Scaling Few-Shot Learning for the Open World

Zhipeng Lin[1], Wenjing Yang[1], Haotian Wang[1], Haoang Chi[2, 1], Long Lan[1], Ji Wang[1]*

[1] State Key Laboratory of High Performance Computing, National University of DefenseTechnology
[2] Intelligent Game and Decision Lab, Academy of Military Science
{linzhipeng13, wenjing.yang, wanghaotian13, long.lan, wj}@nudt.edu.cn, haoangchi618@gmail.com

Abstract

Few-shot learning (FSL) aims to enable learning models with the ability to automatically adapt to novel (unseen) domains in open-world scenarios. Nonetheless, there exists a significant disparity between the vast number of new concepts encountered in the open world and the restricted available scale of existing FSL works, which primarily focus on a limited number of novel classes. Such a gap restricts the practical applicability of FSL in realistic scenarios. To narrow this gap, we propose a new problem named **F**ew-**S**hot **L**earning with **M**any **N**ovel **C**lasses (FSL-MNC) by substantially enlarging the number of novel classes, exceeding the count in the traditional FSL setup by over 500-fold. This new problem exhibits two major challenges, including the increased computation overhead during meta-training and the degraded classification performance by the large number of classes during meta-testing. To overcome these challenges, we propose a Simple Hierarchy Pipeline (SHA-Pipeline). Due to the inefficiency of traditional protocols of EML, we re-design a lightweight training strategy to reduce the overhead brought by much more novel classes. To capture discriminative semantics across numerous novel classes, we effectively reconstruct and leverage the class hierarchy information during meta-testing. Experiments show that the proposed SHA-Pipeline significantly outperforms not only the ProtoNet baseline but also the state-of-the-art alternatives across different numbers of novel classes.

Introduction

The remarkable progress achieved by Few-Shot Learning (FSL) (Wang et al. 2021) has equipped learning models with capabilities for rapid exploration of the open world, e.g., adapting to new visual concepts. However, traditional scenarios for FSL commonly explore with restrictions on the "scale" of the open world, as the number of novel classes is limited from 5 to 160 (Dhillon et al. 2020). On the contrary, the realistic open world often exhibits a large amount of unseen knowledge (classes), which is far beyond the protocols adopted by previous FSL scenarios. Furthermore, taking the number of novel classes into consideration is important for promoting the practicality of FSL, as it is nearly impossible to master prior knowledge of the number of novel classes in practice.

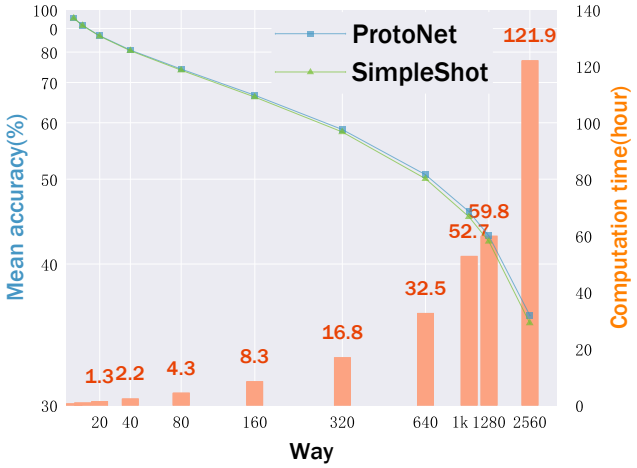


Figure 1: Mean accuracy and computation overhead of ProtoNet and SimpleShot with different scales of novel class (5shot, Vit-Small DINO pre-trained). For traditional fsl methods, as the number of ways increases, the average accuracy experiences a rapid decline. The computation cost of ProtoNet (red bar) in the meta-training stage increases significantly.

Pointed by recent studies of Willes et al. (2022) and Parmar et al. (2023), a realistic open world learner for the FSL should handle over thousands of novel classes, e.g., iNaturalist (Parmar et al. 2023). In response to such viewpoints, we narrow the gap between FSL and a realistic open world by developing a new problem named **F**ew-**S**hot **L**earning with **M**any **N**ovel **C**lasses (FSL-MNC). The core advance of our FSL-MNC compared to traditional FSL is that the open world exhibits a large scale of unseen knowledge with more than 2000 novel classes[1], surpassing the traditional FSL setting by over **10-fold**. The ratio of novel class set

*Lin and Yang are co-first authors of the article. Corresponding author (J. Wang).

[1]Note that few-shot tasks are organized as N-way K-shot episodes. "way" and "shot" are the number of novel classes and annotated samples for each novel class respectively. For the sake of clarity, we will use "ways" to denote the number of novel classes throughout this paper.

size to base class set size in traditional is smaller than 1, but in FSL-MNC the ratio is larger than 10. To make this advance more intuitive, we study the few-shot generalization performance of learning models pre-trained from ImageNet-1k (Russakovsky et al. 2015) on no-overlapped classes of ImageNet-21K (Deng et al. 2009) as a motivating example.

In the motivating example, we first examine the performance of representative algorithms for traditional FSL problems, where the performance and efficiency for different way numbers of ProtoNet and SimpleShot are shown in Fig. 1. We first observe that the efficiency, i.e., the computation overhead of meta-training, increases dramatically from half an hour to more than one hundred hours, as the number of ways increases. Meanwhile, regarding the performance, we observe that the accuracies of traditional FSL methods degrade quickly when the number of ways grows from tens, i.e., traditional FSL's scale, to thousands, i.e., our FSL-MNC's scale. To sum up, typical FLS paradigms suffer from inefficient training with sub-optimal performances under the FSL-MNC problem, leaving a significant gap to study the effectiveness of few-shot learning at a realistic scale.

