Contrastive Prototypical Network with Wasserstein Confidence Penalty

Haoqing Wang Zhi-Hong Deng*

School of Artificial Intelligence, Peking University
wanghaoqing@pku.edu.cn zhdeng@pku.edu.cn

Abstract. Unsupervised few-shot learning aims to learn the inductive bias from unlabeled dataset for solving the novel few-shot tasks. The existing unsupervised few-shot learning models and the contrastive learning models follow a unified paradigm. Therefore, we conduct empirical study under this paradigm and find that pairwise contrast, meta losses and large batch size are the important design factors. This results in our CPN (Contrastive Prototypical Network) model, which combines the prototypical loss with pairwise contrast and outperforms the existing models from this paradigm with modestly large batch size. Furthermore, the one-hot prediction target in CPN could lead to learning the sample-specific information. To this end, we propose Wasserstein Confidence Penalty which can impose appropriate penalty on overconfident predictions based on the semantic relationships among pseudo classes. Our full model, CPNWCP (Contrastive Prototypical Network with Wasserstein Confidence Penalty), achieves state-of-the-art performance on miniImageNet and tieredImageNet under unsupervised setting. Our code is available at https://github.com/Haoqing-Wang/CPNWCP.

Keywords: Unsupervised few-shot learning · Contrastive learning · Confidence penalty · Wasserstein distance

1 Introduction

Humans have the ability to learn from limited labeled data, yet it is still a challenge for modern machine learning systems. Few-shot learning [15, 40, 27, 53, 44] is proposed to imitate this ability and has attracted significant attention from the machine learning community recently. Before solving novel few-shot tasks, most models typically learn task-shared inductive bias from sufficient labeled data (base dataset). However, obtaining sufficient labeled data for certain domains may be difficult or even impossible in practice, such as satellite imagery and skin diseases. When only the unlabeled data from the same domain as the novel tasks is available, we can learn the inductive bias in the unsupervised manner, which is formalized as the unsupervised few-shot learning.

Existing unsupervised few-shot learning models focus on constructing pseudo training tasks from unlabeled dataset with clustering based methods [21, 23] or

* Corresponding author

data augmentation based methods [25, 54, 26]. The latter ones usually achieve better performance, which randomly select a batch of samples from the unlabeled dataset and generate in-class samples for each one to form the support and query set via hand-craft or learnable data augmentations [25, 54, 26]. The synthetic pseudo tasks are used to train the meta-learning models [40, 27, 43].

Concurrently, contrastive learning [9, 16, 51, 11] achieves outstanding success in the field of self-supervised representation learning and can be directly used in unsupervised few-shot learning. These models also randomly select a batch of samples from the unlabeled dataset and generate different views for each one via well-designed data augmentations. The key motivation is to push the views of the same sample (positive pair) close to each other and the views of different samples (negative pair) away from each other in embedding space. To this end, they propose different contrastive losses, such as a lower bound on the mutual information [9, 18], the asymmetric similarity loss [16] or the difference between the cross-correlation matrix and the identity matrix [51].

If we consider the contrastive loss as a meta-learning objective where the augmented views form a training sub-task, the contrastive learning models are essentially the same as the data augmentation based unsupervised few-shot learning models. Therefore, we take a closer look at this paradigm and conduct empirical study to find the key design factors, as shown in Section 4.3. Concretely, the contrastive learning models typically can achieve better performance than unsupervised few-shot learning models, while we find their superiority actually comes from the larger batch size and pairwise contrast. Although the key motivation is appropriate, the specific contrastive losses (e.g., a lower bound on the mutual information) are not as suitable for few-shot learning as some meta losses, which limits its performance. Therefore, we combine the prototypical loss [40] with pairwise contrast and get CPN (Contrastive Prototypical Network). With the modestly large batch size, CPN outperforms both the unsupervised few-shot learning models and the contrastive learning models.

Furthermore, some negative pairs could be semantically similar or even belong to the same semantic class in CPN. This problem is also referred to as "class collision" [3] or "sampling bias" [12, 48] in contrastive learning, as shown in Fig. 1(a). Using one-hot prediction target could overly push view b_2 close to view b_1 while away from view a_1 and make the representations ignoring the semantic information about birds and learning the sample-specific information, like background and color distribution. Therefore, we make the prediction distribution in CPN approximating a latent distribution (e.g., the uniform distribution) to prevent overconfident one-hot prediction during training. The most related methods, Label Smoothing [41, 33] and Confidence Penalty [34], use f-divergence to measure the difference between the prediction distribution and the latent distribution. However, f-divergence does not consider the semantic relationships among classes, since the difference in the prediction probability of each class is computed independently. Instead, we use the Wasserstein distance [7] which can impose appropriate penalty based on the semantic relationships among classes. We solve the optimal transport problem using Sinkhorn iteration [13, 1] which

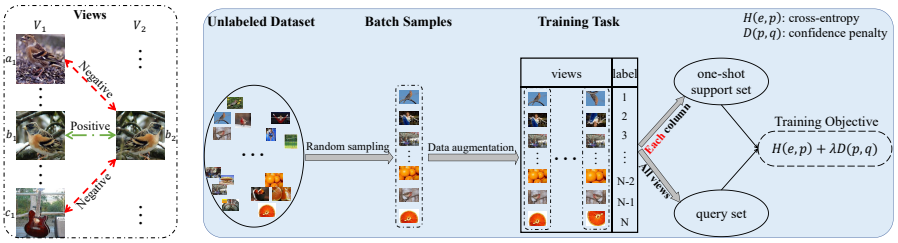


Fig. 1. (a) Some negative pairs (e.g., (a_1, b_2)) may be semantically similar or even belong to the same semantic class. (b) Overview of our CPNWCP model. We select a batch of samples and generate in-class samples for each one via data augmentations. We combine the prototypical loss with pairwise contrast. To alleviate the problem of learning sample-specific information, we propose Wasserstein Confidence Penalty which make the prediction distribution approximating the uniform distribution.

works well with the automatic differentiation libraries for deep learning, without computing the second-order gradients. This method is denoted as Wasserstein Confidence Penalty, which effectively alleviates the problem of learning sample-specific information.

