with the potential impact and application even beyond it. Specifically, our contributions include: 1) We empirically show why MLP head helps contrastive instance learning and visualize it using a feature inversion approach. 2) We present the semantic label shift problem caused by strong data augmentation in supervised learning and study the difference between supervised and contrastive learning. 3) We investigate on negative samples and find that good practices can help to eliminate the need of using large number of them, thereby could simplify the framework design.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

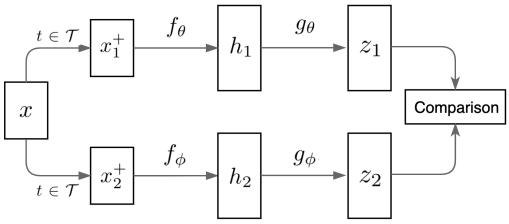


Figure 1: A generic visual depiction of recent contrastive instance learning methods.

Method	Top-1 Acc (%)
MoCov2	64.4
(a) no MLP head	59.2
(b) fixed MLP head	62.9
(c) deeper MLP head	63.9
(d) narrower MLP head	62.2

Table 1: Investigation of nonlinear projection head in MoCov2 [21] on ImageNet.

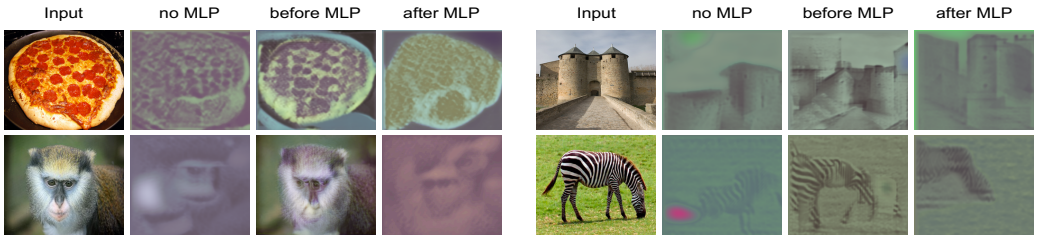


Figure 2: Visualization of feature inversion results by DIP [22]. Each example contains: (a) original image, (b) reconstructed image using h from a model trained without MLP head, (c) using h from a model trained with MLP head and (d) using z from a model trained with MLP head.

2 Nonlinear Projection Head

Before diving into the details, we first revisit recent contrastive instance learning methods and present a generic visual depiction of its framework in Figure 1. More formally, given a set of images $\mathcal{X}$, an image x is uniformly sampled from $\mathcal{X}$ and is augmented using various augmentation techniques $t \in \mathcal{T}$ to generate the positive pairs x_1^+ and x_2^+. f_θ and f_ϕ are encoders which map the images to visual representations h_1 and h_2. These visual representations are then projected via g_θ and g_ϕ, which are often MLP heads, to lower-dimensional features z_1 and z_2 for similarity comparison. By optimizing InfoNCE loss [14], the model learns to map similar instances closer and dissimilar instances farther apart in the embedding space.

A main goal of unsupervised representation learning is to learn features that are transferable to downstream tasks. Typically the outputs of the penultimate layer h are considered for transferring to other tasks. Recently, [20, 19, 21] have shown that simply adding a MLP head as shown in Figure 1 can significantly improve the quality of learned feature representation. However, adding such a shallow MLP head is often not so effective in supervised learning regime. So the question arises, *why adding a nonlinear projection head is so important for contrastive instance learning*?

