# Personalized Clustering via Targeted Representation Learning

Anonymous Author

## 1. Abstract

Clustering traditionally aims to reveal a natural grouping structure model from unlabeled data. However, this model may not always be ideal for users' preference. In this paper, we propose a personalized clustering method which explicitly performs targeted representation learning by interacting with users via modicum task information (e.g., must-link or cannot-link pairs) to guide the clustering orientation. First, we query users with the the most informative pairs, i.e., those pairs most hard to cluster and those most easy to miscluster, to facilitate the representation learning in terms of the clustering preference. And then, by exploiting attention mechanism the targeted representation is learned and augmented. By leveraging the targeted representation and constraint clustering loss as well, personalized clustering is obtained. Theoretically, we verify that the risk of personalized clustering is tightly bounded, that guarantees active queries to users do mitigate the clustering risk. Experimentally, extensive results show that our method performs well across different clustering tasks and datasets, even with a limited number of queries.

## 2. Problem Defination

- (Section 1) In the domain of image clustering, varying tasks often entail distinct clustering objectives. It is evident that unsupervised methods may not consistently excel across all clustering dimensions. This paper introduces a weakly supervised clustering approach, augmented by active learning, to adaptively address clustering tasks aligned with different clustering orientations, as illustrated in Figure 1.

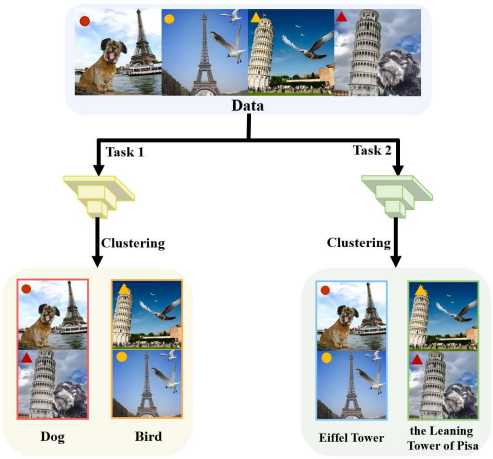


*Figure 1: The diversity of clustering orientation. Different tasks have different orientations for feature learning and image clustering.*

- (Subsection 3.1) For the image datasets, data is initially defined by generating original positive instance pairs through data augmentation techniques. Subsequently, based on the clustering results from each training iteration, the most informative sample pairs are selected for query. The clustering orientation is then learned from the existing positive and negative instance pairs. Ultimately, this process yields clustering results tailored to meet the specific objectives of the task at hand.

# 3.Illustration of Our Framework

Our method mentioned above is presented in Section 3.2 & Section 3.3 & Appendix A . A deep neural network creates representations from two random augmentations of the data. By assessing the position of images in the feature space, the framework selects the most informative sample pairs to guide the training orientation of the model. The model is then retrained by querying whether these pairs are must-link or cannot-link. This approach allows the final model to concentrate on features relevant to the desired clustering orientation, resulting in accurate clustering outcomes. Our framework can be illustrated in Figure 2.

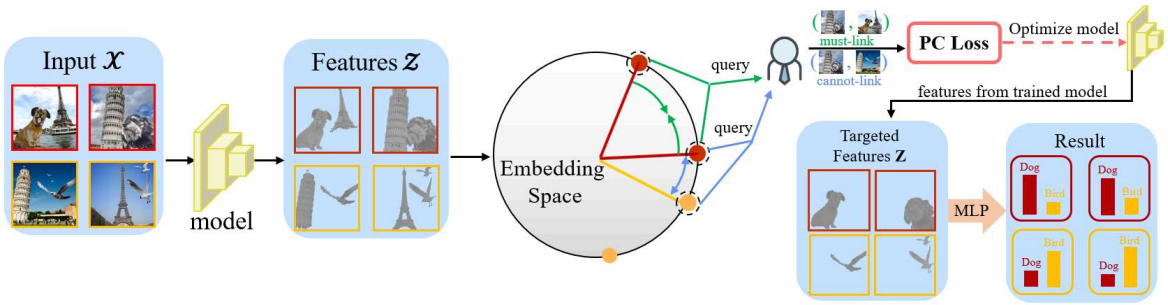


*Figure 2: The framework of Personalized Clustering.*

# 4. Theoretical Analysis

We propose a theoretical analysis to verify effectiveness of active query in constraint clustering. Simultaneously, the tight upper bound of generalization risk of PCL is given as well.

