Unsupervised Visual Representation Learning by Online Constrained K-Means

Qi Qian[1] Yuanhong Xu[2] Juhua Hu[3] Hao Li[2] Rong Jin[1]
[1]Alibaba Group, Bellevue, WA 98004, USA
[2]Alibaba Group, Hangzhou, China
[3]School of Engineering and Technology, University of Washington, Tacoma, WA 98402, USA
{qi.qian, yuanhong.xuyh, lihao.lh, jinrong.jr}@alibaba-inc.com, juhuah@uw.edu

Abstract

*Cluster discrimination is an effective pretext task for unsupervised representation learning, which often consists of two phases: clustering and discrimination. Clustering is to assign each instance a pseudo label that will be used to learn representations in discrimination. The main challenge resides in clustering since prevalent clustering methods (e.g., k-means) have to run in a batch mode. Besides, there can be a trivial solution consisting of a dominating cluster. To address these challenges, we first investigate the objective of clustering-based representation learning. Based on this, we propose a novel clustering-based pretext task with online **Co**nstrained **K**-**m**eans (**CoKe**). Compared with the balanced clustering that each cluster has exactly the same size, we only constrain the minimal size of each cluster to flexibly capture the inherent data structure. More importantly, our online assignment method has a theoretical guarantee to approach the global optimum. By decoupling clustering and discrimination, CoKe can achieve competitive performance when optimizing with only a single view from each instance. Extensive experiments on ImageNet and other benchmark data sets verify both the efficacy and efficiency of our proposal.*

1. Introduction

Recently, many research efforts have been devoted to unsupervised representation learning that aims to leverage the massive unlabeled data to obtain applicable models. Different from supervised learning, where labels can provide an explicit discrimination task for learning, designing an appropriate pretext task is essential for unsupervised representation learning. Many pretext tasks have been proposed, e.g., instance discrimination [12], cluster discrimination [3], invariant mapping [9,14], solving jigsaw puzzles [20], patch inpainting [21], etc. Among them, instance discrimination that identifies each instance as an individual class [12] is popular due to its straightforward objective. However, this pretext task can be intractable on large-scale data sets. Consequently, contrastive learning is developed to mitigate the large-scale challenge [6, 15, 28] with a memory bank [15] or training with a large mini-batch of instances [6], which requires additional computation resources.

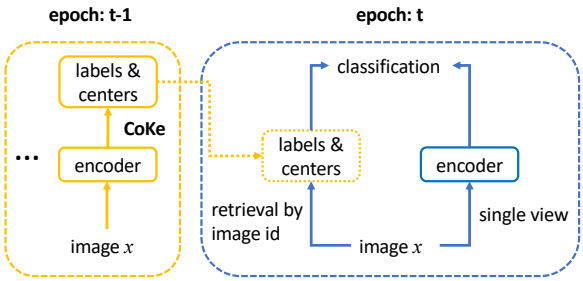

Figure 1. Illustration of CoKe. When a mini-batch arrives, each instance will be assigned to a cluster with our online assignment method. Then, in epoch t, representations from the encoder network are optimized by discrimination using pseudo labels and cluster centers obtained from epoch $t-1$. The pseudo labels from epoch $t-1$ were stored to be retrieved in epoch t using the unique id for each image.

Besides instance discrimination, cluster discrimination is also an effective pretext task for unsupervised representation learning [1, 3–5, 18, 30, 31]. Compared with instance discrimination that assigns a unique label to each instance, cluster discrimination partitions data into a predefined number of groups that is significantly less than the total number of instances. Therefore, the classification task after clustering becomes much more feasible for large-scale data. Furthermore, learning representations with clusters will push similar instances together, which may help explore potential semantic structures in data. Unfortunately, the clustering phase often needs to run multiple iterations over the entire data set, which has to be conducted in a batch mode to access representations of all instances [3]. Therefore, online clustering is adopted to improve the efficiency, while the collapsing problem (i.e., a dominating

cluster that contains most of instances) becomes challenging for optimization. To mitigate the problem, ODC [30] has to memorize representations of all instances and decompose the dominating large cluster with a conventional batch mode clustering method. Instead, SwAV [4] incorporates a balanced clustering method [1] and obtains assignment with a batch mode solver for instances from only the last few mini-batches, which outperforms the vanilla online clustering in ODC [30] significantly. However, using only a small subset of data to generate pseudo labels can fail to capture the global distribution. Besides, balanced clustering constrains that each cluster has exactly the same number of instances, which can result in a suboptimal partition of data.

To take the benefits of cluster discrimination but mitigate the challenge, we, for the first time, investigate the objective of clustering-based representation learning from the perspective of distance metric learning [22]. Our analysis shows that it indeed learns representations and relationships between instances simultaneously, while the coupled variables make the optimization challenging. By decoupling those variables appropriately, the problem can be solved in an alternating manner between two phases, that is, clustering and discrimination. When fixing representations, clustering is to discover relationship between instances. After that, the representations can be further refined by discrimination using labels from clustering. This finding explains the success of existing cluster discrimination methods. However, most existing methods have to conduct expensive clustering in a batch mode, while our analysis shows that an online method is feasible to optimize the objective.