The framework of our full model, CPNWCP (Contrastive Prototypical Network with Wasserstein Confidence Penalty), is shown in Fig. 1(b). The main contributions of this work are as follows:

- Under a unified paradigm of contrastive learning and unsupervised few-shot learning, we conduct empirical study and find that pairwise contrast, meta losses and large batch size are the key factors to the satisfactory performance on novel few-shot tasks, which results in our CPN model.
- To prevent CPN from learning sample-specific information, we propose Wasserstein Confidence Penalty which significantly improves the performance of CPN and outperforms the f-divergence based confidence penalty methods.
- Our CPNWCP model achieves state-of-the-art performance on standard unsupervised few-shot learning datasets, miniImageNet and tieredImageNet.

2 Related work

Unsupervised few-shot learning. Although various few-shot learning models [40, 15, 27, 53, 5] have achieved impressive performance, they rely on sufficient labeled data during training, which is difficult to acquire in some domains. Unsupervised few-shot learning [21, 23, 25, 54, 2, 50, 26] aims to learn the task-shared inductive bias from the unlabeled dataset. These models generate pseudo few-shot tasks to train the meta-learning models through various techniques. [21, 23] construct the partitions of unlabeled dataset through clustering in the embedding space and thus obtain the pseudo labels. [25, 54, 2, 50] randomly select a batch of samples with small batch size (e.g., 5) and each sample is assumed from the different

class. They generate the in-class samples via data augmentations to construct the support set and query set. LASIUM [26] generate the in-class and out-of-class samples via interpolation in the latent space of a generative model. Besides, ProtoTransfer [30] also constructs few-shot tasks via data augmentations, but uses a large batch size and achieves significant improvement. It means that large batch size may be a key factor to the good performance on novel few-shot tasks.

Contrastive learning. Contrastive learning [4, 18, 9, 51, 45] is a successful self-supervised representation learning [24, 46] framework that learns well-generalizing representations from large scale unlabeled dataset. In the training phase, they typically select a batch of samples from the unlabeled dataset and generate different views for each sample via data augmentations. Generally, they push the views from the same sample (positive pair) close to each other and the views from different samples (negative pair) away from each other in the embedding space. The training objectives are typically to maximize the lower bound on the mutual information [20, 4, 18, 9, 47] or make the cross-correlation matrix between positive pair as close to the identity matrix as possible [51], which indirectly learn semantic relevance among samples (linear separability or proximity in the embedding space) and not as suitable for few-shot learning as some meta-losses. Contrastive learning avoids representation collapse via large batch size and it also adapts pairwise contrast, i.e., each view is compared with all other views from the same sample, which is helpful for learning useful representations. Chuang et al. [12] and Wei et al. [48] point out the "sampling bias" problem, and propose a debiased objective and consistency regularization respectively.

Confidence penalty. For a network, over-confident prediction is a symptom of overfitting. For this end, Label Smoothing [41, 33, 52] relaxes the one-hot label to a soft version which is equivalent to adding uniform label noise. Confidence Penalty [34] penalizes low entropy prediction following the maximum entropy principle [22]. They essentially make the prediction distribution approximating a latent distribution, such as the uniform distribution or a learnable distribution [52]. Both Label Smoothing and Confidence Penalty use f-divergence to measure the difference between distributions, which ignores the semantic relationships among classes. We explore the Wasserstein distance and introduce the semantic relationships with cost matrix. With semantic relationships as prior information, our Wasserstein Confidence Penalty can impose appropriate penalty and outperforms both Label Smoothing and Confidence Penalty in CPN.

3 Methodology

3.1 Preliminaries

Few-shot learning aims to obtain the model which can efficiently and effectively solve novel few-shot tasks. Each few-shot task contains a support set $\mathcal{S}$ and a query set $\mathcal{Q}$. When the support set $\mathcal{S}$ contains N classes with K samples in each class, the few-shot task is called a N-way K-shot task, and $N = 5$ and

$K = 1$ or 5 is the standard setting. The query set $\mathcal{Q}$ contains the samples from the same classes with the support set $\mathcal{S}$. We need to classify the samples in the query set correctly based on the few labeled data from the support set. Since K is typically very small, this is challenging for modern deep learning models. To this end, one can learn the inductive bias from a large base training set $\mathcal{D}$, which has completely disjoint classes with the novel tasks. Meta-learning models [40, 15, 27] adapt *episode* training [42], where few-shot training tasks are constructed from the base training set $\mathcal{D}$. Some non-episodical transfer learning models [10, 14, 55] also achieve comparable performance.

In most cases, the large base training set $\mathcal{D}$ is labeled, so we can easily construct few-shot training tasks based on labels. However, obtaining sufficient labeled dataset is difficult or even impossible for some domains, e.g., satellite imagery and skin diseases, so we assume the training set is unlabeled in this work and learn the inductive bias in the unsupervised manner.

3.2 Contrastive Prototypical Network

We first describe a unified paradigm, and then take the data augmentation based unsupervised few-shot learning models, contrastive learning models and our CPN model as its special cases.

Given an unlabeled dataset $\mathcal{D}$, samples $\{x_i\}_{i=1}^N$ are randomly selected and each x_i represents a pseudo class. For each x_i, in-class samples $\{v_i^j\}_{j=1}^M$ are generated via manually or learnable data augmentations. For a specific problem, the loss function $\mathcal{L}$ is designed and calculated on the sub-dataset $\{v_i^j\}_{i=1,j=1}^{N,M}$ and the sub-task training objective is

$$\min_\theta \mathbb{E}_{p(\{v_i^j\}_{i=1,j=1}^{N,M})}[\mathcal{L}(\{v_i^j\}_{i=1,j=1}^{N,M}, \theta)] \tag{1}$$

where θ represents the model parameters. We denote this paradigm as the *Sampling-Augmentation* paradigm.

Data augmentation based unsupervised few-shot learning models [25, 54, 26] follow this paradigm and achieve outstanding performance. Let $M = S + Q$, the augmented samples $\{v_i^j\}_{i=1,j=1}^{N,M}$ constitute a pseudo S-shot training task with $N \cdot S$ samples as the support set $\mathcal{S}$ and the $N \cdot Q$ samples as the query set $\mathcal{Q}$. These pseudo few-shot tasks can be directly used to train the meta-learning models, which means $\mathcal{L}$ can be various meta losses and the training objective is

$$\min_\theta \mathbb{E}_{p(\{v_i^j\}_{i=1,j=1}^{N,M})}[\mathcal{L}^{meta}(\{v_i^j\}_{i=1,j=S+1}^{N,M}, \psi)], \quad \psi = \mathcal{A}(\{v_i^j\}_{i=1,j=1}^{N,S}, \theta) \tag{2}$$

where $\mathcal{A}$ is a base learner and ψ is the task solution. Following the *episode* training, N is typically set small while M is set large, such as $N = 5$ and $M = 5 + 15$. This setting is popular in the few-shot learning, since it makes the training setting aligning with the test scenario. However, we find that the small batch size is not suitable for unsupervised few-shot learning, as shown in Fig. 2.

