Neighborhood Contrastive Learning for Novel Class Discovery (Supplementary Material)

Zhun Zhong[1]*, Enrico Fini[1]*, Subhankar Roy[1,3], Zhiming Luo[2]†, Elisa Ricci[1,3], Nicu Sebe[1]
[1]University of Trento [2]Xiamen University [3]Fondazione Bruno Kessler

In this supplementary material, we provide more implementation detail, experimental results and discussion. In detail, we provide more dataset detail in Sec. A. Sec. B shows parameter analysis of the proposed Neighborhood Contrastive Learning (NCL) and Hard Negative Generation (HNG). We provide discussion on our method in Sec. C.

A. Dataset

CIFAR-10 [4] contains 50,000 training images from 10 classes, each of which has a size of 32×32. For the setting of novel class discovery, we split the samples of the first five classes (namely airplane, automobile, bird, cat and deer) as the labeled data and the remaining samples of the other five classes as the unlabeled data.

CIFAR-100 [4] has the same number of training images and the same image size as CIFAR-10. The difference is that CIFAR-100 is captured from 100 classes. For evaluation, we regard the samples of the first 80 classes as the labeled data and the remaining samples of the other 20 classes as the unlabeled data.

ImageNet [1] is a large-scale dataset, including 1.28 million training images from 1,000 classes. Following [2, 3], we divide the training data into two splits that are composed of images from 882 classes and 118 classes, respectively. The split with 882 classes is regarded as the labeled data. For the unlabeled data, we randomly sample three subsets from another split with 118 classes. Each subset consists about 30,000 images from 30 classes and is used as an unlabeled data.

B. Parameter Analysis

B.1. Parameter Analysis of NCL

We investigate four important parameters in neighborhood contrastive learning (NCL), *i.e.*, memory size $|M|$, temperature of contrastive learning τ, number of pseudo-positives k_1, and weight of augmented-positive α. For evaluation, we vary one parameter at a time while the other three are set to their default values. Results on CIFAR-10 and CIFAR-100 are shown in Fig. 7.

(1) Sensitivity to memory size. In Fig. 7(a), we vary the memory size $|M|$ in the range $[500, 20000]$. The ACC first increases with the memory size and achieves the best results when memory size is between 2,000 and 10,000. Considering the efficiency, we set memory size to 2,000, which achieves reasonably good performance on both datasets with limited computational cost (computing the similarities between mini-batch samples and memory samples).

(2) Sensitivity to temperature. We vary the temperature τ in the range of [0.01, 0.5] and show the results in Fig. 7(b). We can observe that results are similar when temperature is between 0.02 and 0.1, indicating our NCL is robust to temperature within certain ranges. The best results are obtained when temperature is around 0.05.

(3) Sensitivity to number of pseudo-positives. Since the number of novel classes (C^u) is different in CIFAR-10 and CIFAR-100, we vary the number of pseudo-positives k_1 in different ranges for these two datasets. The range of k_1 is [10, 360] for CIFAR-10 and is [1, 80] for CIFAR-100, respectively. Results are shown in Fig. 7(c). Selecting too few KNNs will ignore most of the positive samples and regard them as negative samples, resulting in worse performance. On the other hand, assigning too many KNNs will include more negative samples. Enforcing a sample to approach too many negative samples will suppress the benefit of true positive samples in the KNNs and will undoubtedly hamper the ACC. Interesting, we find that our NCL achieves consistent good performance when the number of KNNs is equal to the half of $|M|/C^u$ (*i.e.*, 200 for CIFAR-10 and 50 for CIFAR-100).

(4) Sensitivity to weight of augmented-positive. As shown in Fig .7(d), both datasets achieves best results when the weight of augmented-positive α is around 0.2. The ACC will be largely reduced when $\alpha \geq 0.35$ for CIFAR-10. For CIFAR-100, when $\alpha \leq 0.05$, the ACC are clearly lower than those of $\alpha \geq 0.1$.

Based on the above analyses, we set the memory size $|M|$=2,000, temperature τ=0.05, number of pseudo-positives k_1=$|M|/C^u/2$, and weight of augmented-positive α= 0.2 for all datasets in default.

