Coarse-to-Fine Pseudo Supervision Guided Meta-Task optimization for Few-Shot Object Classification

Yawen Cui[1], Qing Liao[2], Dewen Hu[3], Wei An[4], Li Liu[5,1,*]

Abstract

Few-Shot Learning (FSL) is a challenging and practical learning pattern, aiming to solve a target task which has only a few labeled examples. Currently, the field of FSL has made great progress, but largely in the supervised setting, where a large auxiliary labeled dataset is required for offline training. However, the unsupervised FSL (UFSL) problem where the auxiliary dataset is fully unlabeled has been seldom investigated despite of its significant value. This paper focuses on the more general and challenging UFSL problem and presents a novel method named Coarse-to-Fine Pseudo Supervision-guided Meta-Learning (C2FPS-ML) for unsupervised few-shot object classification. It first obtains prior knowledge from an unlabeled auxiliary dataset during unsupervised meta-training, and then use the prior knowledge to assist the downstream few-shot classification task. Coarse-to-Fine Pseudo Supervisions in C2FPS-ML aim to optimize meta-task sampling process in unsupervised meta-training stage which is one of the dominant factors for improving the performance of meta-learning based FSL algorithms. Human can learn new concepts progressively or hierarchically

*Corresponding author. (email: dreamliu2010@gmail.com)
[1]Yawen Cui is with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. (email: yawen.cui@oulu.fi)
[2]Qing Liao is with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. (email: liaoqing@hit.edu.cn)
[3]Dewen Hu is with the College of Intelligent Science, National University of Defense Technology, Changsha, China. (email: dwhu@nudt.edu.cn)
[4]Wei An is with the College of Electronic Science and Technology, National University of Defense Technology, Changsha, China. (email: anwei@nudt.edu.cn)
[5]Li Liu is with the College of System Engineering, National University of Defense Technology, Changsha, China, and is also with the Center for Machine Vision and Signal analysis, University of Oulu, Finland.

following the coarse-to-fine manners. By simulating this human's behaviour, we develop two versions of C2FPS-ML for two different scenarios: one is natural object dataset and another one is other kinds of dataset (*e.g.*, handwritten character dataset). For natural object dataset scenario, we propose to exploit the potential hierarchical semantics of the unlabeled auxiliary dataset to build a tree-like structure of visual concepts. For another scenario, progressive pseudo supervision is obtained by forming clusters in different similarity aspects and is represented by a pyramid-like structure. The obtained structure is applied as the supervision to construct meta-tasks in meta-training stage, and prior knowledge from the unlabeled auxiliary dataset is learned from the coarse-grained level to the fine-grained level. The proposed method sets the new state of the art on the gold-standard *mini*ImageNet and achieves remarkable results on Omniglot while simultaneously increases efficiency.

Keywords: Unsupervised few-shot learning, meta-learning, clustering, object classification.

1. Introduction

In the past decade, deep learning techniques have set the state of the art in nearly every classical computer vision tasks, such as image classification [1, 2], object detection [3, 4], semantic segmentation [5, 6], people re-identification [7], object recognition [8, 9], image synthesis [10], and many others. Much of this huge progress has been achieved by supervised deep learning, which has serious limitations in many real world applications. Firstly, supervised learning is of course constrained to relatively narrow domains defined largely by the training data, and thus leads to limited generalization performance. Secondly, the high performance of deep learning models heavily depends on a large amount of accurately labeled data. However, huge labeled data is expensive and time-consuming to collect. Moreover, in many practical applications like medical image analysis [11], industrial inspection, endangered species recognition and military area, acquiring sufficient labeled training data is extremely expensive

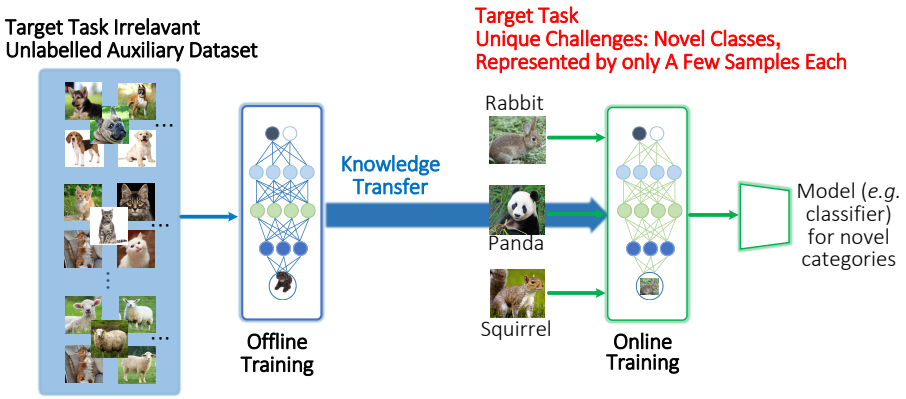


Figure 1: Illustration of the UFSL problem which is the focus of this paper. FSL aims to learn a given task of novel categories presented by just a few samples each. Vanilla FSL task can access a source dataset with annotations, while UFSL is provided a large unlabeled source dataset.

or even prohibitive due to various factors such as privacy or security issues, which imposes significant challenges for applying data and label-hungry deep learning methods. Therefore, there is a pressing need for developing novel methods that are data and label efficient, yet can generalize from limited labeled examples.

Despite the huge progress in Artificial Intelligence (AI) and machine learning, one impressive capability of human intelligence to learn novel concepts from one or several examples has eluded machines as current AI techniques are data hungry and cannot rapidly generalize from a few samples. In order to simulate this humans' learning pattern and learning systems to achieve label efficient learning and the ability of generalizing from very few examples, few-shot learning (FSL) [12, 13, 14, 15, 16], first appeared in [17], has been receiving increasing attention in recent years, and is now a hot research topic. As extreme cases of transfer learning, FSL aims to address the learning of a target task containing a very limited number of labeled examples (like one or several examples), usually using prior knowledge.

Although being a very young topic, many FSL methods [12, 13, 14, 17]

have been proposed in the literature already. A large amount of research in FSL has been devoted to traditional FSL (also referred to vanilla FSL in this paper). Despite of the great value of FSL, researchers cannot deny that FSL is a challenging task due to its intrinsic difficulty which is the easily overfitting problem due to very limited supervised information. In order to alleviate this problem, FSL is provided with a source dataset which is always from the same domain with the novel FSL task. For vanilla FSL, the source dataset contains base categories with a plethora of labeled training instances. The FSL paradigm is first to train the model using a labeled source dataset, and then the learned knowledge is transferred to the novel FSL task. Existing vanilla FSL methods can be categorized into prior knowledge based [18] methods and data augmentation based methods. Data augmentation based methods [19, 20] apply operations on available images in the instance level or the feature level to enlarge the dataset. Prior knowledge based methods can be divied into meta-learning based methods and transfer learning based methods. Recent efforts on meta-learning based vanilla FSL methods contains the following three mainstream approaches: (1) metric learning based methods [21, 22, 12], targeting at learning embedding and metrics functions to measure the distance or similarity among support images and query images; (2) optimization-based methods [14, 23, 24, 25], emphasizing searching for optimal parameter configurations of a given neural network; and (3) memory-based methods [26, 27], modeling the support set of the FSL task as a sequence and the query samples are required to match with the previous obtained knowledge.

Currently, most research in FSL focuses on vanilla FSL problem where a large scale labeled auxiliary dataset is still required. Again, high-quality labels are usually obtained by human workers or even domain experts, and the labeling process is laborious. By contrast, in many scenarios, massive amounts of unlabeled data is easily accessible (such in the Internet). In order to expand the usage of vanilla FSL methods for even more realistic applications, a natural question to ask is: *is it possible to transfer prior knowledge learned from a fully unlabeled large dataset to novel tasks with a few examples?* Therefore,

this practical need stimulates the Unsupervised FSL (UFSL) problem which is illustrated in Fig. 1. In this paper, we attempt to address the more general and challenging UFSL problem, where only a fully unlabeled auxiliary dataset is provided. UFSL is a new emerging problem that has received limited attention [28, 29].

