# Oracle-guided Contrastive Clustering

Mengdie Wang, Liyuan Shang, Suyun Zhao, Yiming Wang, Hong Chen, Cuiping Li, Xizhao Wang

## Abstract

Deep clustering aims to learn a clustering representation through deep architectures. Most of the existing methods usually conduct clustering with the unique goal of maximizing clustering performance, that ignores the personalized demand of clustering tasks.However, in real scenarios, oracles may tend to cluster unlabeled data by exploiting distinct criteria, such as distinct semantics (background, color, object, etc.), and then put forward personalized clustering tasks. To achieve task-aware clustering results, in this study, Oracle-guided Contrastive Clustering(OCC) is then proposed to cluster by interactively making pairwise "same-cluster" queries to oracles with distinctive demands. Specifically, inspired by active learning, some informative instance pairs are queried, and evaluated by oracles whether the pairs are in the same cluster according to their desired orientation. And then these queried same-cluster pairs extend the set of positive instance pairs for contrastive learning, guiding OCC to extract orientation-aware feature representation. Accordingly, the query results, guided by oracles with distinctive demands, may drive the OCC's clustering results in a desired orientation. Theoretically, the clustering risk in an active learning manner is given with a tighter upper bound, that guarantees active queries to oracles do mitigate the clustering risk. Experimentally, extensive results verify that OCC can cluster accurately along the specific orientation and it substantially outperforms the SOTA clustering methods as well. To the best of our knowledge, it is the first deep framework to perform personalized clustering.

## Introduction

Clustering, as one of the most fundamental unsupervised learning techniques[1], has been successively used in a wide range of applications, such as image processing[36], gene analysis[35] and text categories[37]. Recently, by employing highly non-linear latent representations[7], Deep Clustering (DC) is widely studied and achieves promising clustering results[24, 8, 10, 21, 11]. Typically, these existing clustering and DC techniques share a common and unique goal, to maximally enhance the clustering performance. However, the personalized demand of clustering tasks is mostly ignored and dismissed.

In the real applications, there are more than one available cluster demands. Fig. 1 shows an example of the diversity of clustering orientations. Some tasks require clustering in terms of objects, so the oracle indicates that duck and fox belong to the same cluster, categorized as animals. Some other tasks, however, demand distinguishing backgrounds, the oracle then indicates duck and sailboat belong to the same cluster, categorized as river. In such cases with personalized demands, the existing clustering techniques, clustering with default orientation, may decline or even be unworkable without the guide of oracles. Accordingly, it is still a challenging problem to cluster along a desired orientation.

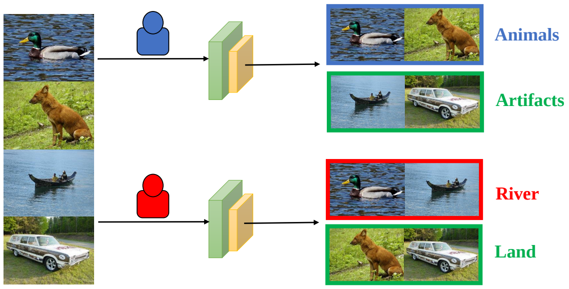

Figure 1: The diversity of clustering orientation. The two oracles make different judgments on the task of dividing the four images into two clusters.

.

To solve this problem, in this study, we propose an Oracle-guided Contrastive Clustering(OCC) model to achieve task-aware clustering results in the guide of oracles with distinctive demands. As active learning, effectively and interactively engaging human for improving annotating performance, benefits us for clustering in the desired orientation, it is exploited to actively select informative instance pairs for oracle to provide answers of same-cluster queries in the forms of yes or no. Thus, the distinctive demand of oracles is embedded in clustering with "queries and answers" in the loop. To catch orientation-aware feature information in deep clustering, the queried same-cluster pairs are regarded as extensions of the positive instance pairs in the process of contrastive learning, prompting the network to learn shared features between same-cluster pairs. Consequently, with the learned orientation-aware features, the clustering solution is achieved along the personalized demands.

Specifically, OCC first constructs the initial positive instance pairs via random data augmentations. Then some instance pairs are selected for annotation according to their similarity and intensity of variation. Subsequently, the same-cluster sample pairs annotated by the personalized oracle extend the positive instance pairs in the contrastive loss. After that, the feature output by the backbone network is projected into the representation space and the assignment space, wherein OCC learns shared features between the positive instance pairs by active contrastive losses. The features are orientation-aware, and thus personalized clustering is achieved in the desired orientation. Our major contributions can be summarized as follows:

- We propose a novel model, named OCC, that exploits active learning joint with contrastive learning to catch the desired feature, and to guide the orientation of clustering according to the personalized demand.
- To the best of our knowledge, it is the first work that concerns the diversity of clustering orientation in deep clustering. Unlike the existing clustering methods, the proposed OCC is bi-objective, that is, to maximize the clustering performance and to accurately cluster in a specific orientation.
- By strict theoretical analysis, a tighter upper bound, that can be mitigated by active query with a specific orientation, is given for the clustering risk in an active learning manner. Simultaneously, extensive experiments demonstrate that OCC can learn clustering features in a targeted manner.