Most contrastive learning models [20, 4, 18, 9] also follow this paradigm and achieve outstanding success in the field of self-supervised representation learning.

It is usually assumed that different views $\{v_i^j\}_{j=1}^M$ share the semantic information from the input x_i, so the view-invariant representations are expected. For example, many contrastive learning models [20, 49, 18, 9] maximize the mutual information between the representations of positive pair, and the training loss is the InfoNCE lower bound estimate [35]. M is typically set 2 for simplicity and the training objective is

$$\min_\theta \mathbb{E}_{p(\{v_i^j\}_{i=1,j=1}^{N,2})}\left[-\frac{1}{2N}\sum_{i=1}^{N}\sum_{j,l=1;j\neq l}^{2}\ln\frac{\exp(h_\theta(g_\theta(v_i^j),g_\theta(v_i^l)))}{\sum_{s=1}^{N}\sum_{t=1,(s,t)\neq(i,j)}^{2}\exp(h_\theta(g_\theta(v_i^j),g_\theta(v_s^t)))}\right] \quad (3)$$

where g_θ is an encoder network and h_θ contains a multilayer perceptron used to calculate the InfoNCE lower bound. Each view v_i^j is compared with all other views $\{v_i^l\}_{l=1}^M, l \neq j$ from the same sample x_i, i.e., pairwise contrast, which is useful in unsupervised few-shot learning, as shown in Table 3. The batch size N is typically set large to avoid representation collapse, such as $N = 4096$ in SimCLR [9]. Although the contrastive learning models can be applied in unsupervised few-shot learning, their training losses indirectly learn the semantic relevance among samples, e.g., using mutual information or cross-correlation matrix [51], which are not as suitable as directly comparing the representations of different views, like prototypical loss, as shown in Fig. 2.

Our CPN model combines the advantages of both the contrastive learning models and the data augmentation based unsupervised few-shot learning models. We adopt a big batch size and introduce pairwise contrast to a widely used meta loss, prototypical loss [40]. The training objective is

$$\min_\theta \mathbb{E}_{p(\{v_i^j\}_{i=1,j=1}^{N,M})}\left[-\frac{1}{NM^2}\sum_{l=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{M}\ln\frac{\exp(-\|g_\theta(v_i^j)-g_\theta(v_i^l)\|^2)}{\sum_{k=1}^{N}\exp(-\|g_\theta(v_i^j)-g_\theta(v_k^l)\|^2)}\right] \quad (4)$$

The l-th view $\{v_i^l\}_{i=1}^N$ is used as the one-shot support set to classify all the views. We directly compare the representations of different views without using the multilayer perceptron to calculate mutual information estimation. With modestly large batch size, our CPN outperforms both the unsupervised few-shot learning models and some classical contrastive learning models.

In addition to the empirical study, we also provide some theoretical or intuitive justification for the above key factors.

1) *Pairwise contrast.* Each positive pair provides supervised information to each other under the unsupervised setting [45], and the model learns semantic knowledge from the shared information between views. Obviously, pairwise contrast brings more shared information between views which contains more useful knowledge. Concretely, assuming there are M views, we have $I_{pc} = \sum_{l=1}^{M}\sum_{k\neq l} I(v_l, v_k) \geq I_{w/o\ pc} = \sum_{k\neq 1} I(v_1, v_k)$, where 'pc' represents 'pairwise contrast' and we set view v_1 as the anchor view when without pairwise contrast.

2) *Meta losses.* Compared with the contrastive losses which indirectly learn the semantic relevance among samples via mutual information or cross-correlation matrix, the meta losses typically directly compare the representations of different views (especially the metric-based meta losses), and are more suitable for few-shot scenarios, i.e., semantic matching between support and query samples.

3) *Large batch size.* In the instance discrimination task, the number of classes is directly related to the batch size. Our CPN and some contrastive learning models adopt the cross-entropy loss and may encounter log-K curse [8]. Concretely, the cross-entropy loss is $\mathbf{H} = -\log \frac{\exp(s_y)}{\sum_{i=1}^{N} \exp(s_i)} = \log(1 + \sum_{i \neq y} \exp(s_i - s_y))$, where $(s_1, \cdots, s_N)$ is the prediction logits. After several training epochs, s_y is significantly larger than $s_i, i \neq y$ and $\exp(s_i - s_y), i \neq y$ are small, so $\mathbf{H} \approx \log(1 + (N-1)\epsilon)$ with ϵ is a small constant. When we use a small batch size, $\mathbf{H} \approx (N-1)\epsilon \approx 0$ and floating-point precision can lead to large gradient variance, hurting performance.

3.3 Wasserstein Confidence Penalty

As shown in Fig. 1(a), some negative pairs in CPN could be semantically similar or even belong to the same semantic class. Using the one-hot prediction target as in Eq. (4) could overly push the semantically similar negative pairs away from each other, which has the risk of learning sample-specific information rather than generalizing semantic information in the representations.

Further, when we use a large batch size, this problem becomes serious since the probability that the samples from the same semantic class appear in the selected batch significantly increases. For this end, we make the prediction distribution approximating a latent distribution during training to prevent overconfident prediction, which we denote as confidence penalty. For CPN, when we use $\{v_i^l\}_{i=1}^N$ as one-shot support set, the prediction distribution of a view v is $p_\theta^l(i|v) = \frac{\exp(-\|g_\theta(v) - g_\theta(v_i^l)\|^2)}{\sum_{k=1}^{N} \exp(-\|g_\theta(v) - g_\theta(v_k^l)\|^2)}$. We introduce a plug-and-play confidence penalty term $D(p_\theta^l(i|v), q)$, where $D(\cdot, \cdot)$ is a distance metric between distributions and q represents a latent distribution. Therefore, the Eq. (4) becomes

$$\min_\theta \mathbb{E}_{p(\{v_i^j\}_{i=1,j=1}^{N,M})} \left[\frac{1}{NM^2} \sum_{l=1}^{M} \sum_{i=1}^{N} \sum_{j=1}^{M} \Big(-\ln p_\theta^l(i|v_i^j) + \lambda D(p_\theta^l(i|v_i^j), q) \Big) \right] \tag{5}$$

where λ is a weight coefficient. The ideal choice for distribution q is the ground-truth class distribution in the instance classification task, which has non-ignorable probability in the semantically similar pseudo classes and we can not obtain it without labels. Therefore, we simply set q as the uniform distribution, i.e., $q_k = 1/N, k = 1, \cdots, N$. For distance metric D, we consider different choices.

