Neighborhood Contrastive Learning for Novel Class Discovery

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

Abstract

*In this paper, we address Novel Class Discovery (NCD), the task of unveiling new classes in a set of unlabeled samples given a labeled dataset with known classes. We exploit the peculiarities of NCD to build a new framework, named Neighborhood Contrastive Learning (NCL), to learn discriminative representations that are important to clustering performance. Our contribution is twofold. First, we find that a feature extractor trained on the labeled set generates representations in which a generic query sample and its neighbors are likely to share the same class. We exploit this observation to retrieve and aggregate pseudo-positive pairs with contrastive learning, thus encouraging the model to learn more discriminative representations. Second, we notice that most of the instances are easily discriminated by the network, contributing less to the contrastive loss. To overcome this issue, we propose to generate hard negatives by mixing labeled and unlabeled samples in the feature space. We experimentally demonstrate that these two ingredients significantly contribute to clustering performance and lead our model to outperform state-of-the-art methods by a large margin (*e.g.*, clustering accuracy +13% on CIFAR-100 and +8% on ImageNet).*

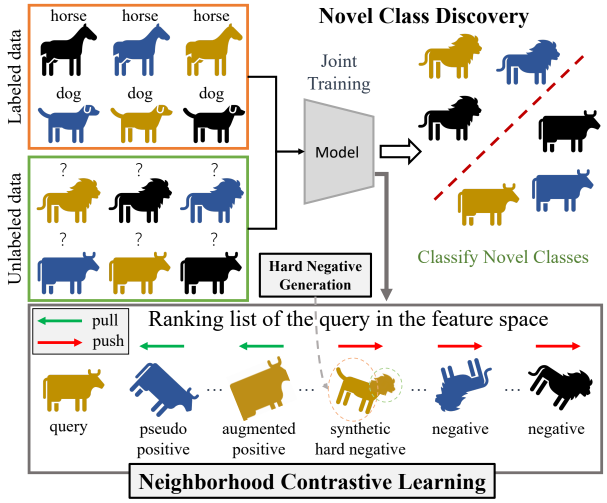

Figure 1. Illustration of novel class discovery (NCD) and the proposed neighborhood contrastive learning (NCL). In NCD, we are given two datasets, a labeled one and an unlabeled one, with disjoint class sets. NCD aims to leverage all data to learn a model that can cluster the unlabeled data. NCL tries to learn discriminative representations by enforcing a query to be close to its correlated view (augmented-positive) and its pseudo-positives (neighbors), as well as to be far from the negatives. We also generate hard negatives by mixing between labeled and unlabeled features, which can further facilitate our NCL.

1. Introduction

Learning from labeled data has been a widely studied topic in the field of machine learning, and more recently in deep learning [15, 21, 26]. Despite tremendous success, supervised learning techniques largely rely on the availability of massive amounts of annotated data [8]. To get rid of the difficulty and expensive cost of annotating, the machine learning community has shifted the attention to techniques that can learn with limited or completely non-annotated data. To this end, many semi-supervised [5, 35] and unsupervised learning [4, 7, 14, 32] methods have been proposed, which achieve promising results compared to supervised learning methods. Nonetheless, not much effort has been made to exploit prior knowledge from existing labeled datasets and use it to discover unknown classes that are not present in the labeled data.

In this paper, we address one such relevant problem, called Novel Class Discovery (NCD) [12, 13], where we are given a labeled dataset and an unlabeled dataset, differing in class label space. The goal of NCD is to learn a model that can cluster the unlabeled data by exploiting the latent commonalities from the labeled data (see the top half of Fig. 1). Importantly, the availability of labeled data does not guarantee transferability because the patterns learned from the labeled data with *off-the-shelf* models might not be useful for the unlabeled data. This poses NCD apart from semi-supervised learning paradigm, where the label space is shared between labeled and unlabeled data, and also makes it more challenging. The NCD task finds relevance in many real-world scenarios where the volume of unlabeled data

*Equal contribution
†Corresponding author

keeps growing (*e.g.*, multimedia). It is desirable to leverage the existing annotated data (collected from known classes) to explore the new unlabeled data from novel classes, rather than in a completely unsupervised fashion from scratch.

With that goal in mind, this work proposes a holistic learning framework that uses contrastive loss [14, 27] formulation to learn discriminative features from both the labeled and unlabelled data, which is absent in most NCD methods [12, 13, 16, 17]. Subsequently, we introduce two key ideas in the paper. The first idea is to exploit the fact that the local neighborhood of a sample (*query*) in the embedding space will contain samples which most likely belong to the same semantic category of the query, and can be considered as *pseudo-positives*. Note that this is specific to the NCD setting, where we can pre-train a feature extractor with supervision. We exploit this observation in the context of contrastive learning to bring the query closer to its pseudo-positives, which is termed as **N**eighborhood **C**ontrastive **L**earning (NCL) (see the bottom half of Fig. 1). These numerous positives allow us to obtain a much stronger learning signal when compared to the traditional contrastive formulation realized with only two views [7, 14]. Our second idea is to address the better selection of *negatives* to further improve the contrastive learning. Peculiar to the NCD task where we are given labeled samples of the known classes (a.k.a *true* negatives of any unlabeled instance), we exploit them, together with the unlabeled samples, to generate synthetic samples in the feature space using a mixing strategy and treat them as *hard* negatives (see Fig. 1). This circumvents the problem of falsely treating the true positives as negatives [14, 18]. We call this process as **H**ard **N**egative **G**eneration (HNG), which is effective and can produce a boost in performance when employed together with NCL.