Based on the observation, we propose a novel pretext task with online **Co**nstrained **K**-m**e**ans (**CoKe**) for unsupervised representation learning. Concretely, in the clustering phase, we propose a novel online algorithm for constrained k-means that lower-bounds the size of each cluster. Different from balanced clustering, our strategy is more flexible to model inherent data structure. In addition, our theoretical analysis shows that the proposed online method can achieve a near-optimal assignment. In the discrimination phase, we adopt a standard normalized Softmax loss with labels and centers recorded from the last epoch to learn representations. By decoupling the clustering and discrimination phases, CoKe can learn representations with a *single* view from each instance effectively and can be optimized with a small batch size. In addition, two variance reduction strategies are proposed to make the clustering robust for augmentations. Fig. 1 illustrates the framework of CoKe, which demonstrates a simple framework without additional components (e.g., momentum encoder [14, 15], batch mode solver [4, 30], etc.). Besides, only one label for each instance is kept in memory, which is an integer and the storage cost is negligible.

Extensive experiments are conducted on both downstream tasks and clustering to demonstrate the proposal. With only a single view from each instance for training, CoKe already achieves a better performance than MoCo-v2 [8] that requires two views. By including additional views in optimization, CoKe demonstrates state-of-the-art performance on ImageNet and clustering.

2. Related Work

Various pretext tasks have been proposed for unsupervised representation learning. We briefly review instance discrimination and cluster discrimination that are closely related to our work, while other representative methods include BYOL [14], SimSiam [9] and Barlow Twins [29].

2.1. Instance Discrimination

Instance discrimination is a straightforward pretext task for unsupervised representation learning, which tries to pull different augmentations from the same instance together but push them away from all other instances. Early work in this category optimizes the instance classification directly (i.e., each instance has one unique label), which implies an N-class classification problem, where N is the total number of instances [12]. Although promising results are obtained, this requires a large classification layer for deep learning. To improve the efficiency, the non-parametric contrastive loss is developed to mitigate the large-scale challenge [28]. After that, many variants such as MoCo [8, 10, 15] and SimCLR [6] are developed to approach or even outperform supervised pre-trained models on downstream tasks.

2.2. Cluster Discrimination

Instance discrimination focuses on individual instances and ignores the similarity between different instances. Therefore, clustering-based method is developed to capture the data structure better, which often consists of two phases: clustering and discrimination. DeepCluster [3] adopts a standard k-means for clustering, while SeLa [1] proposes to solve an optimal transport problem for balanced assignment. After obtaining the pseudo labels, the representation will be learned by optimizing the corresponding classification problem. The bottleneck of these methods is that labels need to be assigned offline in a batch mode with representations of all instances to capture the global information.

To reduce the cost of batch mode clustering, ODC [30] applies the standard online clustering to avoid the multiple iterations over the entire data set, while representations of all instances need to be kept in memory to address the collapsing problem. SwAV [4] extends a batch mode optimal transport solver [1] to do an online assignment to mitigate the collapsing problem. The assignment problem in SwAV is defined within a mini-batch of instances to save

the storage for representations. To improve the effectiveness, the method stores representations from the last few mini-batches of instances to capture additional information. However, this is still a small subset compared to the whole data and thus the global information may not be exploited sufficiently. After that, DINO [5] proposes to have the additional momentum encoder to stabilize the clustering without memorizing representations and can achieve the similar performance as SwAV with ResNet-50.

Besides clustering-based pretext tasks, some work proposes to leverage nearest neighbors for each instance to capture semantic similarity between different instances [13]. However, a large batch size and memory bank are required to capture the appropriate neighbors, which is expensive for optimization. In this work, we aim to improve the clustering phase with an online constrained k-means method, which gives better flexibility on cluster size and has a theoretical guarantee on the online assignment.

3. Proposed Method

3.1. Objective for Clustering-Based Method

We begin our analysis with the supervised representation learning. Given supervised label information, distance metric learning [27] has been studied extensively to learn representations by optimizing triplet constraints including some efficient proxy-based variants [19, 22, 23]. When there are K classes in data, let $C = [\mathbf{c}_1, \dots, \mathbf{c}_K] \in \mathcal{R}^{d\times K}$ denote K proxies with each corresponding to one class. The triplet constraint defined with proxies is $\forall \mathbf{x}_i, \mathbf{c}_{k:k\neq y_i}, \quad \|\mathbf{x}_i - \mathbf{c}_k\|_2^2 - \|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2 \geq \delta$, where y_i is the label of $\mathbf{x}_i$. To maximize the margin, the optimization problem for supervised representation learning can be cast as

$$\min_{\mathbf{x},C} \sum_i \sum_{k:k\neq y_i} \|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2 - \|\mathbf{x}_i - \mathbf{c}_k\|_2^2 \qquad (1)$$

which can be solved effectively with deep learning [22].

Without supervised label information, we assume that there are K clusters in data. Besides proxies for each cluster, we have an additional variable μ such that $\mu_{i,k} = 1$ assigning the i-th instance to the k-th cluster. We constrain the domain of μ as $\Delta = \{\mu | \forall i, \sum_k \mu_{i,k} = 1, \forall i, k, \mu_{i,k} \in \{0, 1\}\}$. It implies that each instance will only be assigned to a single cluster. The objective for the proxy-based unsupervised representation learning can be written as

$$\min_{\mathbf{x},C,\mu\in\Delta} \sum_i \Big((K-1)\sum_{k=1}^K \mu_{i,k}\|\mathbf{x}_i - \mathbf{c}_k\|_2^2 - \sum_{q=1}^K (1-\mu_{i,q})\|\mathbf{x}_i - \mathbf{c}_q\|_2^2\Big) \qquad (2)$$

The coupled variables in Eqn. 2 make the optimization challenging. Hence, we can solve the problem in an alternating way. It should be noted that there are three groups of variables $\{\mathbf{x}, C, \mu\}$ and different decomposition can result in different algorithms.