In this paper we attempt to answer this question by designing experiments to explore different aspects of the nonlinear projection head in MoCov2 during unsupervised pre-training. First, as shown in Table 1 (a), removing the projection head g significantly degrades the classification accuracy compared to baseline. Next, we initialize the nonlinear projection head g with a uniform distribution and freeze its parameters during the unsupervised pre-training. Interestingly, as we can see in Table 1 (b), we obtain better representations compared to removing the projection head. This indicates that the nonlinear projection head is useful beyond its learning capability offered by the extra two layers. In fact, it is the nonlinear transformation itself that somehow benefits the learning process even if the parameters of this transformation is randomly initialized. To strengthen our observation, we investigate two other model variations. We first deepen the nonlinear projection head with more hidden layers, i.e., fc $\rightarrow$ ReLU $\rightarrow$ fc $\rightarrow$ ReLU which in theory adds more learning capacity. As shown in Table 1 (c), this seems not to bring extra benefits. We then narrow the nonlinear projection head by reducing the embedding dimensionality, e.g, $2048 \rightarrow 128$ for a ResNet50. As shown in Table 1 (d), such a drastic dimension reduction indeed lowers the performance compared to baseline, but still outperforms setting (a) by a large margin. With these insights we are more confident to conclude: it is the transformation of the projection head, which separates the pooled convolutional features from the final classification layer, that helps the representation learning. But why is such separation beneficial?

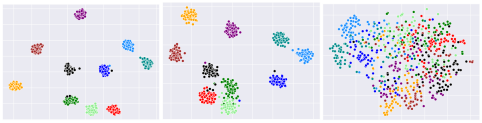
We argue that the nonlinear projection head acts as a filter separating the information-rich features useful for downstream tasks (i.e. color, rotation, or shape of objects) from the more discriminative

features that are more useful for the contrastive loss. This conjecture was previously introduced in [19] and verified by using features to predict transformations applied during the pre-training. In this work, we provide further visual evidence to support this hypothesis.

Inspired by deep image prior (DIP) [22], we perform feature inversion to obtain natural pre-images. By looking at the natural pre-image, we can diagnose which information is lost and which invariances are gained by the network. Specifically, we invert the features before and after the nonlinear projection head, h and z respectively. As we can see in Figure 2, using features before MLP projection head gives the best reconstruction result. Even though we use globally pooled features without spatial dimension, they are able to generate decent image reconstructions, maintaining most color, shape, location and orientation information. However, features learned without projection or features after projection head only preserve the most discriminative information to make classification. This observation supports our claim that layers close to loss computation will lose information due to invariances to data transformations induced by the contrastive loss.

3 Strong Data Augmentation

Data augmentation is an important regularization technique in training most deep learning models, ranging from AlexNet [23] to cutout [24] and autoAugment [25]. Empirical experience however shows that too strong augmentations (i.e., hard positives) are sometimes counterproductive in the supervised setting. In this section, we explore *why unsupervised contrastive instance learning can benefit from hard positive samples*.



(a) (b) (c)

Figure 3: t-SNE plots from 10 randomly chosen ImageNet classes. We use the 1000-d features from a ImageNet-trained ResNet50. The sub-figures use the same images but under: (a) no; (b) weak; (c) strong augmentation.

In contrastive learning without class labels, each image becomes its own class and as a result, there are no clear semantic class boundaries like they exist in supervised training. In order to illustrate how this hurts supervised learning, we provide a t-SNE visualization [26] in Figure 3 to show how the class boundaries in ImageNet break down when strong augmentations are applied. We term this phenomenon as *semantic label shift problem*. However, during instance discrimination, strong augmentation turn samples into hard positive samples that are recently found to be quite helpful in discriminating instance from instance [27].

Table 2: Investigation on different data augmentation settings' effect on MoCov2 [21] versus supervised training. The setting of color jittering strength follows SimCLR [19].

Method	baseline 20%/100%	medium 20%/50%	extreme 2%/10%
MoCo v2	64.4	63.7 −0.7	47.1 −17.3
Supervised	75.5	74.8 −0.7	52.0 −23.5

(a) Rand cropping strength (min/max size)

Method	from weak to strong 1/8	1/4	1/2	1
MoCo v2	63.1	63.9 +0.8	64.2 +1.1	64.3 +1.2
Supervised	75.8	75.7 −0.1	75.6 −0.2	74.6 −1.2

(b) Color jittering strength

Quantitatively, from Table 2a, we can see that as the cropping augmentation becomes more "extreme", MoCov2 performance suffers less than its supervised counterparts. The reason why performance of MoCov2 also drops might be MoCov2 is learning occlusion invariant features, a view that is corroborated by [28]. Such occlusion invariant learning process is hindered by aggressive cropping augmentation. In Table 2b, we show that stronger color jittering benefits MoCov2 but hurts supervised learning which is consistent with the findings from SimCLR [19].

