Closing the Performance Gap between Cumbersome and Lightweight Contrastive Models

Anonymous authors
Paper under double-blind review

Abstract

While self-supervised contrastive learning has made continuous progress utilizing big models, the performance lags far behind when the model size decreases. A common practice to address this problem requires a two-stage training procedure, where a larger model is pretrained in a self-supervised manner first, then its representational knowledge is transferred to a smaller model in the second stage. Despite its effectiveness, this method is highly time-consuming and is inapplicable to some resource-limited scenarios. In this work, we are aiming at directly training a lightweight contrastive model with satisfactory performance in the absence of a pretrained teacher model. Specifically, by incorporating some training techniques (e.g., MLP, lower temperature, et al.), we boost the accuracy of different lightweight models by a large margin. Besides, we observe a common issue in contrastive learning that either the positive views or the negative views can be noisy, and propose a smoothed version of InfoNCE loss to alleviate this problem. With these combined techniques, we successfully improve the linear evaluation results from 36.3% to 62.3% of MobileNet-V3-Large and from 42.2% to 65.8% of EfficientNet-B0 on ImageNet, closing the accuracy gap to ResNet-50 which contains 5× parameters. These results suggest the feasibility to train lightweight self-supervised models without distillation.

1 Introduction

Self-supervised contrastive learning targets at learning semantic representation of instances through pretraining on the instance discrimination pretext task, where a model is trained to align augmented views from the same instance, while pushing away from those of others in the representation space. The leading contrastive learning methods utilizing large models have already shown the superiority to the supervised counterpart, after fine-tuning the pretrained models on downstream tasks (Chen et al., 2020a; He et al., 2020; Grill et al., 2020; Chen & He, 2021).

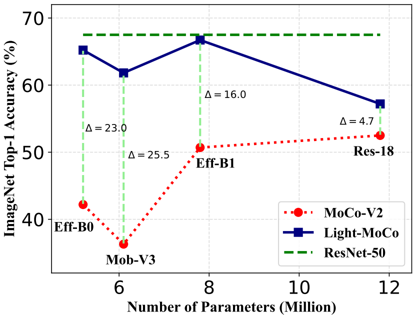

Figure 1: Linear probe top-1 accuracy of Light-MoCo and MoCo v2 on ImageNet. Green dash line is the ResNet-50 result trained by MoCo v2. Our method dramatically boosts the performance based on MoCo v2 without a teacher model. We bridge the gap between lightweight models and the larger ones.

However, their performance all suffers a cliff fall when the model size decreases. For example, the supervised top-1 accuracy on ImageNet of ResNet-50/EfficientNet-B0 is 76.1%/77.1%, but the corresponding linear evaluation counterpart applying MoCo-V2 (Chen et al., 2020c) is 67.5%/42.2%, showing a large gap. To rescue the performance of small models, a widely adopted practice is to leverage a large pretrained model to transfer the representational knowledge through knowledge distillation (Hinton et al., 2015). It first pretrains a large model in the self-supervised contrastive manner serving as a teacher model, and then let the smaller student model mimic the representation distribution of the teacher (Abbasi Koohpayegani et al., 2020; Fang et al., 2020; Xu et al., 2021; Gao et al., 2021). In spite of the effectiveness, this two-stage training pipeline is time-consuming and resource-consuming, especially for contrastive learning which requires a higher training cost. What is worse, it is inapplicable to simultaneously load the student model as well as the teacher model when memory resource is limited. Recently, Shi et al. (2022) attempt to train small self-supervised models without distillation signals. They eventually improve the baselines by a considerable margin, but the performance gap still exists. Moreover, a pretrained large model is still required to guild the lightweight model to a better initialization.

In this work, **we endeavor to enhance the performance of lightweight contrastive models without resorting to a pretrained powerful teacher model**. To begin with, we ask two straightforward questions that have not been answered yet: (1) Besides the canonical contrastive methods, are other line of self-supervised methods (e.g., only-positive-view methods, clustering-based methods) also suffer from the performance degeneration with lightweight models? (2) Is the degenerative performance of lightweight self-supervised models simply caused by suboptimal choice of hyperparameters? To answer the former question, we benchmark a few representative self-supervised methods on ImageNet, and observe that the performance degradation phenomenon exists in all of these methods. This result suggests the necessity to accommodate self-supervised learning (SSL) for models with fewer parameters. For the latter, the answer is partially yes. We extensively explore the best recipe for lightweight contrastive models and largely boost their evaluation accuracy in downstream tasks. Among the explored training techniques, a larger learning rate and a wider MLP head contribute most to reduce the performance gap to large self-supervised models.