We demonstrate a prevalent strategy that optimizes $\{\mathbf{x}\}$ and $\{\mu, C\}$ alternatively. When fixing assignment μ and centers C, the subproblem becomes

$$\min_{\mathbf{x}} \sum_i \Big(\sum_{q:q\neq\tilde{y}_i}^K \|\mathbf{x}_i - \mathbf{c}_{\tilde{y}_i}\|_2^2 - \|\mathbf{x}_i - \mathbf{c}_q\|_2^2 \Big)$$

where $\tilde{y}_i = \arg\max_k \mu_{i,k}$ is the pseudo label of the i-th instance. Given pseudo labels, it can be solved with a supervised method as for Eqn. 1, which is the *discrimination phase* in representation learning.

When fixing representations $\mathbf{x}$, the subproblem can be simplified due to the empirical observation that the distribution of learned representations on the unit hypersphere has a mean that is close to zero [26] as

$$\min_{C,\mu\in\Delta} \sum_i \sum_{k=1}^K \mu_{i,k}\|\mathbf{x}_i - \mathbf{c}_k\|_2^2 \qquad (3)$$

It is a standard k-means clustering problem as the *clustering phase* in representation learning. The analysis shows that decoupling clustering and discrimination [3, 4] is corresponding to an alternating solver for the objective in Eqn 2. In this work, we further decouple μ and C in Eqn. 3 for efficient online clustering.

3.2. Online Constrained K-Means

Since the clustering phase is more challenging, we address the problem in Eqn. 3 first. As indicated in [1], the original formulation may incur a trivial solution that most of instances go to the same cluster. To mitigate the problem, we adopt the constrained k-means [2] instead, that is, controlling the minimal size of clusters to avoid collapsing.

Given a set of N unlabeled data $\{\mathbf{x}_i\}$ and the number of clusters K, the objective for constrained k-means is

$$\min_{C,\mu\in\Delta} \sum_{i=1,k=1}^{i=N,k=K} \mu_{i,k}\|\mathbf{x}_i - \mathbf{c}_k\|_2^2 \quad s.t. \quad \forall k \quad \sum_{i=1}^N \mu_{i,k} \geq \gamma_k \qquad (4)$$

where γ_k is the lower-bound of cluster size for the k-the cluster. Consequently, the final objective for unsupervised representation learning becomes

$$\min_{\mathbf{x},C,\mu\in\Delta} \sum_i \Big((K-1)\sum_{k=1}^K \mu_{i,k}\|\mathbf{x}_i - \mathbf{c}_k\|_2^2 - \sum_{q=1}^K (1-\mu_{i,q})\|\mathbf{x}_i - \mathbf{c}_q\|_2^2\Big)$$
$$s.t. \quad \forall k \quad \sum_{i=1}^N \mu_{i,k} \geq \gamma_k \qquad (5)$$

The problem in Eqn. 4 can be solved in batch mode. However, neural networks are often optimized with stochastic gradient descent (SGD) that can access only a mini-batch of instances at each iteration. Therefore, we propose a novel online algorithm to handle the problem with a theoretical guarantee as follows.

3.2.1 Online Assignment

We consider the alternating solver for the problem in Eqn. 4. When C is fixed, the problem for updating μ can be simplified as an assignment problem

$$\max_{\mu \in \Delta'} \sum_i \sum_k s_{i,k}\mu_{i,k} \quad s.t. \quad \forall k \quad \sum_{i=1}^{n} \mu_{i,k} \geq \gamma_k \tag{6}$$

where the value of μ can be relaxed from the discrete space to the continuous space as $\mu_{i,k} \in [0, 1]$ and $s_{i,k}$ is the similarity between the i-th instance and the k-th cluster. In this work we assume that $\mathbf{x}$ and $\mathbf{c}$ have unit norm and thus $s_{i,k} = \mathbf{x}_i^\top \mathbf{c}_k$.

Let μ^* denote the optimal solution for the problem in Eqn. 6. The standard metric for online learning is

$$\mathcal{R}(\mu) = \sum_i \sum_k s_{i,k}\mu^*_{i,k} - \sum_i \sum_k s_{i,k}\mu_{i,k}$$
$$\mathcal{V}(\mu) = \max_k \{\gamma_k - \sum_i \mu_{i,k}\}$$

where $\mathcal{R}(\mu)$ and $\mathcal{V}(\mu)$ denote regret and violation accumulated over N instances, respectively. Since μ^* can be a solution with continuous values, the regret with μ^* is no less than that defined with a discrete assignment. Consequently, the performance to the optimal integer solution can be guaranteed if we can bound this regret well.

To solve the problem in Eqn. 6, we first introduce a dual variable ρ_k for each constraint $\sum_i \mu_{i,k} \geq \gamma_k$. To be consistent with the training scheme in deep learning, we assume that each instance arrives in a stochastic order. When the i-th instance arrives at the current iteration, the assignment can be obtained by solving the problem

$$\max_{\mu_i \in \Delta'} \sum_k s_{i,k}\mu_{i,k} + \sum_k \rho_k^{i-1}\mu_{i,k} \tag{7}$$

where $\{\rho_k^{i-1}\}$ are the dual variables from the last iteration and $\mu_i = [\mu_{i,1}, \ldots, \mu_{i,K}]$. The problem in Eqn. 7 has a closed-form solution as

$$\mu_{i,k} = \begin{cases} 1 & k = \arg\max_k s_{i,k} + \rho_k^{i-1} \\ 0 & o.w. \end{cases} \tag{8}$$

Note that the domain for the assignment is a continuous space, but our solution implies an integer assignment. Besides, dual variables control the violation over the cluster size constraints. The method degrades to a greedy strategy without the dual variables.

