Relative Contrastive Loss for Unsupervised Representation Learning

Shixiang Tang[1,3], Feng Zhu[3], Lei Bai[2,†], Rui Zhao[3,4], and Wanli Ouyang[2,1]

[1] University of Sydney, Australia
[2] Shanghai AI Laboratory, China
[3] Sensetime Research
[4] Qing Yuan Research Institute, Shanghai Jiao Tong University
stan3906@uni.sydney.edu.au, baisanshi@gmail.com † corresponding author

1 Mathematical Analysis of Relative Contrastive Loss

1.1 Detailed Formulation Analysis of Relative Contrastive Loss.

We start by denoting two different images $\mathbf{x}$ and $\mathbf{x}'$. Given a criteria $\mathcal{M}_i$, we define their label as $\mathcal{Y}_i(\mathbf{x})$ and $\mathcal{Y}_i(\mathbf{x}')$, respectively. Inspired by BYOL [5] and Simsiam [3], the predictor layer aims to predict the expectation of projections $\mathbf{z}$ under transformation $\mathcal{T}_i$, *i.e.,* $\mathbb{E}_{\mathcal{T}_i}(\mathbf{z})$, where $\mathcal{T}_i$ is semantic-invariant on $\mathcal{M}_i$, *i.e.,* $\mathcal{Y}_i(\mathcal{T}_i(\mathbf{x})) = \mathcal{Y}_i(\mathbf{x})$.

To analyze the formulation of our relative contrastive loss given two images $\mathbf{x}$ and $\mathbf{x}'$, we start with its individual component in Eq. (2) of the main paper, *i.e.,*

$$\mathcal{L}_{RCL}\left(\mathbf{x}, \mathbf{x}', \theta; \{\mathcal{M}_i\}_{i=1}^{H}\right) = \sum_{i=1}^{H} \alpha_i \mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i), \tag{1}$$

where α_i is the trade-off parameter among different criteria. The loss $\mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)$ for criterion $\mathcal{M}_i$ can be defined as

$$\mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i) = -\log\left[\frac{\mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')\right] \times \exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) \neq \mathcal{Y}_i(\mathbf{z}')\right]}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}\right]. \tag{2}$$

When $\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')$,

$$\mathcal{L}^{+}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i) = -\log\left[\frac{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right)}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}\right], \tag{3}$$

which pulls the query predictions $\mathbf{q}_{\mathcal{M}_i}$ and projection $\mathbf{z}'$ together.

When $\mathcal{Y}_i(\mathbf{z}) \neq \mathcal{Y}_i(\mathbf{z}')$,

$$\mathcal{L}^{-}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i) = -\log\left[\frac{1}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}\right], \tag{4}$$

which pushs the query predictions $\mathbf{q}_{\mathcal{M}_i}$ and projection $\mathbf{z}'$ apart.

Given a query-key pair $(\mathbf{z}, \mathbf{z}')$ and a set of semantic criteria $\{\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_H\}$, if $\mathcal{Y}_i(\mathbf{z}) \neq \mathcal{Y}_i(\mathbf{z}')$ for $i < h$ and $\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')$ for $i \geq h$, the relative contrastive loss becomes

$$\mathcal{L}_{RCL}\left(\mathbf{x}, \mathbf{x}', \theta; \{\mathcal{M}_i\}_{i=1}^{H}\right) = \sum_{i=1}^{h-1} \alpha_i \mathcal{L}^{-}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i) + \sum_{i=h}^{H} \alpha_i \mathcal{L}^{+}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i), \tag{5}$$

where the positive-negative relation of $(\mathbf{z}, \mathbf{z}')$ is relative and depends on the particular semantic criterion $\mathcal{M}_i$.