For FSL problems, diverse knowledge from source dataset should be obtained to alleviate the negative effect caused by limited labeled data when adapting to novel few-shot object classification tasks. Therefore, the main challenge of UFSL is how to explore underlying knowledge of unlabeled auxiliary dataset and apply this knowledge to assist novel few-shot task learning process. Recently, meta-learning [18] has emerged as one popular paradigm for tackling the FSL problem. It aims at improving the learning algorithm itself by using the experience from multiple episodes, *i.e.*, a number of meta-tasks. By this means, the learning algorithm obtains the transferable knowledge across multiple tasks, and then generalize to novel but similar downstream tasks.

Sampling meta-tasks can be considered as a combinatorial optimization problem. Assuming that source dataset contains C classes and M samples in total, N classes and K samples per class are selected for a specific meta-task. There are $C_C^N \times (C_M^K)^N$ situations for meta-tasks and we iteratively choose meta-tasks for meta-training process from the total meta-tasks, which is a NP-hard problem. In the vanilla supervised FSL setting, meta-tasks are usually sampled randomly based on lots of strong supervision (*i.e.*, labeled samples). However, the random construction for meta-tasks is not the best solution. [30, 31] optimize meta-task sampling process and achieve remarkable performance. It demonstrates that meta-task sampling is one of dominant factors for the whole meta-learning process. When applying meta-learning paradigm to solve UFSL problem, the challenge is how to sample high-quality and diversiform episodic meta-training tasks from unlabeled auxiliary dataset using abundant underlying information. Hsu *et al.* [28] formed the unlabeled data into clusters and constructed meta-tasks from clusters randomly by regarding each cluster as one particular class. However, the clustering process in the meta-training stage

is time-consuming already, and this method also requires too many iterations of episodic training. Siavash *et al.* [29] randomly selected N samples from the unlabeled source dataset, such that the probability of these N samples belonging to different categories is high. Therefore, these N samples were considered as different categories and constituted as a meta-task with one sample per class, while it could only simulate one-shot learning tasks in the meta-training phase.

However, there is no works about combinatorial optimization on mete-task construction for UFSL. To enrich supervisions in the meta-training stage and alleviate the disadvantages of the existing UFSL methods described above, this paper aims to tackle the combinatorial optimization problem on meta-task sampling in UFSL. Human's learning pattern can follow the coarse-to-fine manners. To be specific, human can learn new concepts hierarchically or progressively by organizing the knowledge in a tree-like or pyramid-like structure. In our natural word, objects are arranged into a hierarchical concept tree based on semantic information. In this work, we assume that unlabeled source dataset and the target dataset are from the same domain, as cross-domain FSL is another challenging issue and is not our focus. Therefore, the underlying hierarchical semantics exists in our problem. Moreover, this hierarchical information has prove to be useful for classification tasks. In YOLO9000 [32], a WordTree was constructed with each node representing a category by following the semantic relations of WordNet [33], and then higher performance was achieved by computing losses following the tree structure in a level-by-level manner. The hierarchical information in Yolo9000 and ours are all related to semantics, but the extracting process is different. For Yolo9000, they check the visual nouns in ImageNet and find their paths through the WordNet graph to the root node. In our method, we use a top-to-down hierarchical clustering process to obtain the hierarchical semantics and build a binary tree. However, for other dataset where hierarchical semantic information is not exist, we propose to seek for the similarity information in different perspectives. Since different perspectives do not have direct connections, we construct a pyramid-like structure of the underlying information.

Based on the inspiration and the analysis above, we propose a method named Coarse-to-Fine Pseudo Supervision-guided Meta-Learning(C2FPS-ML) for few-shot object classification. For the two scenarios, we implement two versions of C2FPS-ML. Fig. 2 shows the outline of C2FPS-ML. For natural object dataset, by conducting the hierarchical clustering process, we exploit the underlying hierarchical nature object categories to build a hierarchical tree structure of pseudo visual concepts. This solution is named Hierarchical Pseudo Supervision-guided Meta-learning (HPS-ML). For other dataset (*e.g.*, handwritten characters), underlying progressive categories structure is constructed by a progressive clustering procedure in different similarity levels. This version of C2FPS-ML is nominated as Progressive Pseudo Supervision-guided Meta-learning (PPS-ML).

Such hierarchical/progressive clustering strategy allows us to extract underlying coarse-to-fine semantic knowledge or similarities in shape or appearance of the unlabeled dataset. Each clustering result in a particular level represents a set of pseudo labels (*i.e.*, cluster IDs) describing the input context, and we refer to these cluster IDs as pseudo labels of unlabeled samples. These pseudo labels are used as the supervision for learning the source dataset. After obtaining the pseudo supervision, in the next stage, we randomly select N clusters from a specific layer of the structure and take K training samples together with query samples from each cluster to formulate the N-way-K-shot task. This operation is done iteratively till the meta-training is completed. Since more diversiform pseudo supervison is provided to unsupervised meta-training process, meta-task sampling process is optimized by selecting tasks from different levels of pseudo supervisions.

The contributions of our work are two folds: (1) we propose the C2FPS-ML method for few-shot object classification by exploiting the potential hierarchical/progressive pseudo supervision. (2) Our extensive experiments on the *mini*ImageNet shows state-of-the-art results by using underlying hierarchical information, and on Omniglot illustrates remarkable results by using progressive pseudo supervision of unlabeled data. (3) At the same time, our proposed

C2FPS-ML can improve the efficiency of meta-training stage by decreasing training iterations largely.

2. Related Work

2.1. Vanilla Few-Shot Learning

In the past several decades, researches on FSL can be categorized into data augmentation based methods and prior knowledge based methods. Prior knowledge based methods contains two series of methods: meta-learning based methods and transfer-learning based methods. Data augmentation based methods [19, 20], applying operations on available images in the instance level or the feature level to enlarge the dataset. Chen *et al.* [20] proposed to learn a mapping from a novel sample instance to a concept and relate that concept to the existing ones in the concept space, and new instances were generated by using these relationships.

Recent efforts on meta-learning based FSL are mainly toward the following aspects. (i)Metric learning based methods [21, 22, 12], targeting at learning an embedding and useful metrics function, and using the metrics to measure the distance or similarity among support images and query images. Vinyals *et al.* first applied metric learning on FSL methods, and combined with attention mechanisms. The training image and the test image were mapped into embedding space, and attention mechanisms were used to get the similarity of images. Sung *et al.* proposed a FSL method in which the metric function was learnable. (ii) Optimization-based methods [14, 23], emphasizing searching for parameter configurations of a given neural network such that it can effectively fine-tune on FSL tasks within a few gradient-descent update steps. The main idea of MAML [14] is to obtain optimal initialization parameters of the model through training. For a novel task, the model can achieve better performance with a few gradient steps. (iii) Memory-based methods [26, 27], modeling the support set of the FSL task as a sequence and formulating it as a sequence learning task. The query samples are required to match with the previous

obtained knowledge. The ability of a memory-augmented neural network to rapidly assimilate new data, and leverage this data to make accurate predictions after only a few samples. Santoro *et al.* [26] demonstrated the ability of a memory-augmented neural network to rapidly assimilate new data, and leverage this data to make accurate predictions after only a few samples.

However, transfer learning based methods [34, 35] do not use episodic strategy but use traditional learning strategy by pre-training a model with a large-scale dataset and fine-tuning on FSL task. TransMatch [34] pre-trained a feature extractor on base-class data, then used the feature extractor to initialize the classifier weights for the novel classes, and further updated the model with a semi-supervised learning method. In this paper, we use the meta-learning technique combining optimization-based architecture and metric learning based architecture to implement our proposed UFSL method.

2.2. Semi-Supervised and Unsupervised Few-Shot Learning

Semi-supervised learning focuses on promoting learning performance with labeled data by leveraging a large amount of unlabeled data. It is mainly divided into consistency regularity methods [36], entropy minimization methods [37] and teacher-student model [38]. As for semi-supervised few-shot learning, [39] proposed BR-ProtoNet to enabled metric learning to benefit from readily-available unlabeled data. [40] extended the prototypical network by incorporating unlabeled data to update the prototypes generated by labeled images. [41] used the transductive propagation network to propagate labels from labeled images to unlabeled images along with a constructed graph. [42] adopted self-training by adding unlabeled data to the meta-learning process.