Concurrently, we provide a comprehensive examination of the inefficiency and sub-optimal performance inherent in traditional Few-Shot Learning (FSL) methods when addressing the FSL-MNC problem. The inefficiency arises from the episodic meta-learning (EML) (Baz et al. 2021) framework, which exhibits linear computational overhead with an increasing number of ways (Rajeswaran et al. 2019). This leads to notable inefficiencies, particularly when dealing with a large number of ways. Meanwhile, in the context of FSL-MNC, the sub-optimal performance issue arises when traditional methods encounter an expanding number of ways. The sub-optimal performance can be caused by few-shot tasks that encompass numerous fine-grained classes, presenting challenges for classification. To be specific, a consensus for describing relationships among a large number of fine-grained classes is the nested semantic hierarchy organization of object categories, e.g., in ImageNet-21K and recent efforts on the computer vision (Silla and Freitas 2011; Novack et al. 2023; Guo et al. 2022) have pointed out the fine-grained classification challenge. However, traditional FSL methods neglect such hierarchy structures due to limited scale of novel classes.

To migrate these challenges in FSL-MNC, we propose a novel Simple Hierarchy Aware Pipeline (SHA-Pipeline). To be specific, our SHA-Pipeline primarily contributes two strategies as follows:

Efficiency enhancement. We observe an interesting phenomenon that the performance of EML on FSL-MNC is almost irrelevant to the way number, through extensive experiments using different backbones. Hence, by remaining the way number of episodes as 5, we eliminate the dependency of the complexity of EML on the way number. Furthermore, to accelerate the training of EML, we avoid communication of support samples and overflow of GPU memory by distributing the whole support set and part of the query set.

Performance improvement. We propose to capture the class hierarchy structure with a fast non-parametric hierarchy clustering strategy and leverage such class hierarchy by structured representation learning. We seek to improve representation learning from two levels, including prototype-level and sample-level learning. The former aims to preserve the hierarchy structure on class prototypes by maximizing Cophenetic Correlation Coefficient (CPCC). The latter seeks to enforce similar samples from the different parent classes being far away from each other via the hierarchical triplet loss.

Contributions. We develop a new setting called "Few-Shot Learning with Many Novel Classes (FSL-MNC)" to narrow the gap between few-shot learning and open-world settings, which is more practical and challenging but has not been extensively studied in the community. Our SHA-Pipeline effectively overcomes the computational and performance challenges by reducing the overhead of meta-training to constant and preserving the class hierarchy in meta-testing. Experiments show that the SHA-Pipeline outperforms the SOTA alternatives across different numbers of novel classes.

Related Work

Traditional Few-Shot Learning

Few-shot learning is a prominent and burgeoning research field. Hospedales et al. (2022) and Wang et al. (2021) give comprehensive overviews of the subject respectively. Different kinds of methods (Vinyals et al. 2016; Ravi and Larochelle 2017; Ye et al. 2020; Zhang et al. 2021; Ye and Chao 2022) are proposed to enhance the ability of quickly adapting to unseen few-shot tasks, which show good performance on FSL benchmark.

However, most existing FSL methods are typically designed for few-shot tasks with a small number of novel classes, which limits the application of few-shot learning in a realistic open world (Geng, Huang, and Chen 2021; Parmar et al. 2023). For example, gradient-based methods, like MAML, are not scalable for the extrapolation of different way numbers. The state-of-the-art few-shot metric-based methods, e.g. FEAT (Ye et al. 2020) have an unbearable computation overhead during the meta-training stage on large few-shot tasks. The simple baselines such as SimpleShot (Wang et al. 2019) do not consider how to exploit structure information underlying large tasks efficiently.

Large-Scale Few-Shot Learning

A few studies (Hu et al. 2022; Liu et al. 2022; Dhillon et al. 2020; Li et al. 2019) have investigated FSL on larger scale datasets. However, their studies do not push the limit of novel classes number in the meta-test stage, which is a much more challenging and practical setting. How to achieve a simple and effective baseline for FSL-MNC is still an open and realistic problem. More specifically, Liu et al. (2022) and Li et al. (2019) use an external class hierarchy structure, instead of capturing the structure from few-shot data.

Hu et al. (2022) still relies on the traditional episodic training of base dataset, which limits the scalability of way number. Dhillon et al. (2020) does not explore the impact of a large number of ways in the meta-training stage and gives a method to handle this. Unlike existing works on large-scale FSL, we specifically aim to explore the performance of simple baselines on fsl when the way number goes large.

Few-Shot Learning With Many Novel Classes

Problem Formulation

Few-shot learning. Let C_{base} and C_{novel} denote the set of base and novel classes respectively, which are disjoint, i.e., $C_{\text{base}} \cap C_{\text{novel}} = \emptyset$. Given the dataset D_{base} and D_{novel} containing labeled samples from C_{base} and C_{novel} respectively, the goal of FSL is to train a model f_ϕ on D_{base} that performs well on few-shot tasks sampled from D_{novel}. Each task $\mathcal{T}_i$ consists of a support set $\mathcal{S}^i = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^{N\times k}$ and a query set $\mathcal{Q}^i = \{(\tilde{\mathbf{x}}_j, \tilde{\mathbf{y}}_j)\}_{j=1}^{N\times q}$, where $\mathcal{S}^i$ contains N classes with k labeled examples per class, and $\mathcal{Q}^i$ contains q unlabeled query examples per class. The number N and k are called "way" and "shot" respectively.

Few-shot learning with many novel classes. A larger value of N signifies the extent of "novelness" associated with a few-shot episode. This indicates that as the value of N increases, the few-shot tasks require a more extensive level of adaptation to accommodate the unfamiliar knowledge present within the open-world scenario.

In the context of FSL-MNC, the value of N significantly surpasses that in standard few-shot learning scenarios (e.g., $N > 1000$), leading to notable computational and generalization challenges. Furthermore, we introduce a metric aimed at quantifying the degree of "novelness" inherent within a given few-shot dataset. This metric serves to effectively differentiate FSL-MNC from traditional FSL, underscoring their fundamental distinctions. The novelness ratio, denoted as $\Omega\left(D_{\text{novel}}; D_{\text{base}}\right)$, is defined as the proportion of the size of the novel class set to that of the base class set:

$$\Omega\left(D_{\text{novel}}; D_{\text{base}}\right) = \frac{|C_{\text{novel}}|}{|C_{\text{base}}|} \tag{1}$$

Fig. 2 presents a comparison of the novel class set sizes and the novelness ratio exhibited by both the traditional few-shot dataset and our ImageNet-MNC dataset (15000 novel classes and 1000 classes) designed for the FSL-MNC benchmark. In traditional FSL, novelness ratio $\Omega\left(D_{\text{novel}}; D_{\text{base}}\right)$ is typically smaller than 1, signifying that the quantity of novel classes is relatively limited compared to the number of base classes. In contrast, within the FSL-MNC, $\Omega\left(D_{\text{novel}}; D_{\text{base}}\right) > 10$, indicating a significant increase in the number of novel classes in comparison to the base classes.