1.2 Derivative of Gradients of Relative Contrastive Loss.

We calculate the gradient of relative contrastive loss $\mathcal{L}^{+}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)$ when $\mathcal{Y}_i(\mathbf{x}) = \mathcal{Y}_i(\mathbf{x}')$ and $\mathcal{L}^{-}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)$ when $\mathcal{Y}_i(\mathbf{x}) \neq \mathcal{Y}_i(\mathbf{x}')$, respectively. Specifically,

$$\begin{aligned}
\frac{\partial \mathcal{L}^{+}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{z}} &= -\frac{\partial}{\partial \mathbf{z}}\left[\frac{\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'}{\tau} - \log\left[\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)\right]\right] \\
&= -\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{z}'}{\tau} + \frac{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right)\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{z}'}{\tau} + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{s}_k}{\tau}}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}, \\
\frac{\partial \mathcal{L}^{-}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{z}} &= -\frac{\partial}{\partial \mathbf{z}}\left[-\log\left[\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)\right]\right] \\
&= \frac{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right)\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{z}'}{\tau} + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{s}_k}{\tau}}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}.
\end{aligned} \tag{6}$$

Denote

$$\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right) = \frac{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right)}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}'/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}, \tag{7}$$

$$\mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right) = \frac{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}{\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{z}')/\tau\right) + \sum_{k=1}^{K}\exp\left(\mathbf{q}_{\mathcal{M}_i}^{\top}\mathbf{s}_k/\tau\right)}, \tag{8}$$

where $\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right)$ and $\mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right)$ are always non-negative and $\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right) + \sum_{k=1}^{K}\mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right) = 1$. Therefore, $\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right)$ can viewed as a valid probability of assigning the query prediction $\mathbf{q}_{\mathcal{M}_i}$ to the label of projection $\mathbf{z}'$ and the label of negative samples $\mathbf{s}_k$, respectively.

After substituting Eq. 7 and Eq. 8 into Eq. 6, we get

$$\begin{aligned}
\frac{\partial \mathcal{L}^{+}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{z}} &= \left[\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right) - 1\right]\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{z}'}{\tau} + \sum_{k=1}^{K}\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right)\frac{\mathbf{s}_k}{\tau}, \\
\frac{\partial \mathcal{L}^{-}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{z}} &= \left[\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right)\right]\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\frac{\mathbf{z}'}{\tau} + \sum_{k=1}^{K}\frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}}\mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right)\frac{\mathbf{s}_k}{\tau}.
\end{aligned} \tag{9}$$

Then ,we get the gradient of $\mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)$ in Eq. 2 as Eq. 4 in the main text, *i.e.,*

$$\begin{aligned}\frac{\partial \mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{z}} &= \frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}} \frac{\partial \mathcal{L}(\mathbf{x}, \mathbf{x}', \theta; \mathcal{M}_i)}{\partial \mathbf{q}_{\mathcal{M}_i}} \\ &= \left[\mathbb{P}\left(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}\right) - \mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')\right]\right] \frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}} \frac{\mathbf{z}'}{\tau} + \sum_{k=1}^{K} \frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}} \mathbb{P}\left(\mathbf{s}_k|\mathbf{q}_{\mathcal{M}_i}\right) \frac{\mathbf{s}_k}{\tau}.\end{aligned} \tag{10}$$

Finally, the gradient of relative contrastive loss $\mathcal{L}_{RCL}$ is (discard negative samples $\{\mathbf{s}_k\}_{k=1}^{K}$ in the support set $\mathcal{S}$)

$$\frac{\partial \mathcal{L}_{RCL}}{\partial \mathbf{z}} = \sum_{i=1}^{H} \alpha_i \frac{\partial \mathbf{q}_{\mathcal{M}_i}}{\partial \mathbf{z}} \left(\mathbb{P}(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}) - \mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')\right]\right) \frac{\mathbf{z}'}{\tau}. \tag{11}$$