To summarize, our contributions are threefold:

- We propose Neighborhood Contrastive Learning (NCL) for NCD, which exploits the local neighborhood in the embedding space of a given query. Our NCL recruits more positive samples for the contrastive loss formulation, significantly improving the clustering accuracy.

- We propose to aid the contrastive learning by leveraging the labeled samples to generate hard negative samples through feature mixing. With labeled data from various classes, the proposed Hard Negative Generation (HNG) can obtain consistent improvement.

- Extensive experiments on three NCD benchmarks demonstrate the effectiveness of our method and show that we advance the state-of-the-art approaches by large margins (*e.g.*, clustering accuracy +13% on CIFAR-100 and +8% on ImageNet).

2. Related Work

Novel Class Discovery is a relatively new task that aims to classify the samples in the unlabeled set into different semantic categories. It is different from unsupervised clustering in that one has a labeled set which has completely different classes from the unlabeled set. Typical novel class discovery methods first train a model on the labeled data and use it as an initialization for performing unsupervised clustering on the unlabeled data. The works [16, 17] in this category utilize the labeled data to train a binary classification model by exploiting the pair-wise similarity of images and then use this trained binary classification model as a supervision for clustering on the unlabeled data. Similarly, [13] pretrains the model on the labeled data, followed by an end-to-end clustering technique [32] on the unlabeled data. Deviating from this two-stage training strategy, Han *et. al.* [12] propose to leverage labeled data while performing unsupervised clustering on the unlabeled data. Our proposed NCL also builds on the premise of leveraging labeled data in the unsupervised clustering phase. However, in contrast to [12], NCL uses labeled data not to maintain the basic discrimination of representation, but to aid the contrastive learning process by generating more informative negatives.

Unsupervised Clustering is the task to partition an unlabeled dataset into different semantic categories, where the prior knowledge of a labeled set is not available. To this end, many shallow [1, 23, 34] and deep learning based methods [6, 9, 24, 29, 32, 33, 36] have been proposed. The deep learning based methods can be roughly categorized into two kinds where the first kind exploits pairwise similarity of the samples to generate pseudo-labels for clustering [6, 12, 24]. Whereas, the second kind [29, 36] uses *neighborhood aggregation* of feature embedding to bring closer the similar instances and simultaneously pushing away the dissimilar instances, thereby achieving a clustering effect. Our method also draws inspiration from these two lines of works. Of notable interest to our work, [36] uses a non-deterministic *k-means* algorithm to find a local neighborhood within an iterative optimization process, which however is sensitive to initialization and also computationally expensive. Instead, this paper proposes to adopt an end-to-end clustering technique via the use of pairwise similarity of samples and directly explore neighborhood by k-nearest neighbors, which makes our method much simpler while still retaining the benefits of neighborhood aggregation.

Contrastive Learning is an unsupervised feature representation learning technique that has gained significant momentum in the recent years. The crux of contrastive learning based methods [3, 7, 11, 14, 27, 31] lies in computing a similarity between an input and its correlated view, instead of a fixed target (*e.g.*, one-hot label). Due to the close association between unsupervised learning and NCD, we adopt the contrastive loss [11] formulation to harness its power for

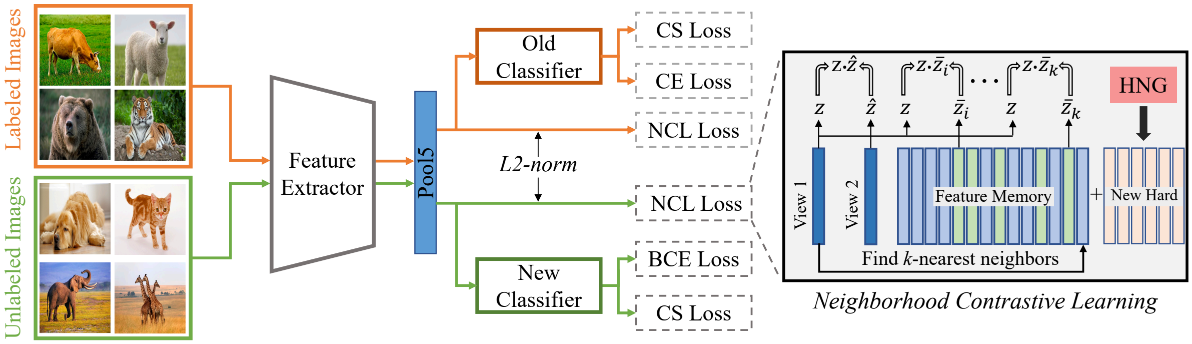

Figure 2. The proposed neighborhood contrastive learning framework for novel class discovery. Given training images sampled from the labeled and the unlabeled data, we forward them into the network to obtain corresponding representations. For the labeled data, the CE loss, CS loss and the proposed NCL loss are calculated with the ground-truth labels. For the unlabeled data, BCE loss and CS loss are computed to optimize the new classifier while the NCL loss is proposed to learn discriminative representation. **CE:** cross-entropy, **BCE:** binary cross-entropy, **CS:** consistency, **NCL:** neighborhood contrastive learning, **HNG:** hard negative generation.

learning discriminative representations. However, different from the above methods, the contrastive loss formulation in NCL exploits both the labeled data and the unlabeled data into one holistic framework, which is well suited for the NCD task. Moreover, in NCL we propose to amalgamate contrastive learning with neighborhood aggregation by considering k-nearest neighbors as pseudo-positives, making our formulation unique in the NCD literature.