Unsupervised learning [43] is a type of machine learning that looks for previously undetected patterns in a dataset with no pre-existing labels and minimal human supervision. For UFSL, [28] applied clustering on source data to formed unlabeled data into clusters first and constructed meta-tasks from clusters randomly by regarding every cluster as one certain class. [29] selected N samples from unlabeled training set randomly, and the probability that these

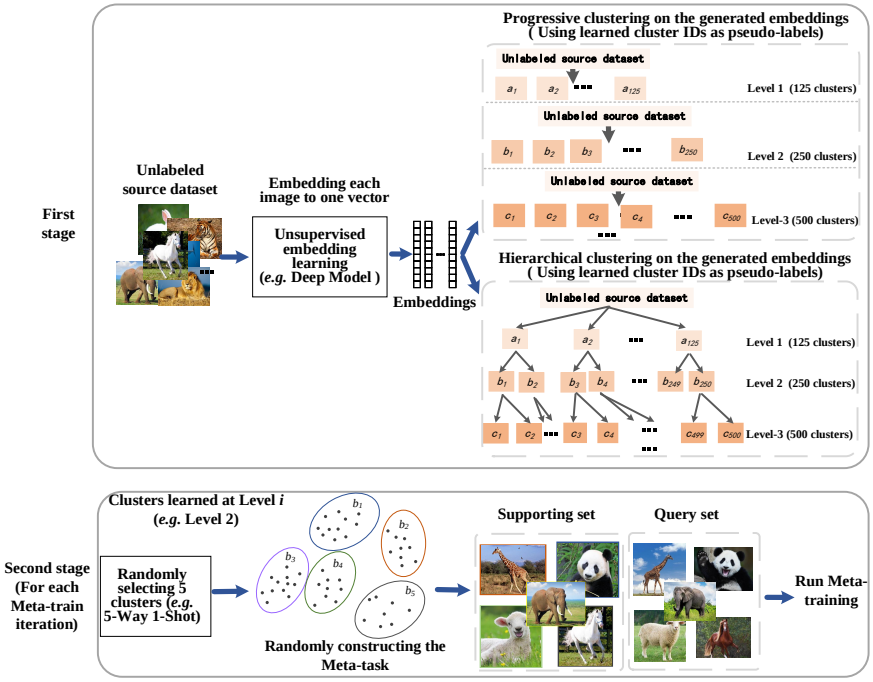


Figure 2: Illustration of meta-training process in C2FPS-ML. First stage: an unsupervised embedding learning algorithm embeds instances, then the underlying hierarchical/progressive pseudo supervision of unlabeled data are obtained by hierarchical/progressive clustering. Second stage: meta-tasks are constructed automatically in every level by regarding cluster IDs as pseudo labels of the unlabeled dataset.

N pictures belonged to different categories was very high, so these N pictures were constituted as one N-way 1-shot meta-task. Progressive clustering and episodic training were used in UFLST [44] to implement unsupervised meta-training. AAL [45] used data augmentation of the unlabeled support set to generate the query data.

3. Problem Setup

FSL formulates as N-way-K-shot classification, *i.e.*, each task includes N classes with K examples for each class. These N-way-K-shot images are known

as the supporting set. In addition, there are more examples of the same classes with the support set known as the query set, which is used for evaluating the performance of the learned task.

Assume an unlabeled dataset $\mathcal{D}^u = \{\boldsymbol{x}_i\}$ used in meta-training phase, and a novel set $\mathcal{D}^n$, which contains unseen classes with only one or a few training samples per class used in the meta-test process. In UFSL, it is crucial that the model $F(.)$, which is primarily trained on unlabeled data, can generalize to the novel classes with a few sample per class. We use N-way K-shot strategy for evaluation in the meta-test process, in which we apply on novel categories $\mathcal{D}^n$.

4. Coarse-to-Fine Pseudo Supervision-guided Meta-Learning(C2FPS-ML)

For few-shot object classification task, there are two types of datasets: one is natural object dataset, like miniImageNet, the rest dataset is other types like character dataset (e.g., Omniglot). For natural object dataset, labeling the objects follows a hierarchical structure. Because we know the unlabeled source dataset and target dataset are from the same domain, the underlying hierarchical semantic information of unlabeled source dataset still exists. For human, the learning pattern follows a coarse-to-fine manner. For example, one book is often organized into a hierarchical manner, i.e., the outline. When people plan to read this book, the best way is to follow the outline which shows the coarse-to-fine contents. The other kind of datasets do not contain hierarchical semantic information. For example, Omniglot consists of handwritten characters from 50 alphabets with 20 instances written by different people from 50 different alphabets. Characters do not contain word level or semantic level information. However, they may have similarities in shape or appearance, like the character 'B' and 'D'.

Toward UFSL problem, we propose a method named C2FPS-ML to solve it, and we provide two solutions for C2FPS-ML to tackle two different kinds of datasets. Fig. 2 shows an illustration of meta-training process in our

proposed C2FPS-ML. The meta-training process of C2FPS-ML is divided into two procedures. In the first stage, we extract potential hierarchical/progressive pseudo supervision of unlabeled samples using a hierarchical/progressive clustering approach. Before clustering, unlabeled source dataset are fed into unsupervised learning algorithms to generate the corresponding embedding first. In the second stage, we exploit this obtained information as pseudo supervision signals for the meta-training in UFSL. In this section, extracting coarse-to-fine pseudo supervision of the two solutions (*i.e.*, HPS-ML and PPS-ML) are first illustrated in detail. Then, the total meta-training process is demonstrated and meta-learning process is illustrated in Fig. 4.

4.1. Extracting Hierarchical Pseudo Supervision

For natural object dataset, annotating the dataset follows a lexical database for English. This kind of lexical dataset is always arranged in a net-like structure and all the annotations in a dataset can be organized in a tree-like structure base on hierarchical semantics from the lexical dataset. As for the common type of FSL, unlabeled source dataset and target dataset are from the same domain. The same domain means that unlabeled source dataset shares the same hierarchical space with the labeled target dataset. Therefore, assuming that there must exist underlying hierarchical semantics in unlabeled source dataset like target dataset and can be organized to the tree-like structure. Finally, we propose Hierarchical Pseudo Supervision-guided Meta-learning (HPS-ML) for few-shot natural object image classification. In order to derive the hierarchical semantics, our model proceeds in a top-down way.

WordNet concept graph in [32] where relationships between labels are represented by a tree has proved that hierarchical semantic information can enhance the performance of classification tasks. An example of the tree for ImageNet [46] is shown in Fig. 3. From the previous analysis, we know that the hierarchical information in YOLO9000 and ours are all hierarchical semantics, just the obtaining process is different. We aim to use the part of the hierarchical structure in ImageNet to illustrate how the hierarchical semantics looks like.

As for one particular image, it belongs to different classes in different levels. In Fig. 3, one image belongs to "plant" in the third level, "natural object" the second level, and "physical object" in the first level. Classification tasks are implemented by computing novel conditional losses level by level:

$$Pr(Airplane) = Pr(Airplane|Air) * Pr(Air|Vehicle) \\ *Pr(Vehicle|PhysicalObject).$$

With the semantic hierarchy, the classification task performs better by computing the conditional loss level by level.