Mathematically, the FSL-MNC problem can be precisely described as follows. Given the datasets D_{base} and D_{novel} where the condition $\Omega\left(D_{\text{novel}}; D_{\text{base}}\right) > 10$ holds, the primary objective of FSL-MNC is to train a model f_ϕ using D_{base}, which exhibits effective performance in the context of few-shot tasks extracted from D_{novel}, considering a way number $N > |C_{\text{base}}|$.

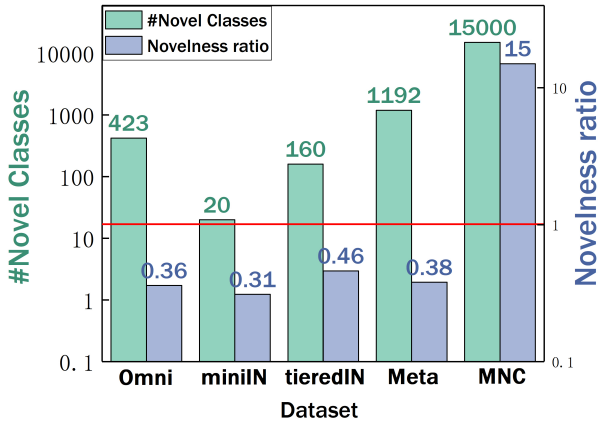


Figure 2: $|C_{\text{novel}}|$ and novelness ratio of Omniglot, miniImageNet, tieredImageNet, Meta-Dataset and ImageNet-MNC.

Class hierarchical structure. In addressing FSL-MNC, our approach involves the utilization of class hierarchy to enhance performance. In the following, we present the formal definition of the class hierarchy. We employ a class hierarchy represented by a Directed Acyclic Graph, denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{C} \cup \mathcal{R}$. Here, $\mathcal{C}$ represents specific classes, and $\mathcal{R}$ represents abstract parent classes. This graph forms a single tree, with leaf nodes corresponding to specific classes from $\mathcal{C}$ and non-leaf nodes from $\mathcal{R}$ representing abstract parent classes encompassing subsets of specific classes. Relationships between nodes are defined by $\mathcal{E} \subseteq \left\{(x, y) \mid (x, y) \in \mathcal{V}^2\right\}$, reflecting parent-child connections.

The tree's height, denoted as H, indicates the length of the path from the root node to its leaf nodes. For any specific class $c_i \in \mathcal{C}$, a set of parent classes $\mathcal{A}^{c_i} = \{\mathcal{P}_1^{c_i}, \mathcal{P}_2^{c_i}, \cdots, \mathcal{P}_H^{c_i}\}$ is derived by finding the shortest path from c_i to the root node. Here, $\mathcal{P}_h^{c_i}$ represents the ancestor node of c_i at the h-th level of $\mathcal{G}$.

Simple Hierarchy Aware Pipeline

To address the computational challenge posed by FSL-MNC, we introduce an efficient meta-training strategy supported by comprehensive experimentation and design a lightweight distributed framework. In pursuit of achieving exceptional performance in the FSL-MNC scenario, we propose an innovative fine-tuning algorithm that harnesses the potential of class hierarchy for improved utilization and integration.

Efficiency Enhancement

In the traditional FSL, a learning function $f(\cdot)$ is trained using a sequence of N-way K-shot few-shot tasks sampled from the base dataset, where $f(\cdot)$ is optimized to minimize the average error across these tasks. The optimization can be expressed as:

$$f^* = \arg\min_f \sum_{(\mathcal{S}^i, \mathcal{Q}^i) \in \mathcal{T}^i} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{Q}^i} \ell\left(f\left(\mathbf{x}; \mathcal{S}^i\right), \mathbf{y}\right), \tag{2}$$

Algorithm 1: A Lightweight Parallel Framework

Require: $p(\mathcal{T})$: distribution over tasks
Require: α: step size hyperparameters, n: GPU numbers
load pre-trained weights θ
while not done **do**
  Sample a few-shot task $\mathcal{T}_i = \left(\mathcal{S}^i, \mathcal{Q}^i\right) \sim p(\mathcal{T})$
  Split $\mathcal{Q}^i$ into $\{\mathcal{Q}_0^i, \mathcal{Q}_1^i, \ldots, \mathcal{Q}_n^i\}$
  Send $\left(\mathcal{S}^i, \mathcal{Q}_j^i\right)$ to GPU j.
  for all GPU i **do**
    Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ with respect to $\left(\mathcal{S}^i, \mathcal{Q}_j^i\right)$
  end for
  Allreduce $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ among GPUs
  Update adapted parameters with gradient descent: $\theta = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
end while

Algorithm 2: Algorithm of fine-tuning

Require: N-way M-shot episodes ($\mathcal{D}^{\mathcal{S}}_{\mathbf{train}}$, $\mathcal{D}^{\mathcal{S}}_{\mathbf{test}}$), learning rate η, fine-tune step I, backbone weights ϕ,
Load model from ϕ
for all iteration = 1,...,I **do**
  $\mathcal{D}^{\mathcal{S}}_{\mathbf{aug}}$ = data_augment($\mathcal{D}^{\mathcal{S}}_{\mathbf{train}}$)
  Feature forward and normalization via Equ. (3)
  Compute prototypes and cosine distance
  Hierarchy clustering
  Compute tree distance via Equ. (5)
  Compute $\mathcal{L}_{train}$ via Equ. (7)
  Update ϕ with $\mathcal{L}_{train}$ use SGD
end for
Compute logits $\sigma_{\mathbf{test}}$ with updated ϕ
return $\sigma_{\mathbf{test}}$.

where $\ell(\cdot)$ represents the loss function, $\mathcal{S}^i$ and $\mathcal{Q}^i$ are the support and query set from $\mathcal{T}^i$ respectively. In traditional FSL, the N of meta-training and meta-testing is usually in the same order of magnitude.