After assignment, dual variables will be updated as

$$\rho^i = \Pi_{\Delta_\tau}(\rho^{i-1} - \eta(\mu_i - \frac{[\gamma_1, \ldots, \gamma_K]}{N}))$$

where Π_{Δ_τ} projects the dual variables to the domain $\Delta_\tau = \{\rho | \forall k, \rho_k \geq 0, \|\rho\|_1 \leq \tau\}$. The performance of online assignment algorithm can be guaranteed in Theorem 1. Complete proofs can be found in the supplementary.

Theorem 1. *If instances arrive in the stochastic order, by setting $\eta = \tau/\sqrt{2N}$, we have*

$$E[\mathcal{R}(\mu)] \leq \mathcal{O}(\sqrt{N}), \quad E[\mathcal{V}(\mu)] \leq \mathcal{O}(\sqrt{N})$$

Remark Theorem 1 indicates that compared with the optimal solution consisting of continuous assignment, the regret of our method with integer assignment can be well bounded. Besides, the violation is also bounded by $\mathcal{O}(\sqrt{N})$ for the constraints accumulated over all instances. It illustrates that for each assignment, the gap to optimum can be bounded by $\mathcal{O}(1/\sqrt{N})$ and our assignment method can achieve a near-optimal result even running online. Moreover, the theorem implies that the violation can be avoided by increasing γ_k with a small factor.

For training with SGD, a mini-batch of instances rather than a single instance will arrive at each iteration. If the size of the mini-batch is b, we will assign pseudo labels for each instance with the closed-form solution in Eqn. 8. The dual variables will be updated with the averaged gradient as

$$\rho^i = \Pi_{\Delta_\tau}(\rho^{i-1} - \eta\frac{1}{b}\sum_{s=1}^{b}(\mu_i^s - \frac{[\gamma_1, \ldots, \gamma_K]}{N})) \tag{9}$$

3.2.2 Online Clustering

With the proposed online assignment, we can update the assignment and centers for constrained k-means in an online manner. Concretely, for the t-th epoch, we first fix C^{t-1} and assign pseudo labels for each mini-batch of instances. After training with an epoch of instances, the centers can be updated as

$$\mathbf{c}_k^t = \Pi_{\|\mathbf{c}\|_2=1}(\frac{\sum_i^N \mu_{i,k}^t \mathbf{x}_i^t}{\sum_i^N \mu_{i,k}^t}) \tag{10}$$

where μ^t is the assignment at the t-th epoch and $\mathbf{x}_i^t$ denotes a single view of the i-th instance at the t-th epoch.

Since our method does not memorize representations of instances, the variables in constrained k-means, especially centers, will only be updated once with an epoch of instances. However, k-means requires multiple iterations to converge as a batch mode method. Fortunately, clustering each epoch of data to optimum is not necessary for representation learning. According to the objective in Eqn. 5, we can further decompose μ and C. When fixing $\mathbf{x}^t$ and C^{t-1}, the assignment can be updated by the proposed online assignment method. When fixing $\mathbf{x}^t$ and μ^t, centers have a closed-form solution as in Eqn. 10. Therefore, a single step of updating is applicable for optimizing the target objective and the cost of clustering can be mitigated. Intuitively, representations are improved with more epochs of training while the clustering is gradually optimized simultaneously.

Furthermore, inspired by mini-batch k-means [25], we can update the centers aggressively to accelerate the convergence of clustering process. Concretely, centers can be updated after each mini-batch as

$$\mathbf{c}_{k:m}^t = \Pi_{\|\mathbf{c}\|_2=1}\left(\frac{\sum_i^m \mu_{i,k}^t \mathbf{x}_i^t}{\sum_i^m \mu_{i,k}^t}\right) \tag{11}$$

where m denotes the total number of received instances in the t-th epoch. After a sufficient training, we may switch to update centers only once in each epoch to reduce the variance from a mini-batch.

3.3. Discrimination

With pseudo labels and centers obtained from the $(t-1)$-th epoch, we can learn representations by optimizing a standard normalized Softmax loss for instances at the t-th iteration as

$$\ell_{\text{cls}}(\mathbf{x}_i^t) = -\log\left(\frac{\exp(\mathbf{x}_i^{t\top}\mathbf{c}_{\tilde{y}_i^{t-1}}^{t-1}/\lambda)}{\sum_{k=1}^K \exp(\mathbf{x}_i^{t\top}\mathbf{c}_k^{t-1}/\lambda)}\right) \tag{12}$$

where $\tilde{y}_i^{t-1}$ is the pseudo label implied by μ^{t-1} and λ is the temperature. Since μ_i is a one-hot vector, we can keep a single label for each instance in the memory, where the storage cost is negligible. $\mathbf{x}_i$ and $\mathbf{c}_k$ have the unit norm. By decoupling clustering and discrimination, our method can optimize the objective in Eqn. 5 effectively in an alternating way. To initialize the pseudo labels and centers for representation learning, we scan one epoch of instances without training the model to obtain μ^0 and C^0.