In UFSL, we suppose that the auxiliary source dataset and target dataset are from the same domain, so the underlying hierarchical categories of the auxiliary source dataset exists. We aim to extract meaningful pseudo labels for each image based on its content in a tree-like (*i.e.*, hierarchical) manner. In the meta-training phase of HPS-ML, extracting underlying hierarchical pseudo supervision involves two steps. In the first step, in order to cluster in spaces where common distance functions correlate to semantic meaning, unlabeled samples in $\mathcal{D}^u$ are fed into the unsupervised learning algorithm $\mathcal{A}$ to generate embedding spaces. Because clustering in the pixel-level is difficult in practice due to the high dimensionality of raw images, and unreasonable due to the distance poorly correlating with semantic meaning [28]. The semantic embedding usually has the strong representative characteristic that can be useful for several downstream tasks, including clustering. In the second step, we further conduct a hierarchical clustering process in the embedding space to extract the underlying hierarchical nature object categories and build a hierarchical tree of pseudo visual concepts (*i.e.*, pseudo-labels). These pseudo labels are used for constructing meta-tasks of UFSL in meta-training stage. A related work [28] required too many iterations to achieve good performance. Too many iterations compromise the efficiency of the UFSL. To alleviate this drawback, we opt to use underlying hierarchical supervision of unlabeled data. This is a simple, but efficient technique to take advantage of privilege underlying information of unlabeled samples to generate pseudo-labels for the meta-training

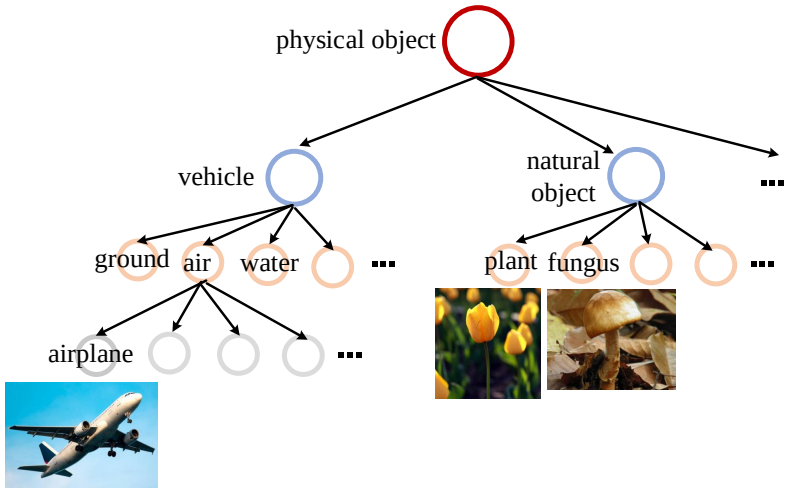


Figure 3: Part of hierarchical structure in ImageNet. One image can be classified into different classes in different semantic levels.

phase.

In order to extract the rich underlying nature object categories of unlabeled data, we use a hierarchical clustering technique in a top-down manner to cluster generated embeddings at different levels. Hence, we generate a tree of hierarchical semantic information representing pseudo-labels for each level (*i.e.*, depth of the tree). Assume the depth of the tree L and the root node containing all unlabeled samples. The level is denoted as L_i, where the subscript i denotes the i-th level. First, we cluster the embeddings and gain a series of C clusters represented by C_1. We generate different branches of the tree by applying the intra-clustering on each cluster to form P partitions, smaller clusters with narrower semantics are gained. Hence, we divide each cluster into more detailed clusters in the next level of the tree. In this way, we are able to create a tree of pseudo nature object categories describing the content of unlabeled images, in which the upper levels contain more abstract categories and the bottom levels represent fine-grained categories.

Fig. 2 illustrates an example of the clustering results: C is 125, P equals to

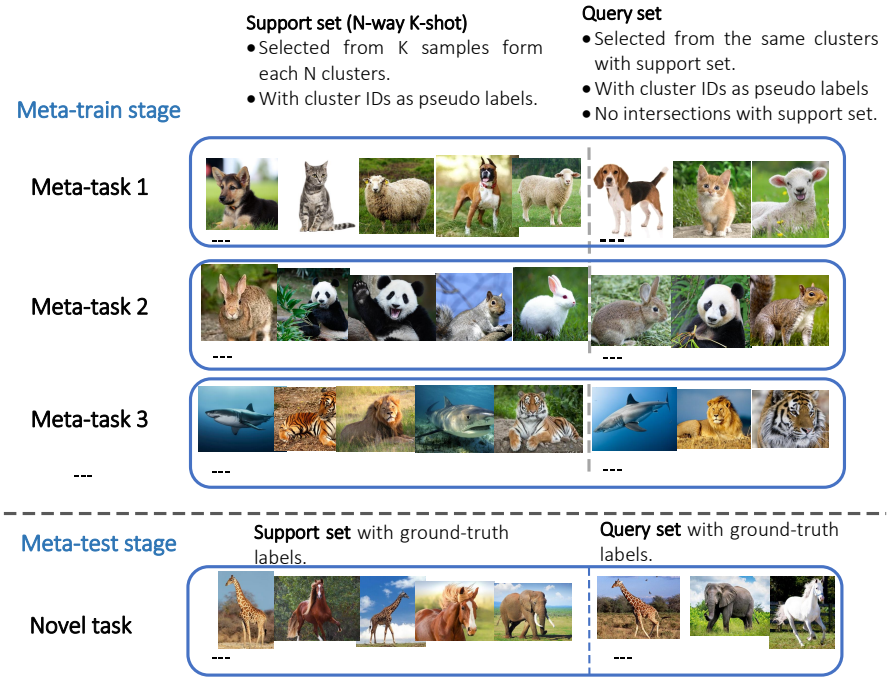


Figure 4: The whole meta-learning process of C2FPS-ML. Meta-learning based UFSL contains unsupervised meta-training stage and meta-test stage. In the meta-training stage, our proposed C2FPS-ML explores coarse-to-fine supervision and use the supervision to learn tasks in an episodic manner.

2 and hierarchical levels L is set as 3. When the process is done, the obtained structure contains: the node in head-level representing the whole unsupervised dataset, 125 first-level nodes, and each of following-levels' nodes with one father node and one brother node.

4.2. Extracting Progressive Pseudo Supervision

As for other kinds of dataset (*e.g.*, handwritten characters), different objects have weak relations in semantic levels, or even they do not contain semantic-level information. Therefore, we cannot use some rules to describe the semantic relations among different samples. However, similarities in shape or appearance from different perspective always exists, such as the layout, the writing style, etc.

Algorithm 1 The unsupervised meta-training in HPS-ML.

Input: $\mathcal{D}^u$, L, $F(.)$, P, $\mathcal{A}$, N, K, C, Q, the meta-training iteration number I

Output:$F(.)$ after meta-training stage

```
Generating embeddings of D^u by unsupervised feature embedding algorithm A;
for l in levels L do
    if l==1 then
        Divide dataset into C parts C_1 by clustering;
    else
        Separating each previous-level cluster C_{l-1} into P parts;
while Meta-training iterations I not done do
    Select N clusters from an certain level of clusters;
    Select K training samples and Q validation samples to construct meta-task;
    Update F(.) by learning the meta-task;
return F(.)
```

From multiple perspective, we can divide the dataset into different partitions. However, direct connections are not contained in different partition aspects, so we cannot conduct splitting process from the one perspective to generate other division results of other perspectives. Toward this situation, we propose Progressive Pseudo Supervision-guided Meta-learning (PPS-ML) to construct the pyramid-like structure of unlabeled data.

In order to form the pyramid structure of unlabeled data, we use a progressive clustering technique to cluster generated embedding at different similarity levels. To be specific, we use clustering algorithm to cluster the embedding into a fixed number of clusters in a progressive fashion. Assume the depth of the tree L and the top node as the head level. Its levels are denoted as L_i, where the subscript i denotes the i-th level. First, for the level l_1, we define C_1 cluster IDs and cluster the embedding into these cluster IDs.

Algorithm 2 The unsupervised meta-training in PPS-ML.

Input: $\mathcal{D}^u$, L, $F(.)$, P, $\mathcal{A}$, N, K, C, Q, the meta-training iteration number I

Output:$F(.)$ after meta-training stage

Generating embeddings of $\mathcal{D}^u$ by unsupervised feature embedding algorithm $\mathcal{A}$;
for l in levels L **do**
    Divide dataset into C, $2C$ and $3C$ parts, respectively;
while Meta-training iterations I *not done* **do**
    Select N clusters from an certain level of clusters;
    Select K training samples and Q validation samples to construct meta-task;
    Update $F(.)$ by learning the meta-task;
return $F(.)$

We increase the number of clusters to C_2 (where $C_2 > C_1$) in the next level l_2 and forme the embedding into C_2. The number of clusters continually increase in the following levels. Fig. 2 illustrates an example of the clustering results: C_1 is 125, C_2 is 250 and C_3 is 500.