A strategy for meta-Learning. In the context of FSL-MNC, we observe an interesting phenomenon that the performance of meta-training beats SimpleShot and the improvement is almost irrelevant to the way number, through extensive experiments using different backbones. The extensive results are shown in Fig. 4 and 5. Table (2) in the appendix shows the specific data. Given the fact that a simple meta-training on $5-$way episodes on FSL-MNC is sufficient to achieve effective performance improvements on FSL-MNC, we use keep $N = 5$ during the meta-training stage of our SHA-Pipeline.

In the following, we give a discussion about the comparison between pre-training and meta-training. The prevailing perspective (Baz et al. 2021) in the traditional FSL suggests that meta-learning might not be as effective as a well-trained embedding model. Simple and no-episodic-trained feature transformation methods on a pre-trained backbone can offer comparable performance to sophisticated methods with meta-training. Simple Baseline approaches are commonly built upon a transfer learning pipeline (Yosinski et al. 2014). The model adapts to the few-shot target data through various feature-transformation methods (Chen et al. 2019; Tian et al. 2020; Fei et al. 2021; Hou and Sato 2022; Wang et al. 2019) using backbone pre-trained from the base class dataset.

Nevertheless, FSL-MNC stands apart from traditional FSL in several ways. Notably, during the meta-training phase, FSL-MNC demonstrates a broader array of tasks, enabling meta-learning to bolster its generalization prowess. This observation aligns with the perspective presented by researchers such as Miranda et al. (2023), suggesting that the efficacy of meta-learning is directly tied to the diversity present within the dataset and meta-learning clearly outperformed Simple Baseline without meta-learning in high diversity settings.

Lightweight parallel framework. For further acceleration of EML training, we propose a parallel framework for the large few-shot tasks by distributing the whole support set and part of the query set to different GPUs to avoid communication of support sets and overflow of GPU memory. The full framework is outlined in Algorithm 1.

Fine-Tuning With Class Hierarchy Capturing

In the following, we focus on the meta-testing stage of SHA-Pipeline. Our three-step fine-tuning is shown in Fig. 3. Given a support set $\mathcal{S}_i$, an augmented support set will be generated for fine-tuning of each few-shot task. We first calculate class prototypes with Z-hubness normalization on features. Next, we conduct parameter-free hierarchical clustering on prototypes. Based on the hierarchical clustering result $\mathcal{H}$, tree-metric distance $\mathcal{T} = \{t(c_i, c_j)\}_{c_i, c_j \in \mathcal{C}}$ of prototype pairs will be calculated. Finally, using $\mathcal{T}$ as supervision, we can fine-tune the backbone guided by cross-entropy loss with CPCC regularization or hierarchy triplet loss. Algorithm 2 presents how to fine-tune based on CPCC normalization. A similar algorithm based on the hierarchy triplet loss can be easily derived.

Z-hubness normalization. To mitigate the hubness problem before hierarchical Clustering, we apply Z-hubness normalization to the feature vectors. The Z-score normalized feature is given by:

$$\mathbf{x}^{(zn)} = \frac{\mathbf{x} - \mu \mathbf{1}}{\sigma} \in \mathbb{R}^D, \tag{3}$$

where μ and σ are the mean and standard deviation of components of the feature vector $\mathbf{x}$ respectively.

Hierarchical clustering with the first neighbor. We utilize a fast hierarchical clustering algorithm on the adjacency link matrix of class prototypes. The adjacency link matrix can be represented as:

$$A(i,j) = \begin{cases} 1 & \text{if } j = \kappa_i^1 \text{ or } \kappa_j^1 = i \text{ or } \kappa_i^1 = \kappa_j^1 \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

where $\mathbf{A}$ is the adjacency matrix, i and j represent sample indices, κ_i^1 symbolizes the first neighbor of point i.

The connected component of the adjacency link matrix is the cluster partition. For hierarchical clustering, we average