Negative Mining plays a crucial role in contrastive learning because the success of the contrastive loss is pivoted on the presence of useful *negatives* [14]. Aside from maintaining a large batch size [7] or a large queue [14] for having ample useful negatives, one can draw inspirations from the semi-supervised learning literature and naturally consider using mixup strategy in the pixel space [35] or the latent space [30] to generate harder negatives [18, 25]. We, therefore, capitalize on the fact that the samples of the known classes in the labeled set are *true negatives* (being disjoint to the novel classes) and their mixing with the farthest features in a queue produces *synthetic* features which are considerably true negatives and harder than the farthest features for the query. Importantly, in NCD, due to the large population of positives in the queue, mixing of two random samples may lead to the generation of false negatives, which can indeed hurt the performance. Hence, our hard negative generation strategy (see Sec. 3.4) alleviates the drawbacks of [18, 25] and is tailor-made to NCD.

3. Method

Problem Definition. The task of Novel Class Discovery (NCD) assumes the availability of two datasets: a labeled dataset D^l and an unlabeled dataset D^u, containing C^l and C^u classes respectively. The sets of classes in the two datasets are disjoint, but some degree of similarity between the two is necessary. The goal of NCD is to cluster the data in D^u, leveraging the knowledge from D^l.

Overall Framework. To discover the latent classes in D^u, we learn a shared feature extractor $\Omega : x \mapsto z \in R^H$ and two linear classifiers ϕ^l and ϕ^u, with C^l and C^u output neurons respectively, each followed by a softmax layer. At each training step, a batch of images is sampled from both D^u and D^l. Using data augmentation we generate two correlated views of the same batch and forward them into the feature extractor. On the one hand, the features of the labeled images are fed to the classifier ϕ^l, which is optimized with the cross-entropy loss using the labels. On the other hand, using the binary cross-entropy loss, the classifier ϕ^u learns to infer the cluster assignments for the unlabeled images. Both classifiers are encouraged to output consistent predictions using the consistency loss. In addition, the representations z are refined by the proposed neighborhood contrastive loss (NCL) on both labeled and unlabeled samples. The overall framework is depicted in Fig. 2.

3.1. Baseline for Novel Class Discovery

For the baseline, we use a three stage learning pipeline similar to [12]. First, we learn a label-agnostic image representation by self-supervision learning [10] using both labeled and unlabeled datasets, which has been shown to be particularly good at extracting low-level features in the first layers of the network [2].

Subsequently, high-level features are learned using supervision on the labeled dataset. Given a sample and its label $(x, y) \in D^l$, we optimize the network using the *cross-entropy* loss:

$$\ell_{ce} = -\frac{1}{C^l}\sum_{i=1}^{C^l} y_i \log \phi_i^l\left(\Omega\left(x\right)\right). \tag{1}$$

Finally, we simplify the cluster discovery step in [12] by using the cosine similarity of the features to estimate pairwise pseudo-labels, instead of ranking statistics. We find this modification can yield similar performance with respect to ranking statistics when applied with our NCL, while being significantly more efficient and easier to implement. Specifically, given a pair of images (x_i^u, x_j^u) sampled from dataset D^u, we extract features (z_i^u, z_j^u) and compute their cosine similarity $\delta\left(z_i^u, z_j^u\right) = {z_i^u}^\top z_j^u / \|z_i^u\| \|z_j^u\|$. The pairwise pseudo-label is then assigned as follows:

$$\hat{y}_{i,j} = \mathbb{1}\left[\delta\left(z_i^u, z_j^u\right) \geq \lambda\right], \tag{2}$$

where λ is a threshold that represents the minimum similarity for two samples to be assigned to the same latent class. Then, the pairwise pseudo-label is compared to the inner product of the outputs of the unlabeled head $p_{i,j} = \phi^u\left(z_i^u\right)^\top \phi^u\left(z_j^u\right)$. The network is optimized using the *binary cross-entropy* loss:

$$\ell_{bce} = \hat{y}_{i,j}\log\left(p_{i,j}\right) + \left(1 - \hat{y}_{i,j}\right)\log(1 - p_{i,j}). \tag{3}$$

The last building block of our baseline is the consistency loss, which enforces the network produce similar predictions for an image x_i and its correlated view $\hat{x}_i$. This is particularly important for unlabeled examples. Nonetheless, we find that consistency helps with both labeled and unlabeled examples. We use *mean squared error*:

$$\begin{aligned}\ell_{mse} = \frac{1}{C^l}\sum_{i=1}^{C^l}\left(\phi_i^l\left(z^l\right) - \phi_i^l\left(\hat{z}^l\right)\right)^2 + \\ \frac{1}{C^u}\sum_{j=1}^{C^u}\left(\phi_j^u\left(z^u\right) - \phi_j^u\left(\hat{z}^u\right)\right)^2.\end{aligned} \tag{4}$$

The overall loss for our baseline reads as:

$$\ell_{base} = \ell_{ce} + \ell_{bce} + \omega\left(t\right)\ell_{mse}, \tag{5}$$

where the coefficient $\omega\left(t\right)$ is a ramp-up function as in [12].