4.3. Unsupervised Meta-Training for Meta-Testing

In the meta-training phase of C2FPS-ML, cluster IDs are regarded as pseudo labels for unlabeled samples in each cluster. Meta-task sampling process is optimized by constructing tasks from the extracted coarse-to-fine pseudo supervisions. The total meta-training process of HPS-ML is illustrated in Algorithm 1. The algorithm of PPS-ML illustrated in Algorithm 2 is the same as that of HPS-ML except for the clustering process. The total meta-learning procedure is presented in Fig. 4. The meta-training performs in an iterative way. In each iteration, we randomly choose N clusters from a particular level of the tree. Then, K samples from each cluster are selected to constitute the supporting set, and we choose Q samples from the rest of samples in each cluster to form the query set. This selection process forms a N-way K-shot

Table 1: Results of HPS-ML on *mini*ImageNet. We obtain state-of-the-art results in 4 settings, and exceed previous best results by 0.19%, 0.54%, 1.23% and 0.5%, respectively.

Algorithm	**N-way K-shot (N, K)**			
	(5,1)	**(5,5)**	**(5,20)**	**(5,50)**
Training from scratch [28]	27.59%	38.48%	51.53%	59.63%
DeepCluster $\mathbf{k_{nn}}$-nearest neighbours [28]	28.90%	42.25%	56.44%	63.90%
DeepCluster linear classifier [28]	29.44%	39.79%	56.19%	65.28%
DeepCluster MLP with dropout [28]	29.03%	39.67%	52.71%	60.95%
DeepCluster clusering matching [28]	22.20%	23.50%	24.97%	26.87%
DeepCluster CACTUs-ProtoNets [28]	39.18%	53.36%	61.54%	63.55%
DeepCluster CACTUs-MAML [28]	39.90%	53.97%	63.84%	69.64%
UMTRA [29]	39.93%	50.73%	61.11%	67.15%
AAL-ProtoNets [45]	37.67%	40.29%	-	-
AAL-MAML++ [45]	34.57%	49.18%	-	-
UFLST [44]	33.77%	45.03%	53.35%	56.72%
HPS-ML-ProtoNets (ours)	39.28%	53.44%	61.57%	63.88%
HPS-ML-MAML (ours)	**40.09**%(↑ 0.19)	**54.51**%(↑ 0.54)	**65.07**%(↑ 1.23)	**70.14**%(↑ 0.5)

learning problem. In each iteration, the model solves meta-tasks by learning to classify images based on pseudo labels. This process repeats till completing unsupervised meta-training. In our proposed method, we construct a meta-task by sampling clusters from the same level, because if we select clusters from different levels, one image may be all selected in different levels. In this case, one unlabeled images have more than one pseudo labels, and it may confuse the classification algorithm. In the meta-test phase, novel FSL tasks, following the same N-way-K-shot learning pattern with meta-tasks in meta-training stage, are learned by fine-tuning the obtained model by using N-way-K-shot samples. The average validation accuracy of novel tasks is considered as the evaluation indicator.

5. Experiment

In this section, we first introduce the experimental setup containing two datasets we used for evaluate our proposed two solutions and parameter configurations. Then, we present the classification results obtained by implementing C2FPS-ML in two widely-used FSL methods. Finally, we conduct ablation

experiments toward two aspects and illustrate the results to demonstrate the efficiency of C2FPS-ML.

5.1. Experimental Setup

Omniglot[13] is frequently-used few-shot learning dataset, which consists of $1,623$ handwritten characters from 50 different alphabets with 20 instances written by different people in every character category. In order to compare our methodology with existing published work, we follow the experimental protocol described in [26]: $1,200$ characters were used for meta-training, 100 characters were used for validation in meta-training and 323 characters were used for meta-testing. In our problem setting, the characters we used for mate-training and validation are all unlabeled and we select supervised novel meta-tasks of meta-test stage in labeled meta-test dataset.

***mini*ImageNet** [22] is a subset of the ImageNet [46] with relatively fewer number of classes. It includes 600 natural object images for each of 100 classes. In this paper, we adopted the class split as in [22], *i.e.*, 64 classes for training, 16 classes for validation, and 20 classes for test. These images are in size of 84×84. In our experiments, we discarded all labels of training and validation data and used the labeled test dataset for meta-test phase.

We implemented C2FPS-ML based on Model-Agonstic Meta-Learning (MAML) [14] and Prototypical Networks (ProtoNets) [12]. MAML is different from previous optimization-based meta-learning algorithms for FSL or other tasks. The core idea of MAML is to train the model's initial parameters such that the model has maximal performance on a new FSL task after the parameters have been updated through a few gradient steps. Moreover, MAML does not constraint the model architecture. ProtoNets learns a metric space in which classification can be performed by computing distances to prototype representations of each class. Compared to recent approaches for FSL, they reflect a more straightforward inductive bias that is beneficial in the limited-data regime and achieve excellent results.

In this paper, we used the same model architectures as [28] for the fair

Table 2: Results on *mini*ImageNet obtained without random scaling and with 50 times random scaling. Compared with [28], HPS-ML performs better in all settings.

Algorithm	**Random scaling**	**N-way K-shot (N, K)**			
		(5,1)	**(5,5)**	**(5,20)**	**(5,50)**
CACTUs-ProtoNets [28]	50 times	39.18%	53.36%	61.54%	63.55%
HPS-ML-ProtoNets (ours)	50 times	**39.28**%	**53.44**%	**61.57**%	**63.88**%
CACTUs-MAML [28]	No	38.75%	52.73%	62.72%	67.77%
HPS-ML-MAML (ours)	No	**38.86**%	**53.06**%	**63.69**%	**69.52**%
CACTUs-MAML [28]	50 times	39.90%	53.97%	63.84%	69.64%
HPS-ML-MAML (ours)	50 times	**40.09**%	**54.51**%	**65.07**%	**70.14**%

comparison. For HPS/PPS-ML-MAML, we used the same four blocks as MAML with 64 filters for each convolutional layer. The outer-loop optimizer was Adam, and the inner-loop optimizer was SGD. For HPS-ML-ProtoNets, we used 4-block convolutional archicture, in which each block consisted of a convolutional layer with 64 3×3 filters, stride 1, and padding 1, followed by BatchNorm, ReLU activation, and 2×2 MaxPooling. Adam optimizer was used in HPS-ML-ProtoNets. Before extracting the underlying structure, unlabeled instances were processed by the unsupervised embedding algorithms. We used DeepCluster [47] to embed *mini*ImageNet, while Adversarially Constrained Autoencoder Interpolation (ACAI) [48] and Bidirectional GAN (BiGAN) [49] to embed Omniglot. We followed different training strategies to evaluate the effectiveness of HPS/PPS-ML, *i.e.* (I) training from scratch [28], only using few labeled instances of the novel task to train the model and get the performance by test images from the same novel categories, (II) $\mathbf{k_{nn}}$-nearest neighbors [28], (III) linear classifier [28], (IV) MLP with dropout [28], (V) clustering matching [28], (VI) CACTUs [28], (VII) UMTRA [29], (VII) AAL [45], and (IX) UFLST [44].

We evaluated the performance of HPS-ML on the *mini*ImageNet with the following settings: 5-way 1-shot, 5-way 5-shot, 5-way 20-shot, and 5-way 50-shot. The number of clusters in [28] was 500. [28] formed pseudo labels of unlabeled

Table 3: Results of PPS-ML-MAML on Omniglot. Our method does not achieve state-of-the-art results in 4 settings, but exceeds the performance of CACTUs-MAML [28] in two settings, which is related to ours.