For simplicity, let e be a one-hot distribution and p be the prediction distribution, a widely-used choice for distance metric $D(p, q)$ is the f-divergence $D(p, q) = \sum_{k=1}^{N} f(p_k/q_k) \cdot q_k$, where f is a convex function with $f(1) = 0$. When $f(z) = z \ln z$, the f-divergence becomes the Kullback–Leibler divergence $D(p, q) = KL(p\|q) = \sum_{k=1}^{N} p_k \ln p_k + \ln N$. This is the regularization term from [34], named Confidence Penalty. It penalizes low entropy prediction distributions following the maximum entropy principle [22] and is also used in reinforcement learning [29, 32]. When $f(z) = -\ln z$, the f-divergence becomes the reverse Kullback–Leibler divergence $KL(q\|p)$ and this regularization term

is equivalent to Label Smoothing [41, 33]. Concretely, $H(e,p) + \lambda KL(q\|p) = H(e+\lambda q, p) - \lambda \ln N$, where $H(e,p) = -\sum_{k=1}^{N} e_k \ln p_k$ is cross-entropy. $e+\lambda q$ is equivalent to $(1-\alpha)e+\alpha q$, which is just the target distribution in Label Smoothing and α is typically set 0.1. When $f(z) = [(z+1)\ln(2/(z+1)) + z\ln z]/2$, we get a symmetric f-divergence, Jensen–Shannon divergence.

The difference in the probability of each class is computed independently in f-divergence and the structural information, i.e., the semantic relationships among different classes, is ignored. For this end, we use the Wasserstein distance [7] $W(p,q)$ to introduces the semantic relationships as the prior knowledge. To calculate $W(p,q)$, we need to solve the optimal transport problem

$$\begin{aligned} \min_{T} \quad & \sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij} \cdot C_{ij} \\ s.t. \quad & T_{ij} \geq 0, i = 1, \cdots, N, j = 1, \cdots, N \\ & \sum_{j=1}^{N} T_{ij} = p_i, i = 1, \cdots, N; \quad \sum_{i=1}^{N} T_{ij} = q_j, j = 1, \cdots, N \end{aligned} \tag{6}$$

where $T \in \mathbb{R}^{N\times N}$ is a transportation matrix and $C \in \mathbb{R}^{N\times N}$ is the cost matrix. C_{ij} represents the cost of transporting per unit probability from class i to class j. T_{ij} represents the amount of the probability transported from class i to class j. Let the solution of the problem (6) be T^*, which is the matching flows with the minimum cost between p and q, we have $W(p,q) = \sum_{i=1}^{N}\sum_{j=1}^{N} T^*_{ij} \cdot C_{ij}$. The cost matrix C is the key to introduce the structural information. C_{ij} can be understood as the cost of misclassifying a sample from class i to class j. Intuitively, the higher the semantic similarity between class i and class j, the smaller the transportation cost C_{ij} should be. Therefore, we define the transportation cost as a decreasing function of class similarity

$$C_{ij} = \gamma \cdot (1 - S_{ij}) + \mathbb{I}_{i=j} \tag{7}$$

where γ is a scaling factor, $\mathbb{I}_{i=j}$ is an indicator function in the condition $i = j$ and is used to avoid zero cost, and S_{ij} represents the semantic similarity between class i and class j. Although this definition can not satisfy $W(p,p) = 0$, we find avoiding zero cost can achieve better performance in practice. In the batch of samples $\{x_i\}_{i=1}^{N}$, we label each x_i as a pseudo class i. Considering that mean can weaken sample differences and highlight intra-class commonality, we use the mean of the representations of views $\{v_i^j\}_{j=1}^{M}$ to represent each pseudo class i and use their cosine similarity as the class similarity, i.e., $S_{ij} = \frac{r_i^T r_j}{\|r_i\|\|r_j\|}$ with $r_i = \frac{1}{M}\sum_{j=1}^{M} g_\theta(v_i^j)$. We thus can measure the difference between distributions in a way that is sensitive to semantic relationships among classes.

To solve the problem (6), we use the Sinkhorn iteration [13, 1] which enforces a simple structure on the optimal transportation matrix and can quickly solve the optimal transport problem. The gradients can be back-propagated along the iteration process, so it works well with the automatic differentiation libraries for

Table 1. Few-shot classification accuracy(%) with 95% confidence interval on 10,000 5-way K-shot tasks randomly sampled from miniImageNet.

Model	1-shot	5-shot	20-shot	50-shot
Train from scratch [21]	27.59 ± 0.59	38.48 ± 0.66	51.53 ± 0.72	59.63 ± 0.74
CACTUs-ProtoNet [21]	39.18 ± 0.71	53.36 ± 0.70	61.54 ± 0.68	63.55 ± 0.64
CACTUs-MAML [21]	39.90 ± 0.74	53.97 ± 0.70	63.84 ± 0.70	69.64 ± 0.63
UMTRA [25]	39.93	50.73	61.11	67.15
ULDA-ProtoNet [36]	40.63 ± 0.61	56.18 ± 0.59	64.31 ± 0.51	66.43 ± 0.47
ULDA-MetaOptNet [36]	40.71 ± 0.62	54.49 ± 0.58	63.58 ± 0.51	67.65 ± 0.48
LASIUM-ProtoNet [26]	40.05 ± 0.60	52.53 ± 0.51	59.45 ± 0.48	61.43 ± 0.45
LASIUM-MAML [26]	40.19 ± 0.58	54.56 ± 0.55	65.17 ± 0.49	69.13 ± 0.49
ArL-RelationNet [54]	36.37 ± 0.92	46.97 ± 0.86	-	-
ArL-ProtoNet [54]	38.76 ± 0.84	51.08 ± 0.84	-	-
ArL-SoSN [54]	41.13 ± 0.84	55.39 ± 0.79	-	-
SimCLR [9]	40.91 ± 0.19	57.22 ± 0.17	65.74 ± 0.15	67.83 ± 0.15
BYOL [16]	39.81 ± 0.18	56.65 ± 0.17	64.58 ± 0.15	66.69 ± 0.15
BarTwins [51]	39.02 ± 0.18	57.20 ± 0.17	65.26 ± 0.15	67.42 ± 0.14
ProtoCLR [30]	44.89 ± 0.58	63.35 ± 0.54	72.27 ± 0.45	74.31 ± 0.45
CPNWCP (ours)	**47.93 ± 0.19**	**66.44 ± 0.17**	**75.69 ± 0.14**	**78.20 ± 0.13**
ProtoNet-Sup [40]	49.42 ± 0.78	68.20 ± 0.66	-	-