3.2. Neighborhood Contrastive Learning

Given a set of stochastic image transforms, we generate two correlated views $(x^u, \hat{x}^u)$ of a generic unlabeled sample to be used as a positive pair. Subsequently, we apply the network Ω to extract $(z^u, \hat{z}^u)$ from the views. This operation is repeated for all the samples of a batch of length B. We also maintain a queue M^u of features stored from past training steps, which initially are regarded as negatives, denoted with $\bar{z}^u$. The contrastive loss for the positive pair can be written as:

$$\ell_{(z^u, \hat{z}^u)} = -\log\frac{e^{\delta(z^u, \hat{z}^u)/\tau}}{e^{\delta(z^u, \hat{z}^u)/\tau} + \sum_{m=1}^{|M^u|} e^{\delta(z^u, \bar{z}_m^u)/\tau}}, \tag{6}$$

where $\delta(\cdot, \cdot)$ is the cosine similarity and τ is a temperature parameter that controls the scale of distribution.

Unfortunately, a well-known drawback of the contrastive loss is that samples belonging to the same class could be treated as negatives, since we have no information about the labels. However, intuitively, the quality of the representations should benefit if the positive and negative pairs correspond to the desired latent classes. One way to mitigate this issue is to use the model itself to generate pseudo-positive pairs of samples, *i.e.*, to consider the *neighbors* of the representation z^u as instances of the same class. The selection of sensible pseudo-positive pairs turns out to be a hard task, especially at the beginning of the training, when the quality of the representations is poor. However, in NCD, we can leverage the labeled dataset D^l to bootstrap the representations, and then use them to infer the relationships between unlabeled examples in D^u.

More formally, given a network Ω pretrained as the first two steps described in Sec. 3.1, we can retrieve the top-k most similar features from the queue for a query z^u:

$$\rho_k = \underset{\bar{z}_i^u}{\operatorname{argtop}_k}\left(\left\{\delta\left(z^u, \bar{z}_i^u\right) \mid \forall i \in \{1, \ldots, |M^u|\}\right\}\right). \tag{7}$$

Assuming the examples in ρ_k are false-negatives (*i.e.*, they belong to the same class as z^u), we can regard them as pseudo-positives and write their contributions in the contrastive loss as follows:

$$\ell_{(z^u, \rho_k)} = -\frac{1}{k}\sum_{\bar{z}_i^u \in \rho_k}\log\frac{e^{\delta(z^u, \bar{z}_i^u)/\tau}}{e^{\delta(z^u, \hat{z}^u)/\tau} + \sum_{m=1}^{|M^u|} e^{\delta(z^u, \bar{z}_m^u)/\tau}}. \tag{8}$$

Finally we can introduce our *Neighborhood Contrastive loss* as follows:

$$\ell_{ncl} = \alpha\ell_{(z^u, \hat{z}^u)} + (1 - \alpha)\,\ell_{(z^u, \rho_k)}, \tag{9}$$

where α controls the weight of the two components.

3.3. Supervised Contrastive Learning

In the case of the labeled dataset D^l, instead of using the network to mine the pseudo-positives, we can directly use the ground-truth labels to retrieve the set of positives from the queue of labeled set M^l for a sample x_i^l with features z_i^l:

$$\rho = \left\{\bar{z}_j^l \in M^l : y_i = y_j\right\} \cup \hat{z}_i^l. \tag{10}$$

Note that ρ contains both the features $\hat{z}_i^l$ of the correlated view $\hat{x}_i^l$ and the other samples belonging to the same class. Using this supervision, our Neighborhood Contrastive loss can be reduced to the *Supervised Contrastive loss* [19]:

$$\ell_{scl} = -\frac{1}{|\rho|}\sum_{\hat{z}_j^l \in \rho}\log\frac{e^{\delta\left(z_i^l, \hat{z}_j^l\right)/\tau}}{e^{\delta\left(z_i^l, \hat{z}_i^l\right)/\tau} + \sum_{m=1}^{|M^l|} e^{\delta\left(z_i^l, \bar{z}_m^l\right)/\tau}}. \tag{11}$$

3.4. Hard Negative Generation

He et al. [14] show the importance of having a large memory that covers a rich set of negative samples for contrastive learning. Recently, other studies [18, 25] find that most of the negatives have very low similarities with the query sample. We experimentally verify that this behavior is also present when contrastive learning is used in the context of Novel Class Discovery (NCD). Specifically, as detailed in Sec. 4.2, we demonstrate that removing the easiest negatives from the queue does not impact performance, indicating that such negative samples contribute less during training. This is not desirable, because we are wasting memory and computation. On the other hand, selecting hard negatives automatically can be difficult since we have no information about the latent classes in the unlabeled set, and therefore we could end up selecting positive samples. However, in NCD we assume that the set of classes in the labeled and unlabeled sets are disjoint. This entails that all the samples from one set are negatives for the samples of the other set, and vice versa. Inspired by the advancements in regularization techniques using image / feature mixtures [35, 30], we use this notion to generate hard negatives by mixing labeled and unlabeled samples.

Given a view x^u of an image belonging to the unlabeled set and its representation in the feature space z^u, we can select easy negatives by looking at the features with minimal similarity in the queue M^u:

$$\varepsilon_k = \underset{\bar{z}_i^u}{\mathrm{argtop}_k} \left(\{ -\delta \left(z^u, \bar{z}_i^u \right) \mid \forall i \in \{1, \dots, |M^u|\} \} \right). \tag{12}$$

Note the negative sign of the similarity. Since the network can confidently distinguish these samples from the query, we can safely assume that they are very likely to be true negatives, *i.e.* they do not belong to the same class as the query. Note that this is in contrast with the recent literature on hard negative mining [18, 25], which samples hard negatives, incurring in the problem of false-negatives.