Finally, we show that our method converges.

Corollary 1. *The proposed method will converge if keeping μ^{t-1} when μ^t provides no loss reduction.*

Although the theory requires to check the optimality of μ^t, we empirically observe that CoKe works well with the vanilla implementation.

3.4. Variance Reduction for Robust Clustering

Variance from different views of each instance provides essential information for representation learning. However, it may perturb the clustering and make the optimization slow. Therefore, we propose two strategies to reduce the variance incurred to the assignment step.

Moving Average Ensemble is an effective way to reduce variance. Therefore, we propose to accumulate clustering results from the second stage. Concretely, for $t > T'$, assignment and centers will be updated as

$$\hat{C}^t = (1-\tfrac{1}{t-T'})\hat{C}^{t-1} + \tfrac{1}{t-T'}C^t;$$
$$\hat{\mathbf{y}}^t = (1-\tfrac{1}{t-T'})\hat{\mathbf{y}}^{t-1} + \tfrac{1}{t-T'}\tilde{\mathbf{y}}^t$$

where C^t and $\tilde{y}^t$ are obtained at the t-th epoch and $\tilde{\mathbf{y}}^t$ denotes the one-hot vector of $\tilde{y}^t$. The formulation averages the clustering results from the last $T-T'$ epochs to reduce the variance from augmentations. Unlike $\tilde{\mathbf{y}}_i$, $\hat{\mathbf{y}}_i$ is not a one-hot vector due to ensemble and can contain multiple non-zero terms. We adopt the loss defined with soft labels as

$$\ell_{\text{cls}}^{\text{soft}}(\mathbf{x}_i^t) = -\sum_k \hat{\mathbf{y}}_{i,k}^{t-1}\log\left(\frac{\exp(\mathbf{x}_i^{t\top}\tilde{\mathbf{c}}_k^{t-1}/\lambda)}{\sum_{j=1}^K \exp(\mathbf{x}_i^{t\top}\tilde{\mathbf{c}}_j^{t-1}/\lambda)}\right)$$

Two Views Learning representations with two views from the same image is prevalent in contrastive learning. Our proposed method can be considered as leveraging two views from different epochs and thus a single view is sufficient for each epoch. Nevertheless, CoKe can be further improved by accessing two views at each iteration.

Given two views of an image, the constraint for assignment is that both views share the same label. Therefore, the assignment problem in Eqn. 7 becomes

$$\max_{\mu_i\in\Delta'} \frac{1}{2}\sum_k \mu_{i,k}\sum_{j=1}^2 s_{i,k}^j + \sum_k \rho_k^{i-1}\mu_{i,k}$$

where $s_{i,k}^j$ denotes the similarity between the j-th view of the i-th instance and the k-th center. Hence, it is equivalent to obtaining a label for the mean vector averaged over two views as

$$\mu_{i,k} = \begin{cases} 1 & k = \arg\max_k \frac{1}{2}\sum_{j=1}^2 s_{i,k}^j + \rho_k^{i-1} \\ 0 & o.w. \end{cases} \tag{13}$$

Then, the loss in Eqn. 12 will be averaged over two views. Compared with the single view, multiple views can reduce the variance from different augmentations and make the assignment more stable.

Besides variance reduction for the one-hot assignment, the other advantage with the additional view is that it can provide a reference label distribution for the other view. Let $p_{i:j}$ denote the predicted probability over labels

$$p_{i:j,q}^{t-1} = \frac{\exp(\mathbf{x}_i^{j\top}\mathbf{c}_q^{t-1}/\lambda)}{\sum_{k=1}^K \exp(\mathbf{x}_i^{j\top}\mathbf{c}_k^{t-1}/\lambda)}$$

We can obtain the soft label for view 1 with the reference from view 2 as

$$\hat{\mathbf{y}}_{i:1}^t = \alpha\tilde{\mathbf{y}}_i^{t-1} + (1-\alpha)\mathbf{p}_{i:2}^{t-1}$$

Then, the cross entropy loss for view 1 can be optimized with $\hat{\mathbf{y}}_{i:1}^t$ instead. Alg. 1 summarizes the pseudo-code of CoKe with two views, which can be extended to multiple views easily.

Algorithm 1 Pseudo-code of CoKe with Two Views.

```
# f: encoder network for input images
# u: pseudo one-hot labels (Nx1)
# C: cluster centers
# rho: dual variable for constraints (Kx1)
# gamma: lower-bound of cluster size
# lambda: temperature
# alpha: ratio between labels

for z in loader: # load a minibatch with b samples
   z_1, z_2 = aug(z), aug(z) # two random views from z
   x_1, x_2 = f(z_1), f(z_2) # encoder representations
   s_1, s_2 = x_1C, x_2C # logits over centers
   y = u(z_id) # retrieve label from last epoch
   # compute reference distribution for each view
   p_1 = softmax(s_1/lambda)
   p_2 = softmax(s_2/lambda)
   # obtain soft label for discrimination
   y_1 = alpha*y + (1-alpha)*p_2
   y_2 = alpha*y + (1-alpha)*p_1
   # loss over two views
   loss = 0.5*(-y_1*log(p_1) -y_2*log(p_2))
   loss.backward() # update encoder
   # update clustering
   x_mean = 0.5*(x_1+x_2) # mean vector of two views
   u(z_id) = update(x_mean, C, rho) # as in Eqn. 13
   C = update(C, x_mean, u(z_id)) # as in Eqn. 11
   rho = update(rho, gamma, u(z_id)) # as in Eqn. 14
```

4. Experiments

We conduct experiments of unsupervised representation learning on ImageNet [24] to evaluate the proposed method. For fair comparison, we follow settings in benchmark methods [4,6,8]. More details can be found in the supplementary.