# Related Work

## Deep Active learing

Active learning[12] aims to query the optimal samples in the unlabeled dataset to reduce the cost of labeling as much as possible while still maintaining performance. The most common query strategies are uncertainty-based approaches, which select samples with high information content to decrease labeling costs. For example, ENS-VarR[13] uses Monte-Carlo dropout and deep ensembles to obtain well-behaved uncertainty estimates from deep neural networks. Ranganathan et al. [14] make efforts to integrate an active learning based criterion in the loss function used to train a deep belief network. Density-based approaches have also been applied to CNNs. Core-Set[17] chooses several scattered center points to minimize the max distance between a data point and its nearest center. TOD[18] is a task-agnostic approach based on the observation that the samples with higher loss are usually more informative to the model than that with lower loss. The method propose an effective loss estimator Temporal Output Discrepancy to query samples with higher loss.

## Active Clustering

Active learning is also widely used in the field of clustering[38]. Dasgupta and Hsu[39] first proposed the idea of guided sampling by querying samples based on the results of hierarchical clustering. ALEC[40] select representative samples drawn on the structure of the data.

Although a number of active methods query the label of a single sample, studies of active clustering prefer pairwise queries[42]. Xiong et al.[41] proposed an active spectral clustering algorithm with k-nearest neighbor graphs, selecting pairwise constraints based on node uncertainty. Ashtiani et al. [19] introduce a semi-supervised active clustering (SSAC) framework asking whether two given instances belong to the same cluster or not and demonstrate that access to simple query answers can turn an otherwise NP-hard clustering problem into a feasible one. Dasgupta and Ng[20] poses the problem that clustering algorithms only group documents along the most prominent dimension without knowing the user's intention, which is similar to our problem of diversity of image clustering criteria. It proposes an active spectral clustering algorithm, which makes it easy for a user to specify the dimension along which she wants to cluster the data points is sentiment.

## Deep Clustering

Deep neural networks are explored to improve clustering performance due to their ability to learn representations on complex high-dimensional datasets[22, 23]. Recent works focus on end-to-end methods to transform the data into clustering-oriented representations. For example, DEC[24] optimizes the cluster centers and embedded features simultaneously by minimizing the KL-divergence for features in the latent subspace. DCN[9] scatter samples in the low-dimensional space around their corresponding cluster centroids to learn a K-means friendly representation. IDFD[31] is a spectral clustering friendly representation learning by reducing correlations within features.

Self-augmentation based methods also achieve good performance. IIC[25] maximizes the mutual information between positive instance pairs to discover clusters. PICA[26] clusters by minimizing the cosine similarity between the cluster-wise assignment vectors to learn the most semantically plausible clustering solution. DCCM[27] introduces the augmentation and utilizes the correlations among representations. Inspired by the above ideas, CC[10] proposes a dual contrastive learning framework. This method is based on the observation of "label as representation", conducting contrastive learning at not only the instance-level but also the cluster-level to learn clustering-favorite representations. GCC[21] selects the positive pairs and negative pairs by the KNN graph constructed on the instance representation.

These methods have achieved excellent results on large datasets in default orientation but cannot cluster personalized. Our approach introduces active learning into deep clustering, where oracles guide the network to learn cluster-oriented features.

# Method

## Proposed model: OCC

We propose a novel model, named OCC, by exploiting active learning to guide the network to cluster in a given orientation and to improve clustering performance as well. One active cycle of OCC is illustrated in Fig. 2. Fig. 2 depicts that active queries to oracles are embedded in the loop of OCC. In terms

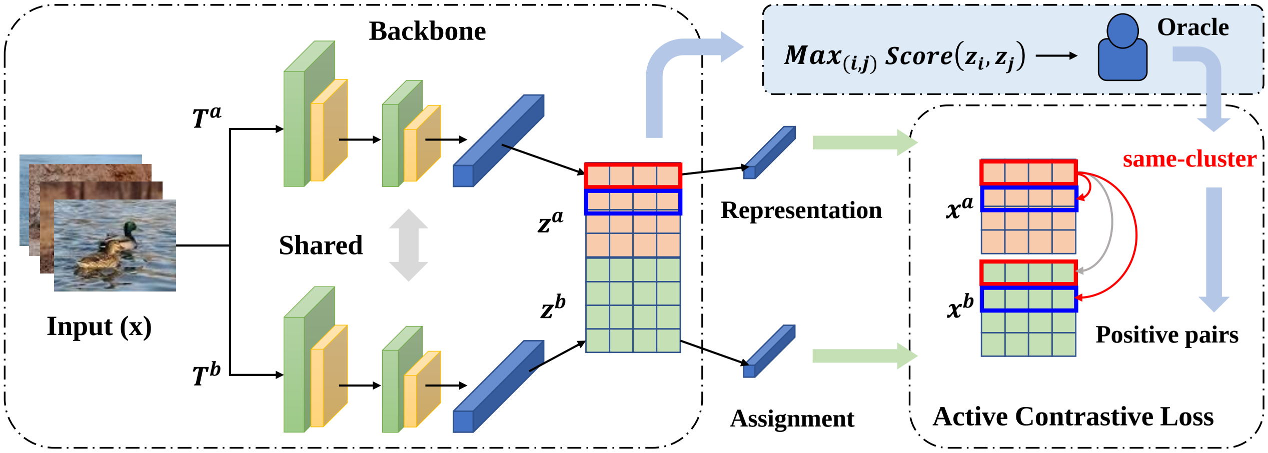

Figure 2: One active cycle of Oracle-guided Contrastive Clustering. A shared deep neural network generates representations from two random augmentations of the data, projected into representation space and assignment space. We explore a scoring function to measure the informativeness of each sample pair and select the largest one to submit a query to the oracle. The obtained same-cluster pairs are leveraged to extend the positive instance pairs in the contrastive loss.

of the designed scoring function on the embedding features, some instance pairs are actively selected for annotation, and they are judged whether belonging to the same cluster in the forms of yes or no. Simultaneously, to learn cluster-friendly features and to compact cluster assignments, respectively, OCC projects the embedding features into representation and assignment spaces. Finally, these two spaces, augmented with annotated instance pairs, are optimized with bi-objectives: better clustering performance along a given orientation by leveraging active contrastive loss.