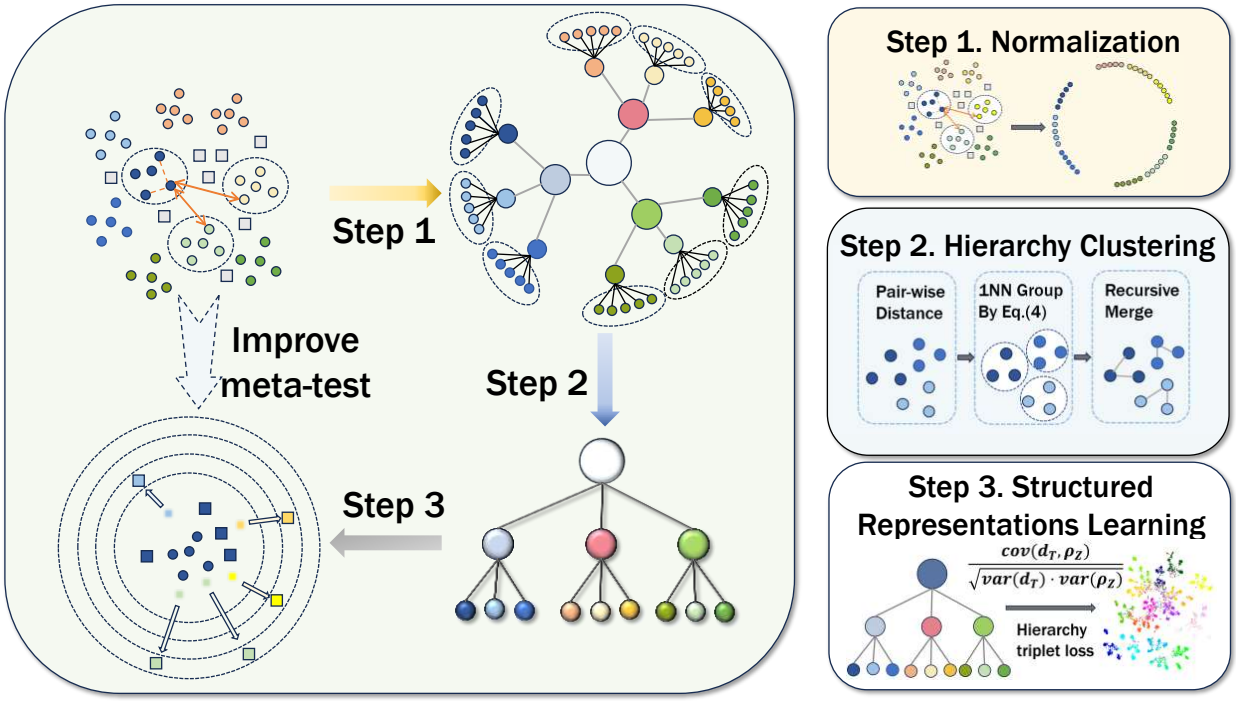


Figure 3: The left sub-part shows the fine-tuning of our SHA-Pipeline. SHA-Pipeline first applies Z-score normalization on features. Then SHA-Pipeline clusters class prototypes with respect to the first neighbor relation. Lastly, based on the results of hierarchy clustering, SHA-Pipeline fine-tunes the backbone. The right three sub-parts illustrate each step.

the feature vectors of each cluster and construct the next-level cluster partition based on these mean vectors.

CPCC regularization. Firstly, we define the distance between prototype i and prototype j on the class hierarchy $d_T(i,j)$ as follow:

$$d_T(i,j) = H - \sum_{k=1}^{H} \left(\mathbb{I}\left(r_i^{(k)} = r_j^{(k)} \right) \right), \tag{5}$$

where r_i is the hierarchy cluster result of the prototype i, the k-th element $r_i^{(k)}$ indicates the cluster id that the prototype i belongs to at level k, H is the overall height of class hierarchy that is larger than 1. $\mathbb{I}$ is the indicator function. The range of $d_T(i,j)$ is the interval $[1, H]$.

Secondly, we incorporate Cophenetic Correlation Coefficient (CPCC) regularization (Sokal and Rohlf 1962) to measure the correspondence between the class hierarchy metric $d_T(\cdot,\cdot)$ and cosine distance $\rho_{\mathcal{Z}}(\cdot,\cdot)$. The CPCC score is computed as:

$$\mathrm{CPCC}\left(d_T, \rho_{\mathcal{Z}}\right) = \frac{\mathrm{cov}\left(d_T, \rho_Z\right)}{\sqrt{\mathrm{var}\left(d_T\right) \cdot \mathrm{var}(\rho_Z)}} \tag{6}$$

where d_T and ρ_Z is the set of pair-wise cosine distance and class hierarchy distance on all class prototypes, $\mathrm{cov}(\cdot)$ denotes covariance, and $\mathrm{var}(\cdot)$ denotes variance.

The overall loss function combines the cross-entropy loss ($\mathcal{L}_{\mathrm{CE}}$) and the CPCC regularization term (CPCC), leading to the following formulation of the total training loss:

$$\mathcal{L}_{train} = \sum_{(x,y)\in\tilde{\mathcal{S}}} \ell_{\mathrm{CE}}\left(y, g\left(f_\theta(x)\right)\right) - \lambda \cdot \mathrm{CPCC}\left(d_T, \rho Z\right), \tag{7}$$

where λ is the weighting factor for the CPCC regularization term, $\tilde{\mathcal{S}}$ is the data-augmented support set of the few-shot task.

Hierarchy triplet loss. In the following, we also give another way to learn structured representation with class hierarchy based on triplet loss with adaptive margin.

We re-scale the $d_T(i,j)$ to the interval $[0,1)$ to obtain the adaptive margin $M(i,j)$ between the anchor prototype i and negative prototype j:

$$M(i,j) = \frac{d_T(i,j)\left(1 - d_{mean}\right)}{H} + d_{mean} - d_i + d_j, \tag{8}$$

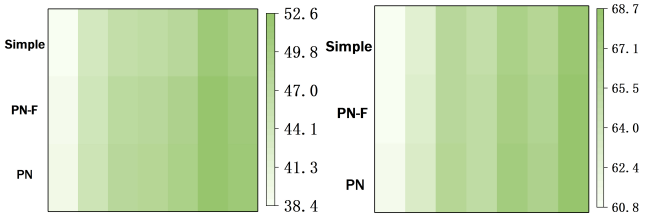
where d_i and d_j are the average cosine distance of the samples from class i and class j respectively. d_{mean} is given by $d_{mean} = \frac{1}{C}\sum_{i=1}^{N} d_i$.

After obtaining the adaptive margin $M(i,j)$, for the sake of computation overhead, we randomly sample triplets $\mathcal{T}_z = (\boldsymbol{x}_a, \boldsymbol{x}_p, \boldsymbol{x}_n)$ from the data-augmented support set $\tilde{\mathcal{S}}$ and get the corresponding adaptive margin $M(i,j)$ for sample x_a and negative sample x_n which comes from i-th class and j-th class respectively. The hierarchy triplet loss is given by:

$$\mathcal{L}_{htl} = \frac{1}{|\mathcal{T}^{\mathcal{M}}|} \sum_{\mathcal{T}^z \in \mathcal{T}^{\mathcal{M}}} \left[\|\boldsymbol{x}_a - \boldsymbol{x}_p\|^2 - \|\boldsymbol{x}_a - \boldsymbol{x}_n\|^2 + M(i,j) \right]_+ \tag{9}$$