Algorithm	**N-way K-shot (N, K)**			
	(5,1)	**(5,5)**	**(20,1)**	**(20,5)**
Training from scratch	52.52%	74.78%	24.91%	47.62%
ACAI $\mathbf{k_{nn}}$-nearest neighbours	57.46%	81.16%	39.73%	66.38%
BiGAN $\mathbf{k_{nn}}$-nearest neighbours	49.55%	68.06%	27.37%	46.70%
ACAI linear classifier	61.08%	81.82%	43.20%	66.33%
BiGAN linear classifier	48.28%	68.72%	27.80%	45.82%
ACAI MLP with dropout	51.95%	77.20%	30.65%	58.62%
BiGAN MLP with dropout	40.54%	62.56%	19.92%	40.71%
ACAI clusering matching	54.94%	71.09%	32.19%	45.93%
BiGAN clusering matching	43.96%	58.62%	21.54%	31.06%
ACAI CACTUs-MAML [28]	68.84%	87.78%	48.09%	73.76%
BiGAN CACTUs-MAML [28]	58.18%	78.66%	35.56%	58.62%
ACAI PPS-ML-MAML (ours)	**69.00**%	**87.88**%	47.62%	72.74%
BiGAN PPS-ML-MAML (ours)	58.20%	78.67 %	39.90%	58.68 %

images in an extreme fine-grained level, since the total number of classes in unlabeled dataset was less than 100. We also set the number of clusters in the last level as 500. Numbers of samples in clusters should be balanced in order to construct meta-tasks, or samples in a particular cluster may not be enough for selecting N-way K-shot tasks. In order to construct balanced clusters and alleviate inadequate situations that the number of samples in a specific cluster was less than K, we represented the potential underlying hierarchical categories by the binary tree. Therefore, we set hierarchical levels L as three and the number of the first level clusters C as 125. The partition number P was two, which meant separating each previous-level cluster into two parts. We used k-means for conducting 3-level clustering: first level with 125 clusters, second level with 250 clusters, third level with 500 clusters. The aim that we use k-means as clustering method is for fair comparison with the baseline method [28].

We evaluated the performance of PPS-ML on the Omniglot with the

Table 4: Results on Omniglot obtained without random scaling and with 100 times random scaling. Compared with [28], PPS-ML performs better in nearly all settings.

Algorithm	Random scaling	N-way K-shot (N, K)			
		(5,1)	(5,5)	(20,1)	(20,5)
ACAI CACTUs-MAML [28]	No	66.49%	85.60%	45.04%	69.14%
ACAI CACTUs-MAML [28]	100 times	68.84%	87.78%	48.09%	73.76%
ACAI PPS-ML-MAML (ours)	No	**68.50**%	**86.42**%	**45.87**%	**71.00**%
ACAI PPS-ML-MAML (ours)	100 times	**69.00**%	**87.88**%	47.62%	72.74%
BiGAN CACTUs-MAML [28]	No	55.92%	76.28%	32.44%	54.22%
BiGAN CACTUs-MAML [28]	100 times	58.18%	78.66%	35.56%	58.62%
BiGAN PPS-ML-MAML (ours)	No	**57.35**%	**77.88** %	**34.54**%	**56.63** %
BiGAN PPS-ML-MAML (ours)	100 times	**58.20**%	**78.67** %	**39.90**%	**58.68** %

following settings: 5-way 1-shot, 5-way 5-shot, 20-way 1-shot, and 20-way 5-shot. The number of clusters in [28] was 500. We also set the number of clusters in the last level as 500, and we set progressive levels L as three. Moreover, C_1 is 125, and C_2 is 250.

5.2. Results

The results obtained by these two implemented frameworks are illustrated in this section, and we also compare our results with the most related method in [28]. Moreover, we present the iteration numbers in meta-training stage to demonstrate the efficiency of C2FPS-ML.

[28] applied random scaling to the dimensions of embedding spaces for inducing different metrics during clustering. The results on *mini*ImageNet in Table 1 are also obtained by 50 times random scaling. By HPS-ML-MAML, we obtain state-of-the-art results for UFSL classification on *mini*ImageNet. Accuracies of 5-way 1-shot and 5-way 5-shot settings are 40.09% and 54.51%. In 5-way 20-shot and 5-way 50-shot settings, we obtain 64.27% and 70.14%.

The clustering was also used in CACTUs [28], while unlabeled source samples were divided into a certain number of clusters. Although the dimensions of generated embeddings were randomly scaled 50 times to induce different metrics and generate diverse meta-tasks, the extracted information only contained simple level supervision. In [28], embeddings were divided into 500 clusters

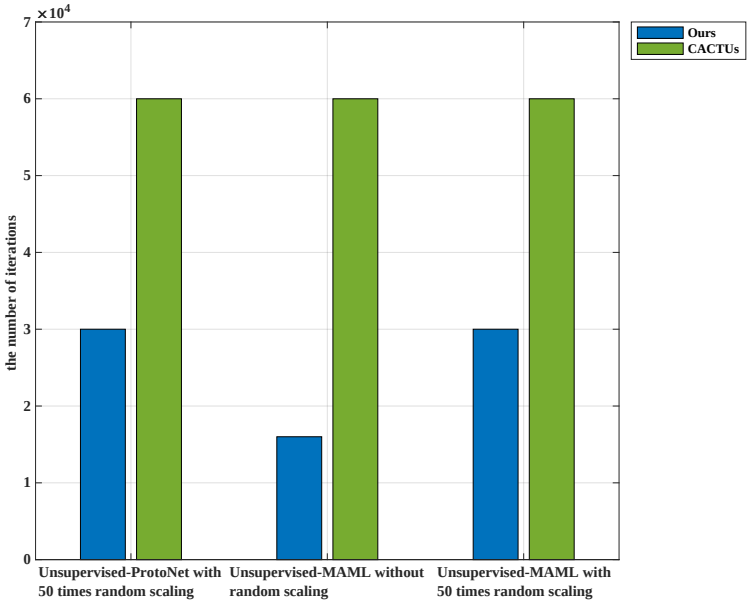
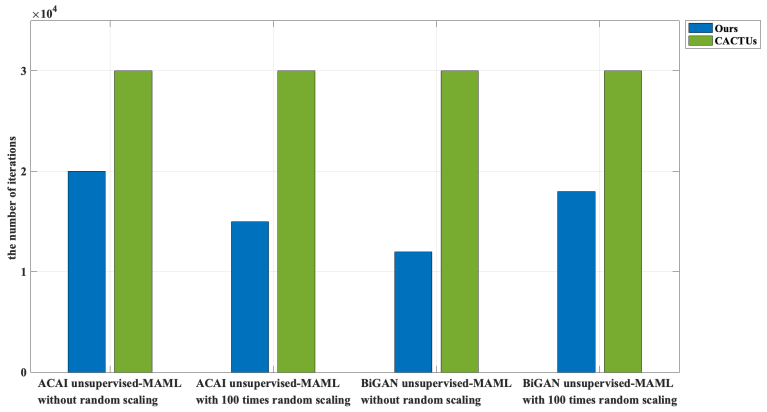


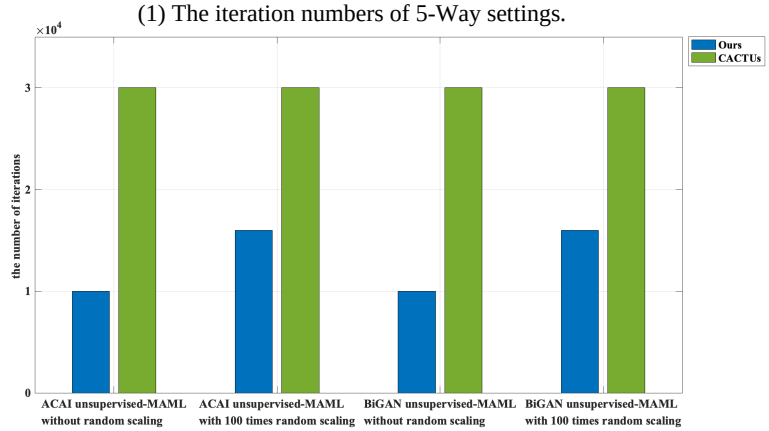
Figure 5: The number of iterations in the meta-training stage of HPS-ML. Our proposed HPS-ML needs less training iterations than [28].

for 50 times. We also conduct the random scaling in proposed method. The comparison results of *mini*ImageNet without random scaling and with 50 times random scaling are summarized in Table 2. Our proposed HPS-ML performs better than [28] in nearly all settings. Therefore, we demonstrate that optimizing meta-task sampling with the underlying hierarchical semantics of unlabeled source dataset can improve the performance of UFSL.