Let us also consider a queue M^l containing labeled samples stored from past training steps. As mentioned above, these are by definition true negatives with respect to x^u. Our insight is that by linearly interpolating the examples in these two sets we can generate new, hopefully more informative negatives. In practice, for each $\bar{z}^u \in \varepsilon_k$ we randomly sample a feature $\bar{z}^l \in M^l$ and compute the following:

$$\zeta = \mu \cdot \bar{z}^u + (1 - \mu) \cdot \bar{z}^l, \tag{13}$$

where μ is the mixing coefficient. This process of cycling through ε_k is repeated N times such that the resulting set of mixed negatives η will contain $k \times N$ features. Then, the hardest negatives are filtered from η, using the cosine similarity as before:

$$\eta_k = \underset{\zeta_i}{\mathrm{argtop}_k} \left(\{ \delta \left(z^u, \zeta_i \right) \mid \forall i \in \{1, \dots, k \times N\} \} \right). \tag{14}$$

This results in a set η_k of hard negatives that have the following two properties: (i) they are most likely true negatives, (ii) it is hard for the network to distinguish them from the query. Finally the queue for x^u is derived by adding the newly generated mixed negatives into the queue M^u:

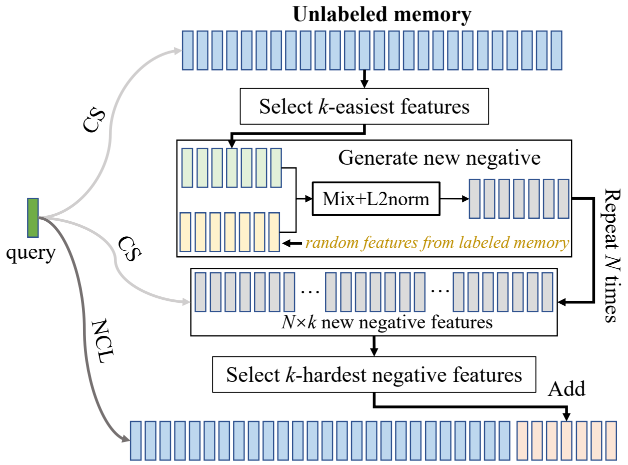
Unlabeled memory
Select k-easiest features
CS
Generate new negative
Mix+L2norm
random features from labeled memory
Repeat N times
query
CS
N×k new negative features
NCL
Select k-hardest negative features
Add

Figure 3. Illustration of hard negative generation (HNG). **CS**: compute similarity, **NCL**: neighborhood contrastive learning.

$$M^{u'} = M^u \cup \eta_k, \tag{15}$$

and the contrastive loss is computed as in Eq. 6 and Eq. 8, but replacing M^u with $M^{u'}$. Note that M^u is not overwritten in the process. This pipeline for hard negative generation (illustrated in Fig. 3) is repeated for each unlabeled sample in the current batch, as they will have different sets of easy negatives. To distinguish between the number of pseudo-positives used in Eq. 7 and number of negatives used in Eq. 12, we denote the former as k_1 and the latter as k_2 respectively.

3.5. Overall Loss

Considering the baseline model, neighborhood contrastive learning on unlabeled data, supervised contrastive learning on labeled data, and the hard negative generation on unlabeled data, the overall loss for our model is:

$$\ell_{all} = \ell_{base} + \ell_{ncl} + \ell_{scl}. \tag{16}$$

Throughout the paper, we refer to the ℓ_{ncl} and ℓ_{scl} collectively as neighborhood contrastive learning.

4. Experiments

4.1. Dataset and Experimental Details

Dataset. We conduct experiments on three datasets that are commonly used in NCD: CIFAR-10 [20], CIFAR-100 [20] and ImageNet [8]. Following [12], we split the training data of each dataset into a labeled set and an unlabeled set, and assume that the the number of classes in the unlabeled set is known. The partitions of the three datasets are reported in Table 1. More details on the datasets can be found in supplementary. Following [12, 13], we report results averaged over 10 runs for CIFAR-10 and CIFAR-100.

Dataset	Labeled Set		Unlabeled Set	
	#image	#class	#image	#class
CIFAR-10	25K	5	25K	5
CIFAR-100	40K	80	10K	20
ImageNet	1.25M	882	≈30K	30

Table 1. Dataset statistics for novel class discovery.

For ImageNet, we report results averaged over 3 runs using three different unlabeled subsets.

Evaluation Metric. We employ average clustering accuracy (ACC) to evaluate the performance of different methods on unlabelled data. The ACC is defined as:

$$\mathrm{ACC} = \max_{perm \in P} \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\left\{y_i = perm\left(\hat{y}_i\right)\right\}, \tag{17}$$

where y_i and $\hat{y}_i$ represent the ground-truth label and clustering predicted label of a sample $x_i^u \in D^u$, respectively. P is the set of all permutations, which can be rapidly computed by the Hungarian algorithm [22].