1.3 Visualization of Relative Contrastive Loss

The relative contrastive loss considers the positive-negative relation depending on a set of criteria $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_H\}$. According to Eq. 11, we define the attractor $\mathcal{A}(\mathbf{z}, \mathbf{z}')$ and repellor $\mathcal{R}(\mathbf{z}, \mathbf{z}')$ to describe the relativeness between the features of a given query-key pair $(\mathbf{z}, \mathbf{z}')$. Without the loss of generality, we add only one predictor $\mathcal{P}(*, \theta_p)$ instead of multiple predictors $\{\mathcal{P}(*, \theta_p^i)\}_{i=1}^{H}$ after query projection in our experiments for visualization, and set the weight $\alpha_i = \frac{1}{H}$. The number of criteria H is set to be 3. For ease of our visualization, we only visualize the pull-push dynamics between query prediction $\mathbf{q}$ and the key projection $\mathbf{z}'$, *i.e.,*

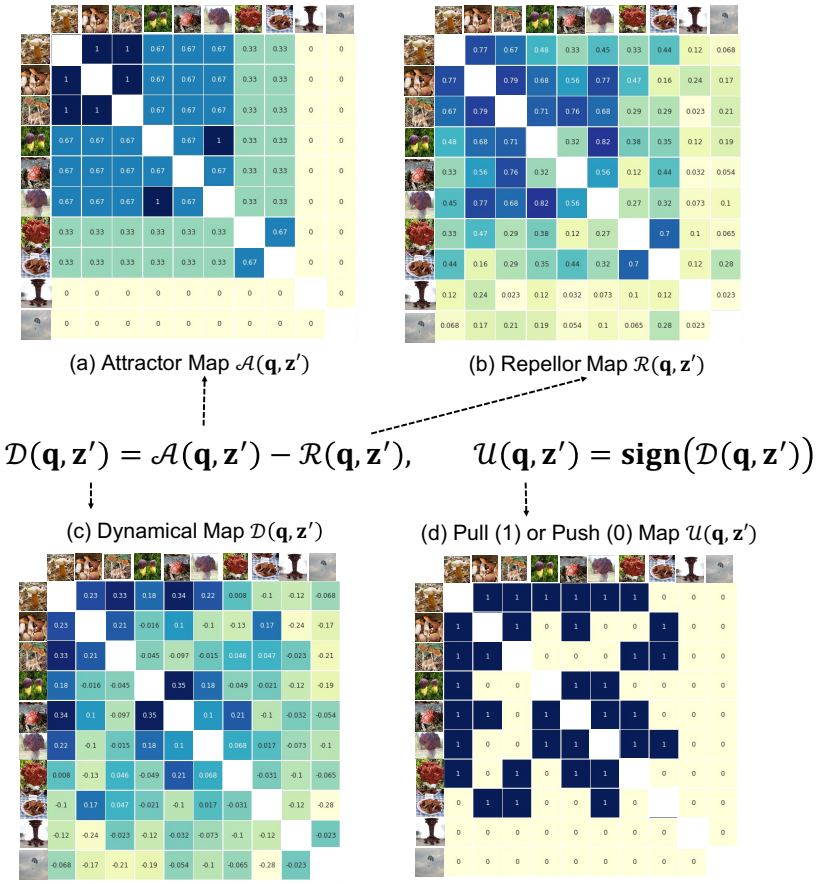
$$\frac{\partial \mathcal{L}_{RCL}}{\partial \mathbf{q}} = \sum_{i=1}^{H} \alpha_i \left(\mathbb{P}(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}) - \mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')\right]\right) \frac{\mathbf{z}'}{\tau}, \tag{12}$$

Concretely, the attractor $\mathcal{A}(\mathbf{q}, \mathbf{z}')$ and the repellor $\mathcal{R}(\mathbf{q}, \mathbf{z}')$ can be defined as

$$\mathcal{A}(\mathbf{q}, \mathbf{z}') = \sum_{i=1}^{H} \alpha_i \mathbb{P}(\mathbf{z}'|\mathbf{q}_{\mathcal{M}_i}) \tag{13}$$

$$\mathcal{R}(\mathbf{q}, \mathbf{z}') = \sum_{i=1}^{H} \alpha_i \mathbb{I}\left[\mathcal{Y}_i(\mathbf{z}) = \mathcal{Y}_i(\mathbf{z}')\right] \tag{14}$$