For the parameters in CoKe, we set the learning rate as 1.6 and temperature $\lambda = 0.1$. Besides the learning rate for model, CoKe contains another learning rate η for updating dual variables as in Eqn. 9. We empirically observe that it is insensitive and set $\eta = 20$. Finally, the batch size is $1,024$ such that all experiments of CoKe except the one with multi-crop can be implemented on a standard server with 8 GPUs and 16G memory on each GPU.

An important parameter in CoKe is the minimal cluster size. To reduce the number of parameters, we assign the same constraint for different clusters as $\gamma_1 = \cdots = \gamma_K = \gamma$. Considering that $\gamma = N/K$ denotes the balanced clustering, we introduce a parameter γ' as $\gamma = \gamma' N/K$ and tune γ' in lieu of γ for better illustration. In the experiments, we observe that the maximal value of dual variables is well bounded, so we simplify the updating criterion for dual variables as

$$\rho_k^i = \max\{0, \rho_k^{i-1} - \eta \frac{1}{b} \sum_{s=1}^{b} (\mu_{i,k}^s - \frac{\gamma'}{K})\} \tag{14}$$

4.1. Ablation Study

First, we empirically study the effect of each component in CoKe. All experiments in this subsection train 200 epochs and each instance has a single view of augmentation at each iteration. After obtaining the model, the learned representations are evaluated by learning a linear classifier on ImageNet. The training protocol for linear classifier follows that in MoCo [15] except that we change the weight decay to 10^{-6} and learning rate to 1 for our pre-trained model.

4.1.1 Balanced vs. Constrained Clustering

In the previous work [1, 4], balanced clustering that constrains each cluster to have the same number of instances demonstrates a good performance for representation learning. Constrained clustering that lower-bounds the size of each cluster is a more generic setting, but has been less investigated. With the proposed method, we compare constrained clustering to balanced clustering in Table 1.

Ratio: γ'	Acc%	#Cons	#Min	#Max
1	63.1	427	403	445
0.8	63.8	342	338	1,301
0.6	64.3	256	254	1,404
0.4	64.5	171	168	2,371
0	41.3	0	0	$449k$

Table 1. Comparison of different ratios γ' in CoKe. The performance is evaluated by linear classification with learned representations on ImageNet as in MoCo [15].

We fix the number of centers as $K = 3,000$ while varying γ' to evaluate the effect of cluster size constraint. When $\gamma' = 1$, each cluster has to contain N/K instances that becomes the balanced clustering. We let "#Cons", "#Min", "#Max" denote the constrained cluster size, the actual size of the smallest cluster and that of the largest cluster from the last epoch of CoKe, respectively. As illustrated in Table 1, the balanced clustering can achieve 63.1% accuracy when training with a single view. It confirms that balanced clustering is effective for learning representations. If decreasing the ratio, each cluster can have a different number of instances that is more flexible to capture the inherent data structure. For example, when $\gamma' = 0.8$, the minimum size of clusters is reduced from 403 to 338 while the largest cluster has more than double of instances in balanced clustering. Meanwhile, the imbalanced partition helps to improve the accuracy by 0.7%. With an even smaller ratio of 0.4, our method surpasses the balanced clustering with a significant margin of 1.4% and it demonstrates that constrained clustering is more appropriate for unsupervised representation learning. The performance will degrade when $\gamma' = 0$ since it may incur the collapsing problem without a sufficient number of instances in each cluster. We will fix $\gamma' = 0.4$ in the following experiments.

Besides the accuracy on linear classification, we further investigate the violation of constraints in Table 1. For balanced clustering, each cluster has the same number of instances which is a strong constraint. Compared to the constraint, the violation of our online assignment is only 5%

when $\gamma' = 1$. If γ' is less than 1, the constraint is relaxed and the violation can be reduced to less than 1%, which illustrates the effectiveness of our method. Compared with the online assignment strategy that only optimizes the constraints over a small subset of data in SwAV [4], we optimize the assignment globally and can explore the distribution of data sufficiently. Interestingly, we find that there is no dominating cluster even when $\gamma' = 0.4$. In that scenario, the largest cluster only contains $2,371$ instances. It illustrates that clustering is effective to learn an appropriate partition for unlabeled data. If $\gamma' = 0$, more than $449,000$ instances will be assigned to the same cluster, which confirms the importance of cluster size constraint to mitigate the collapsing problem.

4.1.2 Coupled Clustering and Discrimination

Then, we study the effect of coupling clustering and discrimination. In CoKe, we decouple clustering and discrimination by collecting clustering results from the last epoch to discriminate data from the current epoch. Table 2 compares the performance with different labels and centers where $\{C^{t-1}, \tilde{y}^{t-1}\}$ and $\{C^t, \tilde{y}^t\}$ are from the last epoch and the current epoch, respectively.

Settings	$\{C^{t-1}, \tilde{y}^{t-1}\}$	$\{C^{t-1}, \tilde{y}^t\}$	$\{C^t, \tilde{y}^{t-1}\}$	$\{C^t, \tilde{y}^t\}$
Acc%	64.5	0.4	51.2	0.1

Table 2. Comparison of labels and centers from different epochs.