In OCC, the contrastive loss is recontructed by extending the positive instance pairs with pairwise annotations given by the oracle. For example, given a data $x_i$, we first select two stochastic data transformations $T^a$, $T^b$ from the same family of augmentations $\mathcal{T}$ and apply them to the data. Therefore, the augmented samples $x_i^a = T^a(x_i)$ and $x_i^b = T^b(x_i)$ constitute the initial positive sample pair $(x_i^a, x_i^b)$. At this time, the oracle informs that the samples of $x_i$ and $x_j$ belong to the same cluster, then the positive instances related to $x_i^a$ include $x_i^b$, $x_j^a$, and $x_j^b$.

In the followed two subsections, we describe the construction of our contrastive loss and query strategy in detail.

## Active Contrastive Loss

Given a mini-batch size of $N$, let $X$ denotes the feature matrix of $2N$ augmented samples $\{x_1^a, ..., x_N^a, x_1^b, ..., x_N^b\}$ in the space. The same-cluster sample pairs given by the oracle is represented by a matrix $C_{N*N}$ and

$$C_{i,j} = \begin{cases} \lambda, & y_i = y_j \\ 0, & others \end{cases} \tag{1}$$

where $\lambda$ is a variable weight parameter that controls the weight between the initial positive pair and the queried positive pair. $C_{i,j}$ indicates if the oracle annotations $x_i$ and $x_j$ belong to the same cluster. We leverage cosine similarity to measure the similarity between $x_i$ and $x_j$, i.e.

$$sim(x_i, x_j) = \frac{(x_i)(x_j)^\top}{\|x_i\|\|x_j\|} \tag{2}$$

Let $s(x_i, x_j) = exp(sim(x_i, x_j)/\tau)$, where $\tau$ denotes a temperature parameter. For an augmented sample $x_i^a$, the relevant positive instances are $x_i^b$ and $x_j^a$, $x_j^b$ if $C_{i,j} > 0$. To narrow the distance between positive instances pairs and learn their similar features, we set the objective function to be the ratio of the similarity between positive pairs to the similarity between negative pairs. The active contrastive loss of $x_i^a$ is defined as

$$l_i^a = \frac{s(x_i^a, x_i^b) + \sum_{j=1}^N C_{i,j}[s(x_i^a, x_j^a) + s(x_i^a, x_j^b)]}{\sum_{j=1}^N (C_{i,j}+1) \sum_{j=1}^N [s(x_i^a, x_j^a) + s(x_i^a, x_j^b)]} \tag{3}$$

where $\sum_{j=0}^N (C_{i,j}+1)$ is a method of data normalization to control the range of the loss. In the denominator, we leverage all of the samples instead of the negative instances for the same reason. Although it seems to affect the similarity to all instances, including itself, it only reduces the distance between negative instances due to the increased distance between positive instances in the molecule.

The loss in the sample space $X$ is the sum of all sample losses, i.e.

$$\mathcal{L}(X, C) = \frac{1}{2N} \sum_{i=1}^N (-log(l_i^a) - log(l_i^b)) \tag{4}$$

The network extracts features $Z$ from the augmented samples and projects the features into the representation space $\hat{Z} \in \mathcal{R}^{2N*M}$ and assignment space $\hat{Y} \in \mathcal{R}^{2N*K}$, where $M$ is a preset feature dimension and $K$ is the number of clusters. $\hat{Z}_i$ is the $M$-dimensional feature of the augmented sample $x_i$ and $\hat{Y}_i$ is the assignment probability vector of the

Algorithm 1: Oracle-guided Contrastive Clustering

**Input**: Training dataset $\mathcal{X}$; Cluster number $K$; Training epochs $E$; Batch size $N$.
**Parameter**: Query times $Q$; Temperature parameter $\tau$; Hyper-parameters $\lambda$.
**Output**: The clustering result $\mathcal{C}$.

```
for $epoch = 1$ to $E$ do
    for a sampled mini-batch $\{x_i\}_{i=1}^{N}$ do
        Generating augmentations for the sampled images;
        Utilize the network to extract feature matrix $Z$;
        Select the sample pair with the highest score according to Eq. 6;
        The oracle indicates the sample pair and records it with Eq. 1;
        Project $Z$ onto the representation space to get $\hat{Z}$;
        Project $Z$ onto the assignment space to get $\hat{Y}$;
        Calculate the active contrastive loss $\mathcal{L}$ through Eq. 2–5;
        Update the network by minimizing $\mathcal{L}$;
    end for
end for
Calculate $\mathcal{C} = argmax(\hat{Y})$
return $\mathcal{C}$
```