- (Subsection 3.4 & Appendix C) **Definition 4.1 (Unsupervised Loss.)**
  We need to measure the effect of clustering by an unsupervised loss. If we can query $q$ times, the overall unsupervised loss is defined as:

$$\hat{\mathcal{L}}_q(f) := \frac{1}{2N}\sum_{i=1}^{2N}\ell_i'(f).$$

  and $\hat{f}_q$ is defined as $\arg\min_{f\in\mathcal{F}}\hat{\mathcal{L}}_q(f)$, $q$ is the total num of query. The unsupervised population loss after $q$ queries is defined as:

$$\mathcal{L}_q(f) := \mathbb{E}_{x\sim\mathcal{X}}[\hat{\mathcal{L}}_q(f)],$$

  and $f_q^*$ is defined as $\arg\min_{f\in\mathcal{F}}\mathcal{L}_q(f)$.
- (Subsection 3.4 & Appendix C) **Theorem 4.1 (Excess Risk)**
  Theorem 3.1 Suppose we can completely query each sample. We use $Q$ as the total number of queries. With probability at least $1-\delta$,

$$|\hat{\mathcal{L}}_Q(\hat{f}_Q) - \mathcal{L}_Q(f_Q^*)| \le \mathcal{O}\left(\frac{\eta\mathcal{R}_{\mathcal{S}}(\mathcal{F})}{N} + \sqrt{\frac{\log\frac{1}{\delta}}{N}}\right).$$

  where $\mathcal{R}_{\mathcal{S}}(\mathcal{F})$ is the Rademacher complexity of $\mathcal{F}$ with respect to training set $\mathcal{S}$, and $\eta$ is the Lipschitz constant. Theorem 3.1 indicates that $\mathcal{L}_Q(f_Q^*)$ can be well bounded if the trained model $\hat{f}_Q$ is good. However, due to the limited queries (less than $Q$), we cannot calculate $\hat{\mathcal{L}}_Q(\hat{f}_Q)$. As a compromise, can we get closer to it?
  (Subsection 3.4 & Appendix C) **Theorem 3.2 (Queries Reduce the Gap)**
  We only consider adding one query. If $q \ll N$:

$$|\hat{\mathcal{L}}_q(\hat{f}_q) - \hat{\mathcal{L}}_Q(\hat{f}_Q)| \ge |\hat{\mathcal{L}}_{q+1}(\hat{f}_{q+1}) - \hat{\mathcal{L}}_Q(\hat{f}_Q)|.$$

  This just matches the process of algorithm: we initially make a new query request based on the existing model $\hat{f}_q$, followed by updating $\hat{\mathcal{L}}_q$ to $\hat{\mathcal{L}}_{q+1}$ based on the query results, and finally optimizing the loss to get $\hat{f}_{q+1}$. Theorem 3.2 guarantees the gap between the loss of the current model and the complete queries loss will be reduced after each query. Now we show that our strategy is indeed effective.
  (Subsection 3.4 & Appendix C) **Theorem 3.3 (Query Strategy Is Effective)**
  Suppose $q \ll N$, the sample pair $(\mathbf{x}_i, \mathbf{x}_j)$ is queried, and the result is cannot-link, we have:

$$|\hat{\mathcal{L}}_q(f) - \hat{\mathcal{L}}_{q+1}(f)| \propto \exp\left(\mathcal{S}_{up}(\mathbf{x}_i, \mathbf{x}_j)\right),$$

and if the result is must-link, we have:

$$|\hat{\mathcal{L}}_q(f) - \hat{\mathcal{L}}_{q+1}(f)| \propto \mathcal{S}_{up}(\mathbf{x}_i, \mathbf{x}_j).$$

Since we want to reduce the gap in Theorem 3.2, we should equivalently maximize $|\hat{\mathcal{L}}_q(f) - \hat{\mathcal{L}}_{q+1}(f)|$ and then train the model according to $\hat{\mathcal{L}}_{q+1}$. Theorem 3.3 states that our query strategy is effective in this sense, because when $q$ is small, $\mathcal{S}_{up}$ matters more in the scoring function. Besides, we have added $\mathcal{S}_{hp}$ in the score and weight it more if $q$ is larger, making our strategy more robust and efficient. The following experiments have proved the effectiveness in practice.