S-Figure 1(a) shows a query image (the first image in the column and row), two images that share the same label with the query image in the hierarchical label bank at all levels $h = 1, 2, 3$, three images that share the same label in the hierarchical label bank at level $h = 2, 3$, two images that only share the same label in hierarchical label bank at level $h = 3$, and two images that are not labeled the same with query image in the hierarchical label bank at any level. The results in S-Figure 1(c) show that the final decision on pull (greater than 0) and push (smaller than 0) is continuous, different from the designs in [6, 2, 4, 1, 5], that are discrete. Besides, the continuous values reflect relative semantic and visual similarities among samples.



S-Figure 1: Visualization of relative contrastive loss. (a) *Attractor Map* $\mathcal{A}(\mathbf{q}, \mathbf{z}')$ in Eq. 13: Attractive map denotes the attractive force of relative contrastive loss that pulls query-key pair $(\mathbf{q}, \mathbf{z}')$ together. (b) *Repellor Map* $\mathcal{R}(\mathbf{q}, \mathbf{z}')$ in Eq. 14: Repellor map denotes the repulsive force of relative contrastive loss that pushes query-key pair $(\mathbf{q}, \mathbf{z}')$ apart. (c) *Dynamical Map* $\mathcal{D}(\mathbf{q}, \mathbf{z}') = \mathcal{A}(\mathbf{q}, \mathbf{z}') - \mathcal{R}(\mathbf{q}, \mathbf{z}')$: the difference of the attractor map and the repellor map. Positive value means the query-key pair $(\mathbf{q}, \mathbf{z}')$ should be pulled together, the negative value means the query-key pair $(\mathbf{q}, \mathbf{z}')$ should be pushed apart. The absolute value of *dynamical map* means the strength of force. (d) *Pull or Push Map* $\mathcal{U}(\mathbf{q}, \mathbf{z}') = \text{sign}\,(\mathcal{D}(\mathbf{q}, \mathbf{z}'))$: Pull or Push map denotes the final attractive or repulsive force between a query-key pair $(\mathbf{q}, \mathbf{z}')$. 0 denotes pushing two features apart and 1 denotes pulling two feature together.

Method	Object Detection		Instance Segmentation	
	AP-all bb	AP-50 bb	AP mk	AP-50 mk
Supervised	38.2	58.2	33.3	54.7
MoCo v2	**39.3**	58.9	**34.4**	55.8
SwAV	37.9	57.6	33.1	54.2
Simsiam	39.2	**59.3**	**34.4**	56.0
Barlow Twins	39.2	59.0	34.3	56.0
RCL	**39.3**	59.1	34.3	**56.1**

S-Table 1: Transfer learning from ImageNet with standard ResNet50 to COCO object detection and instance segmentation. All methods are evaluated on the test-dev dataset. bb: bounding box. mk: segmentation mask.

2 Transfer to Detection and Segmentation Tasks

In this section, we provide the detection and segmentation results[5], when we transfer our model to detection and segmentation tasks. We strictly follow the evaluation protocol in MOCO [6]. Specifically, we do not freeze the batch normalization layer, and finetune the whole network by the COCO training set. We report the results on the COCO evaluation dataset in S-Table 1.

3 Hierarchical Clustering

In this paper, hierarchical clustering is a natural instantiation for multiple criteria. To elaborate the process of hierarchical clustering, we first describe two implementations, *i.e.,* label propagation and average linkage, for deciding two clusters to merge or not.