clusters of $x_i$. In general, contrastive methods learn the similar features of positive instance pairs in the representation space and narrow the intra-cluster distance while expanding the inter-cluster distance in the assignment space. Here we also implement contrastive learning of instances in the assignment space though these two spaces are related to a certain extent. It will encourage the positive pairs to be allocated in the same cluster and achieve better performance in practice. The overall objective function is defined as

$$\mathcal{L} = \mathcal{L}(\hat{Z}, C) + \mathcal{L}(\hat{Y}, C) + \mathcal{L}(\hat{Y}^{\top}, O) - H(\hat{Y}) \qquad (5)$$

where $C$ is the query matrix defined in Eq. 1 so that $\mathcal{L}(\hat{Z}, C)+\mathcal{L}(\hat{Y}, C)$ denotes the contrastive loss for instances on representation space and assignment space. Matrix $O$ is a zero matrix and $\mathcal{L}(\hat{Y}^{\top}, O)$ denotes the contrastive loss for clusters on assignment space. According to Eq. 3, the active loss function degenerates into the initial contrastive loss if $C = O$. Positive pairs at the clustering level consist only of the same column of the cluster assignment matrix of the augmented samples, so the contrastive loss is not expanded. $H(\hat{Y}) = \sum_{i=1}^{K} P(\hat{y}_i) log P(\hat{y}_i)$ where $P(\hat{y}_i) = \sum_{j=1}^{2N} \hat{Y}_{j,i}/||\hat{Y}||_1$, which is leveraged to balance the number of samples in each cluster.

## Pairwise Query Strategy

In the pairwise query strategy, oracles judge whether the sample pairs belong to the same cluster to get pairwise constraints. Thus the network clusters along to the desired orientation and achieves better clustering performance. The scoring function Cyclic Similarity Discrepancy(CSD) of a sample pair $(x_i, x_j)$ is defined as

$$score(x_i, x_j) = s_c(x_i, x_j)|s_c(x_i, x_j) - s_{c-1}(x_i, x_j)| \quad (6)$$

where $s_c(x_i, x_j) = sim(x_i, x_j|c)$ denotes the similarity of sample pair $(x_i, x_j)$ in the $c-th$ iteration. The method is inspired by TOD[18], which argues that samples with the most significant change in features also have the highest true loss. We tend to select sample pairs that are likely to be similar and have a significant discrepancy in the similarity during the iterative process. Such samples have been shown to have large losses in TOD. In the following section, we will prove that selecting these samples helps to receive a tighter bound of clustering risk.

## Generalization bound

Clustering aims to divide the samples into several clusters such that samples lying in the same cluster have more similarities than those in others. We formally define the problem of clustering as minimizing the following criterion:

$$\mathbb{E}_{z \sim \mathcal{Z}} \left[ l\left(z; h\right) \right] = \mathbb{E}_{z \sim \mathcal{Z}} [1 - s(z_i, z_j; h)] \qquad (7)$$

where $Z$ is the population of same-cluster pairs, $s$ is a function measures the similarity of two samples in sample pair $z$. This criterion is expected clustering risk proposed by Liu et al.[33].

The optimization objective we define in Eq. 7 is not directly computable since we do not have access to all the information of sample pairs. In order to design an active learning strategy which is effective in pairwise query setting, we consider the following decomposition of expected clustering risk. This is a probabilistic sampling procedure inspired from Pydi et al.[34].

$$\begin{aligned} \mathbb{E}_{z \sim \mathcal{Z}} \left[ l\left(z; h_S\right) \right] \leq & \underbrace{\left| \, \mathbb{E}_{z \sim \mathcal{Z}} \left[ l\left(z, h_S\right) \right) \right] - \frac{1}{|T|} \sum_{z \in T} l\left(z, h_S\right) \right|}_{(A) \quad excess \quad clustering \quad risk} \\ + & \quad \underbrace{\frac{1}{|T|} \sum_{z \in T} \frac{Q_z}{p_z} l\left(z, h_S\right)}_{(B) \quad extended \quad clustering \quad risk} \\ + & \underbrace{\left| \frac{1}{|T|} \sum_{z \in T} l\left(z, h_S\right) - \frac{1}{|T|} \sum_{z \in T} \frac{Q_z}{p_z} l\left(z, h_S\right) \right|}_{(C) \quad active \quad clustering \quad risk} \end{aligned} \qquad (8)$$

where $T$ denotes the same-cluster pairs in our samples defined as target sample pairs, $p_z$ denotes the probabilities of sample pairs $z$ being queried, $Q_z$ denotes a set of Bernoulli random variables such that $P(Q_z = 1) = p_z$. Sample pairs $z$ will be queried when $Q_z = 1$.

Term (A) corresponds to the excess clustering risk of the algorithm $h_S$. It denotes the difference between expected clustering risk and empirical clustering risk. (B) corresponds to active clustering risk for $h_S$ over the sample pairs queried. This term could be minimized directly during the training process. (C) corresponds to active clustering risk. It is the absolute difference between the average clustering risk over all target sample pairs and the active clustering risk of the queried sample pairs.