Implementation Details. For a fair comparison with existing methods, we use ResNet-18 [15] as the backbone of our method for all three datasets. We follow [12] to initialize the model with self-supervised learning on the whole data and fine-tune the model with supervised learning on the labeled data, more training details can be found in [12]. In the step of novel class discovery on the unlabeled data, we use SGD optimizer to update the network. Note that, in the steps of supervised fine-tuning and novel class discovery, we only update the last convolutional block of the ResNet and the two classifiers. The initial learning rate is set to 0.1 and is divided by 10 after every 170/30 epochs for $\{$CIFAR-10, CIFAR-100$\}$/ImageNet. We train the model with 200/90 epochs in total for $\{$CIFAR-10, CIFAR-100$\}$/ImageNet. We randomly sample training samples from both the labeled and unlabeled data, where the batch size is set to 128/512 for $\{$CIFAR-10, CIFAR-100$\}$/ImageNet. For the consistency loss, we apply the ramp-up function with weight $\gamma = \{5, 50, 10\}$ and ramp-up length $T = \{50, 150, 50\}$ for CIFAR-10, CIFAR-100 and ImageNet, respectively. For the binary-cross entropy loss, we set $\lambda = 0.95$.

For our method, we introduce the neighborhood contrastive learning (NCL) and hard negative generation (HNG) at the $2th$ and $4th$ epoch, respectively. In default, we set memory size $|M|$ = 2,000, temperature τ = 0.05, number of pseudo-positives k_1 = $|M|/C^u/2$, weight of augmented-positive α = 0.2, number of negative samples k_2 = 400, and number of HNG iterations N = 5. For each mixing process, we generate new negatives with μ = 1/3 and μ = 2/3. That is, each mixing process will be performed twice using these two values of μ. We find the above parameter settings can consistently achieve stable and well performance across datasets. The parameter analysis can be found in the supplementary material.

Method	CIFAR-10	CIFAR-100
Basel. w/o SSL	85.0±0.4%	66.5±4.0%
Basel. w/o CE	83.9±9.4%	62.6±3.6%
Basel. w/o BCE	39.5±4.2%	18.1±0.8%
Basel. w/o CS	84.1±0.9%	61.6±3.2%
Baseline	**87.9±0.7%**	**69.4±1.4%**

Table 2. Ablation study of the baseline method on CIFAR-10 and CIFAR-100. **SSL**: self-supervised learning, **CE**: cross-entropy loss on the labeled data, **BCE**: binary cross-entropy loss on the unlabeled data, **CS**: consistency loss.

Method	CIFAR-10	CIFAR-100
Baseline	87.9±0.7%	69.4±1.4%
+ NCL w/o PP	61.8±7.6% (↓ 26.1%)	68.5±1.9% (↓ 0.9%)
+ NCL w/o AP	90.9±2.1% (↑ 3.0%)	79.7±5.7% (↑ 10.3%)
+ NCL w/o LA	93.3±0.1% (↑ 5.4%)	80.3±0.9% (↑ 10.9%)
+ NCL	93.4±0.2% (↑ 5.5%)	82.3±2.6% (↑ 12.9%)
+ NCL + HNG	93.4±0.1% (↑ 5.5%)	86.6±0.4% (↑ 17.2%)

Table 3. Evaluation of the effectiveness of the proposed neighborhood contrastive learning (NCL) and hard negative generation (HNG). **NCL w/o PP**: NCL without pseudo-positives, **NCL w/o LA**: without applying NCL on labeled data. **NCL w/o AP**: removing augmented-positive during NCL.

4.2. Evaluation

Ablation study on the baseline. We first evaluate the effectiveness of the components of the baseline, including self-supervised learning (SSL), cross-entropy (CE) loss on the labeled data, binary cross-entropy (BCE) loss on the unlabeled data, and consistency (CS) loss. We individually remove each of them from the baseline and evaluate the performance. Results are reported in Table 2. We mainly make the following four observations: (1) Removing each component will reduce the results of the baseline. (2) BCE is the most important component. When removing BCE, the results decrease substantially. Without BCE, the classifier is only learned with a weak supervision (*i.e.*, consistency loss) and therefore fails to cluster the samples. (3) Removing SSL from the baseline will decrease the performance. This is due to the fact that SSL improves the generality of the representations and thus benefits the learning of the BCE. (4) CS is also beneficial in novel class discovery, since it encourages the classifier to be more robust to intra-class variations. The above observations verify the effectiveness and importance of each component in the baseline.

Evaluation of the neighborhood contrastive learning. To study the effectiveness of neighborhood contrastive learning (NCL), we implement NCL in four ways. 1) NCL: The proposed NCL. 2) NCL w/o PP: NCL without pseudo-positives, which reduces to the vanilla contrastive learning; 3) NCL w/o LA: NCL without enforcing contrastive learning on the labeled data; 4) NCL w/o AP: NCL without approaching a query to its augmented-positive. Results on CIFAR-10 and CIFAR-100 are reported in Table 3. First, without neighborhood mining, the model will regard all the

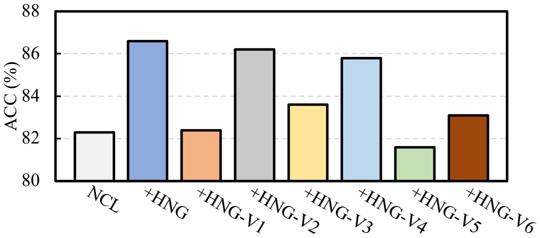

Figure 4. Comparison of the proposed hard negative generation (HNG) and its variants on CIFAR-100.

positive features in the memory as negative samples and push the query sample far away from its positive features, which will certainly damage the performance. Second, implementing NCL on the labeled data can help improve the discrimination of the model, which can facilitate the process of neighborhood mining and thus improve the ACC, especially given a larger labeled dataset (CIFAR-100). Third, the augmented-positive sample is important to improve the performance since it can mitigate the influence caused by the negative samples that are included in the selected KNNs. Fourth, our proposed NCL significantly improves the ACC of baseline. Specifically, NCL gains +5.5% on CIFAR-10 and +12.9% on CIFAR-100, demonstrating the effectiveness of the proposed NCL.