First, we can observe that with labels and centers from the last epoch, CoKe demonstrates the best performance. It verifies that CoKe solves the problem in Eqn. 5 effectively in an alternating way. Second, with current centers C^t, the performance decreases more than 10%, which shows the importance of keeping centers from the last epoch in CoKe. Finally, the other two variants with $\tilde{y}^t$ fail to learn meaningful representations. It is consistent with our analysis for the objective in Eqn. 5. Note that μ is the additional variables introduced by unsupervised learning and decoupling $\mathbf{x}$ and μ is crucial for clustering-based representation learning.

4.1.3 Number of Clusters

The number of clusters is a key parameter in k-means. When K is small, the relationship between similar instances may not be exploited sufficiently. However, additional noise can be introduced with a large K. Instance classification can be considered as a special case when $K = N$. Table 3 summarizes the performance with different K's. We observe that CoKe with $1,000$ clusters is about 1% worse than that with $K = 3,000$. It is because that a small K is hard to capture all informative patterns due to a coarse granularity.

However, obtaining an appropriate K for clustering is a challenging problem in k-means. Moreover, clustering can provide different results even with the same representations, which is researched in multi-clustering [16, 17]. This phenomenon is due to the fact that objects can be similar in different ways (e.g., color, shape, etc.). Multi-clustering has been explored in previous representation learning work [1, 18] and we also apply it to learn representations with a multi-task framework. Each task is defined as a constrained k-means problem with a different K, while the final loss will be averaged over multiple tasks. This strategy mitigates the parameter setting problem in k-means by handling multiple k-means problems with diverse parameters simultaneously.

K	Acc%	#Cons	#Min	#Max
1,000	63.4	512	512	4,639
3,000	64.5	171	168	2,371
5,000	64.3	102	98	1,982

Table 3. Comparison of number of clusters K in k-means.

$K(\times 1,000)$	3	2+3	3+4	3+3+3	3+4+5
Acc%	64.5	65.0	65.2	65.2	65.3

Table 4. Multi-clustering with different K combinations.

Table 4 shows the results of learned representations with multi-clustering. When including a task with $K = 2,000$, the accuracy is improved from 64.5% to 65.0%. With a more fine-grained task of $K = 4,000$, the performance of learned representations is even better and achieves 65.2% in accuracy. Then, we evaluate the triple k-means task with two different settings, that is, same K and different K's. It can be observed that different K's can further improve the performance. We will adopt the strategy in rest experiments for the explicit multi-clustering. More ablation study can be found in the supplementary.

4.2. Comparison with State-of-the-Art on ImageNet

In this subsection, we compare our proposal with state-of-the-art methods by learning a linear classifier on learned representations for ImageNet. All methods have ResNet-50 as the backbone. The results of methods with similar configurations (e.g., 2-layer MLP, 128-dimensional representations, etc.) are summarized in Table 5.

Explicitly, baseline methods have to learn representations with two views of augmentations from an individual instance at each iteration for the desired performance. On the contrary, CoKe can work with a single view using online optimization. It can be observed that the accuracy of representations learned by CoKe with 800 epochs can achieve 71.4%, which performs slightly better than MoCo-v2 but with only a half number of views for optimization. It illustrates that leveraging relations between in-

Methods	#View	#Epoch	#Dim	Acc%
SimCLR	2	1,000	128	69.3
MoCo-v2	2	800	128	71.1
DeepCluster-v2	2	400	128	70.2
SwAV	2	400	128	70.1
CoKe	1	800	128	**71.4**

Table 5. Comparison with methods that have the similar configuration on ImageNet by linear classification.

stances can learn more informative patterns than instance discrimination. Second, compared to the clustering-based methods, CoKe outperforms SwAV and DeepCluster by 1% when training with the same number of views. This further demonstrates the effectiveness of CoKe. Finally, we compare the running time for training one epoch of data in Table 6. With a single view for optimization, the learning efficiency can be significantly improved as in CoKe.

MoCo-v2	SwAV	CoKe	CoKe*
18.3	20.8	11.1	8.4

Table 6. Comparison of running time (mins) for training one epoch of data on ImageNet. All methods are evaluated on the same server. CoKe* applies automatic mixed precision training provided by PyTorch.

Then, we apply more sophisticated settings proposed by recent methods [7, 14] for CoKe and compare with methods using different settings. Concretely, we include 3-layer MLP, an additional 2-layer prediction head and $1,000$ epochs for training. More details can be found in the supplementary. Table 7 summarizes the comparison.

Methods	#V	Bs	#D	ME	MB	Acc%
SimSiam [9]	2	256	2,048	✓		71.3
SwAV [9]	2	4,096	128			71.8
MoCo-v2+ [9]	2	256	128	✓	✓	72.2
Barlow Twins [29]	2	2,048	8,192			73.2
MoCo-v3 [10]	2	4,096	256	✓		73.8
BYOL [14]	2	4,096	256	✓		74.3
NNCLR [13]	2	1,024	256	✓	✓	72.9
NNCLR [13]	2	4,096	256	✓	✓	75.4
DeepCluster-v2 [4]	8	4,096	128			75.2
SwAV [4]	8	4,096	128			75.3
DINO [5]	8	4,096	256	✓		75.3
NNCLR [13]	8	4,096	256	✓	✓	75.6
CoKe	1	1,024	128			72.5
CoKe	2	1,024	128			74.9
CoKe	8	1,024	128			**76.4**

Table 7. Comparison with state-of-the-art methods on ImageNet by linear classification. ME and MB denote momentum encoder and memory bank, respectively.