To finally close the gap, we further make an investigation on the representation space of the small models. The empirical result is in accordance with the observation from Shi et al. (2022), that smaller models hold weaker intra-class alignment compared to larger models (See the detail in (Shi et al., 2022)). In other words, different instance of images are aggressively allocated into separate clusters, regardless of their semantic similarities. We hypothesize that it is caused by susceptibility to noisy views (i.e., false positive views and false negative views) during self-supervised training for tiny models (Kalantidis et al., 2020; Chen et al., 2021; Huynh et al., 2022; Chuang et al., 2022). To alleviate this problem, we propose SoftNCE, a novel loss generalizing the widely used InfoNCE loss to a smoothed one. Instead of aligning with a one-hot distribution in InfoNCE, our loss boosts the instance discrimination objective to match a softened probability distribution, by assigning some weight to similar instances. The proposed smoothing technique prevents models from over-confidence, thus is more robust to noisy labels and generalize better.

Our main contributions are summarized as below:

- We benchmark some representative contrastive learning methods with ResNet-18 on ImageNet, and observe that the inferior performance is not rescued by other line of paradigm like only-positive-view methods or clustering-based methods.
- We empirically show that the default training setting is far from optimal for lightweight models in contrastive learning, which is inappropriately adopted as a weak baseline in previous research (Fang et al., 2020; Gao et al., 2021; Xu et al., 2021; Zheng et al., 2022). We thus provide a better recipe for training lightweight contrastive models.
- We propose a novel loss, SoftNCE to mitigate the side effect from noisy views. We eventually show the feasibility to train lightweight contrastive models without the help from large models. For example, the linear evaluation accuracy of EfficientNet-B0 is improved from 42.2% to 65.8%, which is pretty close to ResNet-50 (67.4%), with only 16.3% of the number of parameters. Our method even yields better transferability than distillation counterpart SEED (Fang et al., 2020).

2 Related Work

2.1 Self-supervised Representation Learning

There is a surge in research on self-supervised contrastive learning in recent years. Dosovitskiy et al. (2014) first introduce the instance-level discrimination pretext task into self-supervised learning. It encourages a model to learn invariance to augmented views from an instance. Following research based on this pretext task continuously pushes the limit of self-supervised learning, even surpasses the supervised counterpart in downstream tasks (He et al., 2020; Chen et al., 2020c;a;b). These methods all require a large amount of negative pairs. To relieve the burden, algorithms are designed

Table 1: 20-NN and linear probe benchmarking results of ResNet-18 on ImageNet-1K with 512-batch self-supervised training. All the self-supervised learning methods are trained for 200 epochs.

Method	ResNet-50		ResNet-18	
	20-NN	Top-1	20-NN	Top-1
MoCo v2 (Chen et al., 2020c)	-	67.5	44.6	52.5
SimCLR v2 (Chen et al., 2020b)	-	71.7	43.8	51.9
SimSiam (Chen et al., 2020a)	-	70.0	23.0	32.7
BYOL (Grill et al., 2020)	59.2	69.3	44.8	52.6
PCL v2 (Li et al., 2021)	-	67.6	44.2	52.1
MSF (Koohpayegani et al., 2021)	64.9	72.4	22.1	29.7
Supervised	-	*76.1*	-	*70.2*

with competitive performance even without negative views Grill et al. (2020); Chen & He (2021); Zbontar et al. (2021). Apart from them, another line of methods, namely clustering-based methods, learn visual representations through clustering semantically similar instances based on the embedding space of the models (Caron et al., 2018; Asano et al., 2019; Caron et al., 2020; Li et al., 2021; Koohpayegani et al., 2021).

2.2 Lightweight Contrastive Learning

Although self-supervised learning leveraging large models has gained a great success, it seems to lose the magic equipped with lightweight models. To address this problem, Abbasi Koohpayegani et al. (2020) and Fang et al. (2020) introduce a pretrained self-supervised large model to transfer representation knowledge to the small models, and greatly boost their performance. Gao et al. (2021) further promote it by letting the student model perform ℓ_2 regression on the teacher's embedding space after the linear projector. Xu et al. (2021) fully exploit the representation knowledge by packing related instances together according to similarities calculated by the teacher. These methods share the same two-stage training procedure. In contrast, Choi et al. (2021) design one-stage online distillation for self-supervised training, and Shi et al. (2022) move forward to relax the need for distillation. However, they either bear a heavy resource burden by co-training two models concurrently, or show a lag in performance while still require a teacher for better initialization. In this work, we completely get rid of any help from a larger model, whilst exhibiting a competitive performance.