*Equal contribution
†Corresponding author

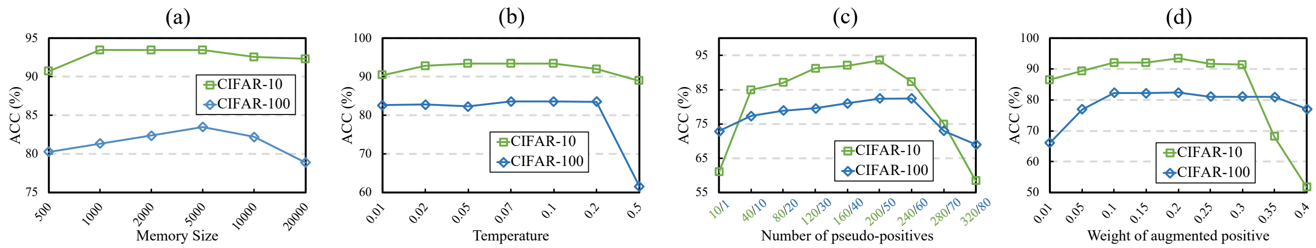

Figure 7. Parameter analysis of the proposed neighborhood contrastive learning on CIFAR-10 and CIFAR-100. Sensitivities to (a) memory size, (b) temperature of contrastive learning, (c) number of pseudo-positives, and (d) weight of augmented-positive.

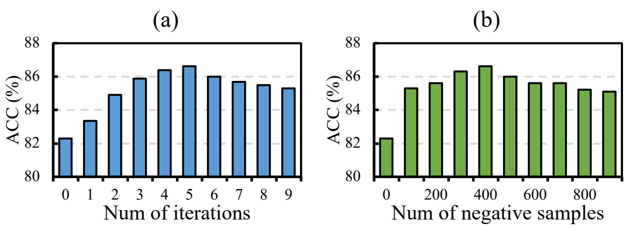

Figure 8. Parameter analysis of the proposed hard generative generation (HNG) on CIFAR-100. Sensitivities to (a) number of iterations N, and (b) number of negative samples k_2.

B.2. Parameter Analysis of HNG

We evaluate two parameters for hard negative generation (HNG), *i.e.*, number of iterations N and number of negative samples k_2. For evaluation, we vary one parameter and fix the other one to its default value. Results on CIFAR-100 are shown in Fig. 8. When $N = 0$ (in Fig. 8(a)) or $k_2 = 0$ (in Fig. 8(b)), the model reduces to the baseline trained only with NCL that does not consider the hard negative samples. As shown in Fig. 8, all values of N and k_2 achieve higher ACC than the model trained only with NCL, demonstrating the effectiveness of the proposed HNG. The ACC first increases with N / k_2 and achieves best results when $N \approx 5$ / $k_2 \approx 400$. Performing the HNG with too many iterations or selecting too many negative samples does not lead to further improvement. Considering the above factors, we set $N = 5$ and $k_2 = 400$ for all datasets in default.

C. Discussion of Our Method

C.1. Different Impact on CIFAR-10 and CIFAR-100

From Table 6, we find that removing pseudo-positives (NCL w/o PP) and adding hard negative generation (NCL + HNG) have inconsistent effects on performance between CIFAR-10 and CIFAR-100. We conjecture that this phenomenon is caused by the difference of number of labeled classes C^l and number of unlabeled classes C^u. 1) We have C^u=5 for CIFAR-10 and C^u=20 for CIFAR-100. NCL w/o PP will regard much more positives as negatives in the memory for CIFAR-10 than CIFAR-100, and thus the performance of CIFAR-10 will be degraded more than of CIFAR-100. 2) We have C^l=5 for CIFAR-10 and C^l=80 for CIFAR-100. CIFAR-10 contains a small C^l. In this context, mixing between labeled and unlabeled samples cannot generate diverse hard negative samples and thus fails to facilitate contrastive learning.

Method	CIFAR-10	CIFAR-100
Baseline	87.9±0.7%	69.4±1.4%
NCL	93.4±0.2% (↑ 5.5%)	82.3±2.6% (↑ 12.9%)
NCL w/o PP	61.8±7.6% (↓ 26.1%)	68.5±1.9% (↓ 0.9%)
NCL + HNG	93.4±0.1% (↑ 5.5%)	86.6±0.4% (↑ 17.2%)

Table 6. Evaluation of the effectiveness of the proposed neighborhood contrastive learning (NCL) and hard negative generation (HNG). **NCL w/o PP**: NCL without pseudo-positives.

C.2. Positive Selection for BCE and NCL

In our method, we use different strategies to select positives for BCE and NCL. The decision is mainly dependent on the number of samples in the batch and memory. 1) The number of samples in batch is much smaller than the size of the memory bank, so the class-balance cannot be ensured. When there are few or no samples of class-i in the batch, we may select overmuch false-positives for a sample of class-i by top-k, which will hamper the performance. 2) Using the memory with larger size, the class distribution will be close to uniform and each class has roughly equal number of positives. Thus, selecting top-k is more suitable for NCL, which potentially leverages class-balance property and helps a robust training. In our experiment, we find that using top-k for BCE and threshold for NCL lead to lower results.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 1

[2] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. In *Proc. ICLR*, 2018. 1

[3] Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. Multi-class classification without multi-class labels. In *Proc. ICLR*, 2019. 1

[4] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *University of Tronto*, 2009. 1