deep learning. Besides, there is no gradient calculation in the forward iterations, so the optimization process does not need to calculate the second-order gradients. This regularization method is denoted as Wasserstein Confidence Penalty.

We apply Wasserstein Confidence Penalty to CPN and get CPNWCP (Contrastive Prototypical Network with Wasserstein Confidence Penalty) model which has the training objective of Eq. (5).

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate the models on two standard few-shot learning benchmarks, miniImageNet [42] and tieredImageNet [38]. The miniImageNet dataset is a subset of ImageNet [39] and consists of 100 classes with 600 images per class. Following the commonly-used protocol from [37], we use 64 classes as the base training set, 16 and 20 classes as validation set and test set respectively. The tieredImageNet dataset is a larger subset of ImageNet, composed of 608 classes grouped into 34 high-level categories. These categories are divided into 20 categories for training, 6 categories for validation and 8 categories for test, which corresponds to 351, 97 and 160 classes respectively. In order to simulate the unsupervised setting, we do not use the labels in the training set during training, nor do we use the labeled validation set to select the best checkpoint.

Table 2. Few-shot classification accuracy(%) with 95% confidence interval on 10,000 5-way K-shot tasks randomly sampled from tieredImageNet.

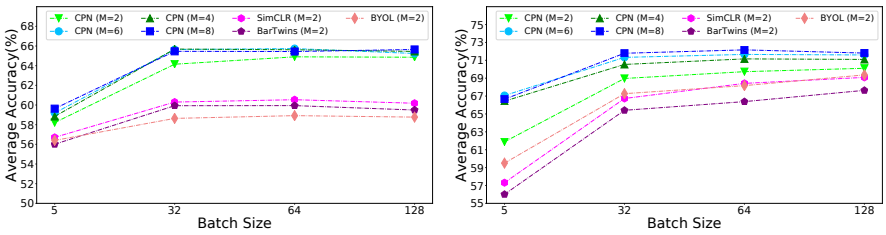
Model	1-shot	5-shot	20-shot	50-shot
Train from scratch [36]	26.27 ± 1.02	34.91 ± 0.63	38.14 ± 0.58	38.67 ± 0.44
ULDA-ProtoNet [36]	41.60 ± 0.64	56.28 ± 0.62	64.07 ± 0.55	66.00 ± 0.54
ULDA-MetaOptNet [36]	41.77 ± 0.65	56.78 ± 0.63	67.21 ± 0.56	71.39 ± 0.53
SimCLR [9]	35.60 ± 0.17	52.88 ± 0.19	61.09 ± 0.17	63.47 ± 0.17
BYOL [16]	37.11 ± 0.18	52.71 ± 0.19	60.56 ± 0.17	62.68 ± 0.16
BarTwins [51]	35.39 ± 0.17	52.01 ± 0.18	60.19 ± 0.17	62.57 ± 0.16
CPNWCP (ours)	**45.00 ± 0.19**	**62.96 ± 0.19**	**72.84 ± 0.17**	**76.03 ± 0.15**
ProtoNet-Sup [40]	53.31 ± 0.89	72.69 ± 0.74	-	-

Implementation details. For fair comparison with previous models [21, 25, 36, 26, 54, 30], we use Conv4 as the backbone which consists of four convolutional blocks with 64 filters for each convolutional layer. We also perform a series of experiments on ResNet12 [19] for comprehensive study. For all experiments, we use random cropping, flip and random color distortion as the data augmentations, like in [25, 30]. The models are trained for 600 epochs using Adam optimizer with the learning rate of 1e-3. We set the batch size $N = 64$ for miniImageNet and $N = 192$ for tieredImageNet. The number of augmented views for each sample is set $M = 4$ for Conv4 and $M = 8$ for ResNet12. For different datasets and backbones, we set the regularization coefficient $\lambda = 1$ and choose scaling factor γ from $\{6, 8, 10, 12\}$. We set the number of Sinkhorn iteration as 5. For Label Smoothing, we choose the label relaxation factor α from $\{0.1, 0.01, 0.001\}$.

Evaluation protocol. We evaluate the models in the standard 5-way 1-shot/5-shot tasks and the tasks with more support samples, i.e., 5-way 20-shot/50-shot tasks. We use 10,000 randomly sampled few-shot tasks with 15 query samples per class, and report the average accuracy (%) as well as 95% confidence interval. Unless otherwise specified, all non-meta-learning models, including our models and contrastive learning models, use a prototype-based nearest neighbor classifier [40] to solve the novel few-shot tasks based on the pre-trained backbone.

4.2 Comparison with State-Of-The-Arts

We compare our full model, CPNWCP, with existing unsupervised few-shot learning models [21, 25, 36, 30, 26, 54] and classical contrastive learning models [9, 16, 51]. Among them, Medina et al. [30] uses a large batch size prototypical loss but not pairwise contrast and Wasserstein Confidence Penalty. We report its results without further fine-tuning for fairness, i.e., the results of ProtoCLR. The results of training a classifier on the support set from scratch is used as the lower bound for performance. The results of supervised ProtoNet [40] are also provided as the supervised baseline. Table 1 and Table 2 provide the results on miniImageNet and tieredImageNet respectively, and the best unsupervised result in each



(a) Backbone: Conv4 (b) Backbone: ResNet12

Fig. 2. Average few-shot classification accuracy across four different settings (5-way 1-shot/5-shot/20-shot/50-shot) on miniImageNet with varying batch size.