4 Negative Samples

Computing contrastive loss requires sampling negative pairs to avoid learning collapsed representations. Recently, [13, 17, 19] have empirically shown that using a large number of negative samples is beneficial to learn good features in contrastive instance learning. However, one needs to design sophisticated mechanisms to store the negative examples and ways to update them. So we ask, *is it possible to use less negative examples during contrastive loss computation without performance degradation*?

Quantity of Negative Samples: In order to answer the question, we first run experiments on recent contrastive instance learning approaches using different number of negative examples. We choose InstDisc [13], MoCo [17], SimCLR [19] and MoCov2 [21] as illustrating methods. Although they

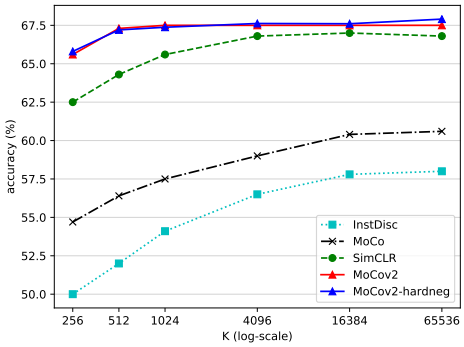


Figure 4: Comparison of different algorithms with varying number of negative samples. K denotes the number of negative samples. We show that MoCov2 performs the same ranging from $K = 512$ to $K = 65536$.

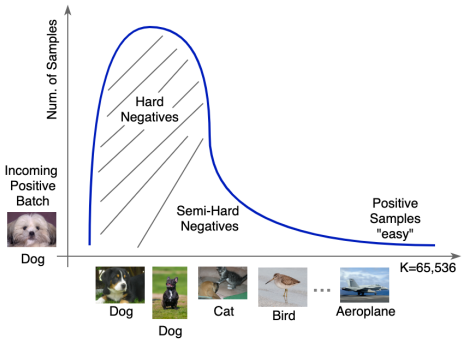


Figure 5: Illustration of how we perform dynamic hard-negative sampling on MoCov2 during training time for each mini-batch. We sort the entire dynamic queue based on similarity with incoming positive samples. We use positive skew-normal distribution to select more of hard and semi-hard samples and less of "easy" positive samples.

use negative examples in different manners, e.g., memory bank in InstDisc, queue in MoCo and MoCov2, and large mini-batch in SimCLR, they all use large number of negative examples. We adopt their official released code and follow the same hyper-parameter setting.

We find, that the performance of InstDisc and MoCo drops when number of negatives decreases unlike for SimCLR and MoCov2 where the impact is less, consistent with [13, 17, 19]. In fact, MoCov2 performs consistently with respect to the number of negatives and its accuracy on ImageNet stays the same ranging from $K = 512$ to $K = 65536$, where K denotes the number of negative samples. Hence, the *quantity* of negative samples during contrastive loss computation has little impact on final linear probe performance. We also find that AP on PASCAL VOC object detection task does not regress due to lower negatives, indicating the quality of the learned features remain the same no matter how many negatives were used (512 or 65536) during contrastive instance pretraining.

Quality of Negative Samples: What if we alter the *quality* of negative samples by performing dynamic hard-negative mining in the MoCov2 queue? Specifically, for a positive batch of images B and an existing dynamic queue Q, we sort Q for each batch based on cosine-similarity between the query-branch feature q and the latent features in Q. We mine hard and semi-hard negative exemplars from dynamic Q using a skew-norm distribution. Note, we over-sample in order to keep the overall queue size same as MoCov2. The intuition for this experiment can be seen in Figure 5. Contrasting among these hard-negative samples intuitively should improve the learned representation. However, as shown in Figure 4 (blue line), in spite of over-sampling hard-negative samples, performance of MoCov2 does not change for various sizes of K.