Method	Shot	Way											Average
		5	10	20	40	80	160	320	640	1000	1280	2560	
ProtoNet	1	86.21	78.74	70.74	61.38	52.43	43.73	35.86	28.78	24.78	22.68	17.56	47.54
	5	95.30	91.55	86.75	80.83	74.08	66.45	58.60	50.68	45.65	42.85	35.69	66.22
ProtoNet-Fix	1	86.21	78.44	70.19	61.27	52.17	43.42	35.50	28.3555	24.41	22.35	17.26	47.23
	5	95.30	91.55	86.64	80.68	73.90	66.25	58.37	50.41	45.34	42.57	35.43	66.04
SimpleShot	1	85.34	77.07	68.76	59.87	50.95	42.36	34.65	27.68	23.80	21.84	16.91	46.29
	5	95.20	91.39	86.52	80.50	73.71	66.02	58.09	50.09	45.02	42.25	35.11	65.81
few-shot-baseline	1	86.21	78.59	70.47	61.32	52.30	43.57	35.77	28.68	24.68	22.60	17.49	47.43
	5	95.30	91.55	86.70	80.75	73.99	66.35	58.54	50.61	45.57	42.78	35.62	66.16
P>M>F	1	86.82	79.91	72.13	62.81	53.83	45.02	37.02	29.83	25.71	23.48	18.18	48.61
	5	95.78	92.49	87.86	81.97	75.20	67.49	59.53	51.52	46.40	43.49	36.18	67.08
SHA-Pipeline(CPCC)	1	88.24	81.59	73.60	64.25	55.31	46.38	38.23	30.98	26.83	24.40	19.02	**49.89**
	5	97.06	94.00	89.18	83.27	76.54	68.71	60.62	52.55	47.41	44.32	36.93	68.24
SHA-Pipeline(Triplet Loss)	1	89.74	82.01	73.49	63.35	53.82	44.75	36.80	29.88	26.08	23.85	18.91	49.33
	5	97.63	94.30	89.29	83.26	76.68	69.03	61.21	52.88	47.90	44.89	37.25	**68.57**

Table 1: Performance on ImageNet-MNC (under different numbers of ways) in comparison with other FSL algorithms.



(a) 1shot (b) 5shot

Figure 4: Mean accuracy of different ways (from 5 to 2560 ways) of ProtoNet (PN), ProroNet-Fix (PN-F) and SimpleShot (Simple) on ResNet50(sup & DINO), ViT-small(DINO,Deit) and Vit-base(DINO,Deit & MAE).

Experiments

Experimental Setup

Datasets. We curated a new dataset ImageNet-MNC. We use the ImageNet-1k((ILSVRC 2012-2017) as the base class set and construct a novel class dataset *ImageNet-21K-MNC* based on ImageNet-21K (winter'21 release), which consists of 19167 classes. We eliminated 1000 classes overlapping with ImageNet-1k and additional 1455 classes with less than 20 samples per class. Among the remaining 16712 classes, 15000 classes are randomly selected for the meta-test and 1712 classes are used for the validation of meta-training.

Experimental protocols. To avoid over-engineering, we use public backbones pre-trained on ImageNet-1k. For the backbone, we selected ViT (Dosovitskiy et al. 2021) and ResNet50 (He et al. 2016), along with diverse pre-training strategies, including Deit (Touvron et al. 2021), DINO (Caron et al. 2021), and MAE (He et al. 2022). The numbers of ways exponentially increase from 5 to 2560. For $N < 160$, the number of episodes for meta-testing is 80 following a non-transductive setting of Dhillon et al. (2020). For $N \geq 160$, the number of episodes is 600, following the standard setting of Vinyals et al. (2016). The number of query samples is 15 per class. All experiments are conducted on 8 NVIDIA A100 nodes with 8 GPUs each.

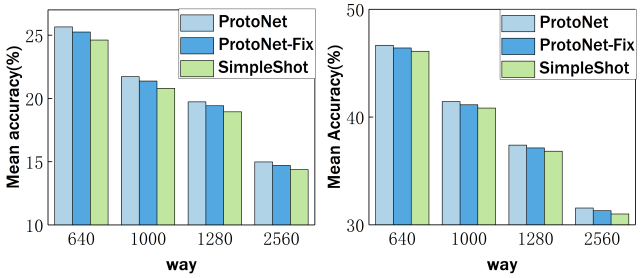
Baselines. We compare with the following approaches. **ProtoNet** (Snell, Swersky, and Zemel 2017) is known as a strong task-agnostic embedding baseline model (Chen et al. 2019). **ProtoNet-Fix** is a simple ProtoNet with $N = 5$ during the meta-training. **SimpleShot** (Wang et al. 2019) is a baseline with centered l2 normalization for few-shot learning without meta-learning and fine-tuning. **few-shot-baseline** (Dhillon et al. 2020) is a baseline fine-tuning the backbone with the cross-entropy loss on the support set. **P>M>F** (Hu et al. 2022) is the SOTA method that adopts ProtoNet for meta-training while fine-tuning the meta-trained backbone with data augmentation.

Technical Details. For meta-training, we use SGD optimizer without weight decay and a momentum of 0.9. The linear lr scaling rule of Goyal et al. (2017) are adopted: $lr = base\_lr \times$ way / 5. For the learning rate schedule, we employ a cosine annealing learning rate schedule with a warm-up epoch of 5, from a base learning rate of 10^{-6} to 5×10^{-5}. For ProtoNet, P>M>F and ProtoNet-Fix, the backbone network was meta-trained for 100 epochs, with each epoch consisting of 600 episodes. Note that the number of ways for meta-training is the same as that of meta-testing for ProtoNet and P>M>F. For the sake of simplicity, strong regularization, including logits scaling (Oreshkin, López, and Lacoste 2018), mixup (Zhang et al. 2018) and label smoothing (Szegedy et al. 2016), were omitted. Early stopping was determined based on the performance on the validation set. For meta-testing of P>M>F, few-shot-baseline and SHA-Pipeline, we fine-tune 50 steps with a fixed learning rate of 10^{-6}. SHA-Pipeline employs the same data augmentation as P>M>F in the original setting Hu et al. (2022).

Efficiency Analysis

Meta-learning comparison. We first compare the performance of different meta-training strategies on FSL-MNC. Fig. 4 shows the mean accuracy of different ways (increasing exponentially from 5 to 2560) on ResNet50, ViT-small and Vit-base.