Table 3. Few-shot classification accuracy(%) with 95% confidence interval on 10,000 5-way K-shot tasks sampled from miniImageNet. 'PC' represents 'pairwise contrast'.

Model	Backbone	1-shot	5-shot	20-shot	50-shot
CPN w/o PC	Conv4	46.08 ± 0.19	63.89 ± 0.17	72.59 ± 0.14	74.81 ± 0.14
CPN		46.96 ± 0.19	64.75 ± 0.17	73.31 ± 0.14	75.63 ± 0.14
CPN w/o PC	ResNet12	48.80 ± 0.19	69.09 ± 0.16	78.54 ± 0.13	80.83 ± 0.12
CPN		50.01 ± 0.18	70.73 ± 0.16	80.33 ± 0.13	82.74 ± 0.11

setting is in bold. All models use Conv4 as the backbone for fair comparison. As we can see, our CPNWCP outperforms all previous unsupervised few-shot learning models and classical contrastive learning models with a large margin, and achieves the performance much closer to the supervised baseline. Besides, many unsupervised methods (e.g., ULDA, SimCLR and our CPNWCP) have a larger gap to the supervised baseline on tieredImageNet than that on miniImageNet. The reason may be that, the classes in tieredImageNet are structural and there are many similar classes under the same category. These classes rely on label information to be well discriminated for learning relevant semantic knowledge, which is not friendly to unsupervised learning. Despite tieredImageNet is a more difficult dataset, our CPNWCP still significantly outperforms existing baselines.

4.3 Empirical Study on Sampling-Augmentation Paradigm

As shown in Table 1, contrastive learning models [9, 16, 51] outperform the models [21, 25, 36, 26, 54] which construct pseudo few-shot tasks with small batch size, but are inferior to the model [30] using large batch size and prototypical loss. This inspires us that when some meta losses are combined with large batch size, the contrastive losses have no advantage any more. Therefore, we empirically study three important factors in the Sampling-Augmentation paradigm: the loss function $\mathcal{L}$, the batch size N and the view number M. We consider three classical contrastive losses [9, 16, 51] and a meta loss [40]. N is chosen from $\{5, 32, 64, 128\}$ and M is chosen from $\{2, 4, 6, 8\}$. For fair comparison, we intro-

Table 4. Few-shot classification accuracy(%) with 95% confidence interval on 10,000 5-way K-shot tasks sampled from miniImageNet. We provide the results of Consistency Regularization ('+CR') [48], Label Smoothing ('+LS') [41], Confidence Penalty ('+CP') [34], Jensen–Shannon Confidence Penalty ('+JSCP') and our Wasserstein Confidence Penalty ('+WCP').

Model	Backbone	1-shot	5-shot	20-shot	50-shot
CPN	Conv4	46.96 ± 0.19	64.75 ± 0.17	73.31 ± 0.14	75.63 ± 0.14
+ CR [48]		47.33 ± 0.19	65.15 ± 0.17	73.28 ± 0.14	75.50 ± 0.14
+ LS [41]		47.19 ± 0.19	65.22 ± 0.17	74.21 ± 0.14	76.71 ± 0.13
+ CP [34]		47.22 ± 0.19	65.46 ± 0.17	74.52 ± 0.14	77.05 ± 0.13
+ JSCP		46.82 ± 0.19	64.89 ± 0.17	73.92 ± 0.14	76.37 ± 0.13
+ WCP (ours)		**47.93 ± 0.19**	**66.44 ± 0.17**	**75.69 ± 0.14**	**78.20 ± 0.13**
CPN	ResNet12	50.01 ± 0.18	70.73 ± 0.16	80.33 ± 0.13	82.74 ± 0.11
+ CR [48]		51.85 ± 0.19	72.23 ± 0.16	81.35 ± 0.12	83.28 ± 0.11
+ LS [41]		50.41 ± 0.19	71.10 ± 0.16	80.97 ± 0.12	83.61 ± 0.11
+ CP [34]		50.71 ± 0.18	71.29 ± 0.16	81.11 ± 0.12	83.91 ± 0.11
+ JSCP		49.87 ± 0.18	70.53 ± 0.16	81.01 ± 0.13	83.19 ± 0.11
+ WCP (ours)		**53.56 ± 0.19**	**73.21 ± 0.16**	**82.18 ± 0.12**	**84.35 ± 0.11**

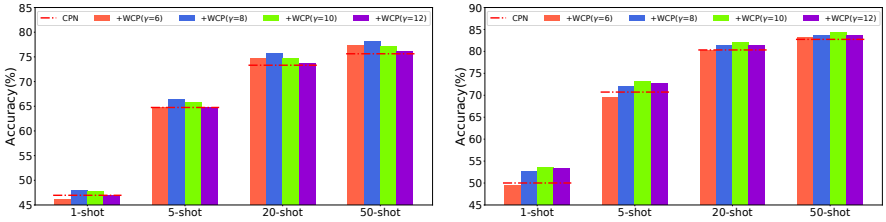
duce pairwise contrast from contrastive learning to the prototypical loss, i.e., each view is used as the one-shot support set to classify other views.

Considering that using the prototype-based nearest-neighbor classifier seems unfair for the comparison between the prototypical loss and contrastive losses, we provide the results with ridge regression classifier [6] in Fig. 2 which examines linear separability. 'CPN' represents prototypical loss with pairwise contrast, 'SimCLR', 'BYOL' and 'BarTwins' represent contrastive losses with pairwise contrast. Similar to Table 1, contrastive learning models with large batch size $N = 64$ outperform the prototypical loss with small batch size $N = 5$. But when the batch size is the same, contrastive learning models consistently perform worse than the prototypical loss. Besides, more augmented views lead to better performance due to increased view diversity, especially for large backbone.

To explore the effect of pairwise contrast, we evaluate the performance of CPN without pairwise contrast, i.e., only using one randomly selected view $\{v_i^l\}_{i=1}^N$ as the one-shot support set to classify all views. The comparison results are shown in Table 3, where we still use a prototype-based nearest neighbor classifier [40]. We set ($N = 64$, $M = 4$) for Conv4 and ($N = 64$, $M = 8$) for ResNet12. As we can see, pairwise contrast achieves consistent performance improvement under different task settings and backbones, especially for large backbone. This shows its potential for larger backbones.