2.3 Contrastive Learning with Noisy Labels

Contrastive learning treats augmented views as positive pairs and any views from other instances as negative views. Although effective, this treatment can be unreliable. The augmented views can be semantically unrelated or distorted ones, and other instances may contain related views (Chuang et al., 2020). Recent research sees quality benefits for representation from resisting this noise. Kalantidis et al. (2020) draw inspiration from data mixing technique to mix hard negative samples in the embedding space. Chuang et al. (2022) modify the InfoNCE to a symmetric loss, demonstrating that the symmetric form is more robust to noisy labels. Chen et al. (2021) and Huynh et al. (2022) seek to identify false negatives, and then either recognize them as positives or directly remove them. In our work, we propose a simple and effective strategy, to soften the one-hot distribution in InfoNCE objective according to the hardness of negative samples. Our method is supplementary to that of Kalantidis et al. (2020)'s and Chuang et al. (2022)'s, while transcending methods proposed by Chen et al. (2021) and Huynh et al. (2022).

3 Establishing a Stronger Baseline

In this section, we ask two questions: (1) Is the much lower performance of lightweight models already rescued by existing self-supervised algorithms, just that we do not know? (2) Is the problem simply the result of uncurated hyperparameters? We answer the questions as follows.

Table 2: Extensive hyperparameters searching results on ResNet-18. We report 20-NN evaluation accuracy on ImageNet after 100-epoch self-supervised training. We perform hyperparameters tuning additively. "Mom.": momentum, "Cos.": cosine, "Sym.": symmetric.

Setting	Wide MLP	τ	LR	Warmup	Mom.	Cos. Mom.	Sym. Loss	**20-NN**	Δ
Base	×	0.2	0.06	×	0.999	×	×	42.9	-
+ Wide MLP	✓	0.2	0.06	×	0.999	×	×	43.1	+0.2
+ Lower τ	✓	**0.1**	0.06	×	0.999	×	×	43.7	**+0.6**
	✓	0.07	0.06	×	0.999	×	×	42.9	-0.2
+ Larger LR	✓	0.1	**0.12**	✓	0.999	×	×	44.7	**+1.0**
	✓	0.1	0.2	✓	0.999	×	×	43.3	-0.4
+ Lower Mom.	✓	0.1	0.12	✓	0.998	×	×	44.7	+0
	✓	0.1	0.12	✓	0.996	×	×	44.7	+0
	✓	0.1	0.12	✓	**0.993**	×	×	45.2	+0.5
	✓	0.1	0.12	✓	**0.99**	×	×	45.2	**+0.5**
+ Cos. Mom.	✓	0.1	0.12	✓	0.99	✓	×	45.3	+0.1
+ Sym. Loss	✓	0.1	0.12	✓	0.99	✓	✓	**46.2**	**+0.9**

3.1 Benchmarking Representative Contrastive Methods

To answer the first question, since most of the related research does not report the results with lightweight models, we fill the gap to benchmark contrastive learning methods with ResNet-18 on ImageNet. However, considering the exploding development of self-supervised learning, it is infeasible to test them all. We thus select a few representative methods mainly categorized as follows: *canonical contrastive methods* (MoCo v2 (Chen et al., 2020c), SimCLR v2 (Chen et al., 2020b)), *only-positive-view methods* (BYOL (Grill et al., 2020), SimSiam (Chen et al., 2020a)), and *clustering-based methods* (PCL v2 (Li et al., 2021), MSF (Koohpayegani et al., 2021)). We benchmark all the methods using their official codes with default hyperparameters, except for scaling the learning rate linearly with batch size.

The results are shown in Table 1, suggesting that the performance degeneration problem is ubiquitous in SSL methods, and is not rescued by any of the tested methods. Among them, SimSiam and MSF get worse accuracy. We hypothesize it is due to the instability of training lightweight SSL model.

3.2 A Better Recipe for Training Lightweight SSL method

In this part, we comprehensively search for the best training setting on lightweight models based on MoCo v2. We first tune the hyperparameters based on ResNet-18, and then verify them on EfficientNet-B0. Following Fang et al. (2020), we set the weight decay as 0.0001 for ResNet and 0.00001 for EfficientNet. The results are shown in Table 2 and Table 3 respectively. We mainly consider the following settings:

Wider MLP We follow Shi et al. (2022)'s suggestion to adopt a wider MLP. Specifically, We set the dimension of the two-layer MLP projector to [2048,128]. On ResNet-18, we see minor improvement, but on EfficientNet-B0, the accuracy increase is drastic. This is due to the fact that previous work only add one-layer projection head on EfficientNet-B0 (Fang et al., 2020).