According to Liu et al.[33], the excess clustering risk bounds in Eq. 8 are mostly of order $\mathcal{O}(\sqrt{K}/\sqrt{n})$ provided that the underlying distribution has bounded support. In a large data set, $n$ is much more large than $K$, term (A) could be small. Moreover, it is widely observed that the deep neural networks are highly expressive leading to very low training risk on selected sample pairs. Empirically, the active clustering risk is small. Hence, the critical part for active clustering are active clustering risk. The following theorem is presented to analyze its upper bound.

**Theorem 1.** *Define $D_p := \sum_{z\in T}\frac{l(z,h_S)}{p_z}$. Let $c_\delta > 0$ be a constant that depends on $\delta$. Active clustering risk can be bounded as follows, with probability at least $1-\delta$*

$$|\frac{1}{|T|}\sum_{z\in T} l\left(z,h_S\right) - \frac{1}{|T|}\sum_{z\in T}\frac{Q_z}{p_z} l\left(z,h_S\right)| \leq c_\delta \frac{D_p}{|T|} \tag{9}$$

Proof of Theorem 1. Define $H_z = l(z,h_S) - \frac{Q_z}{p_z} l\left(z,h_S\right)$, $M = \sum_{z\in T} E[H_z^2]$, we get the following bound on $M$

$$\begin{aligned} M &= \sum_{z\in T} Var[H_z^2] \\ &= \sum_{z\in T} l(z,h_S)^2(\frac{1}{p_z}-1) \\ &\leq \sum_{z\in T}\frac{l(z,h_S)^2}{p_z^2} = D_p \end{aligned} \tag{10}$$

We further use the Bernstein's inequality and conclude that with probability at least $1-\delta$

$$\begin{aligned} &|\frac{1}{|T|}\sum_{z\in T} l\left(z,h_S\right) - \frac{1}{|T|}\sum_{z\in T}\frac{Q_z}{p_z} l\left(z,h_S\right)| \\ &= \frac{1}{|T|}\sum_{z\in T} H_z \\ &\leq \frac{D_p}{3|T|}\log\frac{1}{\delta}\left(1+\sqrt{1+\frac{18}{\log\frac{1}{\delta}}}\right) \end{aligned} \tag{11}$$

□

Unlike the one proposed by Pydi et al. [34], our bound is tighter because we choose true loss instead of pseudo-loss. It is clear from Eq. 9 that choosing $p$ so as to minimize $D_p$ will result in the tightest bound for the expected clustering loss. In the next theorem, we present the optimal sampling probability distribution $p^*$ that minimizes $D_p$.

**Theorem 2.** *The optimal distribution $p^*$ for minimizing $D_p := \sum_{z\in T}\frac{l(z,h_S)}{p_z}$ is given by*

$$p_z^* = \frac{l(z,h_S)^{1/2}}{\sum_{z\in T} l(z,h_S)^{1/2}}$$

Proof of Theorem 2 is the same with proof of Theorem 6.3 in Pydi et al.[34].

| Dataset | Samples | Classes | Target Clusters |
|---|---|---|---|
| CIFAR-10 | 60,000 | 10 | 2 |
| CIFAR-100 | 60,000 | 100 | 4 |
| ImageNet-10 | 13,000 | 10 | 2 |
| ImageNet-Dogs | 19,500 | 15 | 2 |

Table 1: A summary of the datasets.

This theorem suggests that to minimize the clustering risk, it is workable to design a query strategy by selecting instance pairs with higher true loss. As the true losses of the instance pairs are distinct under various clustering criteria, the instance pairs with high cluster loss indicate that they are key instances that signify clustering tasks. Our pairwise query strategy is a CSD-based data sampling strategy, theoretically, instance pairs with large clustering risk can be obtained in the unlabeled pool, so as to minimize the expected clustering risk by active learning.

# Experiments

## Experimental Settings

**Datasets** To evaluate the effectiveness of the proposed method, we conduct experiments on four widely-used image datasets, including CIFAR-10, CIFAR-100[30], ImageNet-10 and ImageNet-Dogs[32]. We use the training and test sets of CIFAR-10 and CIFAR-100, and only make use of the training set of ImageNet-10 and ImageNet-Dogs.

Artificially, we divide the initial classes of the dataset into several target-clusters according to two distinct demands to simulate the diversity of clustering orientation. As depicted in Fig. 1, two distinct oracle demands cluster the images along distinct orientations. It is worth noting that not every demand matches the semantics of reality. Tab. 1 illustrates the details of the adopted datasets.

**Implementation Details** For fairness, ResNet34 is adopted as the backbone network without any modification. The parameters related to deep contrastive clustering are set following previous methods[10, 21]. Adam with an initial learning rate of 0.0003 is adopted to optimize the network and the batch size is set as 256. In addition, We resize all images uniformly to the size of $224 \times 224$. The feature dimensionality $M$ of the instances in representation space is set to 128.

Similar to the decay mechanism, the parameter $\lambda$ in Eq. 1 is set to $(1000-e)*0.05$ where $e$ is the current epoch num. Thus, the proportion of each positive pair gradually decreases as the number of queries increases. We query about $25\%$ of the instance pairs for each dataset and twice per batch. Moreover, after a number of iterations, we extend annotation by pseudo labeling the instances similar to annotated ones with a high confidence.

All comparative experiments are implemented with three NVIDIA TU102 RTX 2080 Ti GPUs on PyTorch platform.