# 5. Experiments

- (Subsection 4.1 & Appendix B) **Datasets.** Our experiments are conducted on three datasets (CIFAR10-2, CIFAR100-4, Imagenet10-2) with two different orientations, encompassing three benchmark datasets (CIFAR-10, CIFAR-100, Imagenet-10), along with an artificial cross-dataset. The details of the datasets are as follows.

| Dataset | Instances | Classes | Clusters |
|---|---|---|---|
| CIFAR10-2 | 60,000 | 10 | 2 |
| CIFAR100-4 | 60,000 | 100 | 4 |
| ImageNet10-2 | 12,818 | 10 | 2 |

*Table 1: A summary of the datasets.*

Figure 3 shows an example of two clustering orientations from the same original dataset. The details of creating our target dataset can be found at (Appendix B)

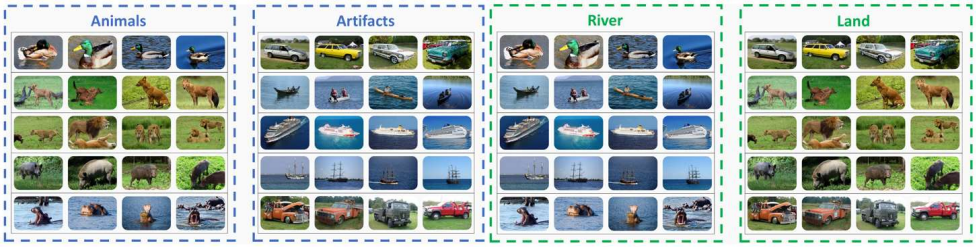


*Figure 3: The default and personalized orientation designed artificially.*

- (Subsection 4.2 ) **Partial Experimental Results.**

| | | CIFAR10-2 | | | | CIFAR100-4 | | | | ImageNet10-2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Default | | NMI | ARI | F | ACC | NMI | ARI | F | ACC | NMI | ARI | F | ACC |
| Unsupervised | kmeans | 0.054 | 0.079 | 0.550 | 0.641 | 0.038 | 0.033 | 0.281 | 0.358 | 0.078 | 0.103 | 0.553 | 0.661 |
| | MiCE | 0.664 | 0.755 | 0.881 | 0.934 | 0.230 | 0.228 | 0.423 | 0.590 | 0.762 | 0.849 | 0.925 | 0.960 |
| | CC | 0.575 | 0.662 | 0.835 | 0.907 | 0.176 | 0.166 | 0.379 | 0.515 | 0.508 | 0.615 | 0.807 | 0.892 |
| | GCC | 0.653 | 0.753 | 0.880 | 0.934 | **0.344** | **0.337** | **0.503** | **0.604** | **0.908** | **0.952** | **0.976** | **0.988** |
| | SPICE | 0.600 | 0.638 | 0.823 | 0.899 | 0.250 | 0.197 | 0.399 | 0.496 | 0.642 | 0.725 | 0.864 | 0.925 |
| Constrained | DCC | 0.572 | 0.686 | 0.848 | 0.914 | 0.134 | 0.120 | 0.345 | 0.449 | 0.432 | 0.536 | 0.768 | 0.866 |
| | ADC | 0.544 | 0.662 | 0.838 | 0.907 | 0.047 | 0.055 | 0.319 | 0.390 | 0.453 | 0.550 | 0.775 | 0.870 |
| | SDEC | 0.355 | 0.430 | 0.720 | 0.828 | 0.070 | 0.048 | 0.317 | 0.380 | 0.192 | 0.254 | 0.627 | 0.752 |
| | DC-GMM | 0.542 | 0.657 | 0.838 | 0.905 | 0.107 | 0.121 | 0.349 | 0.431 | 0.490 | 0.598 | 0.799 | 0.886 |
| | Ours | **0.737** | **0.830** | **0.918** | **0.955** | 0.197 | 0.208 | 0.407 | 0.575 | 0.722 | 0.817 | 0.908 | 0.952 |

*Table 2: The clustering performance under the default orientation, on three object image benchmarks. The default orientation completely follows the tendency of deep clustering. The final results are shown in boldface.*

| | | CIFAR10-2 | | | | CIFAR100-4 | | | | ImageNet10-2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Personlized | | NMI | ARI | F | ACC | NMI | ARI | F | ACC | NMI | ARI | F | ACC |
| Unsupervised | kmeans | 0.009 | 0.015 | 0.519 | 0.563 | 0.028 | 0.022 | 0.274 | 0.314 | 0.027 | 0.037 | 0.519 | 0.597 |
| | MiCE | 0.028 | 0.045 | 0.535 | 0.607 | 0.159 | 0.135 | 0.355 | 0.399 | 0.024 | 0.032 | 0.516 | 0.591 |
| | CC | 0.022 | 0.035 | 0.528 | 0.594 | 0.132 | 0.115 | 0.342 | 0.405 | 0.032 | 0.043 | 0.521 | 0.605 |
| | GCC | 0.035 | 0.057 | 0.542 | 0.620 | 0.230 | 0.203 | 0.404 | 0.508 | 0.025 | 0.034 | 0.517 | 0.592 |
| | SPICE | 0.005 | 0.007 | 0.513 | 0.543 | 0.232 | 0.215 | 0.413 | 0.504 | 0.068 | 0.088 | 0.548 | 0.648 |
| Constrained | DCC | 0.244 | 0.334 | 0.679 | 0.789 | 0.099 | 0.095 | 0.340 | 0.473 | 0.204 | 0.267 | 0.634 | 0.758 |
| | ADC | 0.206 | 0.291 | 0.665 | 0.771 | 0.024 | 0.023 | 0.374 | 0.322 | 0.102 | 0.138 | 0.570 | 0.686 |
| | SDEC | 0.013 | 0.021 | 0.521 | 0.573 | 0.092 | 0.083 | 0.354 | 0.417 | 0.075 | 0.102 | 0.552 | 0.660 |
| | DC-GMM | 0.060 | 0.074 | 0.671 | 0.655 | 0.072 | 0.060 | 0.311 | 0.384 | 0.000 | 0.000 | 0.675 | 0.512 |
| | Ours | **0.453** | **0.558** | **0.765** | **0.873** | **0.234** | **0.247** | **0.437** | **0.597** | **0.409** | **0.510** | **0.755** | **0.857** |