Lower τ A τ closer to 0 results in larger gradients, and may enable faster training (Zhang et al., 2021). We test different values of τ, 0.2, 0.1 and 0.07 on ResNet-18. The selection of value 0.1 brings 0.6% and 1.5% absolute accuracy increase to ResNet-18 and EfficientNet-B0 separately.

Larger learning rate (LR) According to Shi et al. (2022), small models tend to have slow convergence in SSL. An immediate thought is to enlarge the learning rate. However, we observe the training instability phenomenon when directly increasing the LR. The problem is addressed after introducing warmup technique. We linearly increase the LR from 0 to the setting value for 5 epochs.

Table 3: Results of additive study on EfficientNet-B0. We report linear probe top-1 accuracy on ImageNet after 200-epoch self-supervised training. “Mom.”: momentum, “Sym.”: symmetric. The final result shows an absolute accuracy increase of 17.9% in total.

Setting	MLP	LR	Warmup	τ	Mom.	Sym. Loss	Top-1	Δ
Base	×	0.06	×	0.2	×	×	42.2	-
+ MLP	✓	0.06	×	0.2	0.999	×	53.3	+11.1
+ Lower τ	✓	0.06	×	0.1	0.999	×	54.8	+1.5
+ Larger LR	✓	0.12	✓	0.1	0.999	×	58.4	+3.6
+ Mom.	✓	0.12	✓	0.1	0.99	×	57.8	-0.6
+ Symmetric Loss	✓	0.12	✓	0.1	0.99	✓	**60.1**	+2.3

Apart from faster convergence, a larger learning rate also takes a bias towards converging to a flatter minimum, which is the key to generalize (Lewkowycz et al., 2020; Li et al., 2019). In our experiments, the most significant improvement comes along this strategy.

Lower momentum for the moving average encoder MoCo updates the momentum encoder by: $\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}$, where θ_{k} are the parameters of the momentum encoder k, and θ_{q} are updated by back-propagation. A smaller m enables quicker evolution of k, and may lead to faster convergence. We also follow the good practice to incorporate cosine momentum strategy (Grill et al., 2020). While this strategy benefits ResNet-18, on EfficientNet-B0, it slightly decreases the accuracy.

Symmetric loss In self-supervised learning, symmetric loss has demonstrated improved performance by swapping two augmented views as key and query views respectively. Inspired by BYOL (Grill et al., 2020), we exchange the roles of the augmented views, and then pass them to encoder q and k separately to form a symmetric loss. This technique reuses the input images and further lifts the results.

To summarize, we systematically test a few factors that may influence the final results, and provide a better practice to train lightweight models based on MoCo v2 (See the last row in Table 2 and 3).

4 Towards Closing the Performance Gap

Our previous empirical study improves the performance of lightweight SSL models to some extent. In this section, we are aiming towards finally reach the performance of ResNet-50 trained by MoCo v2, which has 67.5% top-1 linear probe accuracy evaluated on ImageNet.

We first turn to inspect the representation space learned by lightweight models and large models. Wang & Isola (2020) demonstrate that a good representation should possess two qualities, well uniformity and well alignment. Wang & Liu (2021) supplement another factor, namely intra-class alignment, indicating that a model should cluster same-class instances. We conduct primary experiments to analyze these three factors. The conclusion is the same with that of Shi et al. (2022), that a lightweight model holds weaker intra-class alignment compared to large models.

We conjecture that it is due to the susceptibility to noisy views in contrastive learning. We visualize the augmented views and find the common existence of unwanted views, i.e., false positive views and false negative views. See the example in Figure 2. The original picture is describing a goose on the grass. While we are interested in the *goose* in the downstream task, the SSL framework treat another augmented view focusing on the background *grass* as the positive, whilst contrasting negative views including the *goose* falsely. We propose to smooth the one-hot labels in accordance with the model’s representation.