**Compared Methods** We compared the proposed method with both traditional and deep learning based meth-

|  | CIFAR-10 |  |  | CIFAR-100 |  |  | ImageNet-10 |  |  | ImageNet-dogs |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Default** | NMI | ARI | ACC | NMI | ARI | ACC | NMI | ARI | ACC | NMI | ARI | ACC |
| K-means | 0.034 | 0.063 | 0.629 | 0.043 | 0.038 | 0.356 | 0.007 | 0.008 | 0.544 | 0.000 | -0.002 | 0.518 |
| PICA | 0.550 | 0.629 | 0.897 | 0.15 | 0.157 | 0.439 | 0.887 | 0.939 | 0.985 | 0.207 | 0.234 | 0.742 |
| GCC | 0.675 | 0.763 | 0.937 | 0.269 | 0.259 | 0.544 | 0.897 | 0.947 | 0.986 | 0.233 | 0.313 | 0.78 |
| CC | 0.602 | 0.646 | 0.902 | 0.369 | 0.385 | 0.701 | **0.924** | **0.963** | **0.991** | 0.179 | 0.221 | 0.735 |
| IDFD | 0.879 | 0.937 | 0.984 | 0.329 | 0.221 | 0.511 | 0.923 | 0.962 | 0.99 | 0.338 | 0.391 | 0.815 |
| OCC | **0.884** | **0.939** | **0.985** | **0.479** | **0.512** | **0.782** | 0.918 | 0.959 | 0.990 | **0.512** | **0.602** | **0.888** |

| **Personalized** | NMI | ARI | ACC | NMI | ARI | ACC | NMI | ARI | ACC | NMI | ARI | ACC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| K-means | 0.002 | 0.003 | 0.503 | 0.046 | 0.049 | 0.362 | 0.003 | 0.004 | 0.533 | 0.000 | -0.000 | 0.506 |
| PICA | 0.000 | 0.001 | 0.517 | 0.250 | 0.253 | 0.587 | 0.025 | 0.034 | 0.593 | 0.032 | 0.041 | 0.601 |
| GCC | 0.037 | 0.036 | 0.616 | 0.283 | 0.267 | 0.582 | 0.023 | 0.031 | 0.588 | 0.063 | 0.089 | 0.649 |
| CC | 0.006 | 0.009 | 0.546 | 0.214 | 0.202 | 0.485 | 0.025 | 0.035 | 0.594 | 0.035 | 0.047 | 0.609 |
| IDFD | 0.021 | 0.038 | 0.599 | 0.175 | 0.095 | 0.469 | 0.025 | 0.035 | 0.593 | 0.280 | 0.266 | 0.758 |
| OCC | **0.690** | **0.791** | **0.945** | **0.538** | **0.569** | **0.801** | **0.732** | **0.810** | **0.950** | **0.702** | **0.801** | **0.948** |

Table 2: The clustering performance under two clustering orientations, default and personalized, on four object image benchmarks. 'default' perform clustering along the default orientation, while 'personalized' along a given and personalized orientation. The best results are shown in boldface.

ods, including K-means[2], PICA[26], GCC[21], CC[10], IDFD[31]. We set the target number of clusters to the number of target clusters for all methods and run them in a unified environment. Moreover, two known active query strategies are compared with our adopted strategy in our designed deep framework.

**Evaluation Metrics** We adopted three standard clustering metrics to evaluate our method including Normalized Mutual Information (NMI), Accuracy (ACC), and Adjusted Rand Index (ARI). These metrics reflect the performance of clustering from different aspects, and higher values indicate better performance.

## Experimental Results

**Clustering Performance** We presented the clustering performance of OCC and the compared methods on four datasets with two distinct cluster orientations in Tab. 2. From Table 2, we observe the following facts. i) In the default clustering orientation,, the clustering performance of our OCC is obviously better than the comparison deep clustering algorithms in most. It is observed that on ImageNet-10, only IDFD slightly performs better than OCC. than . These shows that, without personalized clustering demands, our OCC can outperform the compared SOTA clustering methods. ii) In the personalized clustering orientation, the performance of the SOTA clustering methods is relatively low. This shows that the existing SOTA clustering methods are unworkable and not applicable for the clustering tasks with personalized demands. iii) In the personalized clustering orientation, the performance of OCC is high and extremely better than the compared SOTA clustering methods. This shows that unlike the existing methods, OCC is apt to clustering with personalized demands. it also shows that the active query is workable to guide clustering along a given orientation.

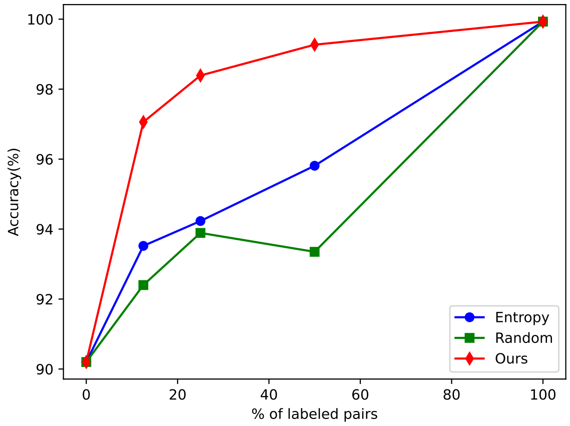

Figure 3: The accuracy of three query strategies available on CIFAR-10.