*Table 3: The clustering performance under the personalized orientation, on three object image benchmarks. The personalized target orientation is artificially designed opposite to the default one. The final results are shown in*

*boldface.*

- (Subsection 4.3) **Visualization Results.**
  Subfigure (a) depicts that the instance points before clustering is chaos. Subfigure (b) depicts that the yellow and orange instance points cluster together, while in subfigure (c) the dark blue and dark orange dots are clustered together. The above facts show that it is workable for PCL to cluster according to the demand of tasks.

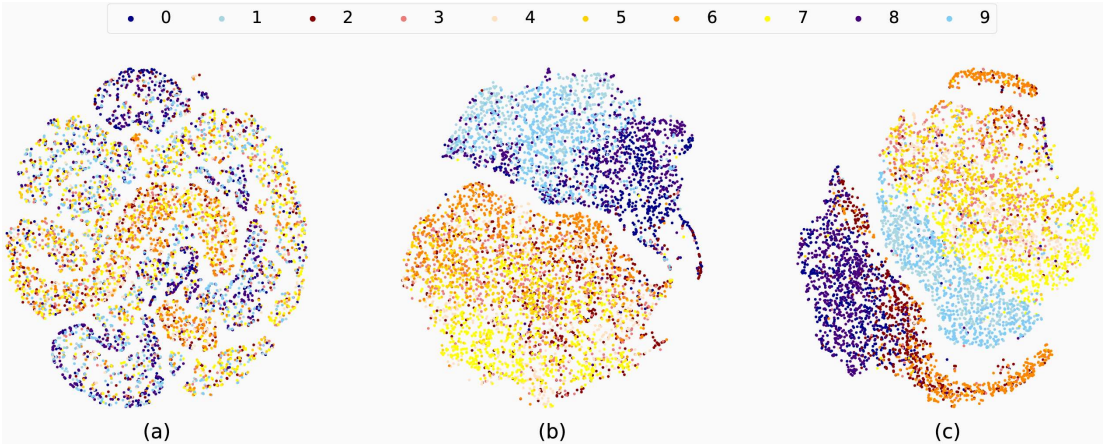


*Figure 4: The distribution of instances in CIFAR10-2 after clustering.*

- (Subsection 4.3) **Ablation Studies.**
  We performed two ablation analysis by removing components of the query strategy and cross-attention module.

| Query strategy | NMI | ARI | F | ACC |
|---|---|---|---|---|
| random | 0.172 | 0.170 | 0.380 | 0.487 |
| $S_{up}$ | 0.196 | 0.211 | 0.411 | 0.569 |
| $S_{hp}$ | 0.160 | 0.154 | 0.369 | 0.481 |
| $S_{up} + S_{hs}$ | **0.234** | **0.247** | **0.437** | **0.597** |

*Table 4:Effect of query strategy in personlized clustering on CIFAR100-4*

| Module | NMI | ARI | F | ACC |
|---|---|---|---|---|
| CNN only | 0.127 | 0.122 | 0.344 | 0.445 |
| CNN+Attention | **0.234** | **0.247** | **0.437** | **0.597** |

*Table 5:Effect of Cross-Attention Module in personlized clustering on CIFAR100-4*

# 6. Contribution

- We propose a novel model, called PCL, for personalized clustering task, which leverages active query to control the targeted representation learning.
- A theoretical analysis on clustering is conducted to verify effectiveness of active query in constraint clustering. Simultaneously, the tight upper bound of generalization risk of PCL is given as well.
- Extensive experiments show that our model outperforms four deep clustering approaches, three semi-supervised clustering approaches, and two active clustering approaches on three image datasets. Our active query strategy also performs well compared to other methods.

# 7. Code

Please access our code through an anonymous link: PCL.