Evaluation of hard negative generation. We first evaluate our proposed hard negative generation (HNG) in Table 3. We find that HNG significantly increases the ACC for CIFAR-100. However, there is no boost for CIFAR-10. This is likely due to the fact that the labeled set in CIFAR-10 contains a small number of classes. In such a context, mixing between labeled and unlabeled samples is unable to generate diverse hard negative samples and thus fails to facilitate contrastive learning. In Table 4, we show that HNG can also improve the ACC for ImageNet, where the labeled set contains a large amount of classes. This further verifies the effectiveness of our HNG when given a rich labeled dataset. Another beneficial side-effect of HNG is the fact that it helps in stabilizing the training, reducing the variance of the results across all datasets (see Table 4).

To further study the advantage of our HNG, we compare HNG with 6 variants and, based on the results in Fig. 4 we make the following observations. (**HNG-V1**): Directly removing k-easiest unlabeled samples when computing NCL for each query rarely affects the ACC, supporting our point that easy negative samples contribute less to contrastive learning; (**HNG-V2**): Replacing k-easiest unlabeled samples with generated hard negative samples produces similar ACC to directly adding generated hard features to the feature queue (HNG); (**HNG-V3**): Directly using k randomly selected labeled samples as hard negative samples can slightly improve the ACC; (**HNG-V4**): Generating hard negative samples by mixing only on k randomly selected labeled samples can achieve further improvement over “HNG-V3”; (**HNG-V5**): Generating hard negative samples by mixing only on k-easiest unlabeled samples fails to improve the performance; (**HNG-V6**): Generating hard negative samples by mixing on k-easiest unlabeled samples and k-nearest labeled samples is suboptimal w.r.t using randomly selected labeled feature (HNG). This is because the k-nearest labeled features mostly are of the same class, limiting the variety of the generated hard features.

Taking the above observations, the proposed HNG can generate more variety and hard negative samples, which are effective in improving contrastive learning.

4.3. Comparison with State-of-The-Art Methods

We compare the proposed approach with one classical method (k-means [23]) and four state-of-the-art methods (*i.e.*, KCL [16], MCL [17], DTC [13] and RS [12]). For method based on k-means [23], we first use the labeled data to pre-train the model by supervised learning loss (*i.e.*, cross-entropy loss). Then, we use the trained model to extract features for the unlabeled data without further learning on the unlabeled data. Finally, we perform k-means clustering on these extracted features to obtain the clustering results. Except RS [12], all the other compared methods do not apply self-supervised learning technique. In order to make a fair comparison, we implement these methods (except RS [12]) with two settings depending on whether to utilize self-supervised learning to pre-train the model. With self-supervised learning, we first initialize the model by the rotation loss [10] using both labeled data and unlabeled data and then implement the methods with their own algorithms. Note that, since ImageNet has sufficient training samples from various classes, we directly use the labeled data to pre-train the model with cross-entropy loss for both settings. Comparison results are reported in Table 4.

We can obtain the following two conclusions. First, using self-supervised learning generally can improve the results of all methods, except when evaluated k-means [23] on CIFAR-100. For example, when using self-supervised learning, the ACC of KCL [16] is increased from 66.5% to 72.3% and from 14.3% to 42.1% on CIFAR-10 and CIFAR-100, respectively. This indicates the effectiveness of self-supervised learning. Second, two versions of our method outperform the state-of-the-art methods (whether using self-supervised learning or not) by a large margin on all datasets, especially on CIFAR-100 and ImageNet. Specifically, our full method achieves **ACC=93.4%** on CIFAR-10, **ACC=86.6%** on CIFAR-100 and **ACC=90.7%** on ImageNet, respectively. These results are higher than the current best method (RS [12]) by +3% on CIFAR-10, +13.4% on CIFAR-100 and +8.2% on ImageNet, respectively. This demonstrates that our method produces the new state-of-the-art results for novel class discovery.

Method	Venue	CIFAR-10	CIFAR-100	ImageNet
Methods without self-supervised learning				
k-means [23]	Classic	65.5±0.0%	56.6±1.6%	71.9%
KCL [16]	ICLR'18	66.5±3.9%	14.3±1.3%	73.8%
MCL [17]	ICLR'19	64.2±0.1%	21.3±3.4%	74.4%
DTC [13]	ICCV'19	87.5±0.3%	56.7±1.2%	78.3%
Methods with self-supervised learning				
k-means [23]*	Classic	72.5±0.0%	56.3±1.7%	71.9%
KCL [16]*	ICLR'18	72.3±0.2%	42.1±1.8%	73.8%
MCL [17]*	ICLR'19	70.9±0.1%	21.5±2.3%	74.4%
DTC [13]*	ICCV'19	88.7±0.3%	67.3±1.2%	78.3%
RS [12]*	ICLR'20	90.4±0.5%	73.2±2.1%	82.5%
Ours* w/o HNG	CVPR21	**93.4±0.2%**	**82.3±2.6%**	**89.5%**
Ours*	CVPR21	**93.4±0.1%**	**86.6±0.4%**	**90.7%**

Table 4. Comparison with state-of-the-art methods on CIFAR-10, CIFAR-100 and ImageNet for novel class discovery. Clustering accuracy is reported on the unlabelled set. "*" indicates methods that initialize models with self-supervised learning, except when evaluated on ImageNet. **Ours**: our method with both neighborhood contrastive learning and hard negative generation, **Ours w/o HNG**: our method without hard negative generation.