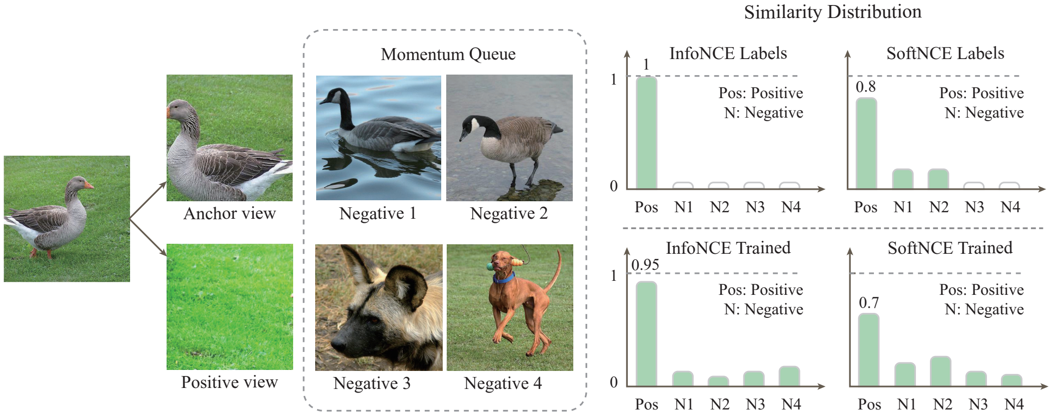

Figure 2: Illustration of noisy views and similarity distribution compared between InfoNCE and our SoftNCE. In this example, Two augmented views are generated from a given instance labelled as “goose”. The positive view is a false positive, and Negative 1 and Negative 2 are false negatives. Our SoftNCE smooths the hard labels of InfoNCE, by assigning some weight to K hardest negatives.

4.1 InfoNCE

Before describing the detail, we start by introducing the InfoNCE loss. Instance-wise contrastive learning commonly uses the InfoNCE loss as the contrastive loss function, which is defined as:

$$\mathcal{L}_i^{\text{InfoNCE}} = -\log \frac{\exp\left(z_i \cdot z_i'/\tau\right)}{\exp\left(z_i \cdot z_i'/\tau\right) + \sum_{n=0}^{N} \exp\left(z_i \cdot z_n'/\tau\right)}, \tag{1}$$

where z_i' is a positive embedding for an embedded sample z_i, and z_n' is one of the N negative embeddings from other samples. τ denotes a temperature hyperparameter. The number of N is large to approximate the distribution in the whole dataset. This is implemented either by a large batch size in SimCLR (Chen et al., 2020a) or a momentum queue in MoCo (He et al., 2020).

Mathematically, the training objective can be viewed as a cross entropy loss, by assigning probability 1 to align two augmented views for an instance, i.e., $z_i \cdot z_i'$, and probability 0 to all the other instances, i.e., $z_i \cdot z_n'$. We call the traditional InfoNCE loss as a “hard” one.

4.2 SoftNCE

For dealing with noisy labels, smoothed labels have already been demonstrated to be more robust compared with hard labels, and label smoothing is shown to be simple yet very competitive to other techniques (Szegedy et al., 2016; Song et al., 2022; Lukasik et al., 2020). Inspired by this fact, we attempt to transfer the success from supervised learning to self-supervised learning with simple modification. The definition of our proposed SoftNCE is given as follows:

$$\mathcal{L}_i^{\text{SoftNCE}} = -\alpha \log \frac{\exp\left(z_i \cdot z_i'/\tau\right)}{Z} - \sum_n \beta_n \log \frac{\exp\left(z_i \cdot z_n'/\tau\right)}{Z}, \tag{2}$$

where $\alpha + \sum_n \beta_n = 1$, and Z acts as a normalizing item, $Z = \exp\left(z_i \cdot z_i'/\tau\right) + \sum_n \exp\left(z_i \cdot z_n'/\tau\right)$. We will discuss different strategies of assigning values of α and β as follows.

Label Smoothing without Accommodation The original label smoothing technique is designed for supervised learning. Assume a cross-entropy loss between the true labels y_t and the network’s outputs p_t, as $L(y,p) = \sum_t -y_t log(p_t)$. Label smoothing technique assign $y_t = \alpha$ for the correct class and $y_t = (1-\alpha)/(T-1)$ for total T classes. In self-supervised learning, the discrimination task is performed in instance level, constituting a very large number of “classes”. In MoCo, the size of negatives N can be very large, e.g., 65536. directly dividing the number of class leads to a value diminishing to zero.