The results on Omniglot in Table 3 are also obtained by 100 times random scaling. By PPS-ML-MAML, we do not achieve the best results for UFSL classification on Omniglot. However, compared with CACTUs [28] which is most related with PPS-ML-MAML, our results outperform its results in nearly all settings. The comparison results of Omniglot without random scaling and with 100 times random scaling are illustrate in Table 4. Our proposed PPS-ML performs better than [28] in most cases. For UFLTS [45] which obtains the state-of-art results, progressive clustering is also used. UFLTS focuses on applying advanced and complex progressive cluster algorithms on the extracted features



(1) The iteration numbers of 5-Way settings.



(2) The iteration numbers of 20-Way settings.

Figure 6: The number of iterations in the meta-training stage. Our proposed PPS-ML needs less training iterations than [28].

to obtain better partitions than CACTUs [28]. For the intermediate stage results of progressive clustering, UFLTS does not use them in meta-training stage, which is similar with CACTUs except for the complicated clustering methods. Compared with our methods, UFLTS uses complex clustering methods, while we apply simple k-means algorithm. Moreover, we use the intermediate stage results of progressive clustering, which is totally different from CACTUs.

Compared with CACTUs [28], Fig. 5 and Fig. 6 shows the number of iterations in the meta-training stage. The method in [28] needs nearly 60,000 iterations to achieve its optimal performance on *mini*ImageNet. Although the

Table 5: Compare the result of *mini*ImageNet obtained by PPS-ML-MAML with that of HPS-ML-MAML. HPS-ML-MAML outperforms PPS-ML-MAML on all settings, and the results illustrated that the performance would become bad without hierarchical knowledge.

Algorithm	**Random scaling**	**N-way K-shot (N, K)**			
		(5,1)	**(5,5)**	**(5,20)**	**(5,50)**
PPS-ML-MAML (ours)	No	38.85%	52.26%	63.32%	68.53%
HPS-ML-MAML (ours)	No	**38.86**%	**53.06**%	**63.69**%	**69.52**%
PPS-ML-MAML (ours)	50 times	38.94%	53.05%	63.73%	68.91%
HPS-ML-MAML (ours)	50 times	**40.09**%	**54.51**%	**65.07**%	**70.14**%

performance of FSL is measured by novel tasks in the meta-test stage, the training process might be hindered by the computational resources resulting in longer training. However, it takes maximum 30,000 iterations for HPS-ML to reach its highest performance, which is half of the iteration number in [28]. This shows the efficiency of our proposed HPS-ML. For omniglot, CACTUs needs 30000 iterations to obtain the satisfying results. PPS-ML only requires maximum 20000 iterations in four settings.

5.3. Ablation Study

We conduct ablation experiments toward two different aspects. The first aspect is PPS-ML on *mini*ImageNet and HPS-ML on Omniglot. The second part is the study on clustering levels and iterations in our proposed C2FPS-ML, which are hyper-parameters.

5.3.1. PPS-ML on mini*ImageNet and HPS-ML on Omniglot*

To explore the effect of various coarse-to-fine supervisions on different classification tasks, we analyze the performance of PPS-ML on *mini*ImageNet and HPS-ML on Omniglot.

For *mini*ImageNet, if we use PPS-ML for the classification task, the hierarchical information were not extracted or stored using the progressive manner. We discarded the clustering results of the former level and formed new

Table 6: Compare the result of Omniglot obtained by HPS-ML-MAML with that of PPS-ML-MAML. PPS-ML-MAML outperforms HPS-ML-MAML on the two settings, and the results illustrated that the performance would become bad with hierarchical knowledge.

Algorithm	**Random scaling**	**N-way K-shot (N, K)**	
		(5,1)	**(5,5)**
ACAI HPS-ML-MAML (ours)	No	41.38%	61.22%
ACAI PPS-ML-MAML (ours)	No	**69.00**%	**87.88**%

clusters from the embedding in the later level. In our experiment, we adopted 3-level progressive clustering with 125, 250 and 500 clusters, respectively. The results implemented on PPS-ML-MAML are illustrated in Table 5. Accuracies obtained by progressive clustering strategy are lower than results achieved by HPS-ML. Therefore, this comparison demonstrates that hierarchical pseudo supervision is useful for few-shot colored image classification.

For omniglot, we also conducted experiment by using HPS-ML. We also adopted 3-level hierarchical clustering with 125, 250 and 500 clusters, respectively. The results are presented in Table 6. We adapted the same hyper-parameters when conducting experiment using these two solutions. PPS-ML-MAML outperforms HPS-ML-MAML in these two settings by a large percentage.

5.3.2. Ablation Study on Clustering Levels and iterations

Clustering levels and iterations taken in each level are hyper-parameters in C2FPS-ML. In this section, we analyse the effect caused by different numbers of clustering levels and iterations on the classification performance.

Fig. 7 presents the results obtained by different iterations and sampling tasks from different clustering levels. Fig. 7(a) illustrates the results obtained by HPS-ML-MAML with 50 times random scaling on *mini*ImageNet. Fig. 7(b) shows the results obtained by HPS-ML-ProtoNets with 50 times random scaling on *mini*ImageNet. The results of four settings are all enhanced by increasing the iteration numbers and selecting meta-tasks from deeper cluster levels. The

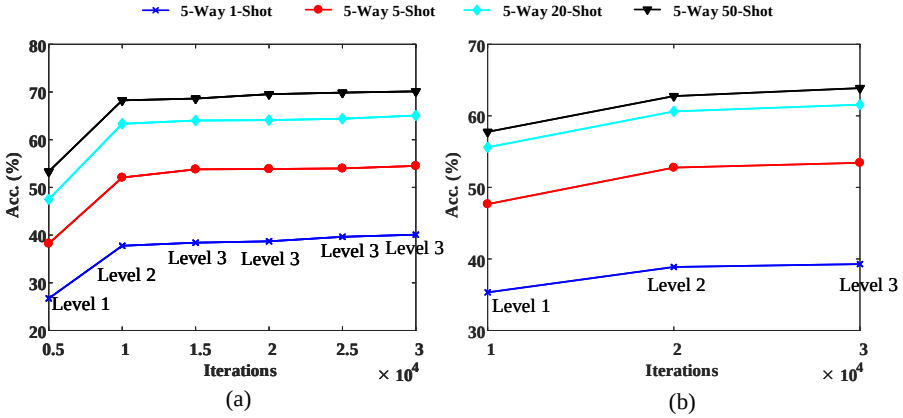


Figure 7: Ablation study results on cluster levels and iterations. Level i under the nodes means that the corresponding result is updated by sampling meta-tasks from this level. (a) This figure illustrates the ablation study results of HPS-ML-MAML with 50 times random scaling on *mini*ImageNet. (b) This line chart shows the ablation study results of HPS-ML-ProtoNets with 50 times random scaling on *mini*ImageNet.

hyper-parameter settings in our experiment achieve the best results.

6. Conclusion

FSL follows the pattern that models learn new tasks with only a few labeled samples per class. This paper focuses on a more general and challenging UFSL problem where the auxiliary source dataset is fully unlabeled. We present a novel method named C2FPS-ML for few-shot object classification. For natural object dataset, the underlying hierarchical nature object categories of the unlabeled source dataset are extracted by hierarchical clustering, and build a hierarchical tree of pseudo visual concepts. For other kinds of dataset without semantic relations between objects, the potential progressive pseudo information is obtained by progressive clustering in different similarity levels. We exploit this information as supervisions in meta-training stage and optimize meta-task sampling process. In terms of accuracy and high efficiency, our extensive experiments reveal state-of-the-art results on the *mini*ImageNet and remarkable results on Omniglot.

7. Acknowledgements

The work of Yawen Cui was partially supported by China Scholarship Council (CSC) under grant 201903170129. This work was partially supported by the Academy of Finland under grant 331883, the National Natural Science Foundation of China under Grant 61872379, 71701205 and 62022091.

References

[1] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (6) (2017) 84–90.

[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[3] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, M. Pietikäinen, Deep learning for generic object detection: A survey, International journal of computer vision (IJCV) 128 (2) (2020) 261–318.

[4] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2015, pp. 91–99.

[5] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1520–1528.

[6] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431–3440.