3.1 Label Propagation

Label Propagation [7] is a widely-adopted method of computing the possibility that two samples/clusters belong to the same class. Given n units $\mathcal{U} = \{\mathcal{U}_i\}_{i=1}^n$, *i.e.,* clusters or samples to be split/merged, we first estimate its pairwise similarities by the dot product of the unit prototypes $\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_n)$, *i.e.,* features for single images or cluster feature centers. Mathematically, it can be formulated as

$$\mathbf{A} = \mathbf{U}^\top \mathbf{U}. \tag{15}$$

Following [7], we can obtain the normalized affinity matrix $\hat{\mathbf{A}}$ by

$$\hat{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{1/2}, \tag{16}$$

[5] These experiments are not the improved version of the method RCL, just the generalization ability evaluation of the method.

where $\mathbf{D}$ is a diagonal matrix with elements $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{A}_{ij}$. We denote the predicted probabilities of samples/clusters as $\mathbf{P}^t = (\mathbf{p}_1^t, \mathbf{p}_2^t, ..., \mathbf{p}_n^t) \in \mathbb{R}^{n \times k}$ after t-th propagation (defined in Eq. 18), where k is the number of classes (clusters) that n units may belong to, $\mathbf{p}_*^t = (p_{*,1}^t, p_{*,2}^t, ..., p_{*,k}^t)$ and $p_{*,k'}^t$ denotes the probability of the sample belong to the k'-th class. For the i-th unit, we would like to propagate the class predictions from other units j as

$$\mathbf{p}_i^{t+1} = \gamma \sum_{j \neq i} \hat{\mathbf{A}}_{ij} \mathbf{p}_j^t + (1-\gamma)\mathbf{p}_i^0 = \gamma \hat{\mathbf{A}}_i \mathbf{P}^t + (1-\gamma)\mathbf{p}_i^0, \tag{17}$$

where γ is a propagation strength parameter, $\mathbf{P}^0 = (\mathbf{p}_1^0, \mathbf{p}_2^0, ..., \mathbf{p}_n^0)$, $\mathbf{p}_i^0$ is the initial label prediction of the i-th unit that we will specifically define in the following cluster split and cluster merge.

Intuitively, if the i-th sample and the j-th sample are similar with a high affinity $\hat{\mathbf{A}}(i,j)$, the prediction $\mathbf{p}_j^t$ of the jth sample would have a larger weight to be propagated to the prediction $\mathbf{p}_i^{t+1}$ of the i-th sample. Propagating the predictions between all samples in parallel can be formulated as

$$\mathbf{P}^{t+1} = \gamma \hat{\mathbf{A}} \mathbf{P}^t + (1-\gamma)\mathbf{P}^0, \tag{18}$$

which is an iterative algorithm. The closed solution $\mathbf{P}^\infty$ after conducting Eq. 18 for multiple times until convergence is

$$\mathbf{P}^\infty = (\mathbf{I} - \gamma \hat{\mathbf{A}})^{-1} \mathbf{P}^{\mathbf{0}}. \tag{19}$$

For each unit to be split or merged, we estimate its class prediction $\mathbf{p}_i^\infty = (p_{i,0}^\infty, p_{i,1}^\infty, ..., p_{i,k}^\infty)$ by propagating the neighboring information with Eq. 19, which is used to merge the i-th unit to j-th cluster when $p_{i,j}^\infty > \sigma_m$. Here σ_m is the manually designed threshold for cluster merge.

Initialize $\mathbf{P}_0$ in Cluster Split. As described in Cluster Split part in Sec. 4.3 in the main text, we split a cluster $\mathcal{C}_i^{h+1}$ into m clusters, and uses the clusters at h-th level at its split units, *i.e.,* $\mathcal{U}_i^{h+1} = \{\mathcal{C}_j^h | \mathcal{C}_j^h \subset \mathcal{C}_i^{h+1}, j = 1, 2, ..., k^h\}$, where k^h is the number of clusters at h-th level. We re-denote $\mathcal{U}_i^{h+1} = \{\mathcal{O}_1, \mathcal{O}_2, ..., \mathcal{O}_n\}$, where $n = |\mathcal{U}_i^{h+1}|$. We first select m the most dissimilar split units in $\mathcal{U}_i^{h+1}$ as the prototypes of each class. Then, we initialize $(\mathbf{p}_{jk}^0)$ as 1 if $\mathcal{O}_j$ is selected as the prototype of the k-th class, and as 0 otherwise, *i.e.,*

$$\mathbf{p}_{jk}^0 = \begin{cases} 1, \text{ if } \mathcal{O}_j \text{ is selected as the prototype of the } k\text{-th class,} \\ 0, \qquad\qquad\qquad\qquad \text{otherwise,} \end{cases} \tag{20}$$