Label Smoothing with Hard Negative Mining Dwibedi et al. (2021); Koohpayegani et al. (2021) avoid directly attracting augmented views, instead, they utilize the momentum queue to sort the most similar images according to the model's embedding to be positive views. The success suggests that the model to some extent is able to distinguish semantic relation between instances. Motivated by this fact, we propose to only smooth the top-K nearest negative instances in the momentum queue. In this way, $\beta_n = 0$, if $z'_n \notin \mathrm{KNN}(z_i)$. But how about $z'_n \in \mathrm{KNN}(z_i)$? We come up with three different patterns, the simplest *average pattern*, the *exponential weight decay pattern*, and the *weight decay with range pattern*.

Average Pattern The average pattern denoted as Strategy 1 takes the simplest form similar to label smoothing:

$$\beta_k = \frac{1-\alpha}{K},\ z'_n \in \mathrm{KNN}(z_i).$$

Exponential Weight Decay Pattern The exponential weight decay pattern denoted as Strategy 2 takes the following definition:

$$\beta_k = \frac{1-\alpha}{2^k},\ z'_n \in \mathrm{KNN}(z_i),$$

where k is the ranking of embedding similarity.

Weight Decay with Range Pattern The weight decay with range pattern denoted as Strategy 3 is defined as:

$$\beta_k = \frac{1-\alpha}{3 \cdot D_j},\ z'_n \in \mathrm{KNN}(z_i),\ \sum_j D_j = K,$$

where D_j is the length of the range. Take an example, when $K = 10$, when k locates in a range (0,1], or (1,4], or (4,10], its corresponding β_k equals to $(1-\alpha)/3$, $(1-\alpha)/9$, $(1-\alpha)/18$, respectively. These ranges are set heuristically, but we do not perform any tuning on them. Just try, and work. We scale all the range accordingly with different K, e.g., (0,2], (2,8], (8,20] for $K = 20$. The principle behind the heuristics is to smoothly decay the weight for farther instances in the embedding space. Our experiments suggest that the weight decay with range pattern work out the best, so we adopt it as our default setting.

Static Smoothing vs. Incremental Smoothing For now, we have discussed on the value β. As for the α, things become simpler. We propose two mechanisms. The first is just using a static α, the second, which is more intuitive, is to increasingly smooth the labels, i.e., gradually decrease α with longer training, scheduled by a cosine function. The idea is that in the initial state, the model possesses no discriminative ability, but with longer training, the model learns something useful, so more smoothing may help to resist noisy labels. Intriguingly, the experiments suggest that static smoothing performs slightly better. We assume it is because the model already learns semantic information very early after the first few epochs. We adopt the static smoothing as the default setting. We suggest setting α as a value between 0.6 and 0.9.

4.3 Training Cost

Our SoftNCE loss introduces only a little extra training cost. We find a larger momentum queue works better, and set the queue size as 256K. It costs about 3% of the total memory. Another extra cost comes from the top-K sorting of the queue with O(N) complexity, costing less than 1% of the total computation. Indeed, we investigate the training time, but observe no significant difference in our experiments, all around 50 hours for 200-epoch training on ResNet-18 with an 8-card V100 server.

4.4 Hard Negative vs. False Negative

The InfoNCE loss is shown to have a hardness-aware property, that negative examples located closer to the anchor point in the embedding space contribute more gradients during training (Wang & Liu, 2021). That is to say, harder negatives are more useful in contrastive learning. However, if the negative samples are extremely hard, they otherwise hinder the final performance of the model. It

Table 4: Evaluation of lightweight models on ImageNet-1K.

Method	EfficientNet-B0		EfficientNet-B1		MobileNet-V3		ResNet-18	
	20-NN	Top-1	20-NN	Top-1	20-NN	Top-1	20-NN	Top-1
MoCo V2	-	42.2	-	50.7	-	36.3	44.6	52.5
Light-MoCo	**54.8**	**65.8**	**57.0**	**66.9**	**51.3**	**62.3**	**48.8**	**57.6**

Table 5: Object detection and instance segmentation results of ResNet-18.

Method	VOC Obj. Det.			COCO Obj. Det.			COCO Inst. Segm.		
	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
MoCo V2	46.1	74.5	48.6	35.0	53.9	37.7	31.0	51.1	33.1
SEED	46.1	74.8	49.1	35.3	54.2	37.8	31.1	51.1	33.2
MoCo V2 (repr.)	50.5	77.3	54.4	35.1	54.0	37.7	31.1	51.0	33.3
Light-MoCo	**51.6**	**77.9**	**56.2**	**35.4**	54.0	37.8	**31.5**	**51.3**	**33.6**

is because that the model is able to gradually learn to capture semantic similarity during training, so the hardest negatives are likely to be semantically positive ones, i.e., false negatives. So there is a false&hard dilemma. Previous methods dealing with the hardest negatives by directly removing them or directly treat them as positives (Chen et al., 2021; Huynh et al., 2022). We argue the importance of the smoothing technique by finding a sweet point to benefit the model to learn from hard negatives, as well as cancel the side effect from false views. Our ablation studies on the smoothing factor also suggest the importance to take a softened weight.