To summarize, in spite of *quantity* and *quality* of negative samples in queue Q, MoCov2 performance appears to be stable. We hypothese that the good practices introduced in MoCov2 (MLP head, GaussianBlur augmentations, cosine learning rate scheduling) are responsible for this effect. Thus, we conduct ablation studies to find out the which good practice(s) helps in eliminating the need for large number of negatives.

Good Practices: From Table 3, we can see that the MLP head and momentum encoder have the biggest impact (row a and d). Without these two techniques, the performance quickly drops as the number of negative samples decreases. This also explains why SimCLR is robust to the number of negatives with $K = 4096$ and $K = 65536$, however performance quickly degrades with $K = 256$ due to the lack of such mechanisms. We hope our empirical evidence can provide insights and better understanding of recent progress, as well as advance future development of self-supervised representation learning.

Table 3: Investigation of good practices' impact on the number of negative examples (K) being used in computing contrastive loss (wo: without, w: with).

Method	K=512	K=65536
MoCov2	67.3	67.5
(a) wo MLP projection head	61.7	63.6
(b) Data Aug: wo GaussianBlur	65.5	66.4
(c) wo cosine learning schedule	67.2	67.3
(d) w smaller mom. (0.5) in mom. encoder	59.8	64.5

References

[1] Gidaris, S., P. Singh, N. Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*. 2018.

[2] Noroozi, M., H. Pirsiavash, P. Favaro. Representation learning by learning to count. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5898–5906. 2017.

[3] Doersch, C., A. Gupta, A. A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430. 2015.

[4] Pathak, D., P. Krahenbuhl, J. Donahue, et al. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544. 2016.

[5] Noroozi, M., P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.

[6] Zhang, R., P. Isola, A. A. Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016.

[7] Caron, M., P. Bojanowski, A. Joulin, et al. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149. 2018.

[8] Donahue, J., P. Krähenbühl, T. Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[9] Agrawal, P., J. Carreira, J. Malik. Learning to see by moving. In *Proceedings of the IEEE international conference on computer vision*, pages 37–45. 2015.

[10] Grill, J.-B., F. Strub, F. Altché, et al. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. *arXiv preprint arXiv:2006.07733*, 2020.

[11] Caron, M., I. Misra, J. Mairal, et al. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[12] Dosovitskiy, A., J. T. Springenberg, M. Riedmiller, et al. Discriminative Unsupervised Feature Learning with Convolutional Neural Networks. In *NeurIPS*. 2014.

[13] Wu, Z., Y. Xiong, S. Yu, et al. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. In *CVPR*. 2018.

[14] Oord, A. v. d., Y. Li, O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[15] Tian, Y., D. Krishnan, P. Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[16] Misra, I., L. v. d. Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717. 2020.

[17] He, K., H. Fan, Y. Wu, et al. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*. 2020.

[18] Khosla, P., P. Teterwak, C. Wang, et al. Supervised Contrastive Learning. *arXiv preprint arXiv:2004.11362*, 2020.

[19] Chen, T., S. Kornblith, M. Norouzi, et al. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*. 2020.

[20] Bachman, P., R. D. Hjelm, W. Buchwalter. Learning Representations by Maximizing Mutual Information Across Views. In *NeurIPS*. 2019.

[21] Chen, X., H. Fan, R. Girshick, et al. Improved Baselines with Momentum Contrastive Learning. *arXiv preprint arXiv:2003.04297*, 2020.

[22] Ulyanov, D., A. Vedaldi, V. Lempitsky. Deep Image Prior. In *CVPR*. 2018.

[23] Krizhevsky, A., I. Sutskever, G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*. 2012.

[24] DeVries, T., G. W. Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[25] Cubuk, E. D., B. Zoph, D. Mané, et al. Autoaugment: Learning augmentation strategies from data. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 113–123. 2019.

[26] van der Maaten, L., G. Hinton. Visualizing Data using t-SNE. *IMLR*, 2008.

[27] Khosla, P., P. Teterwak, C. Wang, et al. Supervised contrastive learning, 2020.

[28] Purushwalkam, S., A. Gupta. Demystifying Contrastive Self-Supervised Learning: Invariances, Augmentations and Dataset Biases? *arXiv preprint arXiv:2007.13916*, 2020.