Initialize $\mathbf{P}_0$ in Cluster Merge. We treat every cluster in the merge units $\mathcal{V}_i^{h+1}$ in Sec. 4.3 Cluster merge in the main text as an individual class, and then use label propagation to determine to merge two clusters if their prediction belonging to the same class is larger than σ_m. Specifically, we initialize $\mathbf{P}^0$ as

$$\mathbf{P}^0 = \mathbf{I}_{n' \times n'}, \tag{21}$$

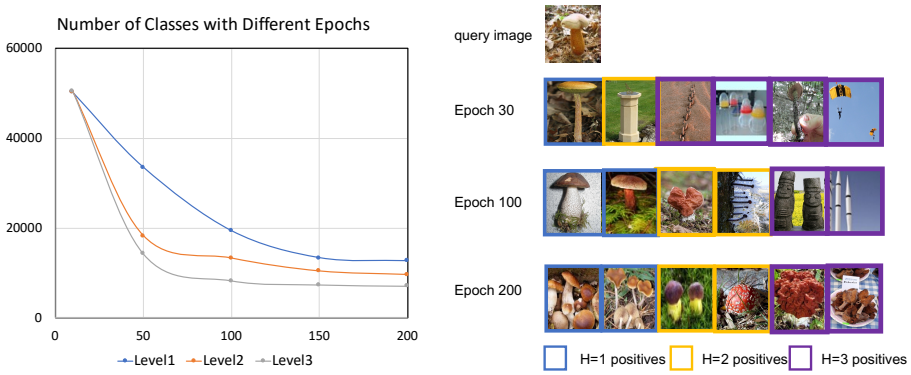
where $n' = |\mathcal{V}_i^{h+1}|$ is number of merge units in $\mathcal{V}_i^{h+1}$.

Effectiveness of Label Propagation. Label propagation serves as a cornerstone in cluster split and cluster merge for estimating the possibility that two samples/clusters labeled the same. To evaluate the effectiveness of label propagation in hierarchical clustering, we replace the label propagation by typical implementation, *i.e.,* feature similarity, in hierarchical clustering to justify whether two units belong to the same class. Average linkage based hierarchical clustering [8] determines merge and split by pairwise similarity only, thus can not consider the neighboring information in the data distribution. The detailed implementation of hierarchical clustering by average linkage is specifically described in supplementary materials. Comparing Exp. 2 with Exp. 3 and comparing Exp. 4 with Exp. 5 in S-Table 2, we find the accuracy with label propagation is about 6% higher than that clustered by average linkage if we set the hierarchy of clustering to 3. Comparing Exp 3, 5 and Exp 1, 2, 4 in S-Table 2, we find different trends when implementing the label propagation and average linkage, *i.e.,* the accuracy increases as the number of hierarchies increases for label propagation (Exp 1, 2, 4) but obviously drops for average linkage (Exp 3, 5). We attribute this to the failure of average linkage based clustering, and therefore the criteria by hierarchical clustering with label propagation determine query-key pair positive and negative incorrectly. This analysis shows the potential of designing more appropriate criteria as the future work when implementing relative contrastive loss in the feature.

3.2 Clusters with Different Epochs.

To explore how the number of clusters changes with the increase of training epochs, we depict the number of clusters in the support set $\mathcal{S}$ with different training epochs in S-Figure 2(left). We find the number of clusters decreases consistently during the network training, which demonstrates that the network can learn semantic knowledge from the dataset. Besides, with the increase of hierarchical level, the number of clusters decrease, which obeys the *cluster preserve property* of the hierarchical label.