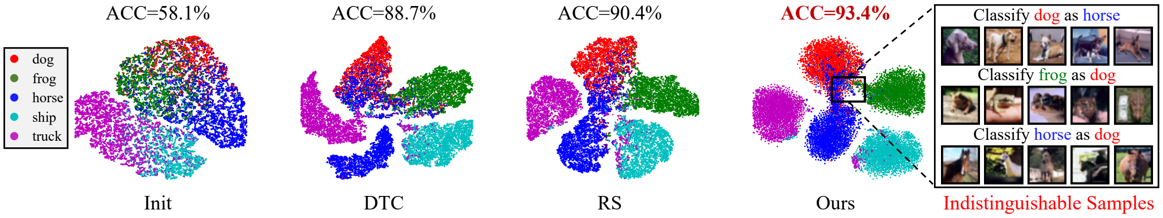

Figure 5. Feature visualization on CIFAR-10. We extract the output of the last pooling layer as the feature for all unlabeled data and use t-SNE [28] to map the features into a 2D embedding space. We compare our method with the initialized model (trained only with self-supervised learning and supervised learning), DTC [13] and RS [12]. We also show examples that are visually similar to samples of other classes and are classified to wrong classes.

4.4. Visualization

To better understand the proposed method, we visualize the feature embeddings of the unlabeled samples on CIFAR-10 using t-SNE [28]. In Fig. 5, we compare our method with the initial model and two state-of-the-art methods (DTC [13] and RS [12]). The initial model is trained with self-supervised learning on all data and supervised learning on the labeled data. As we can see, the initial model can roughly separate samples into 5 clusters. However, there are also many samples of different classes clustered together, resulting in low clustering accuracy (ACC=58.1%). Compared to the initial model, the other three methods (DTC, RS and our method) generate more discriminative representations, which produce significantly better clustering results. Since DTC, RS and our method all achieve very high clustering results (ACC>88%), we cannot observe obvious difference in clustering visualization between them. However, for our method, the samples of the same class are mostly clustered in a circular area, which is mainly caused by the constraint of enforcing neighbors to be close. We also show some indistinguishable samples that are located at the class decision boundaries. We find that these samples are visually similar, such as in terms of color (frog and dog) and pose (dog and horse), leading the model fail to distinguish them.

5. Conclusion

In this paper, we propose a holistic learning framework for Novel Class Discovery (NCD), which adopts contrastive learning to learn discriminate features with both the labeled and unlabeled data. Specifically, we propose the Neighborhood Contrastive Learning (NCL) to effectively leverage the local neighborhood in the embedding space, enabling us to take the knowledge from more positive samples and thus improve the clustering accuracy. In addition, we also introduce the Hard Negative Generation (HNG), which leverages the labeled samples to produce informative hard negative samples and brings further advantage to NCL. Experiments on three datasets demonstrate the significant superiority of our method over state-of-the-art NCD methods.

Acknowledgements This work is supported by the EU H2020 SPRING No. 871245 and AI4Media No. 951911 projects; the Italy-China collaboration project TALENT:2018YFE0118400; and the Caritro Deep Learning Lab of the ProM Facility of Rovereto. Zhun Zhong thanks Wenjing Li for encouragement.

References

[1] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006. 2

[2] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *Proc. ICLR*, 2020. 3

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Proc. NeurIPS*, 2019. 2

[4] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Sutter, and Björn Ommer. Cliquecnn: deep unsupervised exemplar learning. In *Proc. NeurIPS*, 2016. 1

[5] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Proc. NeurIPS*, 2019. 1

[6] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep adaptive image clustering. In *Proc. ICCV*, 2017. 2

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proc. ICML*, 2020. 1, 2, 3

[8] 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, 5

[9] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proc. ICCV*, 2017. 2

[10] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. CVPR*, 2018. 3, 7

[11] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proc. CVPR*, 2006. 2

[12] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *Proc. ICLR*, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[13] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proc. ICCV*, 2019. 1, 2, 5, 7, 8

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. CVPR*, 2020. 1, 2, 3, 4

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 1, 6

[16] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. Learning to cluster in order to transfer across domains and tasks. In *Proc. ICLR*, 2018. 2, 7, 8

[17] Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. Multi-class classification without multi-class labels. In *Proc. ICLR*, 2019. 2, 7, 8

[18] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *Proc. NeurIPS*, 2020. 2, 3, 4, 5

[19] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Proc. NeurIPS*, 2020. 4

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *University of Tronto*, 2009. 5

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NeurIPS*, 2012. 1

[22] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955. 6

[23] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proc. BSMSP*, 1967. 2, 7, 8

[24] Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Lsd-c: Linearly separable deep clusters. *arXiv*, 2020. 2

[25] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *Proc. ICLR*, 2021. 3, 4, 5

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 1

[27] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Proc. ECCV*, 2020. 2

[28] Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *JMLR*, 2014. 8

[29] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Proc. ECCV*, 2020. 2

[30] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *Proc. ICML*, 2019. 3, 5

[31] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, 2018. 2

[32] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *Proc. ICML*, 2016. 1, 2

[33] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *Proc. ICML*, 2017. 2

[34] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Proc. NeurIPS*, 2005. 2

[35] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proc. ICLR*, 2018. 1, 3, 5

[36] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proc. ICCV*, 2019. 2