5 Experiments

5.1 Settings

We utilize ResNet-18, EfficientNet-B0/B1, and MobileNet-V3-Large as our backbones. Based on MoCo-V2, we deploy the SGD optimizer with learning rate as 0.2 and weight decay as 1e-4 for ResNet-18 and 1e-5 for the rest of backbones. All the networks are pre-trained on 8 Nvidia-32G-V100 GPUs with a batch size of 512 for 200 epochs, with 5-epoch warm-up. The K in *SoftNCE* is 20. For the momentum queue, we set the queue size as 256K, $m = 0.99$ and temperature as 0.1. We utilize a two-layer MLP projector with dimension [2048,128]. We follow the data augmentation strategy of MoCo v2.

5.2 Linear and KNN Evaluation on ImageNet.

Following SimSiam (Chen et al., 2020a), we finetune the pretrained networks for 90 epochs with a batch size of 4096 on ImageNet-1K. We utilize LARS as an optimizer with initial learning rate 0.1.

Results are reported in Table 4. In terms of Top-1 accuracy, Light-MoCo outperforms MoCo V2 by an average of 17.72% in accuracy. The application of SoftNCE finally close the performance gap between lightweight models and large models like ResNet-50. Our SoftNCE brings an improvement with 3.5% in accuracy.

5.3 Transferring to Detection and Segmentation.

Following the previous work Fang et al. (2020), we explore the effectiveness of Light-MoCo on two downstream tasks, object detection on VOC (Everingham et al., 2010) and instance segmentation on COCO(Lin et al., 2014). As shown in Table 5, Light-MoCo performs competitively on COCO and outperforms both the original reported MoCo V2 and the results reproduced by us in the settings mentioned above.

Table 6: Ablation study of the strategy, smoothing value α and K in SoftNCE. We report top-1 linear probe accuracy on ImageNet.

Strategy	Stra. 3	Stra. 3	Stra. 3	Stra. 3	Stra. 3	Stra. 3	Stra. 1	Stra. 2
K	5	10	20	30	20	20	20	20
α	0.8	0.8	0.8	0.8	0.95	0.6	0.8	0.8
Top-1	56.5	56.6	**57.6**	57.2	55.6	55.0	56.5	56.7

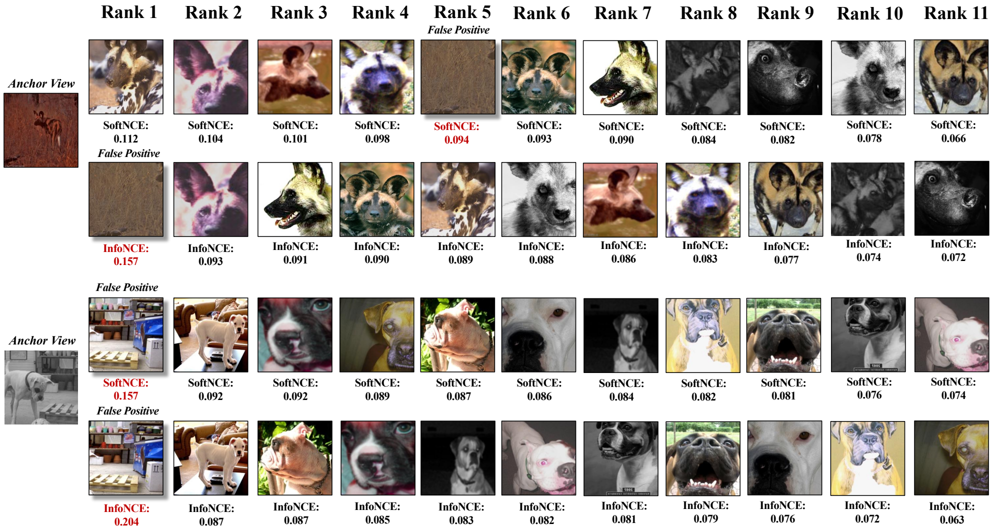

Figure 3: Qualitative comparison between SoftNCE and InfoNCE with ResNet-18 as the backbones.