To further understand the process of hierarchical clustering, we illustrate the clustering results of the support set $\mathcal{S}$ in S-Figure 2(right). As shown in the figure, positive samples at lower hierarchical level (e.g. $H = 1$) are more similar to each other, while positive samples at higher hierarchical level (e.g. $H = 3$) are more dissimilar to each other visually and semantically. For example, at epoch 200, the $H = 1$ positives share the same color, shape, and semantic meaning (mushroom) with query image, but the $H = 3$ positives only share the similar shape with the query image but have different colors and possibly different semantic meanings (mushroom v.s. rockets). With the increase of training epochs, samples in the same hierarchical level are more similar to each other visually and semantically, because the CNN features are learned better.



S-Figure 2: **Left:** Number of classes with different epochs. Blue line, orange line and gray denote the number of clusters in the hierarchical label bank at $H = 1$, $H = 2$ and $H = 3$, respectively. **Right:** Visualization of positives samples at different levels in the hierarchical label bank. Comparing with images in the hierarchical label at different epochs (epoch=30, 100, 200), the samples in the hierarchical label bank at all levels are becoming more and more visually similar with the query image. When we focus on the samples in only one epoch, we find that with the increase of level of the hierarchical label bank, the number of images increases, the images are less visually similar to the query image than those in the hierarchical label bank at relative low levels. Specifically, we find $H = 1, 2$ positives are more similar to the query image than $H = 3$ positives.

S-Table 2: The effectiveness of using label propagation and number of hierachies.

No.	#Predictors	#Hierarchies	LP	Acc
1	1	1	Yes	70.2
2	1	2	Yes	**71.3**
3	1	2	No	68.5
4	1	3	Yes	**71.8**
5	1	3	No	65.5
6	1	4	Yes	**71.4**
7	1	4	No	67.8
8	3	3	Yes	**72.6**

S-Table 3: Sensitivity of Cluster Merge Threshold σ_m

σ_m	num of clusters (H=3)	linear evalutaion
0.10	32	15.4
0.30	565	37.2
0.40	2752	43.2
0.50	5253	64.3
0.55	8795	70.9
0.60	9321	**72.6**
0.70	23246	72.3
0.80	38842	72.2
0.90	89642	71.5

3.3 Sensitivity of the Cluster Merge Threshold σ_m.

The cluster merge threshold is a hyper-parameter and determines when two clusters can be merged. In this part, we analysis the influence of σ_m to the model's performance under the linear evaluation setting. To ease the hyper-parameter tuning process, we simply set the merge threshold the same throughout all levels in the hierarchical clustering. As can be observed in S-Table 3, when σ_m is small, there are only a few clusters in the last level and the linear classification results are very bad. We attribute the failure to too many samples wrongly grouped in same cluster. Even when we set $\sigma_m = 0.4$ (the number of clusters equal to 2752), when linear evaluation results are still poor, which indicates the large number of noisy labels in the hierarchical label bank. The model achieves best performance when setting σ_m to 0.6, which leads to a moderate cluster size compared to $\sigma_m = 0.1$ and $\sigma_m = 0.9$. The results demonstrate that it is important to make a good balance between learning more diverse semantic variance and maintain suitable discriminative ability. Besides, we also observe that the accuracy change is small when the threshold σ_m is larger than 0.55, showing that the model is not sensitive to the value of σ_m when it is large enough.

References

1. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
2. Chen, T., Kornblith, S., Swersky, K., Norouzi, M., Hinton, G.: Big self-supervised models are strong semi-supervised learners. arXiv preprint arXiv:2006.10029 (2020)
3. Chen, X., He, K.: Exploring simple siamese representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15750–15758 (2021)
4. Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., Zisserman, A.: With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. arXiv preprint arXiv:2104.14548 (2021)
5. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., et al.: Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint arXiv:2006.07733 (2020)
6. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020)
7. Liu, Y., Lee, J., Park, M., Kim, S., Yang, E., Hwang, S.J., Yang, Y.: Learning to propagate labels: Transductive propagation network for few-shot learning. arXiv preprint arXiv:1805.10002 (2018)
8. Murtagh, F., Contreras, P.: Algorithms for hierarchical clustering: an overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **2**(1), 86–97 (2012)