.

**Query efficiency** To verify the effectiveness of the query strategy, we compare our adopted query strategy with two known ones, including entropy-based query and random query, in our designed deep clustering framework. The random query strategy is to select instance pairs randomly. The entropy-based query strategy selects the instance with the maximum entropy and paired it with another medium similar one. The results of the comparison on CIFAR-10 are shown in Fig. 3. The percentage of queried ones to the total instance pairs (about 200,000 for CIFAR-10) is seen as the query cost. And it is headed as '% of labeled pairs' as the title of the vertical axis in Fig. 3 . Note that the results of unsupervised(0%) and full-supervised(100%) learning are independent from query strategy.

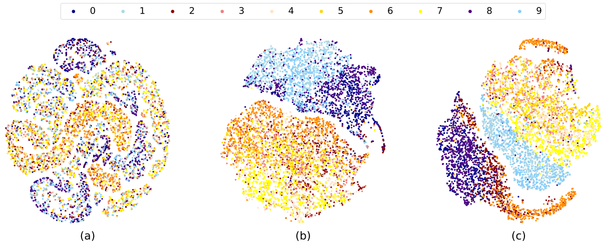

Figure 4: The evolution of features across the training process under two clustering orientations: (a) Initial distribution of instances, (b) Instance distribution after clustering along the default orientation and (c) Instance distribution after clustering along a given personalized orientation. The colors of the dots denote the 10 class labels of CIFAR-10.

We have the following observations from Fig. 3. i) it is observed that the red trendline is always higher than the blue and green ones. This indicates that our adopted query strategy outperforms these two known active query ones. ii) At 25% of the query cost, our query strategy almost reaches the highest accuracy. This shows that OCC saves a lot of annotation cost.

## Visualization

To vividly display the personalized clustering results, we visualize the distribution of instances in CIFAR-10 after clustering. In Fig. 4, we have the following observations. i) Subfigure (a) shows that the instance points before clustering is chaos. ii) Subfigure (b) shows that the yellow and orange instance points cluster together, while in subfigure (c) the dark blue and dark orange dots are clustered together. This shows that OCC realizes personalized clusters.

## Ablation Studies

**Effect of contrastive learning** We perform ablation analysis by removing the contrastive part of each one of the representation and assignment spaces. Along both default and personalized clustering orientation of CIFAR-10, the results are shown in Tab. 3. We observe that the clustering performance with contrastive learning in both spaces always obtains the highest values. This shows it is necessary to conduct contrastive learning in both spaces and it is effective in accurately catching the desired features by contrastive learning.

**Effect of label extension** We perform ablation analysis by removing the operation of label extension. Along both default and personalized clustering orientation of CIFAR-10, the results are shown in Tab. 4. It is observed that it can obtain a substantially higher performance with label extension. This shows that label extension benefits clustering performance improvement by increasing the annotated instance pairs.

| Orientation | Contrastive space | NMI | ARI | ACC |
|---|---|---|---|---|
| Default | R+A | 0.884 | 0.939 | 0.985 |
| | R only | 0.631 | 0.683 | 0.913 |
| | A only | 0.589 | 0.628 | 0.896 |
| Personalized | R+A | 0.690 | 0.791 | 0.945 |
| | R only | 0.005 | 0.006 | 0.539 |
| | A only | 0.492 | 0.499 | 0.853 |

Table 3: Effect of contrastive learning in representation(R) and assignment(A) space on CIFAR-10.

| Orientation | Label Extension | NMI | ARI | ACC |
|---|---|---|---|---|
| Default | YES | 0.884 | 0.939 | 0.985 |
| | NO | 0.631 | 0.683 | 0.913 |
| Personalized | YES | 0.690 | 0.791 | 0.945 |
| | NO | 0.524 | 0.542 | 0.868 |

Table 4: Effect of label extension on CIFAR-10.

# Conclusion

We have presented a model, named Oracle-guided Contrastive Clustering(OCC), for task-aware clustering that incorporate active query to oracles with personalized demand. Moreover, contrastive learning is joint with active learning to catch orientation-aware features and achieve desired clustering solutions. This is first deep framework for personalized image clustering. In the near future, we would like to extend this framework to any clustering domain in general.

# References

[1] D. Xu and Y. Tian, “A comprehensive survey of clustering algorithms,” *Annals of Data Science*, vol. 2, pp. 165–193, 2015.

[2] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967, p. 281–297.

[3] T. Zhang, R. Ramakrishnan, and M. Livny, “Birch: an efficient data clustering method for very large databases,” in *SIGMOD ’96*, 1996.

[4] J. C. Bezdek, R. Ehrlich, and W. E. Full, “Fcm: The fuzzy c-means clustering algorithm,” *Computers & Geosciences*, vol. 10, pp. 191–203, 1984.

[5] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *KDD*, 1996.

[6] C. E. Rasmussen, “The infinite gaussian mixture model,” in *NIPS*, 1999.

[7] E. Min, X. Guo, Q. Liu, G. Zhang, J. Cui, and J. Long, “A survey of clustering with deep learning: From the perspective of network architecture,” *IEEE Access*, vol. 6, pp. 39 501–39 514, 2018.

[8] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, “Deep clustering for unsupervised learning of visual features,” in *ECCV*, 2018.