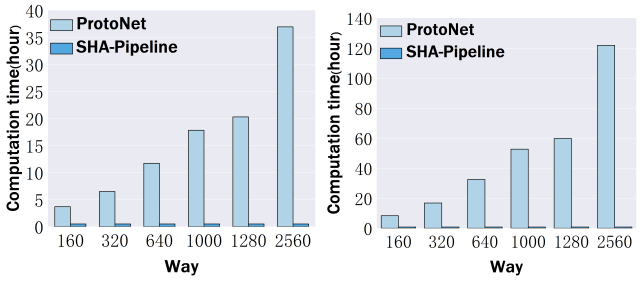
Fig. 4 demonstrates how meta-training strategies affect downstream few-shot learning performance. The employ of ProtoNet shows a positive impact on the downstream few-



(a) ResNet50 Sup.(1shot) (b) ResNet50 Sup.(5shot)

(c) ViT-S DINO. (1shot) (d) ViT-S DINO. (5shot)

Figure 5: Accuracy of ProtoNet, ProtoNet-Fix and SimpleShot on different ways.

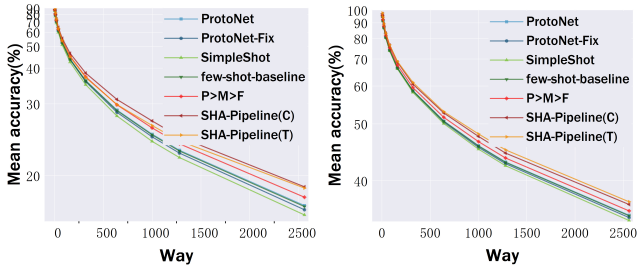


(a) 1shot (b) 5shot

Figure 6: Computation overhead.(ViT-S, DINO)

shot learning performance in the 1-shot setting. But the benefits of ProtoNet diminish as the number of shots increasing from 1 to 5. More importantly, ProtoNet-Fix with a small number of classes can also lead to significant performance improvements in the 1-shot scenario. Fig. 5 also shows the accuracy in different ways, which highlights that the ProtoNet-Fix achieves a trade-off between accuracy and computation overhead.

Lightweight parallel framework. From Fig. 6, it is evident that our lightweight parallel framework effectively reduces computation time, achieving a parallel speedup ratio of over 150 when $N = 2560$. Additionally, when $N < 160$, the traditional parallel framework has already exceeded the GPU memory capacity, whereas our framework continues to smoothly conduct meta-training until N = 2560. Combined with the meta-training approach of ProtoNet, our meta-training process is remarkably lightweight and fast.



(a) 1shot (b) 5shot

Figure 7: Mean accuracy for different ways (1 and 5 shot). SHA-Pipeline(C) is SHA-Pipeline implemented by CPCC regularization and SHA-Pipeline(T) is SHA-Pipeline implemented by hierarchy triplet loss.

FSL-MNC Performance

In this experiment, SHA-Pipeline uses the episodically trained ProtoNet with a fixed number of ways (5) for meta-training. The backbone of all methods is the Vit-small (DINO pre-trained). Table 1 shows the resultant mean accuracy achieved by different methods.

From the table, we can see that the SHA-Pipeline method achieves the highest average accuracy across different numbers of classes. In the 1-shot setting, the average accuracy of SHA-Pipeline (CPCC) is 49.89%, outperforming the best competitor P>M>F by 1.28%. In the 5-shot setting, the average accuracy of SHA-Pipeline (CPCC) is 68.24%, which is 1.16% higher than P>M>F.

Fig. 7 further depicts the mean accuracy trends of all methods in both 1-shot and 5-shot settings with varying numbers of ways. Notably, the SHA-Pipeline consistently exhibits improved classification accuracy compared to other methods across all class number settings, including scenarios with a smaller number of classes.

Conclusions

We introduced a new problem named **F**ew-**S**hot **L**earning with **M**any **N**ovel **C**lasses (FSL-MNC) that drives the exploration of few-shot learning in real open-world scenarios, which poses computational and generalization challenges. In FSL-MNC, we showed that meta-training with fix way number can achieve a trade-off between computation overhead and performance. When the number of ways is large, we find that effectively extracting and utilizing the class hierarchy structure can significantly improve performance. We also design a lightweight distributed framework for FSL-MNC to compare baselines. We verified that our proposed SHA-Pipeline achieves very competitive performance in FSL-MNC.

Limitations and future work. Our study has primarily focused on image data and has not leveraged text data to address FSL-MNC. Future work could investigate methods that incorporate both modalities to improve performance and apply in real-life scenario like visual tracking (Tan et al. 2021; Lan et al. 2020). Our current SHA-Pipeline is rela-

tively simplistic, as we have not fully integrated it with meta-training. Exploring more sophisticated meta-training techniques may yield further performance gains. In addition, the SHA-Pipeline approach is rather computationally expensive at meta-test time. Future research should focus on optimizing the computational efficiency of fine-tuning.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Key Program Grant No. 62032024 and General Program Grant No. 62376282, 62372459)

References

Baz, A. E.; Ullah, I.; Alcobaça, E.; de Carvalho, A. C. P. L. F.; Chen, H.; Ferreira, F.; Gouk, H.; Guan, C.; Guyon, I.; Hospedales, T. M.; Hu, S.; Huisman, M.; Hutter, F.; Liu, Z.; Mohr, F.; Öztürk, E.; van Rijn, J. N.; Sun, H.; Wang, X.; and Zhu, W. 2021. Lessons learned from the NeurIPS 2021 MetaDL challenge: Backbone fine-tuning without episodic meta-learning dominates for few-shot learning image classification. In Kiela, D.; Ciccone, M.; and Caputo, B., eds., *NeurIPS 2021 Competitions and Demonstrations Track, 6-14 December 2021, Online*, volume 176 of *Proceedings of Machine Learning Research*, 80–96. PMLR.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging Properties in Self-Supervised Vision Transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 9630–9640. IEEE.

Chen, W.; Liu, Y.; Kira, Z.; Wang, Y. F.; and Huang, J. 2019. 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.

Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, 248–255. IEEE Computer Society.

Dhillon, G. S.; Chaudhari, P.; Ravichandran, A.; and Soatto, S. 2020. A Baseline for Few-Shot Image Classification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Fei, N.; Gao, Y.; Lu, Z.; and Xiang, T. 2021. Z-Score Normalization, Hubness, and Few-Shot Learning. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 142–151. IEEE.

Geng, C.; Huang, S.; and Chen, S. 2021. Recent Advances in Open Set Recognition: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(10): 3614–3631.

Goyal, P.; Dollár, P.; Girshick, R. B.; Noordhuis, P.; Wesolowski, L.; Kyrola, A.; Tulloch, A.; Jia, Y.; and He, K. 2017. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *CoRR*, abs/1706.02677.

Guo, Y.; Xu, M.; Li, J.; Ni, B.; Zhu, X.; Sun, Z.; and Xu, Y. 2022. Hcsc: Hierarchical contrastive selective coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9706–9715.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. B. 2022. Masked Autoencoders Are Scalable Vision Learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, 15979–15988. IEEE.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 770–778. IEEE Computer Society.

Hospedales, T. M.; Antoniou, A.; Micaelli, P.; and Storkey, A. J. 2022. Meta-Learning in Neural Networks: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(9): 5149–5169.

Hou, M.; and Sato, I. 2022. A Closer Look at Prototype Classifier for Few-shot Image Classification. In *NeurIPS*.

Hu, S. X.; Li, D.; Stühmer, J.; Kim, M.; and Hospedales, T. M. 2022. Pushing the Limits of Simple Pipelines for Few-Shot Learning: External Data and Fine-Tuning Make a Difference. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, 9058–9067. IEEE.

Lan, L.; Wang, X.; Hua, G.; Huang, T. S.; and Tao, D. 2020. Semi-online Multi-people Tracking by Re-identification. *Int. J. Comput. Vis.*, 128(7): 1937–1955.

Li, A.; Luo, T.; Lu, Z.; Xiang, T.; and Wang, L. 2019. Large-Scale Few-Shot Learning: Knowledge Transfer With Class Hierarchy. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 7212–7220. Computer Vision Foundation / IEEE.

Liu, L.; Zhou, T.; Long, G.; Jiang, J.; and Zhang, C. 2022. Many-Class Few-Shot Learning on Multi-Granularity Class Hierarchy. *IEEE Trans. Knowl. Data Eng.*, 34(5): 2293–2305.

Miranda, B.; Yu, P.; Goyal, S.; Wang, Y.; and Koyejo, S. 2023. Is Pre-training Truly Better Than Meta-Learning? *CoRR*, abs/2306.13841.

Novack, Z.; McAuley, J.; Lipton, Z. C.; and Garg, S. 2023. Chils: Zero-shot image classification with hierarchical label sets. In *International Conference on Machine Learning*, 26342–26362. PMLR.

Oreshkin, B. N.; López, P. R.; and Lacoste, A. 2018. TADAM: Task dependent adaptive metric for improved few-shot learning. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds.,

Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, 719–729.

Parmar, J.; Chouhan, S. S.; Raychoudhury, V.; and Rathore, S. S. 2023. Open-world Machine Learning: Applications, Challenges, and Opportunities. *ACM Comput. Surv.*, 55(10): 205:1–205:37.

Rajeswaran, A.; Finn, C.; Kakade, S. M.; and Levine, S. 2019. Meta-Learning with Implicit Gradients. In Wallach, H. M.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E. B.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 113–124.

Ravi, S.; and Larochelle, H. 2017. Optimization as a Model for Few-Shot Learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. S.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.*, 115(3): 211–252.

Silla, C. N.; and Freitas, A. A. 2011. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22: 31–72.

Snell, J.; Swersky, K.; and Zemel, R. S. 2017. Prototypical Networks for Few-shot Learning. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 4077–4087.

Sokal, R. R.; and Rohlf, F. J. 1962. The comparison of dendrograms by objective methods. *Taxon*, 33–40.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2818–2826. IEEE Computer Society.

Tan, H.; Zhang, X.; Zhang, Z.; Lan, L.; Zhang, W.; and Luo, Z. 2021. Nocal-Siam: Refining Visual Features and Response With Advanced Non-Local Blocks for Real-Time Siamese Tracking. *IEEE Trans. Image Process.*, 30: 2656–2668.

Tian, Y.; Wang, Y.; Krishnan, D.; Tenenbaum, J. B.; and Isola, P. 2020. Rethinking Few-Shot Image Classification: A Good Embedding is All You Need? In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J., eds., *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, 266–282. Springer.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 10347–10357. PMLR.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; and Wierstra, D. 2016. Matching Networks for One Shot Learning. In Lee, D. D.; Sugiyama, M.; von Luxburg, U.; Guyon, I.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 3630–3638.

Wang, Y.; Chao, W.; Weinberger, K. Q.; and van der Maaten, L. 2019. SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning. *CoRR*, abs/1911.04623.

Wang, Y.; Yao, Q.; Kwok, J. T.; and Ni, L. M. 2021. Generalizing from a Few Examples: A Survey on Few-shot Learning. *ACM Comput. Surv.*, 53(3): 63:1–63:34.

Willes, J.; Harrison, J.; Harakeh, A.; Finn, C.; Pavone, M.; and Waslander, S. 2022. Bayesian embeddings for few-shot open world recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Ye, H.; and Chao, W. 2022. How to Train Your MAML to Excel in Few-Shot Classification. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Ye, H.; Hu, H.; Zhan, D.; and Sha, F. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 8805–8814. Computer Vision Foundation / IEEE.

Yosinski, J.; Clune, J.; Bengio, Y.; and Lipson, H. 2014. How transferable are features in deep neural networks? In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N. D.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, 3320–3328.

Zhang, H.; Cissé, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Zhang, X.; Meng, D.; Gouk, H.; and Hospedales, T. M. 2021. Shallow Bayesian Meta Learning for Real-World Few-Shot Recognition. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 631–640. IEEE.