4.4 Ablation Study on Confidence Penalty

As we analyze in Section 3.3, since some negative pairs could be semantically similar or even from the same semantic class, using one-hot prediction target has



(a) Backbone: Conv4 (b) Backbone: ResNet12

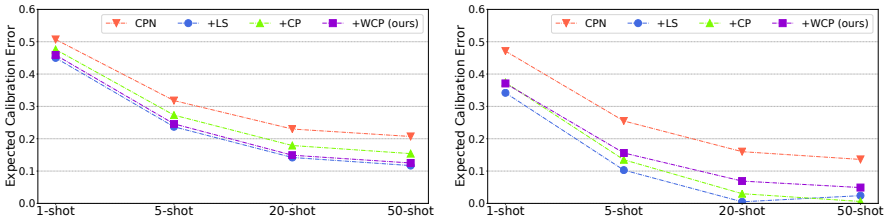
Fig. 3. Few-shot classification accuracy (%) on 10,000 5-way 1-shot/5-shot/20-shot/50-shot tasks sampled from miniImageNet. We provide the results of Wasserstein Confidence Penalty with different scaling factor $\gamma = \{6, 8, 10, 12\}$. The results of CPN are also provided for clear comparison.

the risk of learning sample-specific information instead of generalizing semantic information. This problem is also referred as 'sampling bias' [12] in contrastive learning, and Wei et al. [48] proposes Consistency Regularization to alleviate this problem, which can be directly used in CPN. In this work, we propose to make the prediction distribution approximating a latent distribution to prevent overconfident one-hot prediction. We consider different distance metrics between distributions and compare their effect in CPN, including classical f-divergences and Wasserstein distance. We conduct experiments on miniImageNet and use both Conv4 and ResNet12 as backbone for comprehensive study.

The few-shot classification results are shown in Table 4 and the best result in each setting is in bold. Most regularization methods consistently improve the performance of CPN. Using Jensen-Shannon divergence ('+JSCP') is inferior to using Kullback–Leibler divergence ('+CP' and '+LS'), which means symmetry is not required for the distance metric between the prediction distribution and the latent distribution. Using the Wasserstein distance ('+WCP') outperforms all f-divergence based confidence penalty methods ('+LS', '+CP' and '+JSCP'), which means imposing appropriate penalty based on the semantic relationships among classes helps to learn more general semantic information. Our Wasserstein Confidence Penalty also significantly outperforms the Consistency Regularization ('+CP') [48] in unsupervised few-shot learning.

We also explore the performance of Wasserstein Confidence Penalty with different scaling factor γ, which controls the probability transportation cost between classes. We conduct experiments under the same settings as above and the results are shown in Fig. 3. We provide the results of CPN for clear comparison. As we can see, Wasserstein Confidence Penalty is robust to different γ and can achieve consistent improvement with most candidate scaling factors.

Furthermore, Guo et al. [17] points that although many modern neural networks have better performance, they are poorly calibrated and over-confident. In other words, the confidence of their predictions cannot accurately represent their accuracy, which is potentially harmful in many real-world decision making systems. To measure calibration, the authors propose the estimated Expected



(a) Backbone: Conv4 (b) Backbone: ResNet12

Fig. 4. Expected Calibration Error on different backbones (Conv4 and ResNet12) and settings (5-way 1-shot/5-shot/20-shot/50-shot). We provide the results of CPN model and various confidence penalty methods: Label Smoothing ('+LS') [41], Confidence Penalty ('+CP') [34] and our Wasserstein Confidence Penalty ('+WCP').

Calibration Error (ECE), and it is widely used nowadays. Some works [33, 28, 52] shows that Label Smoothing and Confidence Penalty can improve the calibration of the models in supervised learning. Here we explore whether various confidence penalty methods can improve the calibration of CPN under unsupervised setting. For fair comparison, we calculate the estimated Expected Calibration Error without temperature scaling which not only improves calibration, but also hides the trends in calibration among models [31]. The results calculated on 10,000 sampled tasks are shown in Fig. 4 and the corresponding accuracy is provided in Table 4. As we can see, the models have better calibration property as the support shot increases, which means we can use more support data to simultaneously increase accuracy and improve calibration. Various confidence penalty methods not only improve the few-shot classification accuracy, but also improve the calibration of CPN. Although Label Smoothing has the best calibration property, our Wasserstein Confidence Penalty achieves comparable calibration results and better accuracy on novel few-shot tasks.

5 Conclusions

In this work, we investigate existing unsupervised few-shot learning models and contrastive learning models and find a unified paradigm above them. To find the key design factors for unsupervised few-shot learning, we conduct empirical study and propose CPN model which combines the prototypical loss with pairwise contrast from contrastive learning. Besides, we also provide theoretical or intuitive justification for these key factors. Furthermore, when using a large batch size, CPN has the risk of learning sample-specific information. To this end, we propose Wasserstein Confidence Penalty to prevent overconfident one-hot prediction. Our full model, CPNWCP (Contrastive Prototypical Network with Wasserstein Confidence Penalty), achieves state-of-the-art performance in unsupervised few-shot learning. Besides, Wasserstein Confidence Penalty can also be used in contrastive learning to alleviate the 'sampling bias' problem and we will explore its effect in the future work.