First, we can observe that CoKe with single view performs slightly better than MoCo-v2 again and it demonstrates that optimizing with single view is able to obtain an applicable pre-trained model. Second, by equipping with two views, CoKe can achieve 74.9% accuracy on ImageNet, which is a competitive result but with much lighter computational cost. Furthermore, the superior performance of NNCLR and CoKe shows that capturing relations between instances can learn better representations. However, NNCLR has to obtain appropriate nearest neighbors with a large memory bank and is sensitive to the batch size. On the contrary, CoKe learns relationship by online clustering, which is feasible for small batch size and leads to a simple framework without memory bank and momentum encoder. Finally, with the standard multi-crop trick, CoKe can achieve 76.4% accuracy on ImageNet that is close to the supervised counterpart, i.e., 76.5%. In summary, CoKe is more resource friendly (e.g., a standard server with 8 GPUs is sufficient) with superb performance.

4.3. Comparison on Downstream Tasks

Besides linear classification on ImageNet, we evaluate CoKe on various downstream tasks in Table 8. Methods with public available pre-trained models are included for comparison. For a fair comparison, we search parameters for all baselines with the codebase from MoCo. Evidently, CoKe provides a better performance than the strong baselines with multi-crop training, which confirms the effectiveness of our method. Detailed empirical settings and additional experiments on clustering are in the supplementary.

	VOC	COCO		C10	C100
Methods	Ap_{50}	Ap^{bb}	Ap^{mk}	Acc%	Acc%
Supervised	81.3	38.9	35.4	97.3	86.6
MoCo-v2	<u>83.0</u>	39.6	35.9	97.9	86.1
Barlow Twins	81.5	40.1	36.9	98.0	87.4
BYOL	82.9	<u>40.5</u>	36.9	<u>98.1</u>	<u>87.9</u>
SwAV*	82.1	40.4	<u>37.1</u>	97.7	87.5
DINO*	82.0	40.2	36.8	97.7	87.6
CoKe	<u>83.2</u>	<u>40.9</u>	<u>37.2</u>	<u>98.2</u>	<u>88.2</u>

Table 8. Comparison on downstream tasks. * denotes the multi-crop training trick. Top-2 best models are underlined.

5. Conclusion

In this work, we propose a novel learning objective for cluster discrimination pretext task. An online constrained k-means method with theoretical guarantee is developed to obtain pseudo labels, which is more appropriate for stochastic training in representation learning. The empirical study shows that CoKe can learn effective representations with less computational cost by leveraging the aggregation information between similar instances. Recently, Transformer [11] shows the superior performance, evaluating CoKe on the new architecture can be our future work.

References

[1] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020. 1, 2, 3, 6, 7

[2] Paul S Bradley, Kristin P Bennett, and Ayhan Demiriz. Constrained k-means clustering. *Microsoft Research, Redmond*, 20(0):0, 2000. 3

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 1, 2, 3

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020. 1, 2, 3, 6, 7, 8

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *CoRR*, abs/2104.14294, 2021. 1, 3, 8

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, volume 119, pages 1597–1607, 2020. 1, 2, 6

[7] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020. 8

[8] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. 2, 6

[9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, pages 15750–15758. Computer Vision Foundation / IEEE, 2021. 1, 2, 8

[10] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *CoRR*, abs/2104.02057, 2021. 2, 8

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*. OpenReview.net, 2021. 8

[12] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(9):1734–1747, 2016. 1, 2

[13] 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. *CoRR*, abs/2104.14548, 2021. 3, 8

[14] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *NeurIPS*, 2020. 1, 2, 8

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9726–9735, 2020. 1, 2, 6

[16] Juhua Hu and Jian Pei. Subspace multi-clustering: a review. *Knowledge and information systems*, 56(2):257–284, 2018. 7

[17] Juhua Hu, Qi Qian, Jian Pei, Rong Jin, and Shenghuo Zhu. Finding multiple stable clusterings. *Knowl. Inf. Syst.*, 51(3):991–1021, 2017. 7

[18] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. *CoRR*, abs/2005.04966, 2020. 1, 7

[19] Yair Movshovitz-Attias, Alexander Toshev, Thomas K. Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *ICCV*, pages 360–368, 2017. 3

[20] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *ECCV*, volume 9910, pages 69–84, 2016. 1

[21] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016. 1

[22] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In *ICCV*, pages 6449–6457, 2019. 2, 3

[23] Qi Qian, Jiasheng Tang, Hao Li, Shenghuo Zhu, and Rong Jin. Large-scale distance metric learning with uncertainty. In *CVPR*, pages 8542–8550, 2018. 3

[24] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015. 6

[25] D. Sculley. Web-scale k-means clustering. In Michael Rappa, Paul Jones, Juliana Freire, and Soumen Chakrabarti, editors, *WWW*, pages 1177–1178, 2010. 5

[26] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, volume 119, pages 9929–9939, 2020. 3

[27] Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, 10:207–244, 2009. 3

[28] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, pages 3733–3742, 2018. 1, 2

[29] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In Marina Meila and Tong Zhang, editors, *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 2021. 2, 8

[30] Xiaohang Zhan, Jiahao Xie, Ziwei Liu, Yew-Soon Ong, and Chen Change Loy. Online deep clustering for unsupervised representation learning. In *CVPR*, 2020. 1, 2

[31] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *ICCV*, pages 6001–6011, 2019. 1