[7] Y. Shen, W. Lin, J. Yan, M. Xu, J. Wu, J. Wang, Person re-identification with correspondence structure learning, in: Proceedings of

the International Conference on Computer Vision (ICCV), 2015, pp. 3200–3208.

[8] M. Liang, X. Hu, Recurrent convolutional neural network for object recognition, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3367–3375.

[9] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823.

[10] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, H. Lee, Generative adversarial text to image synthesis, in: Proceeding of the International Conference on Machine Learning (ICML), 2016, pp. 1060–1069.

[11] D. Shen, G. Wu, H. Suk, Deep learning in medical image analysis, Annual Review Of Biomedical Engineering 19 (2017) 221–248.

[12] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2017, pp. 4080–4090.

[13] B. Lake, R. Salakhutdinov, J. Gross, J. Tenenbaum, One shot learning of simple visual concepts, in: Proceedings of the annual meeting of the cognitive science society, Vol. 33, 2011.

[14] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: Proceeding of the International Conference on Machine Learning (ICML), pages=1126–1135, 2017.

[15] X. Sun, B. Wang, Z. Wang, H. Li, H. Li, K. Fu, Research progress on few-shot learning for remote sensing image interpretation, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 2387–2402.

[16] Song, Yu and Chen, Changsheng, MPPCANet: A feedforward learning strategy for few-shot image classification, Pattern Recognition 113 (2021) 107792.

[17] L. Fei-Fei, R. Fergus, P. Perona, One-shot learning of object categories, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 28 (4) (2006) 594–611.

[18] T. M. Hospedales, A. Antoniou, P. Micaelli, A. J. Storkey, Meta-learning in neural networks: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).

[19] X. Yu, Y. Aloimonos, Attribute-based transfer learning for object categorization with zero/one training example, in: Proceedings of the European Conference on Computer Vision (ECCV), 2010, pp. 127–140.

[20] Z. Chen, Y. Fu, Y. Zhang, Y.-G. Jiang, X. Xue, L. Sigal, Multi-level semantic feature augmentation for one-shot learning, IEEE Transactions on Image Processing (TIP) 28 (9) (2019) 4594–4605.

[21] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, T. M. Hospedales, Learning to compare: Relation network for few-shot learning, in: Proceedings of the Computer Vision and Pattern Recognition (Proceedings of the Computer Vision and Pattern Recognition (CVPR)), 2018, pp. 1199–1208.

[22] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al., Matching networks for one shot learning, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2016, pp. 3630–3638.

[23] Y.-X. Wang, M. Hebert, Learning to learn: Model regression networks for easy small sample learning, in: Proceedings of the European Conference on Computer Vision (ECCV), 2016, pp. 616–634.

[24] K. Fu, T. Zhang, Y. Zhang, Z. Wang, X. Sun, Few-shot sar target classification via metalearning, IEEE Transactions on Geoscience and Remote Sensing.

[25] J. Nie, N. Xu, M. Zhou, G. Yan, Z. Wei, 3d model classification based on few-shot learning, Neurocomputing 398 (2020) 539–546.

[26] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, T. Lillicrap, Meta-learning with memory-augmented neural networks, in: Proceeding of the International Conference on Machine Learning (ICML), 2016, pp. 1842–1850.

[27] P. Shyam, S. Gupta, A. Dukkipati, Attentive recurrent comparators, in: Proceeding of the International Conference on Machine Learning (ICML), 2017, pp. 3173–3181.

[28] K. Hsu, S. Levine, C. Finn, Unsupervised learning via meta-learning, in: Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[29] S. Khodadadeh, L. Bölöni, M. Shah, Unsupervised meta-learning for few-shot image classification, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2019, pp. 10132–10142.

[30] L. Zhang, J. Liu, M. Luo, X. Chang, Q. Zheng, A. G. Hauptmann, Scheduled sampling for one-shot learning via matching network, Pattern Recognition 96 (2019) 106962.

[31] C. Liu, Z. Wang, D. Sahoo, Y. Fang, K. Zhang, S. C. Hoi, Adaptive task sampling for meta-learning, in: Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 752–769.

[32] J. Redmon, A. Farhadi, Yolo9000: better, faster, stronger, in: Proceedings of the Computer Vision and Pattern Recognition (Proceedings of the Computer Vision and Pattern Recognition (CVPR)), 2017, pp. 7263–7271.

[33] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller, Introduction to wordnet: An on-line lexical database, International journal of lexicography 3 (4) (1990) 235–244.

[34] Z. Yu, L. Chen, Z. Cheng, J. Luo, Transmatch: A transfer-learning scheme for semi-supervised few-shot learning, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2020, pp. 12856–12864.

[35] Y. Wang, C. Xu, C. Liu, L. Zhang, Y. Fu, Instance credibility inference for few-shot learning, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2020, pp. 12836–12845.

[36] T. Miyato, S. Maeda, M. Koyama, S. Ishii, Virtual adversarial training: a regularization method for supervised and semi-supervised learning, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 41 (8) (2018) 1979–1993.

[37] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, C. A. Raffel, Mixmatch: A holistic approach to semi-supervised learning, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2019, pp. 5049–5059.

[38] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2017, pp. 1195–1204.

[39] Huang, Shixin and Zeng, Xiangping and Wu, Si and Yu, Zhiwen and Azzam, Mohamed and Wong, Hau-San, Behavior regularized prototypical networks for semi-supervised few-shot image classification, Pattern Recognition 112 (2021) 107765.

[40] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. B. Tenenbaum, H. Larochelle, R. S. Zemel, Meta-learning for semi-supervised few-shot

classification, in: Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[41] Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S. Hwang, Y. Yang, Learning to propagate labels: Transductive propagation network for few-shot learning, in: Proceedings of the International Conference on Learning Representations (ICLR), 2019.

[42] X. Li, Q. Sun, Y. Liu, Q. Zhou, S. Zheng, T.-S. Chua, B. Schiele, Learning to self-train for semi-supervised few-shot classification, in: Proceedings of the Neural Information Processing Systems (NeurIPS), 2019, pp. 10276–10286.

[43] G. E. Hinton, T. Sejnowski, T. A. Poggio, et al., Unsupervised learning: foundations of neural computation, MIT press, 1999.

[44] Z. Ji, X. Zou, T. Huang, S. Wu, Unsupervised few-shot feature learning via self-supervised training, Frontiers In Computational Neuroscience 14.

[45] A. Antoniou, A. Storkey, Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation, arXiv preprint arXiv:1902.09884.

[46] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2009, pp. 248–255.

[47] M. Caron, P. Bojanowski, A. Joulin, M. Douze, Deep clustering for unsupervised learning of visual features, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 132–149.

[48] J. Donahue, P. Krähenbühl, T. Darrell, Adversarial feature learning, in: Proceedings of the International Conference on Learning Representations (ICLR), 2017.

[49] D. Berthelot, C. Raffel, A. Roy, I. Goodfellow, Understanding and improving interpolation in autoencoders via an adversarial regularizer, in: Proceedings of the International Conference on Learning Representations (ICLR), 2019.

Yawen Cui received B.Sc. degree from Jiangnan University, China, in 2016, the M.S. degree from the National University of Defense Technology, China, in 2018. She is currently pursuing the Ph.D. degree in Computer Science from the University of Oulu, Finland. Her research interests include few-shot learning and incremental learning.

Qing Liao received B.Sc. degree from Macau University of Science and Technology, Macau, in 2010, the M.Phil. degree from Hong Kong University of Science and Technology in 2013 and the Ph.D. degree from the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, in 2016.

Dewen Hu received the B.S. and M.S. degrees from Xi'an Jiaotong University, China, in 1983 and 1986, respectively, and the Ph.D. degree from the National University of Defense Technology in 1999. From October 1995 to October 1996, he was a Visiting Scholar with the University of Sheffield, U.K.

Wei An received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 1999. She was a Senior Visiting Scholar with the University of Southampton, Southampton, U.K., in 2016.

Li Liu received the B.E. degree, the M.S. degree and the Ph.D. degree from the National University of Defense Technology, China, in 2003, 2005 and 2012, respectively. From 2016.12 to 2018.11, she worked as a senior researcher at the Machine Vision Group at the University of Oulu, Finland.