References

1. Altschuler, J., Weed, J., Rigollet, P.: Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. Advances in Neural Information Processing Systems **2017**, 1965–1975 (2017)
2. Antoniou, A., Storkey, A.J.: Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation. CoRR (2019)
3. Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., Saunshi, N.: A theoretical analysis of contrastive unsupervised representation learning. In: 36th International Conference on Machine Learning, ICML 2019. pp. 9904–9923. International Machine Learning Society (IMLS) (2019)
4. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. Advances in Neural Information Processing Systems **32**, 15535–15545 (2019)
5. Baik, S., Choi, J., Kim, H., Cho, D., Min, J., Lee, K.M.: Meta-learning with task-adaptive loss function for few-shot learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9465–9474 (2021)
6. Bertinetto, L., Henriques, J.F., Torr, P.H.S., Vedaldi, A.: Meta-learning with differentiable closed-form solvers. In: 7th International Conference on Learning Representations (2019)
7. Bogachev, V.I., Kolesnikov, A.V.: The monge-kantorovich problem: achievements, connections, and perspectives. Russian Mathematical Surveys **67**(5), 785–890 (2012)
8. Chen, J., Gan, Z., Li, X., Guo, Q., Chen, L., Gao, S., Chung, T., Xu, Y., Zeng, B., Lu, W., Li, F., Carin, L., Tao, C.: Simpler, faster, stronger: Breaking the log-k curse on contrastive learners with flatnce. CoRR **abs/2107.01152** (2021)
9. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
10. Chen, W., Liu, Y., Kira, Z., Wang, Y.F., Huang, J.: A closer look at few-shot classification. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net (2019), https://openreview.net/forum?id=HkxLXnAcFQ
11. Chen, X., He, K.: Exploring simple siamese representation learning. In: IEEE Conference on Computer Vision and Pattern Recognition, 2021. Computer Vision Foundation / IEEE (2021)
12. Chuang, C.Y., Robinson, J., Lin, Y.C., Torralba, A., Jegelka, S.: Debiased contrastive learning. Advances in neural information processing systems **33**, 8765–8775 (2020)
13. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. Advances in neural information processing systems **26**, 2292–2300 (2013)
14. Dhillon, G.S., Chaudhari, P., Ravichandran, A., Soatto, S.: A baseline for few-shot image classification. In: 8th International Conference on Learning Representations, 2020 (2020)
15. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1126–1135. JMLR. org (2017)
16. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Pires, B., Guo, Z., Azar, M., et al.: Bootstrap your own latent: A new approach to self-supervised learning. In: Neural Information Processing Systems (2020)

17. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: International Conference on Machine Learning. pp. 1321–1330. PMLR (2017)
18. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
20. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. In: International Conference on Learning Representations (2018)
21. Hsu, K., Levine, S., Finn, C.: Unsupervised learning via meta-learning. In: 7th International Conference on Learning Representations (2019)
22. Jaynes, E.T.: Information theory and statistical mechanics. Physical review **106**(4), 620 (1957)
23. Ji, Z., Zou, X., Huang, T., Wu, S.: Unsupervised few-shot feature learning via self-supervised training. Frontiers Comput. Neurosci. (2020)
24. Jing, L., Tian, Y.: Self-supervised visual feature learning with deep neural networks: A survey. IEEE transactions on pattern analysis and machine intelligence (2020)
25. Khodadadeh, S., Bölöni, L., Shah, M.: Unsupervised meta-learning for few-shot image classification. In: Wallach, H.M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E.B., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems. pp. 10132–10142 (2019)
26. Khodadadeh, S., Zehtabian, S., Vahidian, S., Wang, W., Lin, B., Bölöni, L.: Unsupervised meta-learning through latent-space interpolation in generative models. In: 9th International Conference on Learning Representations (2021)
27. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10657–10665 (2019)
28. Lienen, J., Hüllermeier, E.: From label smoothing to label relaxation. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence, AAAI, Online (2021)
29. Luo, Y., Chiu, C.C., Jaitly, N., Sutskever, I.: Learning online alignments with continuous rewards policy gradient. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2801–2805. IEEE (2017)
30. Medina, C., Devos, A., Grossglauser, M.: Self-supervised prototypical transfer learning for few-shot classification. arXiv preprint arXiv:2006.11325 (2020)
31. Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., Lucic, M.: Revisiting the calibration of modern neural networks. Advances in Neural Information Processing Systems **34** (2021)
32. Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: International conference on machine learning. pp. 1928–1937. PMLR (2016)
33. Müller, R., Kornblith, S., Hinton, G.E.: When does label smoothing help? Advances in Neural Information Processing Systems **32**, 4694–4703 (2019)
34. Pereyra, G., Tucker, G., Chorowski, J., Kaiser, L., Hinton, G.E.: Regularizing neural networks by penalizing confident output distributions. In: 5th International

Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings (2017)
35. Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., Tucker, G.: On variational bounds of mutual information. In: International Conference on Machine Learning. pp. 5171–5180. PMLR (2019)
36. Qin, T., Li, W., Shi, Y., Gao, Y.: Unsupervised few-shot learning via distribution shift-based augmentation. arXiv preprint arXiv:2004.05805 **2** (2020)
37. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: 5th International Conference on Learning Representations (2017)
38. Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J.B., Larochelle, H., Zemel, R.S.: Meta-learning for semi-supervised few-shot classification. In: 6th International Conference on Learning Representations (2018)
39. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
40. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in neural information processing systems. pp. 4077–4087 (2017)
41. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
42. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. Advances in neural information processing systems **29**, 3630–3638 (2016)
43. Wang, H., Deng, Z.H.: Few-shot learning with lssvm base learner and transductive modules. arXiv preprint arXiv:2009.05786 (2020)
44. Wang, H., Deng, Z.H.: Cross-domain few-shot classification via adversarial task augmentation. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (2021)
45. Wang, H., Guo, X., Deng, Z.H., Lu, Y.: Rethinking minimal sufficient representation in contrastive learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16041–16050 (2022)
46. Wang, H., Mai, H., Deng, Z.h., Yang, C., Zhang, L., Wang, H.y.: Distributed representations of diseases based on co-occurrence relationship. Expert Systems with Applications **183**, 115418 (2021)
47. Wang, X., Zhang, R., Shen, C., Kong, T., Li, L.: Dense contrastive learning for self-supervised visual pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3024–3033 (2021)
48. Wei, C., Wang, H., Shen, W., Yuille, A.L.: CO2: consistent contrast for unsupervised visual representation learning. In: 9th International Conference on Learning Representations (2021)
49. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3733–3742 (2018)
50. Ye, H., Han, L., Zhan, D.: Revisiting unsupervised meta-learning: Amplifying or compensating for the characteristics of few-shot tasks. CoRR (2020)
51. Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: Self-supervised learning via redundancy reduction. In: Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. PMLR (2021)

52. Zhang, C.B., Jiang, P.T., Hou, Q., Wei, Y., Han, Q., Li, Z., Cheng, M.M.: Delving deep into label smoothing. IEEE Transactions on Image Processing **30**, 5984–5996 (2021)
53. Zhang, C., Cai, Y., Lin, G., Shen, C.: Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12203–12213 (2020)
54. Zhang, H., Koniusz, P., Jian, S., Li, H., Torr, P.H.S.: Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021 (2021)
55. Ziko, I.M., Dolz, J., Granger, E., Ayed, I.B.: Laplacian regularized few-shot learning. In: Proceedings of the 37th International Conference on Machine Learning, 2020 (2020)