[9] B. Yang, X. Fu, N. Sidiropoulos, and M. Hong, "Towards k-means-friendly spaces: Simultaneous deep learning and clustering," in *ICML*, 2017.

[10] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 10, pp. 8547–8555, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17037

[11] Y. Liu, W. Tu, S. Zhou, X. Liu, L. Song, X. Yang, and E. Zhu, "Deep graph clustering via dual correlation reduction," in *AAAI*, 2022.

[12] P. Ren, Y. Xiao, X. Chang, P. Huang, Z. Li, X. Chen, and X. Wang, "A survey of deep active learning," *ACM Computing Surveys (CSUR)*, vol. 54, pp. 1–40, 2022.

[13] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler, "The power of ensembles for active learning in image classification," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9368–9377, 2018.

[14] H. Ranganathan, H. Venkateswara, S. Chakraborty, and S. Panchanathan, "Deep active learning for image classification," *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3934–3938, 2017.

[15] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1183–1192. [Online]. Available: http://proceedings.mlr.press/v70/gal17a.html

[16] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal, "Deep batch active learning by diverse, uncertain gradient lower bounds," *ArXiv*, vol. abs/1906.03671, 2020.

[17] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=H1aIuk-RW

[18] S. Huang, T. Wang, H. Xiong, J. Huan, and D. Dou, "Semi-supervised active learning with temporal output discrepancy," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3427–3436, 2021.

[19] H. Ashtiani, S. Kushagra, and S. Ben-David, "Clustering with same-cluster queries," in *Advances in Neural Information Processing Systems*, vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/9597353e41e6957b5e7aa79214fcb256-Paper.pdf

[20] S. Dasgupta and V. Ng, "Which clustering do you want? inducing your ideal clustering with minimal feedback," *J. Artif. Intell. Res.*, vol. 39, pp. 581–632, 2010.

[21] H. Zhong, J. Wu, C. Chen, J. Huang, M. Deng, L. Nie, Z. Lin, and X. Hua, "Graph contrastive clustering," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9204–9213, 2021.

[22] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.

[23] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2014.

[24] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proceedings of The 33rd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 48. PMLR, 20–22 Jun 2016, pp. 478–487. [Online]. Available: https://proceedings.mlr.press/v48/xieb16.html

[25] X. Ji, A. Vedaldi, and J. F. Henriques, "Invariant information clustering for unsupervised image classification and segmentation," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9864–9873, 2019.

[26] J. Huang, S. Gong, and X. Zhu, "Deep semantic clustering by partition confidence maximisation," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8846–8855, 2020.

[27] J. Wu, K. Long, F. Wang, C. Qian, C. Li, Z. Lin, and H. Zha, "Deep comprehensive correlation mining for image clustering," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8149–8158, 2019.

[28] D. Zhang, F. Nan, X. Wei, S. Li, H. Zhu, K. McKeown, R. Nallapati, A. O. Arnold, and B. Xiang, "Supporting clustering with contrastive learning," in *NAACL*, 2021.

[29] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[30] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Toronto, Tech. Rep., 2009.

[31] Y. Tao, K. Takagi, and K. Nakata, "Clustering-friendly representation learning via instance discrimination and feature decorrelation," in *9th International Conference on Learning Representations (ICLR)*, 2021.

[32] J. Chang, L. Wang, G. Meng, S. Xiang, and C. Pan, "Deep adaptive image clustering," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5880–5888, 2017.

[33] S. Li and Y. Liu, "Sharper generalization bounds for clustering," in *International Conference on Machine Learning*. PMLR, 2021, pp. 6392–6402.

[34] M. S. Pydi and V. S. Lokhande, "Active learning with importance sampling," *CoRR*, vol. abs/1910.04371, 2019. [Online]. Available: http://arxiv.org/abs/1910.04371

[35] B. J. Gao, O. L. Griffith, M. Ester, and S. J. M. Jones, "Discovering significant opsm subspace clusters in massive gene expression data," in *KDD '06*, 2006.

[36] X. Cai, F. Nie, H. Huang, and F. Kamangar, "Heterogeneous image feature integration via multi-modal spectral clustering," *CVPR 2011*, pp. 1977–1984, 2011.

[37] C. C. Aggarwal and C. Zhai, "A survey of text clustering algorithms," in *Mining Text Data*, 2012.

[38] S. Basu, I. Davidson, and K. L. Wagstaff, *Constrained Clustering: Advances in Algorithms, Theory, and Applications*. Chapman and Hall/CRC, 2008.

[39] S. Dasgupta and D. J. Hsu, "Hierarchical sampling for active learning," in *ICML '08*, 2008.

[40] M. Wang, F. Min, Z. Zhang, and Y. Wu, "Active learning through density clustering," *Expert Syst. Appl.*, vol. 85, pp. 305–317, 2017.

[41] C. Xiong, D. Johnson, and J. J. Corso, "Spectral active clustering via purification of the $k$-nearest neighbor graph," in *European Conference on Data Mining*, 2013.

[42] E. Chien, H. Zhou, and P. Li, "Hs2: Active learning over hypergraphs with pointwise and pairwise queries," in *AISTATS*, 2019.

[43] L. Manduchi, K. Chin-Cheong, H. Michel, S. Wellmann, and J. E. Vogt, "Deep conditional gaussian mixture model for constrained clustering," in *NeurIPS*, 2021.