5.4 Ablation Study

We make ablation studies on the performance of our proposed SoftNCE. The results are shown in Table 6. The results suggest the importance of choosing a proper smoothing value. When α is close to 1, our Light-MoCo degenerates to MoCo v2, without using any smoothing term. When $\alpha = 0.95$, the performance decreases from 57.6% to 55.6%. We also examine our SoftNCE on bigger model ResNet-50. Our primary experiments indicate the same trend. We eval ResNet-50 trained for 150 epochs. The 20-NN accuracy is 59.8% (SoftNCE trained) and 56.1% (MoCo v2 baseline).

6 Limitation

Although Light-MoCo closes the performance gap between cumbersome and lightweight contrastive models, limitations remain in this paper. First, we only adapt our approach on MoCo-v2 and it is still worthwhile to explore how it can be applied with other methods. Second, Figure 1 demonstrates that performance of ResNet-18 is not noticeably improved than the others. We leave these two limitations to future work.

7 Conclusion

Despite its thriving development, the performance of self-supervised learning suffers a dramatic fall as the size of model declines. In this paper, we demonstrate the prevalence of this phenomenon across all lines of the contrastive learning paradigm. Besides, for the lightweight models in contrastive learning, we empirically show how inadequate the existing recipe is and provide a better one. Based on these two issues, we propose SoftNCE, a self-supervised contrastive learning loss mitigating the side effect from noisy views. Extensive experiments show the effectiveness in bridging gap between lightweight and cumbersome models.

References

Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Compress: Self-supervised learning by compressing representations. *Advances in Neural Information Processing Systems*, 33:12980–12992, 2020.

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 132–149, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020b.

Tsai-Shien Chen, Wei-Chih Hung, Hung-Yu Tseng, Shao-Yi Chien, and Ming-Hsuan Yang. Incremental false negative detection for contrastive learning. In *International Conference on Learning Representations*, 2021.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Hee Min Choi, Hyoa Kang, and Dokwan Oh. Unsupervised representation transfer for small networks: I believe i can distill on-the-fly. *Advances in Neural Information Processing Systems*, 34: 24645–24658, 2021.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Ching-Yao Chuang, R Devon Hjelm, Xin Wang, Vibhav Vineet, Neel Joshi, Antonio Torralba, Stefanie Jegelka, and Yale Song. Robust contrastive learning against noisy views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16670–16681, 2022.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. *Advances in neural information processing systems*, 27, 2014.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9588–9597, 2021.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010.

Zhiyuan Fang, Jianfeng Wang, Lijuan Wang, Lei Zhang, Yezhou Yang, and Zicheng Liu. Seed: Self-supervised distillation for visual representation. In *International Conference on Learning Representations*, 2020.

Yuting Gao, Jia-Xin Zhuang, Ke Li, Hao Cheng, Xiaowei Guo, Feiyue Huang, Rongrong Ji, and Xing Sun. Disco: Remedy self-supervised learning on lightweight models with distilled contrastive learning. *arXiv preprint arXiv:2104.09124*, 2021.

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.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Tri Huynh, Simon Kornblith, Matthew R Walter, Michael Maire, and Maryam Khademi. Boosting contrastive self-supervised learning with false negative cancellation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2785–2795, 2022.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in Neural Information Processing Systems*, 33:21798–21809, 2020.

Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Mean shift for self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10326–10335, 2021.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.02218*, 2020.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.

Yuanzhi Li, Colin Wei, and Tengyu Ma. Towards explaining the regularization effect of initial large learning rate in training neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pp. 6448–6458. PMLR, 2020.

Haizhou Shi, Youcai Zhang, Siliang Tang, Wenjie Zhu, Yaqian Li, Yandong Guo, and Yueting Zhuang. On the efficacy of small self-supervised contrastive models without distillation signals. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 2225–2234, 2022.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2495–2504, 2021.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Haohang Xu, Jiemin Fang, Xiaopeng Zhang, Lingxi Xie, Xinggang Wang, Wenrui Dai, Hongkai Xiong, and Qi Tian. Bag of instances aggregation boosts self-supervised distillation. In *International Conference on Learning Representations*, 2021.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pp. 12310–12320. PMLR, 2021.

Oliver Zhang, Mike Wu, Jasmine Bayrooti, and Noah Goodman. Temperature as uncertainty in contrastive learning. *arXiv preprint arXiv:2110.04403*, 2021.

Kai Zheng, Yuanjiang Wang, and Ye Yuan. Boosting contrastive learning with relation knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 3508–3516, 2022.

A APPENDIX